Machine Learning based Short Term Electric Load Forecasting for Domestic Users

By

Anum Niaz

00000274147

Supervisor

Dr. Hashir Moheed Kiani

Department of Computing

School of Electrical Engineering and Computer Science (SEECS)

National University of Sciences and Technology (NUST)

Islamabad, Pakistan

July 2022

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS thesis written by ***Ms. Anum Niaz***, 00000274147, of School of Electrical Engineering and Computer Science (SEECS) have been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for the award of MS degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature:

Name of Supervisor: **Dr. Hashir Moheed Kiani**

Date: 05-Jul-2022

Signature (HOD): ________________________

Date: ________________________

Signature (Dean/Principal): ________________________

Date: ________________________

Approval

It is certified that the contents and form of the thesis entitled “Machine Learning based Short Term Electric Load Forecasting for Domestic Users" submitted by Anum Niaz have been found satisfactory for the requirement of the degree.

Advisor: **Dr. Hashir Moheed Kiani**

Signature:

Date: 05-Jul-2022

Committee Member 1: **Dr. Zuhair Zafar**

Signature: ___________________________

Date: 06-Jul-2022

Committee Member 2: **Dr. Muhammad Daud Abdullah Asif**

Signature: ____________________

Date: 07-Jul-2022

Dedicated to my parents and my husband for their never-ending support

Certificate of Originality

I hereby declare that this submission titled "Machine Learning based Short Term Electric Load Forecasting for Domestic Users" is my own work. To the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST SEECS or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECS or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics, which has been acknowledged. I also verified the originality of contents through plagiarism software.

Student Name: ANUM NIAZ

Student Signature:

Acknowledgments

I am thankful to my Creator Allah Subhana-Watala to have guided me throughout this work at every step and for every new thought which You set up in my mind to improve it. Indeed, I could have done nothing without Your priceless help and guidance.

Whosoever helped me throughout the course of my thesis, whether my parents or any other individual was Your will, so indeed none be worthy of praise but You. I am utterly thankful to my parents and my husband for their support; I am nothing but a piece of everything you have wanted me to be.

I would also like to express my utmost gratitude to my supervisor Dr. Hashir Moheed Kiani for his help throughout my thesis and his guidance, tremendous support, and cooperation. I can safely say that I wouldn't have been able to complete my thesis without his in-depth knowledge of this field. Each time I stumbled because of my personal and professional issues; he was there with a helping hand that got me where I am right now in my professional education. There are not enough words to appreciate his patience and guidance throughout my entire thesis. I am heartily thankful to him. His encouragement, guidance, and support from the initial to the final level enabled me to develop an understanding of the subject. Finally, I would like to express my gratitude to all the individuals who have rendered valuable assistance to my study.

Anum Niaz

Table of Contents

List of Figures

List of Tables

List of Equations

Abstract

Pakistan is facing a lot of problems in generating the required amount of energy and supplying it to the people. Most of the utility companies use load forecasting to anticipate how much power they'll expect to fulfil the demand of the electricity. Load forecasting helps the utility companies to generate the required amount of energy which will benefit the companies economically. Load forecasting can be classified in three ways: Short Term Load Forecasting, Medium-Term Load Forecasting and Long Term Load Forecasting. In this research, we have focused on Short Term Load Forecasting (STLF). STLF is forecasting approach with a period of a couple of hours to a day. In past years, multiple machine learning algorithms and models are implemented and tested to predict the electric load accurately. It's possible that the forecasted outcomes might have yielded better results but the disadvantage is that they have used the whole data set to create the model and when the new data value is added, they reprogrammed the whole system from the start. This has resulted in several issues. When a large set of data is used to create a machine learning algorithm, it takes a lot of time to create and test the model; it requires great storage to store the big data and lastly it consumes power to process the large data set again and again for reprogramming which results in delayed data processing. The purpose of this research is to use an online machine learning method to create a model using STLF technique in which data is presented in a progressive sequence and the model seeks to learn and upgrade for the accurate prediction of new data points at each phase that will minutely forecast the electric load which would result in better power management. With Short-Term Electric Load Forecasting we can foretell the electric load and according to that we can smartly manage the power consumption, generate the electricity as per the demand and improve the situation of load shedding in Pakistan. We have implemented Recursive Least Square with Forgetting Factor algorithm to forecast the electric load for Domestic users using PRECON dataset. The results we have obtained from this model are quite promising. The average MAPE % ranges from 0.6996% to 2.162% in the month of June and in December ranges from 0.1567% to 4.2864%, which shows that our proposed model has outperformed and we are getting the optimal results.

Chapter 1

Introduction

1.1 Machine Learning

This globe has a huge amount of records, which include the Internet of Things (IoT) records, cell records, cybersecurity records, commercial enterprise records, social media records, fitness records, etc. To effectively exploit these records and build intelligent, automated applications, knowledge of artificial intelligence (AI), especially machine learning (ML), is required. There are numerous sorts of ML algorithms in this field, based on supervised, unsupervised, semi-supervised, and reinforcement learning. In current years, ML has extended quickly with regards to information examination and calculation, permitting applications to work keenly. ML is one of the most well-known current innovations in the fourth modern insurgency since it permits frameworks to gain and improve as a matter of fact without being unequivocally coded.

Machine learning is an interesting branch of artificial intelligence that is all around us. Its objective is to comprehend the form of the records and to create models that people can comprehend and use. It allows data to be used in new ways. This wonderful technology allows computer systems to learn and grow from their experiences by designing computer programs that can autonomously access data and complete tasks via predictions and detections. [1]

ML is a domain that is rapidly developing. There are a few factors to consider if you are working with ML techniques or exploring the significance of ML methods. There has been a significant shift in technology and how it is used in everyday life during the last several years. Machine learning is being recognized as the technology that is powering our innovations, from robots to search engines.

The nature of data and the efficiency of the algorithms, evaluate the performance of the created ML model. Multiple algorithms exist to create efficient machine learning models; algorithms are regression, classification evaluation, reinforcement learning, etc.

It's challenging to select an algorithm that is appropriate for the intended application in a specific space. The justification behind this is that distinctive ML algorithms fill various needs, and, surprisingly, the aftereffects of practically identical algorithms can vary depending upon the information and the records. Thus, it's basic to appreciate the standards of different ML algorithms as well as their significance in an assortment of true situations. [2]

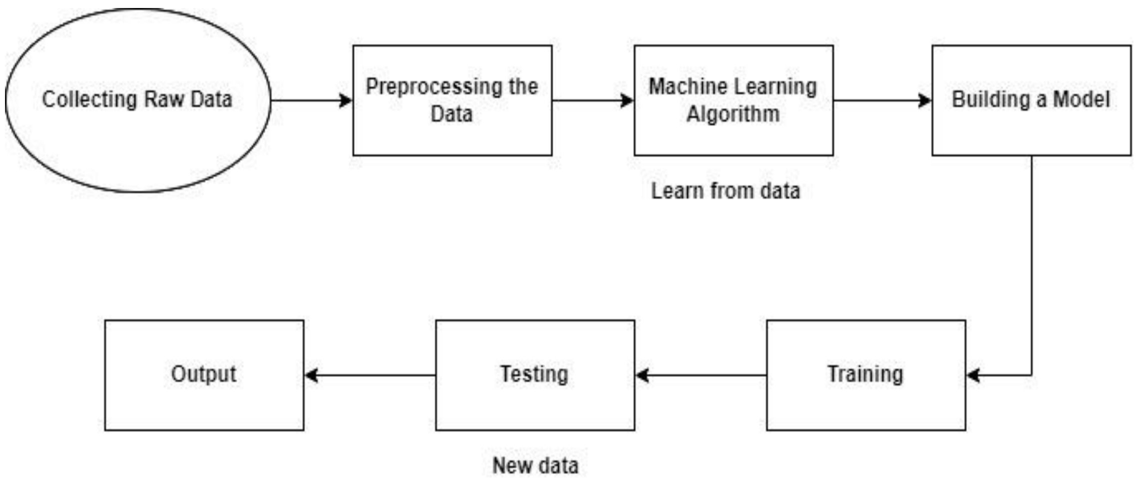

Figure 1: Workflow of Machine Learning Algorithm

1.2 Load Forecasting

Power organizations use load estimating to anticipate how much power they'll expect to keep market interest adjusted. Load forecasting helps the utility companies to generate the required amount of energy which will benefit the companies economically as:

- Shift towards the renewable energy is one of the reasons to forecast the electric load.
- Less fuel will be consumed for generation of power results in less emission and it will benefit to the environment

1.3 Types of Load Forecasting

Load forecasting can be classified in three ways:

1. Short Term Load Forecasting

2. Medium-Term Load Forecasting
3. Long Term Load Forecasting

MTLF (Medium-Term Load Forecasting) is a forecasting approach with a time frame ranging from half a month to a year whereas LTLF (Long-Term Load Forecasting) is a forecasting approach with a time frame ranging from one year to several decades. [3]

1.4 Short-Term Electric Load Forecasting (STLF)

Power utility companies use effective ways to forecast the amount of energy or power is to maintain demand and supply balance. Forecasting accuracy is critical for a power company's operational and managerial load. One of the methods is Short-Term Electric Load Forecasting.

Short-Term Load Forecasting (STLF) is an estimating approach with a period of a couple of hours to a day. STLF is essential for efficient power system planning and its function. Load forecasting has been done in a variety of ways during the last many decades. Load forecasting is a very important activity from both a power system and an economic standpoint; hence there is a lot of research going on in this field.

STLF is a prominent player in the development of cost-effective, dependable, and secure power system operating techniques. The main goal is to offer load estimates or predictions for scheduling operations and analyzing the power system's security at any given time.

Short-term electric load prediction will benefit the utility companies economically. They would know the demand of the power consumption and accordingly, they will generate the electricity. An excessive amount of electricity would not be generated. In this way, resources will be managed smartly and the cost of power consumption will also be reduced. Hence, circular debt will be controlled.

The advantages of Short Term Load Forecasting (STLF) are that it facilitates the minimization of ongoing expenses by giving exact contribution to one day prior scheduling. A power company that can generate up to 10,000 MW can save $1.6million annually if it reduces forecasting error by 1%. [4]

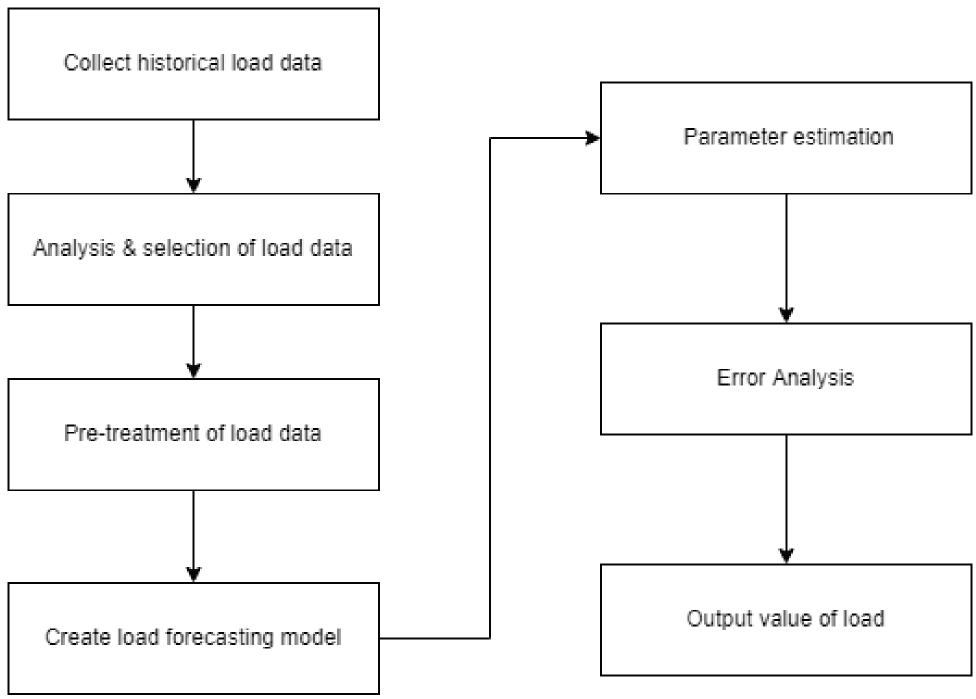

Figure 2: Flowchart of STLF Methodology [2]

1. 5 The System Load

The aggregation of all the demands of the electric load is called system load. If each consumption pattern is known, the system load pattern could theoretically be determined. The demand of the customer's load is quite unpredictable. The factors that influence the system load behavior are:

1. Time
2. Economic
3. Weather

1.5.1 Time Factor

The weekly day cycle, seasonal changes, governmental and religious holidays are all key elements that influence load patterns. The quantity of sunlight hours and change in temperature can cause progressive changes in the load pattern.

1. 5.2 Economic Factor

Power consumption patterns are influenced by the economic factor in which the utility functions. Factors such as demographics in the service area, modern action levels, changes in the cultivating area, the electrical appliances and the devices used according to the location and trends, regulatory climate changes, and, more broadly, global changes all have a sizable effect on system load trends.

1.5.3 Weather Factor

The most notable weather conditions factor is the “temperature”. Significant fluctuations in the load pattern are caused by weather conditions. This is because most utilities have significant weather-sensitive load components, such as those from space heating, air conditioning, and agricultural irrigation. [1]

1.6 Problem Statement

Renewable power assets make up more than a quarter of the world's power today, however as indicated by the IEA (International Energy Agency) this is going to further boost in the future. Pakistan's renewable resources are also increasing day by day. Perhaps, the greatest concern of the renewable energy is, as it relies on the natural resources, there is no fixed amount of energy that can be generated. So, it is uncontrollable and the value fluctuates.

In addition to this, Pakistan is facing a lot of problems in generating the required amount of energy and supplying it to the people. For this, the Short-Term Electric Load Forecasting will minutely forecast the electric load which would result in better power management. With Short-Term Electric Load Forecasting we can foretell the electric load and according to that we can smartly manage the power consumption, generate the electricity as per the demand and improve the situation of load shedding in Pakistan. [5]

1.6.1 High memory requirements

One of the most important factors that effects load forecasting is the high memory requirements. The electric data of the houses are required to be stored and then preprocessing is done on the stored data. It requires a lot of storage which results in complications.

1.6.2 High power consumption

For the big data, a lot of memory is required which result in high power consumption. The purpose of load forecasting is to predict the load and to supply the demand power only. For this, the problem of high power consumption is a major concern.

1.6.3 Processing delay

There would be processing delays if large memory is consumed to store the electric data in the memory, hence machine learning model would take time to build and more power would also be consumed.

1.7 Motivation

In a competitive market, load forecasting is vital for power utility companies. Power generation and purchasing, load switching, transmission, distribution and infrastructure improvement are just a few of the uses. Electric burden determining models that are exact are vital to a utility's proceedings and setting up. Load determining helps an electric utility in settling on basic choices, for example, power buy and age, load exchanging, and foundation development. Energy providers, ISOs, monetary organizations, and different members in electric energy age, transmission, appropriation, and markets depend intensely on load projections.

Load anticipating is basic for utilities since market interest vary, and weather patterns and energy costs can increment by an element of at least 10 during top periods. Short-term load forecasting can help with the assessment of burden streams and the creation of choices to abstain from over-burdening. Carrying out such choices on time further develops network dependability and diminishes the probability of hardware disappointments and power outages.

Load forecasting is also vital for contract assessments and for evaluating the market's different complex financial instruments on energy price. Capital consumption choices in light of long haul estimating are likewise more significant in a liberated economy than in a non-liberated climate where rate increases could be defended by capital use projects. [5] The online load forecasting technique through machine learning will help in less memory and power consumption which would result in less processing delays.

1.8 Research Objective

Putting away electrical energy is incomprehensible. Whenever there is an interest in it, it should be delivered. Subsequently, it is basic for electric power organizations to foresee the interest in their frameworks early. Load forecasting is the term for assessing load early. It is expected for the preparation of force frameworks. The initial phase in power framework extension arranging is to conjecture future burden necessities. A reasonable model for electric power forecasting is expected for the operation and arrangement of a power service organization.

An electric utility can utilize load determining to assist them with settling on choices about producing and buying power, load exchanging, voltage control, network reconfiguration, and foundation advancement. The act of projecting future electric load involving chronicled load and climate information as well as present and anticipated climate information is known as electric load forecasting. A few models have been created in ongoing a very long time to all the more accurately expect electric load. With the liberation of the power business, the members in the energy market have confronted a huge number of new snags. [5]

For electricity management, load forecasting is a core part of any model, particularly in the present changing power framework structure. This demand is decided by the level of forethought and precision. [7]

1.8.1 Efficient resource utilization

Load forecasting is helpful in predicting the necessary assets, for example, fuels to produce the generating plants and different assets, to assure consistent and cost-effective supply and transmission.

1.8.2 Online training of machine learning model

The fundamental target of this research is to create an online machine learning model by training the model by using the most recent data examples and previous weights. It will result in a fast processing and less memory consumption. Hence, dataset is not required to be stored.

1.9 Sustainable Development Goals

Following are the Sustainable Development Goals of our research:

1.9.1 Responsible usage of electric power

Load forecasting empowers the power companies to plan effectively since they get a clear prediction of overall usage or load demand.

1.9.2 Day-ahead power prediction

Load forecasting ensures that power plants are used to their full potential. The forecasting prevents under-generation or over-generation of the electricity.

1.9.3 Effective memory consumption

The aim of this research is to consume the memory effectively by using the most recent data values and discarding the previous ones. This will give in faster results and there would be less processing delays as data would not be stored.

1.9.4 Saving Resources

Load forecasting is used to predict the power required in the future for demand and supply. The goal is to save the power generating resources like fuel; it will result in fewer emissions that will benefit the environment as pollution would be reduced. [6]

Chapter 2

Literature Review

Load Forecasting

Power organizations use load estimating to anticipate how much power they'll expect to keep market interest adjusted. Load forecasting helps the utility companies to generate the required amount of energy which will benefit the companies economically.

Types of Load Forecasting

Load forecasting can be classified in three ways:

1. Short Term Load Forecasting
2. Medium-Term Load Forecasting
3. Long Term Load Forecasting

In this research we have used Short Term Load Forecasting (STLF) to predict the electric load as we are predicting with a period of a couple of hours to a day. There are multiple methods and techniques in Machine Learning to forecast the electric load using STLF. After doing a detailed literature review, we have come up with the following techniques.

1. Neural Networks
2. Extreme Gradient Boosting
3. Particle Swarm Optimization (PSO) Algorithm
4. Deep Learning
5. Multiple Linear Regression
6. Support Vector Regression
7. Support Vector Machines
8. Internet of Things

Following are the papers that are related to Neural Networks:

Since power assumes an urgent part in nations' modern foundations, power organizations are attempting to screen and control frameworks to further develop energy the board and booking. Exact estimating is a basic assignment for a steady and proficient load supply, in which demand and contribution of electricity are coordinated. In this article, for the investigation of STLF, convolutional neural network (CNN) and long short-term memory (LSTM) are consolidated with deep learning model and different calculations. Model that is presented in this article is called equal LSTM-CNN Network. “Malaysia’s hourly load utilization" as well as “Germany’s day to day load utilization", are utilized to access and analyze the introduced models. In order to test the models' exhibition, Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), and R-squared were utilized. Altogether, article is isolated into two sections. The initial segment shows that there are multiple AI models, which also includes PLCNet, whose purpose is to foresee time step load. The subsequent part shows the presentation of the model that presents the most dependable outcomes in the initial segment. The outcomes presented by convolutional neural network models, particularly PLCNet, are a great possibility for utilizing as short-term forecasting. For the German information, PLCNet worked on the precision from 83.17% to 91.18%, and in Malaysian information, 98.23% exactness was accomplished, an amazing outcome in power estimating. [8]

An effective administration and planning by the energy organizations are of extraordinary importance for exact power load forecasting. There is an elevated degree of vulnerabilities in load time series, which is trying to make the precise and long-term load forecasting (LTLF), medium-term load estimate (MTLF), short-term load forecasting (STLF). To separate the neighborhood patterns and to catch similar examples of medium and short load time series, we suggested Long Short-Term Memory (LSTM), Multilayer perceptron, and Convolutional Neural Network (CNN) to gain proficiency. These models are proposed to further develop the precision. The models were created and tested in light of this present reality case by leading point-by-point examinations to approve their dependability and reasonableness. The results were tested by MAE, MAPE, R2, and RMSE. [5]

As a significant help for the advancement of the public economy, the power business assumes a part in guaranteeing monetary tasks. Time series expectation can handle dynamic information, is broadly utilized in financial aspects and designing, and particularly is of incredible reasonable worth in utilizing verifiable information to anticipate future turn of events. In the direction of time series hypothesis and extreme learning machine, this paper put in the extreme learning machine to the investigation of time series, and fabricated a model. Load anticipating assumes a significant part in power arranging, influencing arranging activity modes, power trade plans, and so on, so load estimating is extremely vital in power arranging. In the first place, laid out an extreme learning machine model; secondly, the STLF is performed by various functions after experimental examination, the function with the best prediction is acquired. [9]

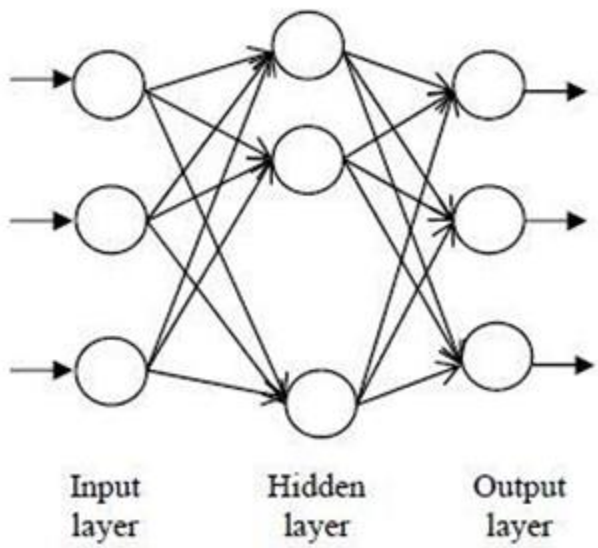

Figure 3: Network structure of the model

In accurate prediction and development of power, Short-term Load Forecasting (STLF) is a vital component. The motivation behind power estimating is, to adjust the demand and power supply. The load changes throughout the time so the electric load is dynamic. The arrangement of power is likewise powerful following the example of load changes. Load anticipating is expected to guarantee a precise choice of electric plant booking, section/unit responsibility, and electric load conveyance. This paper presents Recurrent Neural Network (RNN) model with Levenberg-Marquardt and Bayesian regularization was prepared for Short-term load forecasting (STLF). The exactness standard utilized is the Mean Absolute Percentage of Error (MAPE). The RNN model makes great expectations as the outcome has presented this. The training algorithm of RNN model along with the Bayesian regularization preparing has great precision. [10]

Extreme Gradient Boosting is one of the methods of STLF which gives tremendous results. One of the papers related to this technique is:

Forecasting has incredible significance in the electrical power framework. It informs the framework ahead of time about the future burden demand and helps the framework administrator to settle on choices with respect to turning saves, monetary dispatch, unit responsibility, and request side administration, and so forth. Frameworks of Educational establishments request power with great quality and dependability. Load forecast of Educational organizations varies from different customers in light of delicate lab contraptions, complex instruments, and hardware. To meet the prerequisites of Educational foundations, mix of sustainable power sources with a public matrix require information on the precise interest of energy; a vital aspect for building supervisors and organizers. Remembering this, we attempted to conjecture the load of an Educational Institute of Pakistan as a contextual investigation by investigating decision tree-based calculation (XGBoost). The outcomes were amazing, and forecasts appear to be close to ideal. Beforehand, this information has never been utilized in forecasting. Notwithstanding, via preparing a XGBoost model, short-term load forecasting (STLF) was finished. Results are determined by MAE, MAPE, R2, and RMSE of anticipated and real qualities. By effectively utilizing the renewables during the pinnacle hour, the top burden request can be managed. [11]

Following paper shows how PSO is used as a STLF technique:

Power load forecasting assumes a pivotal part in energy framework development and operational strategies. Up until this point, an assortment of methods has been utilized for load forecasting. In the interim, neural-network-based techniques prompted fewer expectation mistakes because of their capacity to adjust appropriately to its load's secret trademark to consume. Subsequently, these strategies were broadly acknowledged by scientists. As the boundaries of the neural network altogether affect its presentation, in this paper, a transient electrical burden anticipating strategy utilizing neural network and particle swarm optimization (PSO) calculation is proposed, in which some neural network boundaries are explained. Then, the neural network with these improved boundaries is utilized to anticipate the STLF. In this technique, a three-layer feedforward neural network prepared by backpropagation calculation is utilized close to an improved PSO calculation. Likewise, the neural network forecast error is characterized as the PSO calculation cost work. The suggested approach has been examined on the Iranian power matrix utilizing MATLAB programming. The normal of three lists alongside graphical outcomes have been considered to assess the presence of the suggested technique. [12]

Deep Learning is on top trending of all the STLF techniques to forecast the electric load. One of the papers regarding this method is:

Due to smart grids, load forecasting is arising one of the fundamental innovations to conduct ideal preparation and control of network resources. For the demanded energy, a critical push is seen for the enhancement and development of the energy. Anyway subsequently there has been many procedures implemented and presented in the papers that shows the best prediction result. Every other paper that is based on load forecasting reported their one of the best outcomes but for the specific dataset, i.e. the one they are working on. This is one of the most highlighted issues; the training data should be given enough data sample in order to learn. The major focus here is to come up with a load determining procedure that can give the most accurate forecast on multiple datasets, where training data is restricted. The paper gives a clever blend of deep learning along with STLF. Deep Derived Feature Fusion (DeepDeFF), the suggested design, depends on the consecutive model to create the model based on finer expectations and learning. At independent levels, the information and the features are prepared, and after that their separate results are joined to get the final forecast. The viability of the suggested procedure is assessed on assorted data samples from multiple nations with totally various examples. The broad tests and outcome exhibit that the given strategy is better than the current state of the craftsmanship. [13]

The most used technique that gives the best result is the regression technique. Some of the papers regarding multiple linear regression and support vector regression are:

In this paper, the author has presented an examination of short term energy (up 24 hours) estimating the interest for the South's Indonesia electric Utility, utilizing a Multiple Linear Regression (MLR) strategy. The use of polynomial terms and means to form the MLR model will be made sense, following a concise logical conversation of the procedure. Information on execution of MLR calculation utilizing a financially accessible device like MS EXCEL is also explained. As a contextual analysis, previous data comprising hour by hour load demand and weather of South Indonesia power framework is utilized, to forecast the STLF. The outcomes are introduced and broken down as potential for developments in this paper. [14]

The - Support Vector Regression (- SVR) is a powerful relapse learning calculation, which enjoys the benefit of utilizing a boundary on controlling the number of help vectors and changing the width of the cylinder naturally. Be that as it may, contrasted with - Support Vector Classification (- SVC) - SVR brings an extra straight term into its goal work. In this way, straightforwardly applying the precise online - SVC calculation (AONSVM) to - SVR won't produce a compelling beginning arrangement. It is the primary test to plan a gradual - SVR learning calculation. To beat this test, we propose an exceptional technique called starting changes in this paper. This system changes loads of - SVC in light of the Karush-Kuhn-Tucker (KKT) conditions to set up an underlying answer for gradual learning. Joining the underlying changes with the two stages of AONSVM produces an accurate and successful gradual - SVR learning calculation (INSVR). The hypothetical examination has demonstrated the presence of the three vital reverse lattices, which are the foundations of the three stages of INSVR (counting the underlying changes), individually. The outcomes additionally show that INSVR is quicker than clump - SVR calculations with both cold and warm beginnings. [15]

Support Vector Machines are considered to be the most authentic way in prediction. Following paper is about SVM:

The exact forecasting of the short-term load is a significant issue in the power industry. This paper proposes another determining model by incorporating the support vector machines (SVMs) estimating procedure and rough sets (RSs) with diminished credits utilizing developmental calculations (EAs). Recreation results show that this new model can work on the forecast exactness, speed the assembly, and require less computational exertion in correlation with another two strategies, to be specific the customary SVM model and a model joining the SVMs and mimicked tempering calculations (SVMSA). This improvement is connected with the truth that the RS methods can diminish the SVM input factors and work on the convergence. [16]

Internet of Things (IoT) is the most recent technique used with STLF to predict the future. The following paper shows IoT integrated with Machine Learning.

This paper presents an original online load estimating utilizing regulated Machine Learning (ML) algorithm in Internet of Things (IOT). Loading the board and power dispatch, I he fundamental angle of Short Term Load Forecasting (STLF) for smart grids. IOT is an arising mechanization breaking into each fragment of designing. This work hand overs the chance of STLF online with exact forecast models by utilizing ML calculations. Power load utilization information and climate information at an examination Lab, JNTUH, Hyderabad is utilized to prepare ML calculations to carry out STLF. Through distributed computing, ML calculations estimating models are created utilizing MATLAB code. Web based prediction is more modern and successful on account because of its capacity to utilize late information records for preparing and determining on the web. Web based prediction is valuable in Online Home Energy Management Systems (OHEMS) for powerful energy. ML calculations like Gaussian Process Regression (GPR), Support Vector Machines (SVM), Ensemble Boosted (EBo) relapse, Linear Regression (LR), for relapse, Ensemble Bagged (EB) relapse, Gaussian Process Regression (GPR) and Fine Tree (FT) relapse are executed on the cloud to conjecture the energy utilization. Execution boundaries, for example, RMSE, MSE and MAE are inferred to assess the viability of the ML calculations carried out. Financially savvy Arduino Uno, Node MCU/ESP8266, PZEM 004T and DHT 11 sensors are utilized to manufacture the equipment model to get the heap information for the suggested load determining technique. The most ideal ML calculation is recommended for the suggested web based prediction with assisting outcomes. [17]

Chapter 3

Problem Formulation and Proposed System Design

3.1 Overview

In this study, we have used Python libraries to implement and create the models that can predict one day ahead electric load. The libraries used are numpy, pandas, scikit-learn and matplotlib.

3.2 Pakistan Residential Electricity Consumption (PRECON) Dataset

PRECON [16] dataset is collected over a period of one year (2018 – 2019) from domestic users through smart meters. Data from 42 houses was collected for this dataset. The dataset has data points of date and time, and the electric load on that date and time of all the houses. Minute per minute electric load is given in this dataset. So, we need to find out minute by minute temperature and the predicted load. In order to determine minute by minute temperature '*cubic spline interpolation'* has been used. The temperature data is added in the PRECON dataset houses in parallel to the electric load which shows that on a particular time, at this temperature, this was the electric load of that house.

Independent Variables		**Dependent Variable**
Temperature	**Previous Load**	**Predicted Load**
T_n	L_{n-1}	L_n
T_{n-1}	L_{n-2}	L_{n-1}
T_{n-2}	L_{n-2}	L_{n-2}
. . .	. . .	. . .
T_2	L_1	L_{21}
T_1	L_0	L_1

Table 1: Structure of Independent and Dependent variables present in dataset

3.3 Cubic Spline Interpolation

The basic concept of cubic spline interpolation is to draw continuous curves. Weights are fastened to a horizontal plane at the locations to be linked on the curve. The weights are then stretched against each other with a bendable strip, giving in a wonderfully smooth curvature. In essence, the arithmetical spline is comparable. In this scenario, the dots are quantitative data. Cubic polynomial interpolates because of the coefficients that are the weights. The line bends with the help of the coefficients, causing it to pass from one data point to another without breaking.

The mathematical method of cubic spline interpolation is widely used to create new points within the bounds of a set of known points. Interpolation function also known as spline is formed from these new points. New points are the function values of the interpolation function. In cubic spline, the curve is smooth and continuous. These are used to find out the change over an interim.

Here, as we have hourly data of temperature, so new data points are created to get the minute by minute temperature within the boundaries of the known point via interpolation.[18]

The idea is to fit the known points of the form:

$$T(x) = \begin{cases} t_1(x) & if \quad x_1 \leq x < x_2 \\ t_2(x) & if \quad x_2 \leq x < x_3 \\ t_3(x) & if \quad x_3 \leq x < x_4 \\ & \vdots \\ t_{n-1}(x) & if \quad x_{n-1} \leq x < x_n \end{cases}$$

where $\boldsymbol{t_i}$ is a 3rd degree polynomial that is written as defined by

$$\boldsymbol{t_i(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i}$$

where i = 1, 2, 3, ……, n-1.

The spline interpolation graph shown below is between time and temperature. It shows the new data points of temperature ranging between the given intervals of temperature. The graph shows the continuous and smooth curve between the hourly temperatures. Here, cubic spline interpolation is done of 1st of June 2018 of house 1.

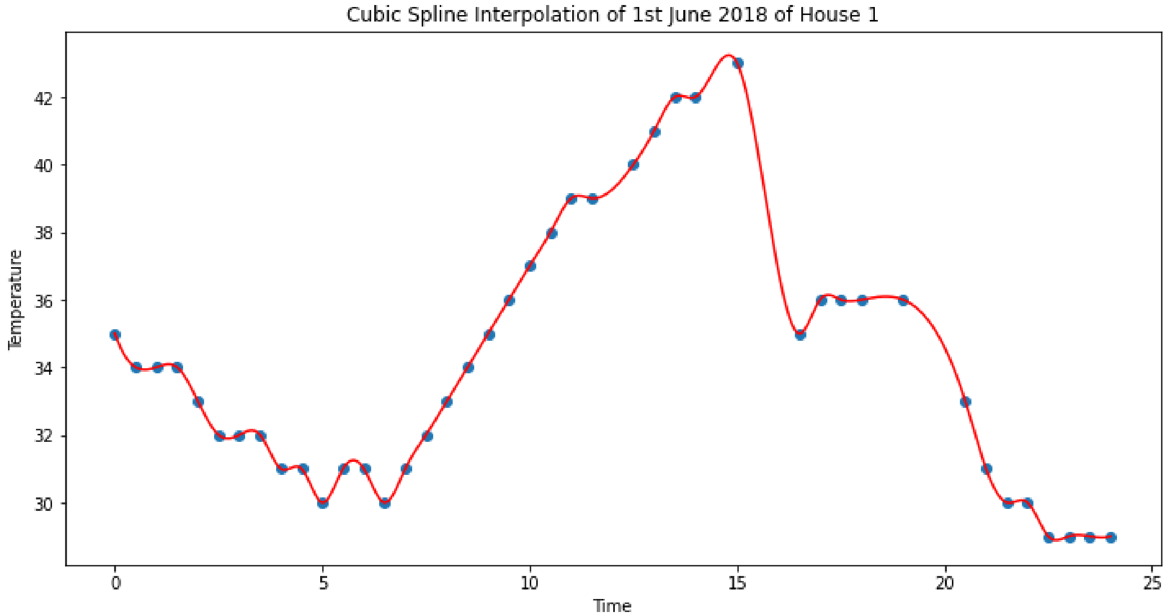

Figure 4: Cubic Spline Interpolation of June 1st, 2018 - House 1

3.4 Problem Formulation

Renewable power assets make up more than a quarter of the world's power today, however as indicated by the IEA (International Energy Agency) this is going to further boost in the future. Pakistan's renewable resources are also increasing day by day [19]. Perhaps, the greatest concern of the renewable energy is, as it relies on the natural resources, there is no fixed amount of energy that can be generated. So, it is uncontrollable and the value fluctuates.

In addition to this, Pakistan is facing a lot of problems in generating the required amount of energy and supplying it to the people. In past years, multiple machine learning algorithms and models are implemented and tested to predict the electric load accurately. The predicted results may have given better results but the disadvantage is that these approaches are offline which use the whole dataset to create the model and addition of new data requires the model to be created again. This has caused a lot of problems. When a large set of data is used to create a machine learning algorithm, it takes a lot of time to create and test the model; it requires great storage/memory to store the big data and lastly it consumes power to process the large data set again and again for reprogramming which results in delayed data processing.

For this, an online Short-Term Electric Load Forecasting technique will minutely forecast the electric load which would result in better power management. With Short-Term Electric Load Forecasting we can foretell the electric load and according to that we can smartly manage the power consumption, generate the electricity as per the demand and improve the situation of load shedding in Pakistan.

One of the most important factors that effects load forecasting is the high memory requirements. The electric data of the houses are required to be stored and then preprocessing is done on the stored data. Secondly, it requires a lot of storage which results in complications. The purpose of load forecasting is to predict the load and to supply the demand power only. Thirdly, there would be processing delays if large memory is consumed to store the electric data in the memory, hence machine learning model would take time to build and more processing power would also be consumed. [20]

These problems can be addressed if instead of using the whole data set for implementing, creating, and testing model, only the most recent data example and the previous model is used to create the new model. Once the new model is created, discard the data that was used to create it. If a new data point is added in the data set, again use that new data point and the current model to create the new model and discard that data as well. In this way, memory would be managed as less storage will be required to store the data, less processing power would be required, and the model would be created in real-time, hence less processing delay. This is called '*Online Machine Learning'*.

3.5 Online Machine Learning

Machine learning algorithms that have been around for a long time operate in batch mode. For instance, in supervised learning, the entire training data is provided ahead in order to train a model to use certain methods. Due to the high expense of training, such a technique necessitates having all of the training data accessible before the learning job. Traditional approaches have many significant shortcomings, including storage and time cost; and limited expandability for extensive applications since the model must often train up again for updated data.

Online learning is a hybrid of multiple ML approaches in which data is presented in a progressive sequence and the model seeks to learn and upgrade for the accurate prediction of new data points at each phase. Online learning overcomes the disadvantages of offline learning by allowing models to be modified instantaneously in response to changes in information. As a result, in real-world data for wide-scale learning tasks, statistics, and diverse applications where records are not only vast in quantity but also come at a fast rate, online learning is considerably faster and more flexible. [21]

3.5.1 Training Level and Complexity

The learner's parameters and properties are maintained while training in batch mode. The model's resilience is attained by ongoing and repetitive training. In online learning, changes in parameters and properties occur at each stage, specifically based on the new sample that is under observation and the present state of the model. As a result, the learner is always viewing the latest data and improves itself; resulting in an online mode it is data-efficient.

3.5.2 Computational Delay

In offline/batch mode, supplying the big data in the stream takes a long time. In contrary to this, online learning adapts and tweaks itself. Sometimes it contributes to higher training costs because the model will demand a huge amount of resources to be trained on a regular basis.

As shown in figure 1, batch learning also called as offline learning is creates and trains the model by using whole dataset. It extract the features from the data and based on that features algorithm is implemented. Model is created and the results are predicted accordingly. For training, new data points are taken from which same features are extracted and the results are predicted according to the trained model. Where as in online learning, shown in figure 2, real time data is taken in iterations, features on which algorithm is implemented and model is trained. Step by step iterations are done of the data and the predicted results give the final output instantly. [22]

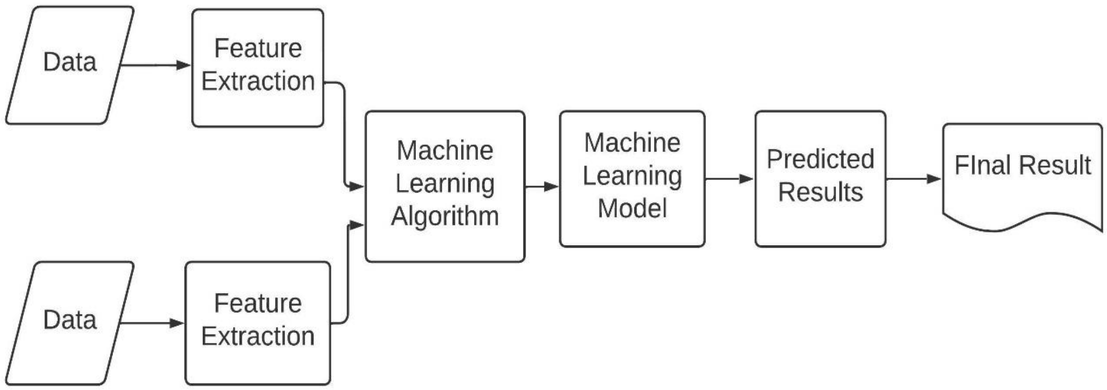

Figure 5: Batch Machine Learning

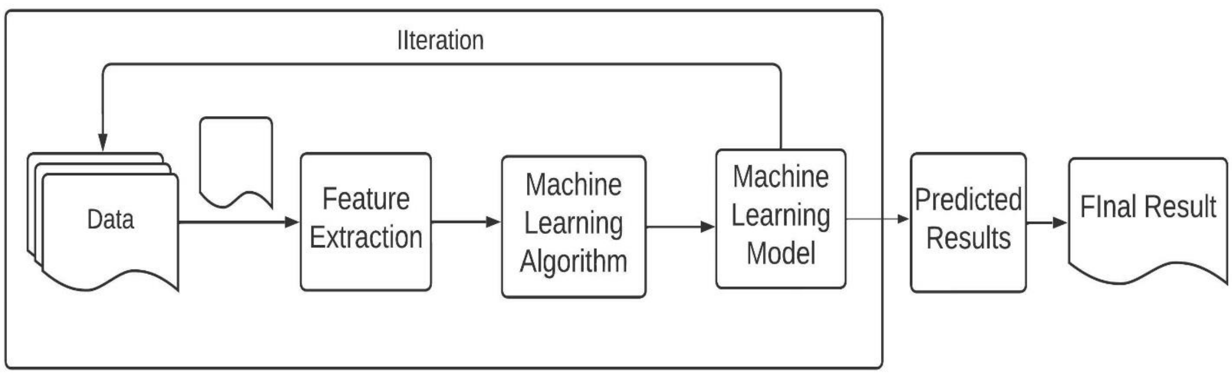

Figure 6: Online Machine Learning

3.6 Multiple Linear Regression (MLR)

By drawing a line according to the observed data, regression techniques are utilized to illustrate the relationships among variables. It is a statistical technique for predicting how much a dependent variable will make a difference when the explanatory variable(s) changes. [23]

The most prevalent type of linear regression analysis is Multiple Linear Regression (MLR). It explains the link between one dependent variable and more than one explanatory variable, to draw a fitted linear line based on the data sample. The value of dependent variables varies according to the change happening in independent variables. Each value of an independent variable (x) is associated with each value of the dependent variable (y).

Linear Regression (LR) has one explanatory (independent) variable and one dependent variable whereas Multiple Linear Regression (MLR) has two or more than two explanatory (independent) variables and one dependent variable.

General Equation

The general equation of MLR is

$$y_{\boldsymbol{i}} = \beta_{\boldsymbol{0}} x_{\boldsymbol{0}} + \beta_{\boldsymbol{1}} x_{\boldsymbol{1}} + \ldots\ldots\ldots + \beta_{\boldsymbol{n}} x_{\boldsymbol{n}} + \mathcal{E}$$

where, i = 1…….n

Symbols	**Meanings**
$y_{\boldsymbol{i}}$	dependent (response) variable
$x_{\boldsymbol{i}}$	independent (explanatory) variables
$\beta_{\boldsymbol{0}}$	y-intercept (when all the parameters are zero)
$\beta_{\boldsymbol{n}} x_{\boldsymbol{n}}$	regression coefficients for the last explanatory variable
ε	error term (residuals)

In this study, we have used Multiple Linear Regression to create a model through the explanatory variables that are temperature and previous load to predict the electric load of the next day. As our aim is to implement an incremental online model to predict the electric load of the next day so we have compared this simple MLR model with the incremental online model to predict the real-time electric load.

Let,

$y_{\boldsymbol{n}}$ = load at time n

$t_{\boldsymbol{n}}$ = temperature at time n

$y_{\boldsymbol{n-1}}$ = load at time n-1

The equation of MLR becomes,

Equation 1

$$y_{\boldsymbol{n}} = [\ y_{\boldsymbol{n-1}} \quad t_{\boldsymbol{n}}\ ] \begin{bmatrix} \theta_{\boldsymbol{1}} \\ \theta_{\boldsymbol{2}} \end{bmatrix} + \mathcal{E}$$

Equation 1 can also be written as,

$$y_{\boldsymbol{n}} = y_{\boldsymbol{n-1}}\,\theta_{\boldsymbol{1}} + t_{\boldsymbol{n}}\,\theta_{\boldsymbol{2}} + \mathcal{E}$$

The matrix of this equation 1 would be

$$\begin{bmatrix} y_{\boldsymbol{1}} \\ \cdot \\ \cdot \\ y_{\boldsymbol{n}} \end{bmatrix} = \begin{bmatrix} y_{\boldsymbol{n-1}} & t_{\boldsymbol{n}} \\ \cdot & \cdot \\ \cdot & \cdot \\ y_{\boldsymbol{0}} & t_{\boldsymbol{1}} \end{bmatrix} \begin{bmatrix} \theta_{\boldsymbol{1}} \\ \theta_{\boldsymbol{2}} \end{bmatrix} + \begin{bmatrix} \varepsilon_{\boldsymbol{1}} \\ \cdot \\ \cdot \\ \varepsilon_{\boldsymbol{n}} \end{bmatrix}$$

The final equation for MLR in our case is

$$Y = X\,\theta + \mathcal{E}$$

Error term can be found out by this equation

$$\mathcal{E} = Y - X\,\theta$$

Where,

$$X = \begin{bmatrix} y_{\boldsymbol{n-1}} & t_{\boldsymbol{n}} \\ \cdot & \cdot \\ \cdot & \cdot \\ y_{\boldsymbol{0}} & t_{\boldsymbol{1}} \end{bmatrix}, = \begin{bmatrix} \theta_{\boldsymbol{1}} \\ \theta_{\boldsymbol{2}} \end{bmatrix}, \ \varepsilon = \begin{bmatrix} \varepsilon_{\boldsymbol{1}} \\ \cdot \\ \cdot \\ \varepsilon_{\boldsymbol{n}} \end{bmatrix}$$

3.7 Least Squares

Least Squares is used to find out best fit line for a given set of data points. For a given set of data points, the best fit line or curve is found by minimizing the sum of the squares of the residual component in least squares. Every point present in the data set represents the relationship between the dependent and the independent variables. It is used to foresee the functioning and the performance of the dependent variable.

The method of least-squares explains why the best fit line should be positioned among the sample points. The goal is to draw a straight line that minimizes the sum of the squares of the residuals caused by the linked equations' outcomes, such as the squared residuals coming from disparities between the observed and anticipated values based on that model. [24]

If (x_1, y_1), (x_2, y_2), (x_3, y_3), $(x_4, y_4) \ldots\ldots (x_n, y_n)$ are the data points of a dataset where x's shows the explanatory or the independent variables and the y's shows the dependent variables. f(x) is the curve that is fitted to create the best fit line and ε is the error of the given points. The error or the offsets can be written as the difference between the dependent variable and the f(x).

$$\varepsilon_1 = y_1 - f(x_1)$$

$$\varepsilon_2 = y_2 - f(x_2)$$

$$\varepsilon_3 = y_3 - f(x_3)$$

$$\varepsilon_4 = y_4 - f(x_4)$$

$$\vdots$$

$$\varepsilon_n = y_n - f(x_n)$$

The least square represents the sum of the squares of all the residuals (offsets). The general equation is

$$S = \sum_{i=1}^{n} \varepsilon^2$$

$$\boldsymbol{S = \sum_{i=1}^{n} [y_i - f(x_i)]^2}$$

$$\boldsymbol{S = \sum_{i=1}^{n} \varepsilon_1{}^2 + \varepsilon_2{}^2 + \varepsilon_3{}^2 + \varepsilon_4{}^2 + \cdots + \varepsilon_n{}^2}$$

In this research, we have used this general equation according to our requirements. So, in order to create a model based on Recursive Least Squares (RLS), first we need to calculate Least Squares.

$$S(\theta) = \sum \varepsilon^2 = \varepsilon^T \varepsilon = (\mathrm{Y} - \mathrm{X}\,\theta)^T (\mathrm{Y} - \mathrm{X}\,\theta)$$

It can be written as,

$$S(\theta) = Y^T Y - Y^T X\theta - X^T \theta^T Y + X^T \theta^T X\theta$$

Now, in order to find out the value of θ, we will take the partial derivate

$$\frac{dS}{d\theta} = 0 - Y^T X - X^T Y + 2X^T X\theta = -2X^T Y + 2X^T X\theta$$

$$X^T Y = X^T X\theta$$

$$\theta = \frac{X^T Y}{X^T X}$$

Equation 2

$$\theta = (X^T X)^{-1} X^T Y$$

This equation gives the values of $\boldsymbol{\theta_1}$ and $\boldsymbol{\theta_2}$. If a new data sample comes in, we have to calculate the θ_{n+1} . This needs repetition of calculations and re-estimation of the inverse. For this, we will take the expression $(X^T X)^{-1}$ and $X^T Y$ to find out the inverse and θ_{n+1} .

3.8 Recursive Least Squares (RLS)

The Recursive Least Squares (RLS) method iteratively handles the least-squares problem. The weights are modified at each step when new data points are available. This saves computation time and convergence is also established at a faster rate.

In this research, we have predicted the electric load using RLS. In step 1, we used simple MLR with two data points to get an initial estimate of θ ($\boldsymbol{\theta_n}$). In step 2, we applied recursive least squares to get $\boldsymbol{\theta_{n+1}}$. Following is the derivation of RLS to achieve step 2.

From the equation 2, let's assume,

$$A^{-1} = X^T X$$

Below is the matrix form of the above equation

$$A_n^{-1} = \begin{bmatrix} y_{n-1} & . & . & . & y_0 \\ t_n & . & . & . & t_1 \end{bmatrix} \begin{bmatrix} y_{n-1} & t_n \\ . & . \\ . & . \\ y_0 & t_1 \end{bmatrix}$$

$$A_n^{-1} = \begin{bmatrix} {y_{n-1}}^2 + {y_{n-2}}^2 + \cdots + {y_0}^2 & y_{n-1} t_n + \quad y_{n-2} t_{n-1} + \cdots + \quad y_0 t_1 \\ t_n y_{n-1} + t_{n-1} y_{n-2} + \cdots + t_1 y_0 & {t_n}^2 + \quad {t_{n-1}}^2 + \cdots + \quad {t_1}^2 \end{bmatrix}$$

This can be written as

Equation 3

$$\boldsymbol{A_n^{-1}} = \sum_{i=1}^{n} \mathbf{X}_i^T \mathbf{X}_i$$

For n-1; equation 3 can be written as,

Equation 4

$$\boldsymbol{A_n^{-1}} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{n-1}} \mathbf{X}_{\boldsymbol{i}}{}^{\boldsymbol{T}}\mathbf{X}_{\boldsymbol{i}} + \mathbf{X}_{\boldsymbol{n}}{}^{\boldsymbol{T}}\mathbf{X}_{\boldsymbol{n}}$$

By putting values in equation 4, it would become as follows

Equation 5

$$A_n^{-1} = A_{n-1}^{-1} + \mathrm{X}_n{}^T\mathrm{X}_n$$

Now, from equation 2, we would take X^TY. The matrix form of X^TY can be written as,

$$\mathrm{X}_n{}^T\mathrm{Y}_n = \begin{bmatrix} \mathrm{y}_{n-1} & . & . & . & \mathrm{y}_0 \\ \mathrm{t}_n & . & . & . & \mathrm{t}_1 \end{bmatrix} \begin{bmatrix} Y_1 \\ . \\ . \\ Y_n \end{bmatrix}$$

After multiplying, we get

$$\mathrm{X}_n{}^T\mathrm{Y}_n = \begin{bmatrix} \mathrm{y}_{n-1}\,\mathrm{Y}_n \quad \mathrm{y}_{n-2}\,\mathrm{Y}_{n-1} + \cdots + \mathrm{y}_0\,\mathrm{Y}_1 \\ \mathrm{t}_n\mathrm{Y}_n + \mathrm{t}_{n-1}\,\mathrm{Y}_{n-1} + \cdots + \mathrm{t}_1\mathrm{Y}_1 \end{bmatrix}$$

Equation 6

$$\mathrm{X}_n{}^T\mathrm{Y}_n = \sum_{i=1}^{n} \mathrm{X}_i{}^T\mathrm{Y}_i$$

For n-1, equation 5 cane be written as

$$\mathrm{X}_n{}^T\mathrm{Y}_n = \sum_{i=1}^{n-1} \mathrm{X}_i{}^T\mathrm{Y}_i + \mathrm{X}_n{}^T\mathrm{Y}_n$$

Hence,

Equation 7

$$\mathrm{X}_n{}^T\mathrm{Y}_n = \mathrm{X}_{n-1}{}^T\mathrm{Y}_{n-1} + \mathrm{X}_n{}^T\mathrm{Y}_n$$

As the equation 2 is,

$$\theta = (X^TX)^{-1}X^TY$$

For θ_n , it would be

Equation 8

$$\theta_n = (\mathrm{X}_n{}^T\mathrm{X}_n)^{-1}\,\mathrm{X}_n{}^T\mathrm{Y}_n$$

From equation 3, we can write like this

$$\mathrm{A}_n = (\mathrm{X}_n{}^T\mathrm{X}_n)^{-1}$$

By placing the value in equation 7, we get

Equation 9

$$\theta_n = \mathrm{A}_n \ (\mathrm{X}_n{}^T\mathrm{Y}_n)$$

By replacing equation 6 in equation 8, we get

Equation 10

$$\theta_n = \mathrm{A}_n \ (\mathrm{X}_{n-1}{}^T\mathrm{Y}_{n-1} + \mathrm{X}_n{}^T\mathrm{Y}_n)$$

Keeping in mind the equation 8, for θ_{n-1} it would be,

$$\theta_{n-1} = \mathrm{A}_{n-1} \ (\mathrm{X}_{n-1}{}^T\mathrm{Y}_{n-1})$$

By solving the above equation,

Equation 11

$$A_{n-1}^{-1}\theta_{n-1} = \mathrm{X}_{n-1}{}^T\mathrm{Y}_{n-1}$$

Putting the values of equation 10 in equation 9, we get,

Equation 12

$$\theta_n = \mathrm{A}_n \ (A_{n-1}^{-1}\theta_{n-1} + \mathrm{X}_n{}^T\mathrm{Y}_n)$$

From, equation 5 it can be written as,

Equation 13

$$A_{n-1}^{-1} = A_n^{-1} - \mathrm{X}_n{}^T\mathrm{X}_n$$

By putting the equation 12 in equation 11 and solving it, we get

$$\theta_n = \mathrm{A}_n \ [(A_n^{-1} - \mathrm{X}_n{}^T\mathrm{X}_n)\theta_{n-1} + \mathrm{X}_n{}^T\mathrm{Y}_n]$$

$$\theta_n = \mathrm{A}_n \ [A_n^{-1}\theta_{n-1} - \mathrm{X}_n{}^T\mathrm{X}_n\theta_{n-1} + \mathrm{X}_n{}^T\mathrm{Y}_n]$$

$$\theta_n = \mathrm{A}_n \ A_n^{-1}\theta_{n-1} + \mathrm{A}_n \ \mathrm{X}_n{}^T\mathrm{X}_n\theta_{n-1} + \mathrm{A}_n \ \mathrm{X}_n{}^T\mathrm{Y}_n$$

$$\theta_n = \theta_{n-1} + \mathrm{A}_n \ \mathrm{X}_n{}^T(\mathrm{X}_n\theta_{n-1} - \mathrm{Y}_n)$$

For n+1, θ would be,

Equation 14

$$\theta_{n+1} = \theta_n + \mathrm{A}_{n+1}\ \mathrm{X}_{n+1}{}^T(\mathrm{X}_{n+1}\theta_n - \mathrm{Y}_{n+1})$$

For ease equation 13 can be written as,

$$\theta_{n+1} = \theta_n + W_n V_n$$

The RLS equations are:

$$\boldsymbol{V_n} = \mathbf{X}_{\boldsymbol{n+1}}\boldsymbol{\theta_n} - \mathbf{Y}_{\boldsymbol{n+1}}$$

$$\boldsymbol{W_n} = \mathbf{A}_{\boldsymbol{n+1}}\ \mathbf{X}_{\boldsymbol{n+1}}{}^{\boldsymbol{T}}$$

$$\boldsymbol{\theta_{n+1}} = \boldsymbol{\theta_n} + \boldsymbol{W_n V_n}$$

$$\mathbf{A}_{\boldsymbol{n+1}} = (\boldsymbol{A_n^{-1}} + \mathbf{X}_{\boldsymbol{n+1}}{}^{\boldsymbol{T}}\mathbf{X}_{\boldsymbol{n+1}})^{\boldsymbol{-1}}$$

To calculate the inverse of the matrix, we will use matrix inversion lemma, where

If B, D, CDE are nonsingular matrix then the inverse exists, as

$$[B + CDE]^{-1} = B^{-1} - B^{-1}C[D^{-1} + EB^{-1}C]^{-1}EB^{-1}$$

So, for

$$\mathrm{A}_{n+1} = (A_n^{-1} + \mathrm{X}_{n+1}{}^T\mathrm{X}_{n+1})^{-1}$$

Assume, B $=\mathrm{A}_n$, C = X_{n+1} , D = 1, E = $\mathrm{X}_{n+1}{}^T$

$$\mathrm{A}_{n+1} = \mathrm{A}_n - \mathrm{A}_n\ \mathrm{X}_{n+1}\left[1 + \mathrm{X}_{n+1}{}^T\mathrm{A}_n\ \mathrm{X}_{n+1}\right]^{-1}\mathrm{X}_{n+1}{}^T\mathrm{A}_n$$

$$\mathrm{A}_{n+1} = \mathrm{A}_n - \frac{\mathrm{A}_n\ \mathrm{X}_{n+1}\ \mathrm{X}_{n+1}{}^T\mathrm{A}_n}{1 + \mathrm{X}_{n+1}{}^T\mathrm{A}_n\ \mathrm{X}_{n+1}}$$

Hence concluded, the final RLS equations are

Equation 15

$$\boldsymbol{V_n} = \mathbf{X}_{\boldsymbol{n+1}}\boldsymbol{\theta_n} - \mathbf{Y}_{\boldsymbol{n+1}}$$

Equation 16

$$\boldsymbol{W_n} = \mathbf{A}_{\boldsymbol{n+1}}\ \mathbf{X}_{\boldsymbol{n+1}}{}^{\boldsymbol{T}}$$

Equation 17

$$\boldsymbol{\theta_{n+1}} = \boldsymbol{\theta_n} + \boldsymbol{W_n V_n}$$

Equation 18

$$\mathbf{A}_{n+1} = \mathbf{A}_n - \frac{\mathbf{A}_n \ \mathbf{X}_{n+1} \ \mathbf{X}_{n+1}{}^{T} \mathbf{A}_n}{\mathbf{1} + \mathbf{X}_{n+1}{}^{T} \mathbf{A}_n \ \mathbf{X}_{n+1}}$$

3.9 Recursive Least Squares with Forgetting Factor

When the parameters change continually but slowly, the forgetting factor is used, in which previous data points are gradually eliminated in favor of newer facts. Forgetting may be thought of as assigning lesser weight to historic data and higher weight to newer data, in the least square technique. The performance and the convergence of the Recursive Least Squares (RLS) are determined by the forgetting factor. [26]

The following are two key points to remember about RLS:

1. In addition to determining how much weight is given to a previous data point, a parameter that is defined by the user, called Lambda (λ), is used in which the previous input data points have much less anticipated weight than new ones. The smaller the value of λ, the less significant the previous input data point will be.

2. The Matrix Inversion Lemma (MIL) could be used instead of performing matrix inversion at every iteration phase, which can be computationally expensive. [25]

To lessen the effect of previous data we would bias the squared error (i.e. minimize the objective function).

$$F(\theta) = \sum_{i=1}^{n} \lambda^{n-i} \quad \varepsilon^2$$

Similarly, we have minimized the A_n matrix; the aim is to give more importance to the recent values.

$$A_n^{-1} = \sum_{i=1}^{n} \lambda^{n-i} \quad \mathrm{X}_i^{T} \ \mathrm{X}_i$$

Where, λ is ranges from 0 to 1. If the λ is less than 0, it means less significance is given to the older data values and if it is 1, all the data values have equal significance. For recursive Least squares, we have used weighted the Matrix Inversion Lemma by adding λ.

$$A_n^{-1} = \sum_{i=1}^{n-1} \lambda^{n-i} \quad \mathrm{X}_i{}^T\mathrm{X}_i + \mathrm{X}_n{}^T\mathrm{X}_n$$

$$A_n^{-1} = \lambda A_{n-1}^{-1} + \mathrm{X}_n{}^T\mathrm{X}_n$$

$$\mathrm{A}_{n+1} = \frac{1}{\lambda}\left(\mathrm{A}_n - \frac{\mathrm{A}_n \ \mathrm{X}_{n+1} \ \mathrm{X}_{n+1}{}^T\mathrm{A}_n}{\lambda + \mathrm{X}_{n+1}{}^T\mathrm{A}_n \ \mathrm{X}_{n+1}}\right)$$

The final equations of RLS with forgetting factor are written below

Equation 19

$$\mathbf{A}_{n+1} = \frac{\mathbf{1}}{\boldsymbol{\lambda}}\left(\mathbf{A}_n - \frac{\mathbf{A}_n \ \mathbf{X}_{n+1} \ \mathbf{X}_{n+1}{}^{\boldsymbol{T}}\mathbf{A}_n}{\boldsymbol{\lambda} + \mathbf{X}_{n+1}{}^{\boldsymbol{T}}\mathbf{A}_n \ \mathbf{X}_{n+1}}\right)$$

Equation 20

$$\boldsymbol{V}_n = \mathbf{X}_{n+1}\boldsymbol{\theta}_n - \mathbf{Y}_{n+1}$$

Equation 21

$$\boldsymbol{W}_n = \mathbf{A}_{n+1} \ \mathbf{X}_{n+1}{}^{\boldsymbol{T}}$$

Equation 22

$$\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}_n - \boldsymbol{W}_n\boldsymbol{V}_n$$

These are the final equations of RLS with forgetting factor that will be used to create the model to minutely predict the electric load for the domestic users.

Where,

A_n = inverse matrix of $\mathrm{X}_n{}^T\mathrm{X}_n$

X_{n+1} = matrix of temperature and previous load

$\mathrm{X}_{n+1}{}^T$ = transpose matrix of X_{n+1}

λ = any value between 0 and 1

Chapter 4

Implementation and Results

This chapter finally shows the implementation of the Recursive Least Squares (RLS) with Forgetting Factor. The implementation is compared with the Recursive Least Squares (RLS) and the Multiple Linear Regression (MLR). All the needed equations are listed in this chapter. As mentioned in chapter 3, the features or the parameters that we are considering in this study are:

Dependent Variable	**Independent Variable**
load	previous load and temperature

4.1 Recursive Least Squares (RLS) with Forgetting Factor Algorithm

4.1.1 Input and Output

The algorithm is ultimately implemented after extensive math proofs and study. It's written in pseudocode. The following is the input and output data that this algorithm requires:

INPUTS

1. X MATRIX { p_load , temp}

   where,

   p_load = previous load at time n-1

   temp = temperature at time n
2. Y MATRIX { y_load}

   where,

   y_load = load at time n
3. INVERSE MATRIX
4. θ MATRIX { $\boldsymbol{\theta_1}$, $\boldsymbol{\theta_2}$ }

The input contains the X and the Y matrices. X matrix contains all the values of the previous load at time n-1 and the values of temperature at time n. The Y matrix contains the values of electric load at time n.

$$X = \begin{bmatrix} y_{n-1} & t_n \\ \vdots & \vdots \\ y_0 & t_1 \end{bmatrix} \quad \text{where,} \quad \text{p\_load} = \begin{bmatrix} y_{n-1} \\ \cdot \\ \cdot \\ y_0 \end{bmatrix} \text{ and temp} = \begin{bmatrix} t_n \\ \cdot \\ \cdot \\ t_1 \end{bmatrix}$$

$$Y = \begin{bmatrix} y_1 \\ \cdot \\ \cdot \\ y_n \end{bmatrix}$$

OUTPUT

1. NEW θ MATRIX { θ_1 , θ_2 }
2. NEW INVERSE MATRIX
3. PREDICTED LOAD at time n

The output contains the θ matrix, inverse matrix and the predicted load values at time n. The θ matrix contains two values of θ { θ_1 , θ_2 } which is used for the prediction of electric load of the next day. The updated inverse matrix of each day is used as an argument of the recursive least square function for the next day. The load is predicted using the θ values, previous electric load at time n-1 and the temperature values at time n.

4.1.2 Global Variables

VAR

1. INVERSE_MATRIX {GLOBAL VARIABLE}
2. LEAST_SQUARES {GLOBAL VARIABLE}
3. INPUT {X MATRIX}{GLOBAL VARIABLE}
4. TEMP {{GLOBAL VARIABLE}}
5. OUTPUT {{GLOBAL VARIABLE}}
6. Y_PREDICTION {GLOBAL VARIABLE}

4.1.3 Two data samples for training Pseudo-code

This for loop ranges from 0 to 1. It loops over first two rows of the data and creates X matrix using temp and p_load and Y matrix using y_load. In order to calculate the two values of θ, two steps are done. First, inverse of the matrix is taken out after taking the transpose of the X matrix and dot product of X transpose matrix and X matrix. This inverse matrix is updated in every step. Secondly, dot product of X transpose matrix is calculated with the Y matrix and the result is multiplied with the inverse matrix. This calculation gives us the first two values of θ, which is taken out from the first two data samples.

```
FOR I IN 0 TO 1

        CREATE X MATRIX         INPUT = X_ MATRIX {p_load , temp}

        CRAETE Y MATRIX         OUTPUT = Y_ MATRIX {y_load}

IF (X MATRIX & Y MATRIX! = NULL)

        COMPUTE TRANSPOSE = X MATRIX TRANSPOSE

        COMPUTE PRODUCT = DOT_PRODUCT(X MATRIX TRANSPOSE, X MATRIX)

        COMPUTE INVERSE_MATRIX = INVERSE (PRODUCT)

        COMPUTE RES = DOT_PRODUCT (TRANSPOSE, OUTPUT)

        COMPUTE LEAST_SQUARES = DOT_PRODUCT (INVERSE_MATRIX, RES)

ELSE

        PRINT RESPONSE

                "X & Y MATRICES ARE EMPTY"

ENF IF

END FOR
```

4.1.4 The Pseudo-code of Recursive Least Squares (RLS) with Forgetting Factor Function

This is the pseudo-code of the Recursive Least Squares (RLS) with Forgetting Factor function.

```
FUNCTION RECURSIVE_LEAST_SQUARES (j, INVERSE_MATRIX, LEAST_SQUARES)

        P_LOAD = PREVIOUS LOAD at Tn-1

        CREATE X MATRIX        INPUT = X_ MATRIX {p_load, temp}

        COMPUTE TRANSPOSE = X MATRIX TRANSPOSE

        COMPUTE D = DOT_PRODUCT (INVERSE_MATRIX, INPUT)

        COMPUTE E = DOT_PRODUCT (TRANSPOSE, INVERSE_MATRIX)

        COMPUTE F = DOT_PRODUCT (D, E)

        COMPUTE G = DOT_PRODUCT (TRANSPOSE, INVERSE_MATRIX)

        COMPUTE H = DOT_PRODUCT (G, INPUT)

        ADD 0.1 AND H  (J = 0.1+H)

        DIVIDE F BYJ     (K = F/J)

        SUBTRACT INVERSE_MATRIX FROM K (L = INVERSE_MATRIX – K)

        DIVIDE 1 BY 0.1 (L_1 = 1/0.1)

        MULTIPLY L_1 AND L (P = L_1*L)

        ASSIGN INVERSE_MATRIX = P

        COMPUTE M = DOT_PRODUCT (INPUT, LEAST_SQUARES)

        SUBTRACT M FROM OUTPUT     (BRAC = M – OUTPUT)

        COMPUTE N = DOT_PRODUCT (INVERSE_MATRIX, TRANSPOSE)
```

```
            COMPUTE O = DOT_PRODUCT (N, BRAC)

            SUBTRACT LEAST_SQUARES FROM O (θ = LEAST_SQUARES – O)

            ASSIGN LEAST_SQUARES = θ

            RETURN LEAST_SQUARES, INVERSE_MATRIX, OUTPUT, TEMP

    END FUNCTION
```

The Recursive Least Squares (RLS) with Forgetting Factor function is called in the following pseudo code and the returned values are used for the prediction of electric load for the next day. Here, the for loop starts from 2 and ends at 1439 data sample which shows that it loops over the data sample of first day (24 hours) where we have per minute data sample.

```
FOR j IN 2 TO 1439

        FUNCTION RECURSIVE_LEAST_SQUARES (j, INVERSE_MATRIX, LEAST_SQUARES)

        ASSIGN R = RECURSIVE_LEAST_SQUARES (j, INVERSE_MATRIX, LEAST_SQUARES)

        LEAST_SQUARES = R[0]

        INVERSE_MATRIX = R[1]

        Y_PREDICTION = R[2]

END FOR
```

This for loop starts from 1440 and ends at 2879 data sample, which shows that it loops over the data sample of second day (24 hours). The purpose of this loop is to predict the electric load after training the model using the first day data sample.

Now, as we are predicting the electric load of second day, the value of previous θ (least_squares), inverse matrix and previous load (y_prediction) is used as arguments of the recursive least squares function. The recursive least square function with forgetting factor will return the value of the updated θ (least_squares),, inverse matrix, load and temperature, which helps in predicting the electric load.

```
FOR j IN 1440 TO 2879

        FUNCTION RECURSIVE_LEAST_SQUARES (j, INVERSE_MATRIX, LEAST_SQUARES)

        ASSIGN R = RECURSIVE_LEAST_SQUARES (j, INVERSE_MATRIX, LEAST_SQUARES)

        LEAST_SQUARES = R[0]

        INVERSE_MATRIX = R[1]

        Y_PREDICTION = R[2]

        TEMP_VALUE = R[3]

        MULTIPLY Y_PREDICTION WITH LEAST_SQUARES [0]

        ASSIGN INP1 = Y_PREDICTION * LEAST_SQUARES [0]

        MULTIPLY TEMP_VALUE WITH LEAST_SQUARES [1]

        ASSIGN INP2 = TEMP_VALUE * LEAST_SQUARES [1]

        ADD INP1 AND INP2

END FOR
```

4.2 Forecasting Metrics

In this study, we have used three forecasting metrics to evaluate the performance of the proposed algorithm. These four forecasting metrics are: [27]

1. Mean Absolute Percentage Error (MAPE)
2. Root Mean Square Error (RMSE)
3. R-squared (R^2)

4.2.1 Mean Absolute Percentage Error (MAPE)

To measures the accuracy of a forecasting system or a model, Mean Absolute Percentage Error (MAPE) is used. The lower the value of the MAPE, the higher is the accuracy of the forecasting system, and the better the model is. [28]

The formula of MAPE **%** is:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right| \text{ x } 100$$

Where,

n	number of sample points
$\sum$	summation notation
A_t	actual value
F_t	forecasted value

4.2.2 Root Mean Square Error (RMSE)

When the predicted errors (residuals) deviate from the regression line fitted data points, it is called as Root Mean Square Error (RMSE). It measures that how far the residuals are from the actual regression line. [28]

The formula of RMSE is:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n} (y_t - \hat{y}_t)^2}{N}}$$

Where,

N	number of sample points
$\sum$	Adds up (summation)
y_t	y_t is the t-th data sample

$\hat{y}_t$	$\hat{y}_t$ is the corresponding predicted data value

4.2.3 R-squared (R^2)

It is one of the accuracy determining metric that tells how efficiently the regression line the actual data. Its values range from 0 to 1. [28]

The approaches formula of $\boldsymbol{R^2}$ is:

$$\boldsymbol{R^2} = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum(y_t - \hat{y}_t)^2}{\sum(y_t - y')^2}$$

SS_{RES}	sum of squared residual error
SS_{TOT}	Total sum of squared error

4.3 Determination of Forgetting Factor

In this research, we have used forgetting factor with the Recursive Least Square (RLS). Why we have used Forgetting Factor in this research? In online forecasting, 'Forgetting Factor' helps to converge more towards the most recent data points and gives less value to the previous data points. In this way, the previous data points are used in less quantity and the most recent data points are used in higher quantity and we get an optimal result. [29]

How we have determined the best forgetting factor value for our study? As mentioned in chapter 3, forgetting factor is a value on which we get the most optimal solution. The value should be between 0 to 1, so in order to get that value we have done mathematical calculations and compared it with the multiple forecasting metrics. Hence, we got the optimal results from this value.

4.4 Comparison of Forgetting Factor with Forecasting Metrics

To get a value of forgetting factor on which we get the optimum results, we have compared multiple values with the forecasting metrics that are MAPE, RMSE and $\boldsymbol{R^2}$. Formula mentioned in chapter 3 of forgetting factor is:

$$\mathbf{A}_{n+1} = \frac{1}{\lambda}\left(\mathbf{A}_n - \frac{\mathbf{A}_n \mathbf{X}_{n+1} {\mathbf{X}_{n+1}}^T \mathbf{A}_n}{\lambda + {\mathbf{X}_{n+1}}^T \mathbf{A}_n \mathbf{X}_{n+1}}\right)$$

Here we have to find out only one value of λ, by using multiple values ranging from 0 to 1 and get the best one by comparing these values with the forecasting metrics results. After taking the average of all the forecasting metrics, we got the value of λ.

Following are the graphical representation of the forgetting factor ranging from 0 to 1 and the forecasting metrics. These graphs show the lowest value of forgetting factor against each forecasting metrics MAPE, RMSE and $\boldsymbol{R^2}$.

4.4.1 Graphical Representation of Forgetting Factor and MAPE

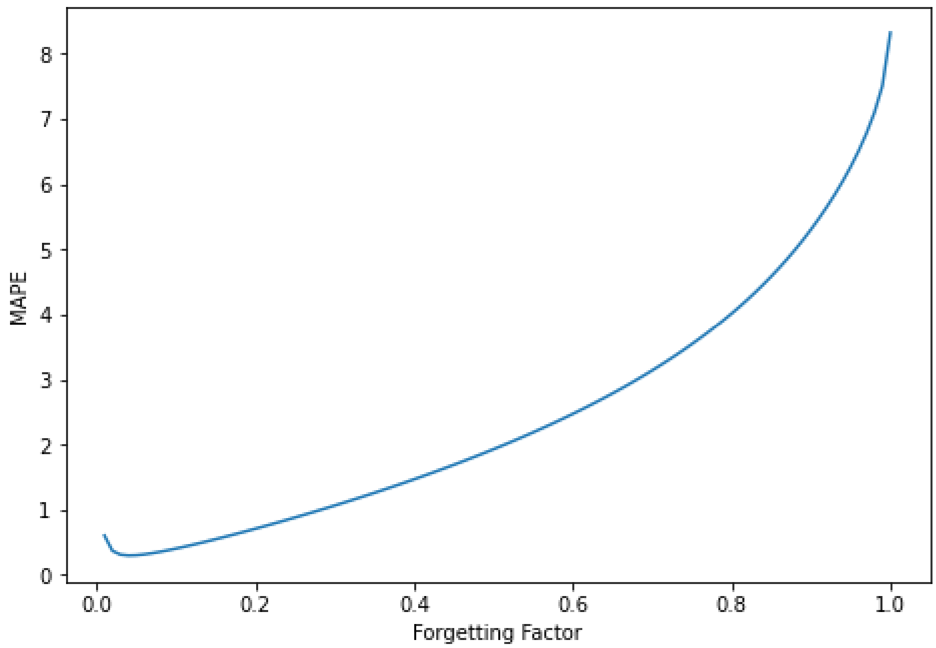

Figure 8: Graphical Representation of Forgetting Factor and MAPE

4.4.2 Graphical Representation of Forgetting Factor and RMSE

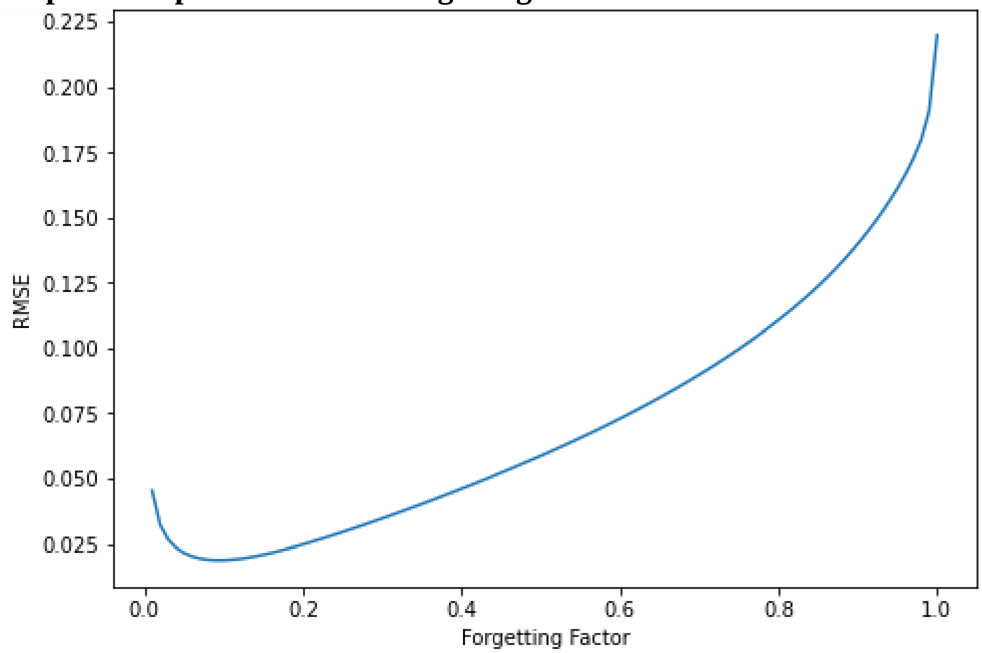

Figure 9: Graphical Representation of Forgetting Factor and RMSE

4.4.3 Graphical Representation of Forgetting Factor and R^2

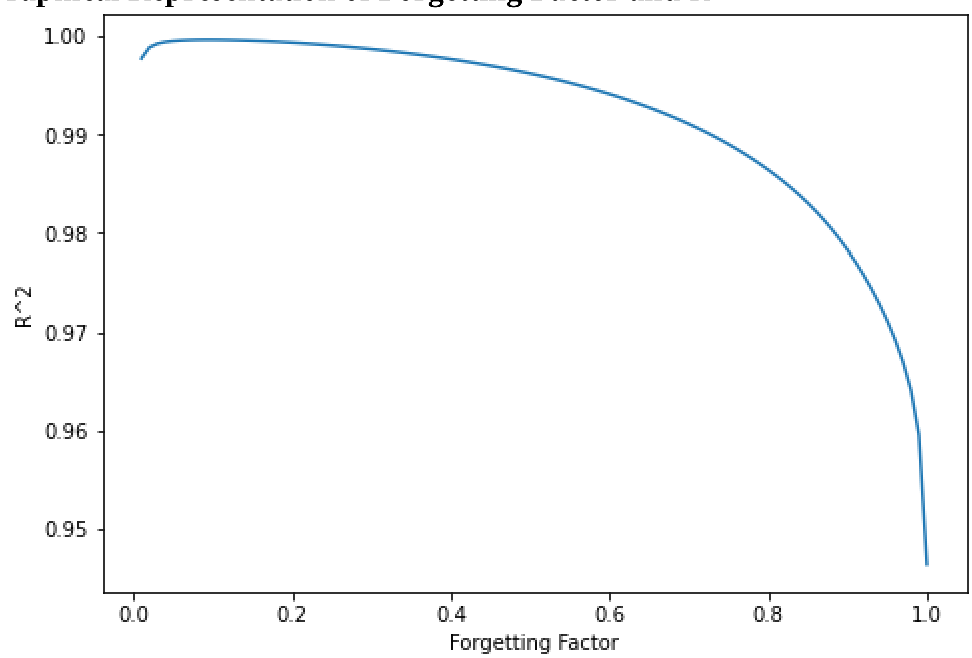

Figure 10: Graphical Representation of Forgetting Factor and $\boldsymbol{R^2}$

The average of the forecasting metrics' values gives the value of the λ. In our proposed algorithm the value of λ is 0.1.

4.5 Results: Forecasting Metrics of 10 Houses for Summer & Winter Season

This research is focused on forecasting the electric load for domestic users so that utility companies would know the usage of power in the domestic areas. We have covered the two extreme cases of the year i.e. the summer season and the winter season. From the PRECON dataset we have used the data points of June 2018 and December 2018 to test and evaluate the performance of our proposed and designed algorithm and the results we got are quite optimal and accurate.

The reason we have tested on June and December data point is that these months experience the extreme hot and cold temperatures. So for the forecasting purposes we have decided to use the month of June and the month of December to forecast the next day prediction for the domestic users.

For summer and winter season, we have tested on 10 different houses with multiple forecasting metrics. For example, house 1 is evaluated for summer season (June) and for the winter season (December). In total, we have 20 tables, 10 for summer and 10 for winter season of each house. The average of all the forecasting metrics have also been calculated to get how accurate the given model is. Fortunately, it gives accurate and precise values for all the forecasting metrics MAPE, RMSE and R^2 respectively.

The following tables show that the values of the forecasting metrics on each day of a specific house over a period of one month. The proposed recursive algorithm with forgetting factor works well for the forecasting of electric load for the domestic users.

Table 2: Forecasting Metrics of House 1 (1st June 2018 – 30th June 2018)

Days	**MAPE %**	**RMSE**	R^2
01-06-2018	0.370661	0.01127	0.999625
02-06-2018	0.501079	0.03577	0.995928
03-06-2018	0.470138	0.029893	0.996808
04-06-2018	0.440386	0.026348	0.997157
05-06-2018	0.433387	0.024288	0.997453
06-06-2018	0.424445	0.02266	0.997724
07-06-2018	0.403187	0.021206	0.997946
08-06-2018	0.86918	0.314174	0.542475
09-06-2018	0.805086	0.296226	0.58206
10-06-2018	0.750457	0.281035	0.638978
11-06-2018	0.724795	0.267998	0.768561
12-06-2018	0.697423	0.256617	0.790027
13-06-2018	0.681773	0.246592	0.801258
14-06-2018	0.654595	0.237637	0.810406
15-06-2018	0.632357	0.229595	0.818441
16-06-2018	0.606338	0.222315	0.831324
17-06-2018	0.59819	0.217479	0.835521
18-06-2018	0.588246	0.212944	0.842081
19-06-2018	0.577363	0.208683	0.848929
20-06-2018	0.577847	0.204681	0.852049
21-06-2018	0.569984	0.200891	0.857566
22-06-2018	0.561419	0.197301	0.86113
23-06-2018	0.556839	0.193904	0.86369
24-06-2018	0.545652	0.190668	0.865981
25-06-2018	0.534322	0.187587	0.867563
26-06-2018	0.524328	0.184651	0.869422
27-06-2018	0.514847	0.183985	0.855001
28-06-2018	0.518298	0.181542	0.858232
29-06-2018	0.521553	0.179208	0.851267
30-06-2018	0.524628	0.176974	0.844122

Table 3: Forecasting Metrics of House 1 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-12-2018	0.121592	0.008592	0.995231
02-12-2018	0.12202	0.007533	0.997368
03-12-2018	0.166155	0.00888	0.99654
04-12-2018	0.156821	0.008831	0.996028
05-12-2018	0.147518	0.008458	0.996293
06-12-2018	0.148928	0.00844	0.995898
07-12-2018	0.152386	0.00847	0.995721
08-12-2018	0.154795	0.008717	0.995559
09-12-2018	0.151175	0.00855	0.995546
10-12-2018	0.152666	0.00863	0.995389
11-12-2018	0.151885	0.008652	0.995218
12-12-2018	0.155668	0.008902	0.995443
13-12-2018	0.155858	0.008992	0.995248
14-12-2018	0.156602	0.008918	0.995252
15-12-2018	0.15555	0.008938	0.995209
16-12-2018	0.159601	0.009169	0.995057
17-12-2018	0.160343	0.009146	0.995038
18-12-2018	0.157511	0.009069	0.995104
19-12-2018	0.16019	0.011363	0.992548
20-12-2018	0.16166	0.011402	0.993444
21-12-2018	0.159915	0.011282	0.993463
22-12-2018	0.1571	0.011119	0.993558
23-12-2018	0.163616	0.011268	0.993458
24-12-2018	0.159574	0.011092	0.993515
25-12-2018	0.156721	0.010935	0.993578
26-12-2018	0.15544	0.010865	0.99364
27-12-2018	0.159218	0.010959	0.993543
28-12-2018	0.156972	0.010854	0.993574
29-12-2018	0.156716	0.010804	0.993589
30-12-2018	0.158617	0.010883	0.993502
31-12-2018	0.161441	0.010969	0.993463
01-12-2018	0.121592	0.008592	0.995231

Table 4: Forecasting Metrics of House 2 (1st Jun 2018 – 30th Jun 2018)

Days	MAPE %	RMSE	R^2
01-06-2018	0.8661	0.026558	0.999751
02-06-2018	0.821313	0.026213	0.99986
03-06-2018	0.881565	0.04224	0.999604
04-06-2018	1.402054	0.195062	0.992255
05-06-2018	1.260824	0.174813	0.993221
06-06-2018	1.175757	0.159842	0.994179
07-06-2018	1.125624	0.148374	0.99492
08-06-2018	1.131411	0.139752	0.995373
09-06-2018	1.104736	0.132188	0.995793
10-06-2018	1.056988	0.125682	0.996093
11-06-2018	1.037566	0.121542	0.996319
12-06-2018	1.033178	0.116907	0.996691
13-06-2018	1.01639	0.112504	0.996911
14-06-2018	0.995019	0.105104	0.997403
15-06-2018	0.985606	0.101951	0.997509
16-06-2018	0.969989	0.099852	0.997564
17-06-2018	0.96168	0.097839	0.997605
18-06-2018	0.945449	0.095924	0.997695
19-06-2018	0.932694	0.094167	0.997737
20-06-2018	0.92252	0.092514	0.997805
21-06-2018	0.91201	0.090983	0.997873
22-06-2018	0.903553	0.08954	0.997907
23-06-2018	0.900153	0.088103	0.997928
24-06-2018	0.891329	0.08673	0.998005
25-06-2018	0.880697	0.08558	0.998038
26-06-2018	0.893557	0.084396	0.998065
27-06-2018	0.886397	0.083178	0.998109
28-06-2018	0.874252	0.082249	0.998127
29-06-2018	0.869264	0.081181	0.998155
30-06-2018	0.8659	0.08100	0.998134

Table 5: Forecasting Metrics of House 2 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-12-2018	1.256768	0.032901	0.998572
02-12-2018	1.297732	0.03266	0.998576
03-12-2018	1.291113	0.032655	0.998508
04-12-2018	1.285644	0.032638	0.998485
05-12-2018	1.296104	0.032899	0.998551
06-12-2018	1.271655	0.032228	0.998624
07-12-2018	1.242758	0.031489	0.998642
08-12-2018	1.245802	0.031606	0.998617
09-12-2018	1.246495	0.0328	0.998502
10-12-2018	1.311057	0.056672	0.995531
11-12-2018	1.302497	0.054856	0.995819
12-12-2018	1.297189	0.053312	0.99601
13-12-2018	1.298501	0.052031	0.996218
14-12-2018	1.300095	0.050906	0.99638
15-12-2018	1.318942	0.050199	0.996504
16-12-2018	1.325784	0.049429	0.996618
17-12-2018	1.300174	0.04857	0.996908
18-12-2018	1.290113	0.047835	0.996981
19-12-2018	1.294207	0.04746	0.997045
20-12-2018	1.306631	0.047165	0.997147
21-12-2018	1.303947	0.046706	0.997187
22-12-2018	1.286597	0.046047	0.997236
23-12-2018	1.289366	0.045735	0.997291
24-12-2018	1.290143	0.045329	0.997336
25-12-2018	1.288744	0.044932	0.997635
26-12-2018	1.300443	0.044711	0.997797
27-12-2018	1.312584	0.044536	0.99782
28-12-2018	1.312283	0.044338	0.998015
29-12-2018	1.317047	0.044085	0.998095
30-12-2018	1.334559	0.044089	0.99812
31-12-2018	1.334000	0.044087	0.94452

Table 6: Forecasting Metrics of House 3 (1st Jun 2018 – 30th Jun 2018)

Days	MAPE %	RMSE	R^2
01-06-2018	0.635021	0.018802	0.999608
02-06-2018	0.641059	0.017489	0.99969
03-06-2018	0.621045	0.01687	0.999765
04-06-2018	0.619059	0.016865	0.99979
05-06-2018	0.619651	0.017282	0.999762
06-06-2018	0.640467	0.017632	0.999752
07-06-2018	0.644362	0.017588	0.999759
08-06-2018	0.648865	0.017659	0.99975
09-06-2018	0.670656	0.018469	0.999714
10-06-2018	0.6778	0.018888	0.999698
11-06-2018	0.680508	0.018743	0.999695
12-06-2018	0.685121	0.018624	0.999697
13-06-2018	0.680794	0.018411	0.999704
14-06-2018	0.684275	0.018553	0.999701
15-06-2018	0.694011	0.02075	0.999629
16-06-2018	0.699969	0.020901	0.999616
17-06-2018	0.709509	0.021127	0.999604
18-06-2018	0.720894	0.02144	0.999586
19-06-2018	0.716088	0.021296	0.999587
20-06-2018	0.710073	0.02116	0.999588
21-06-2018	0.708373	0.021114	0.999584
22-06-2018	0.699399	0.02093	0.999585
23-06-2018	0.69071	0.020765	0.999589
24-06-2018	0.689752	0.020757	0.999586
25-06-2018	0.694927	0.020761	0.999589
26-06-2018	0.686881	0.020523	0.999599
27-06-2018	0.687033	0.02048	0.999595
28-06-2018	0.688693	0.020418	0.999603
29-06-2018	0.689232	0.02083	0.999585
30-06-2018	0.680572	0.020657	0.999586

Table 7: Forecasting Metrics of House 3 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-11-2018	1.665051	0.094401	0.995857
02-11-2018	1.781142	0.087402	0.996608
03-11-2018	1.783018	0.08673	0.996563
04-11-2018	1.916817	0.090362	0.99639
05-11-2018	2.18155	0.097548	0.996411
06-11-2018	2.261715	0.099338	0.996469
07-11-2018	2.417645	0.102676	0.996429
08-11-2018	2.469462	0.105457	0.996226
09-11-2018	2.518246	0.105634	0.996324
10-11-2018	2.556063	0.105648	0.996333
11-11-2018	2.553164	0.105261	0.996368
12-11-2018	2.692082	0.108371	0.996325
13-11-2018	2.686754	0.10864	0.996326
14-11-2018	2.71763	0.109645	0.996306
15-11-2018	2.703078	0.109951	0.996282
16-11-2018	2.707989	0.110234	0.996303
17-11-2018	2.730552	0.110996	0.996338
18-11-2018	2.697193	0.11027	0.996343
19-11-2018	2.703078	0.105648	0.996355
20-11-2018	2.707989	0.105261	0.996365
21-11-2018	2.730552	0.108371	0.996324
22-11-2018	2.703078	0.10864	0.996212
23-11-2018	6.38E+20	0.109645	0.996365
24-11-2018	2.261715	0.109951	0.996387
25-11-2018	2.417645	0.110234	0.996387
26-11-2018	2.469462	0.110996	0.996382
27-11-2018	2.518246	0.11027	0.996356
28-11-2018	2.261715	0.105648	0.996321
29-11-2018	2.417645	0.105261	0.996254
30-11-2018	2.469462	0.108371	0.996375
31-11-2018	2.468684	0.10864	0.995865

Table 8: Forecasting Metrics of House 4 (1st Jun 2018 – 30th Jun 2018)

Days	MAPE %	RMSE	R^2
01-06-2018	1.01788	0.031867	0.999363
02-06-2018	1.135966	0.036459	0.999168
03-06-2018	1.178025	0.043912	0.998894
04-06-2018	1.110791	0.042046	0.998921
05-06-2018	1.096716	0.040744	0.998967
06-06-2018	1.098317	0.040153	0.998998
07-06-2018	1.114763	0.03925	0.998986
08-06-2018	1.098941	0.038677	0.999041
09-06-2018	1.088814	0.0385	0.99905
10-06-2018	1.092057	0.03854	0.999054
11-06-2018	1.099204	0.038226	0.999095
12-06-2018	1.102101	0.038394	0.99909
13-06-2018	1.120275	0.037961	0.999094
14-06-2018	1.116078	0.03741	0.999106
15-06-2018	1.117135	0.037108	0.999105
16-06-2018	1.112295	0.036926	0.999098
17-06-2018	1.104882	0.036758	0.999095
18-06-2018	1.098366	0.0364	0.999108
19-06-2018	1.082077	0.036299	0.999125
20-06-2018	1.069494	0.036172	0.999124
21-06-2018	1.06738	0.036021	0.999131
22-06-2018	1.061157	0.03595	0.99913
23-06-2018	1.061724	0.03577	0.999123
24-06-2018	1.05233	0.035542	0.999135
25-06-2018	1.042565	0.035413	0.999137
26-06-2018	1.035451	0.035273	0.999151
27-06-2018	1.032536	0.035113	0.999162
28-06-2018	1.02876	0.035127	0.999159
29-06-2018	1.024831	0.034903	0.999163
30-06-2018	1.018636	0.034900	0.999347

Table 9: Forecasting Metrics of House 4 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-11-2018	1.064418	0.025243	0.999167
02-11-2018	1.210634	0.028827	0.998748
03-11-2018	1.042391	0.026694	0.999132
04-11-2018	1.073957	0.026088	0.999023
05-11-2018	1.070976	0.025639	0.998997
06-11-2018	1.09333	0.026276	0.998919
07-11-2018	1.090173	0.025713	0.998884
08-11-2018	1.068879	0.025054	0.998912
09-11-2018	1.10161	0.03128	0.998241
10-11-2018	1.109655	0.030787	0.998297
11-11-2018	1.100382	0.030169	0.998304
12-11-2018	1.099311	0.029808	0.998303
13-11-2018	1.113402	0.029972	0.998343
14-11-2018	1.129337	0.030698	0.998369
15-11-2018	1.136443	0.030406	0.998404
16-11-2018	1.138856	0.030138	0.998437
17-11-2018	1.160748	0.03028	0.998416
18-11-2018	1.147333	0.029907	0.998428
19-11-2018	1.148234	0.029678	0.998427
20-11-2018	1.146802	0.029439	0.998438
21-11-2018	1.143338	0.029275	0.998435
22-11-2018	1.127862	0.028906	0.998444
23-11-2018	1.127611	0.028821	0.998447
24-11-2018	1.121779	0.028613	0.998447
25-11-2018	1.122441	0.028551	0.998498
26-11-2018	1.127253	0.02858	0.99857
27-11-2018	1.133308	0.028573	0.998599
28-11-2018	1.127719	0.028399	0.998637
29-11-2018	1.114303	0.028088	0.998649
30-11-2018	1.120347	0.028097	0.99865
31-11-2018	1.126534	0.028115	0.998651

Table 10: Forecasting Metrics of House 5 (1st Jun 2018 – 30th Jun 2018)

Days	MAPE %	RMSE	R^2
01-06-2018	2.819701	0.049396	0.999204
02-06-2018	2.950153	0.060086	0.998931
03-06-2018	2.683449	0.066023	0.998732
04-06-2018	2.712464	0.100341	0.997383
05-06-2018	2.703961	0.092819	0.997668
06-06-2018	2.717782	0.087554	0.99785
07-06-2018	2.761293	0.093609	0.997518
08-06-2018	2.671841	0.088829	0.997709
09-06-2018	2.608394	0.086376	0.997807
10-06-2018	2.625843	0.083755	0.997892
11-06-2018	2.655532	0.081994	0.997958
12-06-2018	2.616832	0.080907	0.997989
13-06-2018	2.562229	0.078578	0.998121
14-06-2018	2.572519	0.076959	0.998393
15-06-2018	2.56696	0.075354	0.99856
16-06-2018	2.581141	0.074041	0.998604
17-06-2018	2.651322	0.073856	0.998616
18-06-2018	2.691649	0.073365	0.998646
19-06-2018	2.704833	0.073042	0.998673
20-06-2018	2.680942	0.072118	0.998689
21-06-2018	2.693925	0.071473	0.99875
22-06-2018	2.730241	0.071278	0.998809
23-06-2018	2.717057	0.070635	0.99884
24-06-2018	2.729439	0.070219	0.998847
25-06-2018	2.747487	0.069996	0.998867
26-06-2018	2.746828	0.069542	0.998881
27-06-2018	2.697979	0.068658	0.998896
28-06-2018	2.705088	0.06828	0.998904
29-06-2018	2.710056	0.068038	0.998906
30-06-2018	2.678472	0.067411	0.998926

Table 11: Forecasting Metrics of House 5 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-11-2018	2.139456	0.059514	0.998422
02-11-2018	1.909341	0.04889	0.998472
03-11-2018	1.854236	0.147636	0.988006
04-11-2018	1.990984	0.133074	0.990433
05-11-2018	1.869665	0.123008	0.993759
06-11-2018	1.881749	0.114998	0.994089
07-11-2018	1.823949	0.108954	0.99473
08-11-2018	1.807569	0.103454	0.995101
09-11-2018	1.775065	0.102009	0.997097
10-11-2018	1.885683	0.098874	0.997674
11-11-2018	1.886388	0.095999	0.998033
12-11-2018	1.877301	0.094931	0.998414
13-11-2018	1.913672	0.093038	0.998659
14-11-2018	1.916686	0.091174	0.998881
15-11-2018	1.905846	0.091153	0.999051
16-11-2018	1.946667	0.090425	0.999103
17-11-2018	1.966164	1.573749	0.737983
18-11-2018	3.840634	1.54225	0.755273
19-11-2018	3.778753	1.512561	0.784623
20-11-2018	3.733696	1.484531	0.800553
21-11-2018	3.686295	1.457998	0.811225
22-11-2018	3.639991	1.432853	0.826736
23-11-2018	3.5993	1.408999	0.834126
24-11-2018	3.593391	1.386279	0.845082
25-11-2018	3.555679	1.364634	0.853331
26-11-2018	3.519726	1.343987	0.863269
27-11-2018	3.502686	1.324243	0.867902
28-11-2018	3.477834	1.305334	0.875653
29-11-2018	3.440927	1.287213	0.885054
30-11-2018	3.404964	1.270053	0.891025
31-11-2018	3.427285	1.270046	0.891023

Table 12: Forecasting Metrics of House 6 (1st Jun 2018 – 30th Jun 2018)

Days	**MAPE %**	**RMSE**	R^2
01-06-2018	2.819701	0.049396	0.999204
02-06-2018	2.950153	0.060086	0.998931
03-06-2018	2.683449	0.066023	0.998732
04-06-2018	2.712464	0.100341	0.997383
05-06-2018	2.703961	0.092819	0.997668
06-06-2018	2.717782	0.087554	0.99785
07-06-2018	2.761293	0.093609	0.997518
08-06-2018	2.671841	0.088829	0.997709
09-06-2018	2.608394	0.086376	0.997807
10-06-2018	2.625843	0.083755	0.997892
11-06-2018	2.655532	0.081994	0.997958
12-06-2018	2.616832	0.080907	0.997989
13-06-2018	2.562229	0.078578	0.998121
14-06-2018	2.572519	0.076959	0.998393
15-06-2018	2.56696	0.075354	0.99856
16-06-2018	2.581141	0.074041	0.998604
17-06-2018	2.651322	0.073856	0.998616
18-06-2018	2.691649	0.073365	0.998646
19-06-2018	2.704833	0.073042	0.998673
20-06-2018	2.680942	0.072118	0.998689
21-06-2018	2.693925	0.071473	0.99875
22-06-2018	2.730241	0.071278	0.998809
23-06-2018	2.717057	0.070635	0.99884
24-06-2018	2.729439	0.070219	0.998847
25-06-2018	2.747487	0.069996	0.998867
26-06-2018	2.746828	0.069542	0.998881
27-06-2018	2.697979	0.068658	0.998896
28-06-2018	2.705088	0.06828	0.998904
29-06-2018	2.710056	0.068038	0.998906
30-06-2018	2.678472	0.067411	0.998926

Table 13: Forecasting Metrics of House 6 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-11-2018	2.139456	0.059514	0.998422
02-11-2018	1.909341	0.04889	0.998472
03-11-2018	1.854236	0.147636	0.988006
04-11-2018	1.990984	0.133074	0.990433
05-11-2018	1.869665	0.123008	0.993759
06-11-2018	1.881749	0.114998	0.994089
07-11-2018	1.823949	0.108954	0.99473
08-11-2018	1.807569	0.103454	0.995101
09-11-2018	1.775065	0.102009	0.997097
10-11-2018	1.885683	0.098874	0.997674
11-11-2018	1.886388	0.095999	0.998033
12-11-2018	1.877301	0.094931	0.998414
13-11-2018	1.913672	0.093038	0.998659
14-11-2018	1.916686	0.091174	0.998881
15-11-2018	1.905846	0.091153	0.999051
16-11-2018	1.946667	0.090425	0.999103
17-11-2018	1.966164	1.573749	0.737983
18-11-2018	3.840634	1.54225	0.755273
19-11-2018	3.778753	1.512561	0.784623
20-11-2018	3.733696	1.484531	0.800553
21-11-2018	3.686295	1.457998	0.811225
22-11-2018	3.639991	1.432853	0.826736
23-11-2018	3.5993	1.408999	0.834126
24-11-2018	3.593391	1.386279	0.845082
25-11-2018	3.555679	1.364634	0.853331
26-11-2018	3.519726	1.343987	0.863269
27-11-2018	3.502686	1.324243	0.867902
28-11-2018	3.477834	1.305334	0.875653
29-11-2018	3.440927	1.287213	0.885054
30-11-2018	3.404964	1.270053	0.891025
31-11-2018	3.427285	1.270050	0.891056

Table 14: Forecasting Metrics of House 7 (1st Jun 2018 – 30th Jun 2018)

Days	MAPE %	RMSE	R^2
01-06-2018	1.945528	0.040017	0.999663
02-06-2018	1.711002	0.037744	0.999716
03-06-2018	1.702619	0.035821	0.999748
04-06-2018	1.68159	0.034166	0.999769
05-06-2018	1.905349	0.032723	0.999784
06-06-2018	1.984295	0.03145	0.999795
07-06-2018	1.814977	0.030321	0.999803
08-06-2018	1.600373	0.029303	0.99981
09-06-2018	1.432221	0.028384	0.999816
10-06-2018	1.297361	0.027774	0.999819
11-06-2018	1.186652	0.0272	0.999822
12-06-2018	1.094421	0.026669	0.999825
13-06-2018	1.016283	0.026299	0.999825
14-06-2018	0.949738	0.026398	0.999818
15-06-2018	0.891684	0.025932	0.999821
16-06-2018	0.841268	0.02549	0.999823
17-06-2018	0.808717	0.025076	0.999824
18-06-2018	0.77859	0.024677	0.999826
19-06-2018	0.752155	0.024875	0.999834
20-06-2018	0.740287	0.025884	0.999844
21-06-2018	0.757617	0.026961	0.99984
22-06-2018	0.733756	0.027535	0.999835
23-06-2018	0.711463	0.028096	0.999827
24-06-2018	0.690872	0.040017	0.999663
25-06-2018	0.671364	0.037744	0.999716
26-06-2018	0.682214	0.035821	0.999748
27-06-2018	0.718153	0.034166	0.999769
28-06-2018	0.774937	0.032723	0.999784
29-06-2018	0.811171	0.03145	0.999795
30-06-2018	0.848585	0.3001	0.999754

Table 15: Forecasting Metrics of House 7 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-11-2018	1.440318	0.056634	0.998202
02-11-2018	1.457369	0.073281	0.997806
03-11-2018	1.950442	0.071546	0.99769
04-11-2018	1.914744	0.073226	0.997656
05-11-2018	2.020867	0.073121	0.99761
06-11-2018	2.040558	0.071821	0.997526
07-11-2018	1.972061	0.07168	0.997383
08-11-2018	2.01073	0.084193	0.997419
09-11-2018	2.788796	0.09039	0.998084
10-11-2018	3.219772	0.102033	0.997905
11-11-2018	4.0689	0.112573	0.997848
12-11-2018	4.899046	0.118154	0.997898
13-11-2018	5.290434	0.120469	0.997853
14-11-2018	5.569293	0.126471	0.997811
15-11-2018	5.973723	0.134086	0.99775
16-11-2018	6.499148	0.137762	0.997706
17-11-2018	6.838462	0.141003	0.997667
18-11-2018	7.14727	0.147935	0.99758
19-11-2018	7.66455	0.148947	0.997664
20-11-2018	7.802851	0.150313	0.997627
21-11-2018	8.002317	0.155188	0.997639
22-11-2018	8.431107	0.156559	0.99765
23-11-2018	8.559418	0.159585	0.997684
24-11-2018	8.835987	0.164149	0.997657
25-11-2018	9.231738	0.168252	0.997794
26-11-2018	9.584585	0.16882	0.997942
27-11-2018	9.64135	0.172964	0.997932
28-11-2018	9.993927	0.177114	0.997985
29-11-2018	10.32145	0.178251	0.998132
30-11-2018	10.50631	0.181248	0.998225
31-11-2018	10.8135	0.181200	0.998200

Table 16: Forecasting Metrics of House 8 (1st Jun 2018 – 30th Jun 2018)

Days	MAPE %	RMSE	R^2
01-06-2018	1.115902	0.031142	0.999085
02-06-2018	1.869737	0.278878	0.951467
03-06-2018	1.667533	0.22876	0.977508
04-06-2018	1.594549	0.199989	0.988534
05-06-2018	1.618709	0.183941	0.989045
06-06-2018	1.624452	0.168805	0.99044
07-06-2018	1.534759	0.156675	0.991607
08-06-2018	1.438811	0.146815	0.992792
09-06-2018	1.382424	0.138674	0.993231
10-06-2018	1.451667	0.132312	0.993653
11-06-2018	1.431658	0.126586	0.994291
12-06-2018	1.399706	0.121687	0.994974
13-06-2018	1.387463	0.117393	0.995937
14-06-2018	1.390247	0.114168	0.996249
15-06-2018	1.358561	0.110501	0.996657
16-06-2018	1.330082	0.107195	0.996829
17-06-2018	1.341098	0.105068	0.996879
18-06-2018	1.351434	0.103133	0.99706
19-06-2018	1.332032	0.101194	0.997147
20-06-2018	1.330542	0.099446	0.997205
21-06-2018	1.320774	0.097734	0.997279
22-06-2018	1.322184	0.096178	0.997321
23-06-2018	1.317608	0.094667	0.99738
24-06-2018	1.312599	0.09323	0.997407
25-06-2018	1.306305	0.091892	0.997449
26-06-2018	1.29191	0.09055	0.997479
27-06-2018	1.294851	0.089363	0.997519
28-06-2018	1.284819	0.088169	0.997556
29-06-2018	1.271086	0.087021	0.997595
30-06-2018	1.269061	0.085936	0.997622

Table 17: Forecasting Metrics of House 8 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-11-2018	1.440318037	0.056634	0.998202
02-11-2018	1.457368711	0.073281	0.997806
03-11-2018	1.950442138	0.071546	0.99769
04-11-2018	1.91474385	0.073226	0.997656
05-11-2018	2.02867248	0.073121	0.99761
06-11-2018	2.80557653	0.071821	0.997526
07-11-2018	1.92061423	0.07168	0.997383
08-11-2018	1.20730306	0.084193	0.997419
09-11-2018	1.78879607	0.09039	0.998084
10-11-2018	1.21977171	0.102033	0.997905
11-11-2018	1.468900479	0.112573	0.997848
12-11-2018	1.499046499	0.118154	0.997898
13-11-2018	1.529043411	0.120469	0.997853
14-11-2018	1.569293144	0.126471	0.997811
15-11-2018	1.573722754	0.134086	0.99775
16-11-2018	1.699147935	0.137762	0.997706
17-11-2018	1.638462157	0.141003	0.997667
18-11-2018	1.747269882	0.147935	0.99758
19-11-2018	1.764549589	0.148947	0.997664
20-11-2018	1.702851128	0.150313	0.997627
21-11-2018	1.802317358	0.155188	0.997639
22-11-2018	1.831107252	0.156559	0.99765
23-11-2018	1.859417747	0.159585	0.997684
24-11-2018	1.835987042	0.164149	0.997657
25-11-2018	1.931737932	0.168252	0.997794
26-11-2018	1.98458497	0.16882	0.997942
27-11-2018	1.94134997	0.172964	0.997932
28-11-2018	1.99926603	0.177114	0.997985
29-11-2018	1.03214461	0.178251	0.998132
30-11-2018	1.05063075	0.181248	0.998225
31-11-2018	1.08134992	0.18145	0.998553

Table 18: Forecasting Metrics of House 9 (1st Jun 2018 – 30th Jun 2018)

Days	MAPE %	RMSE	R^2
01-06-2018	0.49569	0.01581	0.999286
02-06-2018	1.331412	0.035413	0.998298
03-06-2018	1.048255	0.03019	0.998936
04-06-2018	0.9442	0.028889	0.999028
05-06-2018	0.892744	0.02755	0.99906
06-06-2018	0.826285	0.026379	0.999132
07-06-2018	0.809106	0.025925	0.999261
08-06-2018	0.958742	0.030493	0.999301
09-06-2018	1.122636	0.079311	0.994952
10-06-2018	1.084703	0.075499	0.995111
11-06-2018	1.038976	0.072301	0.995324
12-06-2018	1.001474	0.06951	0.995582
13-06-2018	0.97291	0.067	0.996019
14-06-2018	0.930748	0.064793	0.996261
15-06-2018	0.877372	0.062611	0.996582
16-06-2018	0.841629	0.060739	0.996849
17-06-2018	0.81879	0.059508	0.997007
18-06-2018	0.805899	0.058382	0.997034
19-06-2018	0.812634	0.05743	0.997043
20-06-2018	0.805755	0.056515	0.997063
21-06-2018	0.79427	0.055632	0.997068
22-06-2018	0.784462	0.054799	0.997081
23-06-2018	0.778008	0.054003	0.997116
24-06-2018	0.769964	0.053179	0.997129
25-06-2018	0.762953	0.052457	0.997147
26-06-2018	0.816619	0.052505	0.997572
27-06-2018	0.883028	0.052733	0.997729
28-06-2018	0.952116	0.052772	0.997776
29-06-2018	1.022477	0.05294	0.997755
30-06-2018	1.033878	0.052526	0.997755

Table 19: Forecasting Metrics of House 9 (1st Dec 2018 – 31st Dec 2018)

Days	MAPE %	RMSE	R^2
01-11-2018	1.623142	0.044199	0.997316
02-11-2018	2.058659	0.09353	0.990967
03-11-2018	1.869454	0.084528	0.992652
04-11-2018	1.794602	0.078943	0.994139
05-11-2018	1.809885	0.074069	0.994408
06-11-2018	1.725831	0.070283	0.994871
07-11-2018	1.640949	0.067455	0.995386
08-11-2018	1.59868	0.066248	0.995945
09-11-2018	1.665826	0.064832	0.996346
10-11-2018	1.684979	0.07236	0.995699
11-11-2018	1.761319	0.071546	0.995936
12-11-2018	1.831534	0.070341	0.996344
13-11-2018	1.856399	0.069029	0.996454
14-11-2018	1.854383	0.070536	0.996594
15-11-2018	2.039811	0.073027	0.996814
16-11-2018	2.265382	0.304724	0.94805
17-11-2018	2.861634	0.299364	0.952563
18-11-2018	3.061346	0.294292	0.960341
19-11-2018	3.235586	0.289598	0.964694
20-11-2018	3.421947	0.285211	0.968308
21-11-2018	3.603261	0.281031	0.970358
22-11-2018	3.756675	0.276927	0.97187
23-11-2018	3.853552	0.273197	0.973575
24-11-2018	3.988844	0.269731	0.975876
25-11-2018	4.132904	0.266332	0.978388
26-11-2018	4.237547	0.263076	0.979952
27-11-2018	4.329424	0.260058	0.981295
28-11-2018	4.418446	0.256904	0.982551
29-11-2018	4.462672	0.254196	0.983951
30-11-2018	4.57583	0.251773	0.985762
31-11-2018	4.699574	0.241234	0.986721

Table 20: Forecasting Metrics of House 10 (1st Jun 2018 – 30th Jun 2018)

Days	MAPE %	RMSE	R^2
01-06-2018	1.373608755	0.037322652	0.999374266
02-06-2018	1.939720138	0.051971136	0.998693699
03-06-2018	2.005867938	0.052587573	0.998816775
04-06-2018	1.814353932	0.049681345	0.999191519
05-06-2018	1.676300848	0.046505754	0.999298145
06-06-2018	1.69995141	0.046101367	0.999317474
07-06-2018	1.814340384	0.049031182	0.999240251
08-06-2018	1.727738442	0.046966416	0.999273779
09-06-2018	1.70266356	0.046043961	0.999265831
10-06-2018	1.646302684	0.044728292	0.999324108
11-06-2018	1.774807051	0.047401912	0.99923269
12-06-2018	1.814353932	0.052587573	0.998693699
13-06-2018	1.676300848	0.049681345	0.998816775
14-06-2018	1.69995141	0.046505754	0.999191519
15-06-2018	1.814340384	0.046101367	0.999298145
16-06-2018	1.727738442	0.049031182	0.999317474
17-06-2018	1.814353932	0.046966416	0.999240251
18-06-2018	1.676300848	0.052587573	0.998693699
19-06-2018	1.69995141	0.049681345	0.999240251
20-06-2018	1.814340384	0.046505754	0.999273779
21-06-2018	1.69995141	0.046505754	0.999265831
22-06-2018	1.814340384	0.046101367	0.999324108
23-06-2018	1.727738442	0.049031182	0.99923269
24-06-2018	1.70266356	0.046966416	0.998693699
25-06-2018	1.646302684	0.046043961	0.998816775
26-06-2018	1.774807051	0.044728292	0.999191519
27-06-2018	1.814353932	0.047401912	0.999298145
28-06-2018	1.676300848	0.052587573	0.999240251
29-06-2018	1.69995141	0.049681345	0.999273779
30-06-2018	1.814340384	0.046505754	0.999265831

Table 21: Forecasting Metrics of House 10 (1^{st} Dec 2018 – 31^{st} Dec 2018)

Days	**MAPE %**	**RMSE**	R^2
01-11-2018	0.438795	0.016984	0.996125
02-11-2018	0.337918	0.01481	0.997088
03-11-2018	0.38272	0.014394	0.996936
04-11-2018	0.378799	0.015249	0.996864
05-11-2018	0.386499	0.015874	0.996601
06-11-2018	0.393438	0.015745	0.996499
07-11-2018	0.382397	0.015296	0.996608
08-11-2018	0.372697	0.01581	0.996566
09-11-2018	0.389594	0.015817	0.998343
10-11-2018	0.405	0.015857	0.998325
11-11-2018	0.406734	0.016505	0.998303
12-11-2018	0.418635	0.01653	0.998278
13-11-2018	0.418925	0.016639	0.9983
14-11-2018	0.426051	0.016298	0.998296
15-11-2018	0.417626	0.016224	0.998472
16-11-2018	0.41463	0.016437	0.998573
17-11-2018	0.422286	0.0166	0.99865
18-11-2018	0.42898	0.016603	0.998702
19-11-2018	0.428433	0.016657	0.998709
20-11-2018	0.427363	0.016459	0.998739
21-11-2018	0.423003	0.016596	0.998784
22-11-2018	0.427922	0.016638	0.998793
23-11-2018	0.430998	0.016836	0.998754
24-11-2018	0.441686	0.016775	0.998809
25-11-2018	0.441742	0.01655	0.998819
26-11-2018	0.436116	0.016657	0.998844
27-11-2018	0.438716	0.016622	0.998898
28-11-2018	0.438526	0.016572	0.998927
29-11-2018	0.440495	0.016639	0.998943
30-11-2018	0.441484	0.016773	0.998915
31-11-2018	0.445188	0.016819	0.998873

4.6 Average of Forecasting Metrics of Summer Season and Winter Season

The following table shows the average of forecasting metrics of the two months, June and December 2018 of 10 different houses.

Table 22: Average of Forecasting Metrics of Summer Season and Winter Season

Houses	Average of Forecasting Metrics of Summer Season – Jun 2018			Average of Forecasting Metrics of Winter Season – Dec 2018		
	MAPE %	RMSE	R^2	MAPE %	RMSE	R^2
House 1	1.5489	0.0454	0.8432	0.1567	0.0100	0.9942
House 2	0.9839	0.1049	0.9970	1.2882	1.2882	0.9974
House 3	0.6996	0.0269	0.9992	2.4333	0.1156	0.9952
House 4	1.0818	0.0372	0.9990	2.1344	1.0449	0.9986
House 5	2.7162	0.0762	0.9984	2.6629	0.7097	0.9190
House 6	1.2033	0.0498	0.9996	1.8296	1.1732	0.9835
House 7	1.0981	0.0314	0.9997	4.0158	0.1262	0.9977
House 8	1.4132	0.1181	0.9949	1.6285	0.0605	0.9963
House 9	0.8890	0.0516	0.9973	2.8296	0.1732	0.9836
House 10	1.7432	0.0466	0.9992	0.4156	0.0162	0.9981

4.7 Maximum and Minimum Average of June and December 2018

The following table shows the maximum and the minimum averages of forecasting metrics of the two months, June and December 2018.

Table 23: Maximum and Minimum Average of June and December 2018

Season	Maximum Average			Minimum Average		
	MAPE %	RMSE	R^2	MAPE %	RMSE	R^2
Summer Jun 2018	2.7162	0.1181	0.9997	0.6996	0.0269	0.8432
Winter Dec 2018	4.2864	1.2882	0.9986	0.1567	0.0100	0.9190

4.8 Comparison of the proposed algorithm and the other STLF techniques based on PRECON dataset

In this section, multiple techniques and their results from different papers are discussed that are applied on the PRECON dataset in order to compare with our proposed algorithm.

In this paper [30], a deep learning technique that is Deep Derived Feature Fusion (DeepDeFF) is applied on multiple datasets. One of the datasets is PRECON. They have calculated the average MAPE % of different houses and compared the average MAPE % with other models. DeepDeFF average MAPE ranges from 7.67% to 37.61% of multiple houses. Comparing this with our proposed algorithm's results, the average MAPE % in the month of June ranges from 0.6996% to 2.162% and in winter ranges from 0.1567% to 4.2864% which shows that our proposed model has outperformed and we are getting the optimal results.

In this research [31], a combinational load forecasting model in which they have introduced a new auto regression and time series model to analyze the PRECORN dataset. The minimum average MAPE % of summer is 1.70% and of winter is 1.80% whereas, in our scenario the minimum average MAPE % of summer is 0.6996% and of winter is 0.1567$. This shows that our proposed algorithm gives more accurate predicted load with less error.

In this paper [32], Multiple Linear Regression (MLR), Support Vector Machine (SVM) and Artificial Neural Network (ANN) are compared to know which technique is better than in terms of load forecasting keeping the PRECON dataset in mind and analyzing it. Multiple Linear Regression and Support Vector Machine performed better than Artificial Neural Network. The minimum RMSE calculated is 0.22 in MLR case, 0.13 in SVM case and 0.25 in ANN case. In our scenario, RMSE calculated from our proposed algorithm in summers is 0.0269 and in winters is 0.0100, which is less than MLR and SVM. The comparison with other benchmarks shows that our algorithm works well for load forecasting and gives satisfactory results with minimum error.

4.9 Difference between accuracy for winter and summer

Temperature is one of the explanatory variables that we have used in this research work. In Pakistan, the usage of electricity is dependent on the temperature. The hotter the temperature is the more power resources like fans, air conditioners are used for cooling purposes which leads to high power consumption [33]. In winters most of the people do not use electricity for heating purposes. The power consumption is less dependent on the temperature so the power consumption in winters is quite less in Pakistan.

In our case, the maximum average MAPE % of summer season is 2.7162% and the maximum average MAPE % of winter season is 4.2864%. As mentioned earlier the electricity consumption is more in summers because of the high temperature so the average MAPE % we got in summers is more accurate than the winters.

4.10 Comparison of offline MLR and our proposed technique

The main purpose of this research is to use online learning machine learning technique so that we cannot store the data into our memory. As there would be no storing of data in the memory m there would be less processing delays and less power would be consumed to create the model. Not only this, the other main purpose is to come up with the optimal and most accurate results comparing with the offline Multiple Linear Regression (MLR) method. [34]

As we have two explanatory variables (temperature, previous load) and one dependent variable (load) so we have compared our proposed technique with offline RLS method. We have compared 10 houses of the PRECON dataset of our technique with the same 10 houses of the offline MLR method and the results we got are quite accurate and optimal.

The following table shows the comparison of MAPE % of offline Multiple Linear Regression (MLR) and MAPE % of Recursive Least Square with Forgetting Factor for both summer and winter seasons.

Table 24: Comparison of offline MLR and our proposed technique

Houses	Summer Season – Jun 2018		Winter Season – Dec 2018	
	MAPE % of offline MLR	MAPE % of RLS with Forgetting Factor	MAPE % of offline MLR	MAPE % of RLS with Forgetting Factor
House 1	39.1421	1.5489	10.0694	0.1567
House 2	156.5407	0.9839	82.2604	1.2882
House 3	84.2619	0.6996	95.2353	2.4333
House 4	98.9793	1.0818	63.0832	2.1344
House 5	174.3737	2.7162	288.3082	2.6629
House 6	223.1684	1.2033	182.7649	1.8296
House 7	183.0959	1.0981	376.9420	4.0158
House 8	141.4100	1.4132	59.7185	1.6285
House 9	77.3899	0.8890	175.6233	2.8296
House 10	146.5343	1.7432	42.5793	0.4156

4.11 Comparison of RLS and RLS with Forgetting Factor

Similarly, we have also compared the Recursive Least Square (RLS) and Recursive Least Square with Forgetting Factor so check the accuracy level. The advantage of adding forgetting factor is that it gives less value to the previous data points and leads to more accurate results which we can see in the following table.

This table shows the comparison of MAPE % of offline Recursive Least Square (RLS) and MAPE % of Recursive Least Square with Forgetting Factor for both summer and winter seasons.

Table 25: Comparison of RLS and RLS with Forgetting Factor

Houses	Summer Season – Jun 2018		Winter Season – Dec 2018	
	RLS	MAPE % of RLS with Forgetting Factor	RLS	MAPE % of RLS with Forgetting Factor
House 1	41.4252	1.5489	9.6380	0.1567
House 2	168.6999	0.9839	71.2612	1.2882
House 3	89.9737	0.6996	88.5224	2.4333
House 4	102.1829	1.0818	62.9036	2.1344
House 5	168.5711	2.7162	155.8867	2.6629
House 6	242.0621	1.2033	80.5634	1.8296
House 7	178.3881	1.0981	179.0409	4.0158
House 8	142.7986	1.4132	54.1818	1.6285
House 9	74.8956	0.8890	108.6251	2.8296
House 10	151.4854	1.7432	29.5659	0.4156

Chapter 5

Conclusion, Limitations, and Future Work

5.1 Conclusion

Electric power utilization is increasing day by day because of the overuse of electric appliances. Pakistan is facing a lot of problems in generating the required amount of energy and supplying it to the people. In a competitive market, load forecasting is vital for power utility companies. Power generation and purchasing, load switching, transmission, distribution and infrastructure improvement are just a few of the uses. One of the most significant techniques for predicting the electric load is load forecasting which helps the utility companies to generate the required amount of demand energy and supply them to their customers. [35]

Short Term Load Forecasting technique (STLF) is a prominent player in the development of cost-effective, dependable, and secure power system operating techniques. The main goal is to offer load estimates or predictions for scheduling operations and analyzing the power system's security at any given time. Short-term electric load prediction will benefit the utility companies economically. They would know the demand of the power consumption and accordingly, they will generate the electricity. An excessive amount of electricity would not be generated. In this way, resources will be managed smartly and the cost of power consumption will also be reduced. Hence, circular debt will be controlled.

In this research, we have presented a Short Term Load Forecasting technique (STLF) technique to predict the electric load for the domestic users using the PRECON dataset. We have analyzed the data of 10 different houses from this dataset. The dataset contains minute per minute electric load values in each house file. We have used the electric load data and the temperature values to predict the electric load of the next day. The previous electric load data and the temperature give the predicted electric load values.

For this, we have implemented Recursive Least Square (RLS) with Forgetting Factor algorithm to create a model to predict the electric load for the domestic users. Forgetting factor helps in giving less weightage to the previous electric load values and more weightage to the recent electric load data. We have done the research on the months of June 2018 and December 2018 i.e. summer and winter season. Multiple performance metrics MAPE%, RMSE and R^2 are calculated to test the results and to evaluate the error. [36]

Recursive Least Square (RLS) with Forgetting Factor model has shown the optimal results i.e. for summer season the average MAPE % ranges from 0.6996 to 2.7162, the average RMSE ranges from 0.0269 to 1.9742, and the average R^2 ranges from 0.8432 to 0.9997. Whereas, for winter season the average MAPE % ranges from 0.1567 to 4.2864, the average RMSE ranges from 0.0100 to 3.0774 and season the average R^2 ranges from 0.9190 to 0.9986. The results and statistics show that the overall behavior of the Recursive Least Square (RLS) with Forgetting Factor model is the most efficient among the other algorithms while all the algorithms show relatively good behavior.

5.2 Limitations

The limitation of our proposed model is that we have used temperature and the previous load as the factors to forecast the electricity for the next day. There are other factors that has a great impact on the demand of power, for example, humidity, rainfall, wind speed, peak hours, holidays etc.

Other than the factors, age of the data is also one of the limitations in this research. The dataset contains the electrical load of the year 2018 and 2019. Now the demand of the electricity may have increased as the population is increasing day by day. Our proposed algorithm works for the year 2018 and 2019 and results may differ for the current year data points.

5.3 Future Work

The further scope of this research is to work on the PRECON dataset; we can use the online methods of multiple Machine Learning methods. For example, the online machine learning method of Support Vector Regression (SVM) can be implemented via this technique. Similarly, based on online machine learning techniques like decision trees, Logistic Regression, K-means, KNN algorithms etc. can also be implemented and tested to compare the performance of these online ML methods with the proposed model. This will help us to look the data more closely and to forecast the energy with the best online ML method which would consume the less resources, storage and power.

We did our research on the PRECON dataset consisting of domestic users' data but we can also use our proposed model on other datasets to check its performance and accuracy level in multiple areas. For example, we can use any industrial dataset and train it on our proposed model to test the accuracy level of the model for the industries as well as most of the energy is consumed by industries. This will help us in making the required amount of the demand energy for the industries. Multiple datasets regarding manufacturing, mining, construction and agriculture, of Pakistan are available on which we can do our analysis and create a model based on the different suitable online ML technique. This will help us to look over different datasets and the ML techniques to forecast the requirement amount of electricity in various organizations.

In our research, we have focused on two months, June and December. Basically we have forecasted the electricity for summer and winter seasons. To extend this research we can use this model on other months as well to see what would be the forecasted power in other seasons like in spring and autumn that are the months of March, April, October and November. For summer and winter season, January and July data points can also be analyzed and tested on this model to check the precision of the proposed model.

Here we have analyzed and tested on 10 houses from the PRECON dataset to forecast the electricity using one of the STLF methods i.e. Recursive Least Squares with forgetting factor. We can also examine other houses' data points as well on this proposed data points and on the other online Machine Learning algorithms to evaluate the performance of the multiple houses as more data we test on, the more accurate results we will get.

References

[1] Chen H., Canizares C.A., Singh A., Short-Term Load Forecasting (2001), 411-415.

[2] Ekonomou, L., C. A. Chritsodoulou, and Valeri Mladenov. "A short-term load forecasting method using artificial neural networks and wavelet analysis." Int. J. Power Syst 1 (2016): 64-68.

[3] Mohandes M., Introduction_to_Load_Forecasting, International Journal of Energy Research 26 (2002), 335-345.

[4] B. F. Hobbs, S. Jitprapaikulsarn, S. Konda, V. Chankong, K. A. Loparo, and D. J. Maratukulam, "Analysis of the value for unit commitment of improved load forecasts," IEEE Transactions on Power Systems, vol. 14, pp. 1342-1348, 1999

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

[6] Mohandes M., Support Vector Machines for Short-Term Electrical Load Forecasting, International Journal of Energy Research 26 (2002), 335-345.

[7] Chen B.J., Chang M.W., Lin C.J., Load Forecasting using Support Vector Machines: A Study on EUNITE Competition 2001, Technical report, Department of Computer Science and Information Engineering, National Taiwan University (2002).

[8] Farsi, Behnam, et al. "On short-term load forecasting using machine learning techniques and a novel parallel deep LSTM-CNN approach." IEEE Access 9 (2021): 31191-31212.

[9] Ting, Wang, and Li Xueyong. "Research on short-term electric load forecasting based on extreme learning machine." E3S Web of Conferences. Vol. 53. EDP Sciences, 2018.

[10] Wahab, Abdul, et al. "A novel technique for short-term load forecasting using sequential models and feature engineering." IEEE Access 9 (2021): 96221-96232.

[11] Kharal, A. Y., A. Mahmood, and K. Ullah. "Load forecasting of an educational institution using machine learning: The case of nust, islamabad." Pakistan Journal of Science 71.4 (2019): 252.

[12] Butt, Faisal Mehmood, et al. "Artificial Intelligence based accurately load forecasting system to forecast short and medium-term load demands." Mathematical Biosciences and Engineering 18.1 (2021): 400-425.

[13] Park, Dond C., Taehoon Park, and Seungeok Choi. "Short-term electric load forecasting using recurrent neural network." Proc. of ISAP. Vol. 97. 1997.

[14] Shafiei Chafi, Zahra, and Hossein Afrakhte. "Short-term load forecasting using neural network and particle swarm optimization (PSO) algorithm." Mathematical Problems in Engineering 2021 (2021).

[15] Raju, M. Pratapa, and A. Jaya Laxmi. "IOT based online load forecasting using machine learning algorithms." Procedia Computer Science 171 (2020): 551-560.

[16] Wang, Jing-Min, and Li-Ping Wang. "A new method for short-term electricity load forecasting." Transactions of the Institute of Measurement and Control 30.3-4 (2008): 331-344.

[17] Amral, N., C. S. Ozveren, and D. King. "Short term load forecasting using multiple linear regression." 2007 42nd International universities power engineering conference. IEEE, 2007.

[18] Kavitha, S., S. Varuna, and R. Ramya. "A comparative analysis on linear regression and support vector regression." 2016 online international conference on green engineering and technologies (IC-GET). IEEE, 2016.

[19] Yuan, Jie, et al. "Short-term electric load forecasting based on improved Extreme Learning Machine Mode." Energy Reports 7 (2021): 1563-1573.

[20] Yahya, Muhammad Amri, Sasongko Pramono Hadi, and Lesnanto Multa Putranto. "Short-term electric load forecasting using recurrent neural network (study case of load forecasting in central java and special region of Yogyakarta)." 2018 4th International Conference on Science and Technology (ICST). IEEE, 2018.

[21] Saber, Ahmed Yousuf, and AKM Rezaul Alam. "Short term load forecasting using multiple linear regression for big data." 2017 IEEE symposium series on computational intelligence (SSCI). IEEE, 2017.

[22] Nadeem, Ahmad, and Naveed Arshad. "PRECON: Pakistan residential electricity consumption dataset." Proceedings of the tenth ACM international conference on future energy systems. 2019.

[23] Zerta, Martin, et al. "Alternative World Energy Outlook (AWEO) and the role of hydrogen in a changing energy landscape." International journal of hydrogen energy 33.12 (2008): 3021-3025.

[24] Zor, Kasım, Oğuzhan Timur, and Ahmet Teke. "A state-of-the-art review of artificial intelligence techniques for short-term electric load forecasting." 2017 6th international youth conference on energy (IYCE). IEEE, 2017.

[25] Al-Hamadi, H. M., and S. A. Soliman. "Long-term/mid-term electric load forecasting based on short-term correlation and annual growth." Electric power systems research 74.3 (2005): 353-361.

[26] Zhang, Jinliang, et al. "Short term electricity load forecasting using a hybrid model." Energy 158 (2018): 774-781.

[27] Ling, Sai-Ho, et al. "Short-term electric load forecasting based on a neural fuzzy network." IEEE Transactions on industrial electronics 50.6 (2003): 1305-1316.

[28] Li, Jian, et al. "A novel hybrid short-term load forecasting method of smart grid using MLR and LSTM neural network." IEEE Transactions on Industrial Informatics 17.4 (2020): 2443-2452.

[29] Amral, N., C. S. Ozveren, and D. King. "Short term load forecasting using multiple linear regression." 2007 42nd International universities power engineering conference. IEEE, 2007.

[30] Bracale, Antonio, et al. "Short-term industrial reactive power forecasting." International Journal of Electrical Power & Energy Systems 107 (2019): 177-185.

[31] Kong, Xiangyu, et al. "A recursive least squares method with double-parameter for online estimation of electric meter errors." Energies 12.5 (2019): 805.

[32] Martinek, Radek, et al. "Least mean squares and recursive least squares algorithms for total harmonic distortion reduction using shunt active power filter control." Energies 12.8 (2019): 1545.

[33] Tran, Manh-Kien, and Michael Fowler. "Sensor fault detection and isolation for degrading lithium-ion batteries in electric vehicles using parameter estimation with recursive least squares." Batteries 6.1 (2019): 1.

[34] Kim, Yunsun, Heung-gu Son, and Sahm Kim. "Short term electric load forecasting for institutional buildings." Energy Reports 5 (2019): 1270-1280.

[35] Li, Yuancheng, Panpan Guo, and Xiang Li. "Short-term load forecasting based on the analysis of user electricity behavior." Algorithms 9.4 (2016): 80.

[36] Gajowniczek, Krzysztof, and Tomasz Ząbkowski. "Short term electricity forecasting using individual smart meter data." Procedia Computer Science 35 (2014): 589-597.

[37] Yang, Wangwang, et al. "A combined deep learning load forecasting model of single household resident user considering multi-time scale electricity consumption behavior." Applied Energy 307 (2022): 118197.

[38] El-Baz, Wessam, and Peter Tzscheutschler. "Short-term smart learning electrical load prediction algorithm for home energy management systems." Applied Energy 147 (2015): 10-19.

[39] Kuster, Corentin, Yacine Rezgui, and Monjur Mourshed. "Electrical load forecasting models: A critical systematic review." Sustainable cities and society 35 (2017): 257-270.

[40] Hayes, Barry, Jorn Gruber, and Milan Prodanovic. "Short-term load forecasting at the local level using smart meter data." 2015 IEEE Eindhoven PowerTech. IEEE, 2015.