TÉCNICO LISBOA

Forecasting Electricity Consumption Using Simulation Data from Physical Models

Abdul Rehman

Thesis to obtain the Master of Science Degree in

Energy Engineering and Management

Supervisor: Prof. Carlos Augusto Santos Silva

Examination Committee

Chairperson: Prof. Edgar Caetano Fernandes

Supervisor: Prof. Carlos Augusto Santos Silva

Member of the Committee: Dr. Henrique Ramalho Monteiro Latourrette Pombeiro

December 2019

ACKNOWLEDGEMENTS

I want to thank my excellent supervisor Prof. Dr. Carlos Augusto Santos Silva for his insight, pedagogy, patience and support during the development of this work at Instituto Superior Tecnico, Lisbon. It has been an absolute pleasure working with you.

I would like to thank Mr. Ricardo. A. Gomes for his help in simulation part and Ms. Olga Savchuk for her guidance in writing algorithm.

I would like to thank Omais, Nouman and Yousif for being awesome friends and for their moral support during my stay in Lisbon and throughout master's degree.

Finally, I would like to thank my family for always encouraging and cheering me up. Without them it would not have been possible.

RESUMO

O crescimento da população tem vindo a criar um aumento no consumo de eletricidade, o que aumenta a emissão de gases com efeito da estufa devido á produção e consumo. Atualmente, os edifícios são responsáveis por 30% das emissões globais de CO2. Devido a números alarmantes, surgem conceitos de sistemas de gestão energética nos edifícios (BEMS) e redes inteligentes. Os BEMS têm sido alvo de pesquisa desde há quatro décadas, usando métodos numéricos, mas os desenvolvimentos recentes no uso de machine learning (ML) para a previsão de carga têm mostrado um grande potencial na gestão energética. O presente trabalho propõe-se a combinar métodos numéricos (Energy Plus) e de ML na previsão dos serviços de energia num edifício universitário. É utilizado o RandomForest combinado com técnicas de seleção de características, devido à sua capacidade de lidar com dados complexos, comparada com outros algoritmos de ML. Desenvolveram-se quatro modelos de ML usando inputs de simulações de EP e dados meteorológicos de um ano, prevendo o serviço energético para o próximo ano, de hora a hora. Calcularam-se três tipos de erros (MAE, RMSE e CV-RMSE) que foram usados para comparar o desempenho modelo com padrões de previsão hora a hora aceites internacionalmente. O CV-RMSE, sendo adimensional, oferece uma boa comparação entre os modelos e o seu valor é inferior a 30%, excepto no modelo HVAC, em que é 37%. No geral, os modelos tiveram um desempenho bastante bom e, com eventuais melhorias, poderão ajudar em serviços de precisão energética com erro mínimo.

Palavras-chave: Machine learning, Ramdom Forest, serviços energéticos, Energy plus, previsão de carga

ABSTRACT

Ever growing population has given rise to the electricity demand which results in more Greenhouse gas emissions connected to its generation and consumption. Currently buildings accounts for the 30 % of CO_2 emissions globally. These alarming numbers have given rise to concepts of the building energy management systems (BEMS) and smart grids. BEMS has been the topic of research since last four decades using numerical techniques but the recent developments in the use of machine learning (ML) technologies for load forecasting has shown a great potential in energy management. This work is an effort to combine numerical (Energy Plus) and ML methods for energy services forecasting in a campus building. *RandomForest* predictor, in combination with feature selection techniques, is used owing to its ability to deal with complex data compared to other ML algorithms. Four ML models have been constructed taking the input from EP simulations and meteorological data for one year and predicting the energy service for next year in hourly fashion. Three type of errors (MAE, RMSE, and CV-RMSE) have been calculated and are used to compare the model performance against internationally accepted standards for hourly prediction. CV-RMSE being scale independent provides a good comparison between models and its value is less than 30% except HVAC model where it is 37%. Overall, the models performed significantly well and with further improvements can help in energy services forecasting with minimal error.

Keywords: Machine Learning, Random Forest, Energy Services, Energy Plus, Load forecasting

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

AI	Artificial Intelligence
ARIMA	Auto-Regressive Integrated Moving Average
ARMA	AutoRegressive-Moving Average
ANN	Artificial Neural Networks
CV-RMSE	Coefficient of Variation of Root Mean Square Error
DT	Decision Tree
EP	Energy Plus
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
ML	Machine learning
MTLF	Medium-term load forecasting
RF	Random forest
RMSE	Root Mean Square Error
STLF	Short-term load forecasting
SVM	Support Vector Machine
VSTLF	Very short-term load forecasting

CHAPTER 1

INTRODUCTION

With the increasing trend of the world population and the consequent increase of demand for energy services, the demand of energy in the world is increasing. Only in European Union, the buildings consume the 40% of total energy consumption [1]. On one hand, there is a massive research going on in the field of energy management that includes energy saving by shifting of least important tasks to off-peak hours and smart control of HVAC and lighting equipment. While on the other there a trend of passive housing and smart grids. The fundamental feature of smart grids is load forecasting which helps the operator to take effective and efficient decisions.

The buildings account for 30 % of global CO_2 emissions and 36 % of greenhouse gas (GHG) emissions only in European Union. These GHG emissions give rise to the atmospheric temperature and cause a devastating climate changing effect globally [2]. There is already an abundance of historical and meteorological data of buildings that needs to be utilized smartly to help shape the decarbonizing building strategies.

1.1 Motivation

Load forecasting helps in several operating decisions such as management, planning, scheduling and load dispatching. An accurate result of load forecasting is highly desirable as the procedure takes lot of time and cost. It has been claimed in literature that just 1 % increase in the prediction error can cause a loss of millions of dollars every year [3]. Load forecasting has been categorized in four different categories depending upon their interval of forecasting. Short-term load forecasting (STLF) has major role in controlling the electricity price and demand close to real time, help schedule the fueling and other such operations.

On the other hand, Long-term load forecasting (LTLF) helps balancing the demand and production in case of smart grids or planning the energy policies. LTLF is much more complex compared to the STLF as it is affected by seasonal variation and uncertain future event changing the demand heavily. Medium-term load forecasting (MTLF) is useful for maintenance scheduling, coordination of load dispatch and price settlement so that demand and generation is balanced.

There are numerous methods available for building demand forecast. All the available methods can be classified easily into three categories; Numerical, Analytical and predictive. Numerical methods include TRNSYS, Energy Plus, DOE-2, etc. These modelling techniques need a considerable amount of real data and computation time to build the physical models for simulation of future consumption. Still it is hard to use them for online or real time applications, as it also requires the study of human energy utilization behavior [3]. Analytical model rely on in-depth knowledge of processes and the law governing them but they are advantageous to the numerical methods in terms once calibrated can be used anywhere.

Contrary to them predictive methods like Artificial Neural Networks (ANN), Decision Trees (DT), etc. are highly accurate and quick [4]. Random forests (RF) can be said the extension of DT´s as DT is their binary element. RF outsmarts the most of AI prediction techniques owing to its appealing characteristics which include [4]; (i) it´s interaction between predictors (ii) its basis of ensemble techniques allows it to learn the complex models (iii) it requires less hyper parameter tuning compared to its competitors.

1.2 Objectives and methodological approach

The objective of this work is to combine the numerical and machine learning techniques to have better forecast results. Energy Plus software will be used for numerical part of work and RF machine learning technique for the predictive part.

The EP simulations will be run on an already available physical model for a building of Tecnico´s alameda campus. The default weather data of EP software will be replaced with the real weather data available for campus´s open source meteorological platform. Simulation will be run for two years 2017 and 2018 to get the energy services.

The RF model will be trained for the one year´s (2017) data and the energy services will be forecasted for the next whole year (2018) in hourly fashion. The EP simulated data for 2018 will be used to validate the model.

The prediction will also be compared with the real data of power consumption for the building but only total energy consumption and HVAC consumption is available for this purpose.

Internationally accepted criteria for errors in hourly forecast will be used to check the fitness of used model.

The use of real weather data in EP simulations is supposed to give the better results compared to default weather data. The RF model is thought to produce less error when feature selection techniques will be used for prediction.

1.3 Thesis outline

This document is subdivided into five chapters. The initial chapters provide the context to the work done and some key concepts for understanding the work, while the later chapters exclusively discuss the methodology, implementation of models, the results and conclusions.

The present chapter serves as introduction the work done, the motivation behind it and the objectives of the work.

The second chapter provides the overview of the related work, the technologies used, its application in different fields of science and life. Moreover, it also provides the elaborative theoretical background of the concepts necessary to understand the work. It includes the overview of ML techniques, importance of RF and Feature selection methods.

The third chapter is dedicated to the implementation methods. It explains how the work was done systematically, starting from data acquisition to the training and validation of the model. Four different models were constructed in Python to forecast the energy consumption for whole building, HVAC, the exterior light and the whole facility collectively.

The fourth chapter presents the results of the prediction by the application of models and the comparison of the prediction with simulated and real data. The errors metrics are calculated in this chapter.

In the fifth and final chapter, the conclusions are drawn about results and model performance. Some suggestions for the future work are also enclosed in this chapter.

CHAPTER 2

LITERATURE REVIEW

This chapter of the thesis is dedicated to discussing the state of the art, research papers related to this work and some theoretical concepts behind this work. It is divided in two sections, the related work is discussed in the present one while the next section encompasses the key concepts related to this work.

2.1 Time series load forecasting

Time series load forecasting can be categorized in four groups depending upon the time intervals of forecasting; [5]

- Very short-term load forecasting (VSTLF) has the time period ranging from a few minute to half hour or few hour. Its aim is to adjust and control the demand and price in real time.
- Short-term load forecasting (STLF) has time period ranging from one day to one weak ahead and aims at economic dispatch and optimal generator unit commitment.
- Medium term load forecasting (MTLF) includes the forecasting from one month to one year ahead. Its purpose is to maintain the balance between generation and consumption for maintenance scheduling.
- Long-term load forecasting (LTLF) has the forecasting horizon longer than one year. It is necessary for the future electricity network planning conditions.

Short and medium term forecasting is important for economical operation, schedule the fuelling and maintenance operation while long term forecasting is useful for planning operation and capacity expansion [6]. STLF is slightly affected by weather conditions and the social behaviour of the community and in some cases, such algorithms report less than 1% mean absolute percentage error (MAPE) [7]. LTLF involves the uncertainties introduced by seasonal issues and distant future, which makes LTLF a challenging task [8].

Conventional methods of forecasting include statistical methods, which exhibit a white box model where the internal structure of process is well known and can be interpreted by using mathematical formulas and equations. This mathematical explanation allows the understanding of the relation between input and outputs. Widely used statistical methods include multiple linear regression [9][10], autoregressive-moving average (ARMA) [11], auto-regressive integrated moving average (ARIMA) [12], Kalman filter [13], general exponential technique [14] and stochastic time series [15]. These white box statistical models are easily to implement and interpret but their shortcoming to handle non- linear and large data sets makes the machine learning methods need of the time.

There are several engineering methods and tools to forecast energy consumption in buildings but most of them are complex and require the physical models of the buildings or the space as well as the human behaviour and pattern of energy utilisation. DOE-2, Energy Plus, BLAST ESPr. are some of the tools used for energy efficiency and simulations for proper energy management.

In 1993-1994 American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) organized the first edition of energy prediction contest, the Great Energy Predictor Shootout (GEPS) with the purpose of predicting energy consumption of commercial buildings in hourly fashion. The successful participant developed a machine-learning algorithm using a model based on sensors, which relied on domain knowledge of used building [16].

Following the GEPS, the black box type machine learning models became the topic of research because of their ability to learn and implement the complex patterns with minimal human interaction. Most of the times, the internal mechanism of these black box methods is unknown and difficult to interpret. The significant works include the use of Artificial Neural Network (ANN) by Chae et al. [17] for a next day prediction of electricity consumption in a commercial building with a time period of 15 minutes and Fu et al. [18] using Support Vector Machine (SVM) to forecast the coming day's load forecast of a public building in Shanghai.

In a simplified definition, machine learning is a process of gathering all the available data, extracting the relevant information from it and developing a model which best explains the past and future datasets [19].

Decision tree (DT) is a widely used machine learning technique, which includes the classification and regression trees (CART) [20]. The decision Tree is a kind of an inverted tree where the top most node is called the root node which has all the training data in it. Each decision node applies a test to the input data and the outputs are more than one, giving the different result in each case. The final nodes where a branch ends is called the leaf of tree. There are several leaves in a DT, which store the results of each case. For example, to predict the electricity load of some place considering that it depends on temperature, type of day and season, an exemplary DT is constructed in Figure 2.1.

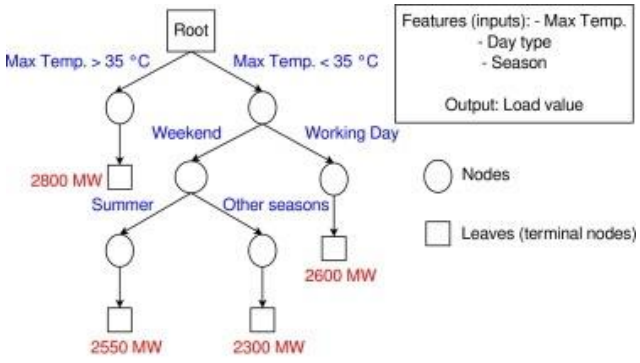

Figure 2.1 Sample regression tree predicting the load based on weather data [21]

Random forest technique was first developed by Breiman in 2001 for both classification and regression [22]. The main purpose of RF was to overcome the shortcomings of DT by using multiple DT´s to generate a forest called RF. Since then it has been used in various fields for forecasting. Olivier [19] used random forest regression and clustering for medical application. He developed a model for

multiple organ localization, segmentation, lesion detection or content-based modality recognition of medical images. Yuan [23] applied the random forest to predict the sales of a specific company based on the advertisement made by it on TV. Although the model did not predict the best results but it helped the writer understand the main features to boost the sales of company´s product.

Onisimo et al. [24] have utilized the random forest regressor to predict the wetland biomass using satellite provided imaging data as the input to model. He found out that RF regressor outperformed the multiple linear regression model by giving three percent less RMSE of prediction.

Paula et al. [25] have combined the existing physics based model with RF to predict the electricity generated by the ocean waves at Mutriki wave farm. The prediction was done for 24 hours ahead with the lead-time of 4 hours and it was concluded that RF based model outperforms the existing model for a forecast of 8-10 hours ahead and it can be applied to other wave energy production farms.

Benali et al. [26] compared the RF with smart persistence and Artificial Neural Network (ANN) for the prediction of solar radiation components (direct normal, diffused horizontal and global horizontal) for a site in France from one to six hours ahead. Their study showed that the RF performed the best out of three applied method and they suggested applying this study to other sites also for solar radiation prediction.

Rui et al. [27] have compared the machine learning methods (Recurrent Neural Network combined with RF) with the traditional atmospheric methods to analyze and forecast the air pollutants in Hangzhou city of China. RF helped to find the relation between atmospheric factors and air pollution by using feature importance technique. For the first time it was found that dew point temperature is more important than relative humidity in shaping the air pollutants, which helped in better policy making to prepare the city for upcoming Asian games.

Wind power generation is highly unpredictable due to unstable weather conditions anywhere for a wind farm. Lahoua et al. [28] have used the RF regressor model for one-hour ahead prediction of wind power generation. RF helped to narrow down the features to be emphasized on by providing a correlation between output and meteorological data. RF model performed well with just 14 percent of mean absolute percentage error (MAPE).

In terms of electrical load forecasting Ali et al. [29] predicted the one hour ahead load for whole Tunisia. The input was the half- hourly load demand, the model was trained with the data for whole year of 2009 and the prediction was made for next half year. The MAPE was less than 2 percent for initial few months but then it was drastic for a few days with immensely high demand. It was learnt that the training set of the model did not have that sudden change in demand which turned to the reason of unexpected errors in forecasting.

Grzegorz [30] used the RF technique for STLF. The input was hourly data from Polish power system for 2002 to 2004 and the target was to predict the consumption for next day. Furthermore, a comparison was made of RF with other models such as CART, ARIMA, exponential smoothing and neural networks. The finding was that RF performs equally accurate to ANN while gives results better than other models. The MAPE was found out to be less than 1.5 percent.

2.2 Machine Learning theoretical background

This section is about the understanding of machine learning techniques, the importance and process of feature selection and the working of random forest decision trees.

2.2.1 Machine learning

Machine learning is commonly divided into three main categories according to their purpose [31].

1. Supervised learning
2. Unsupervised learning
3. Reinforcement learning

2.2.1.1 Supervised learning

It is simplest to understand and easiest to implement type of machine learning. The algorithm is to provide the model with a data set containing a target variable (outcome) and several other variables called features. The model is trained with this input data which means it learns the relationship between the features and target and memorizes it. We train the model repeatedly until the required accuracy is achieved. The next step is testing where we provide the model new set of features which are unseen for it and it predicts the target with these new features using the logic or relation learnt in the learning phase. Supervised learning is further classified into two branches. One deals with regression problems where the prediction is done for continuous values like predicting the temperature of a city or price of the property in a country. While the second is classification where the target is to predict the categorical response. For example, predicting either a day will be rainy or not, an email is spam or not, etc.

2.2.1.2 Unsupervised learning:

There is no target variable in this type of learning and this is the main difference it has with supervised learning. This is mostly used in pattern detection and descriptive modelling. The main task is clustering data into different groups. For example, model is provided with the news articles and it clusters them with respect to the type of news. This type of machine learning is widely used as most of the real-world problems fall into this category.

2.2.1.3 Reinforcement learning:

This can be called as a branch of artificial intelligence. This is the type of learning where the model learns from mistakes to make the decisions. The machine is exposed to an environment to train itself iteratively, it makes decisions, interacts with the environment and changes it until a favorable decision is acquired. All in all, the machine learns to provide the best possible response in every case.

2.2.2 Feature Selection

Feature selection is a process of selecting a subset features and using them in model construction. Feature selection is the key in machine learning algorithms which immensely affects the performance

of the model. Some times in the data, there are numerous features and using all of them can lead to a dead end [32]. The features helps us to;

- Reduce overfitting to avoid the use of noisy data for prediction
- Improve accuracy by removing the misleading data
- Minimize the training time

Feature selection (FS) is a concept of vital importance in Machine Learning. Features are the inputs to any ML algorithm to train the model for learning the data and giving the best results of prediction. In real life algorithms, these features can range into hundreds and thousands. Therefore, a quest to find some key features which best represent the data and use them for the prediction, is must. Using all available feature leads to high computational cost, time and error [33].

Main categories of FS methods are filter, wrapper and embedded methods. Jurado et al. [34] applied the different building STLF techniques like Fuzzy Inductive Reasoning, RF and NNs using some filters for feature selection. These hybrid ML models were compared to statistical model ARIMA. The input data was for the houses in Catalonia, Spain. The results showed that Artificial Intelligence (AI) methods using FS step outperformed the statistical ones by giving 20 percent more accurate results.

It is mostly confused with feature extraction which differs in a way that feature extraction generates the new features by combining the existing ones while the feature selection uses the subsets of the features which greatly increase the accuracy of the model as shown in Figure 2.2. Feature selection is preferred over feature extraction as combination of features mostly does not have any physical significance. That is why feature selection is widely used in machine learning [35].

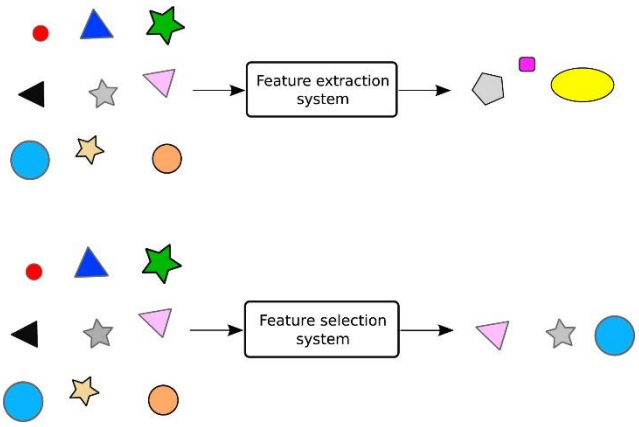

Figure 2.2 Difference between Feature extraction and Feature selection explained [35]

Feature selection methods can be widely classified into three categories.

1. Filter Methods
2. Wrapper methods
3. Embedded methods

2.2.2.1 Filter methods:

Filter methods do not involve any machine learning algorithm in feature selection. It selects the features based on statistical correlations completely. Basically, it scores each feature using some scoring function and based on that score a feature is decided to be selected or dropped. They are mostly univariate methods where the features are considered independently or regarding the target variable. Most commonly used example is *SelectKBest* where a scoring function e.g. *Chi Square*, is used to find the importance of feature. The other filter methods are using *ExtraTreeClassifier* used for feature importance and correlation matrix where a strong relation of a feature helps feature selection process. The broader view of filter methods is shown in Figure 2.3.

Figure 2.3 Filter methods explained sequentially [36]

2.2.2.2 Wrapper methods:

Wrapper methods use the machine learning algorithm to select the best performing subset of the features. It is a kind of greedy search where the algorithm tries many different subsets and checks the model performance to compare those subsets for final selection as shown in Figure 2.4. The drawback of this method is being computationally expensive and this problems is more when the data set is big. Wrapper methods can further be classified into forward Selection, backward selection and recursive feature elimination.

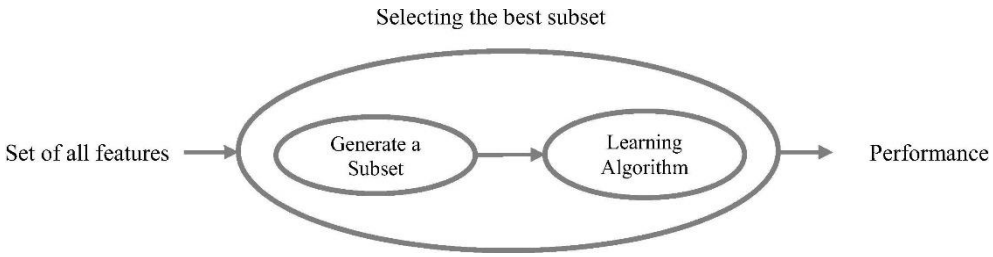

Figure 2.4 Algorithm used by wrapper methods of feature selection [36]

Forward selection tries the performance of each feature in data set and best feature is selected. The next step is to evaluate the performance of each remaining feature in combination with the first best feature and two best performing features are selected in this process which are then again tried with remaining features. This process keeps on going until we get the best combination of features.

Backward selection is opposite to the forward selection. Here the process starts with removing the worst performing feature in round-robin fashion and the performance of all other combinations is evaluated except the removed one. This process is repeated until the required accuracy is achieved.

Recursive feature elimination checks the performance of each subset and removes the least accurate feature in each step and in the end, it ranks the features in the order of their elimination.

2.2.2.3 Embedded methods:

Embedded methods can be called as the optimization of filter and wrapper as they use an algorithm with an inbuilt feature selection method. They tell that which features help in the accuracy of the model while it is being generated as depicted in Figure 2.5.

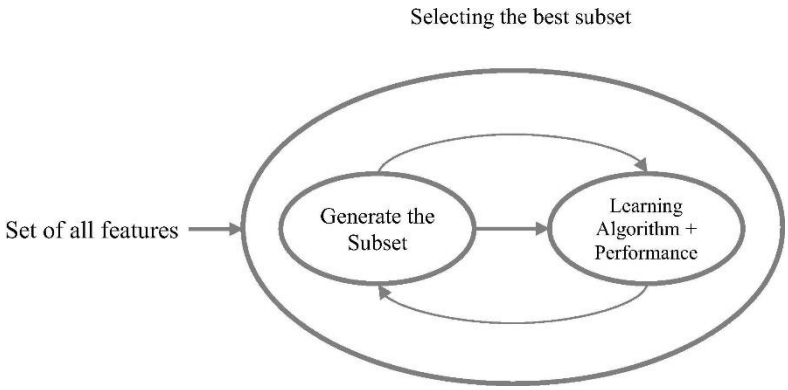

Figure 2.5 Working principle of embedded methods of Feature selection [36]

Feature selection methods discussed above can easily be explained by classification tree shown in Figure 2.6.

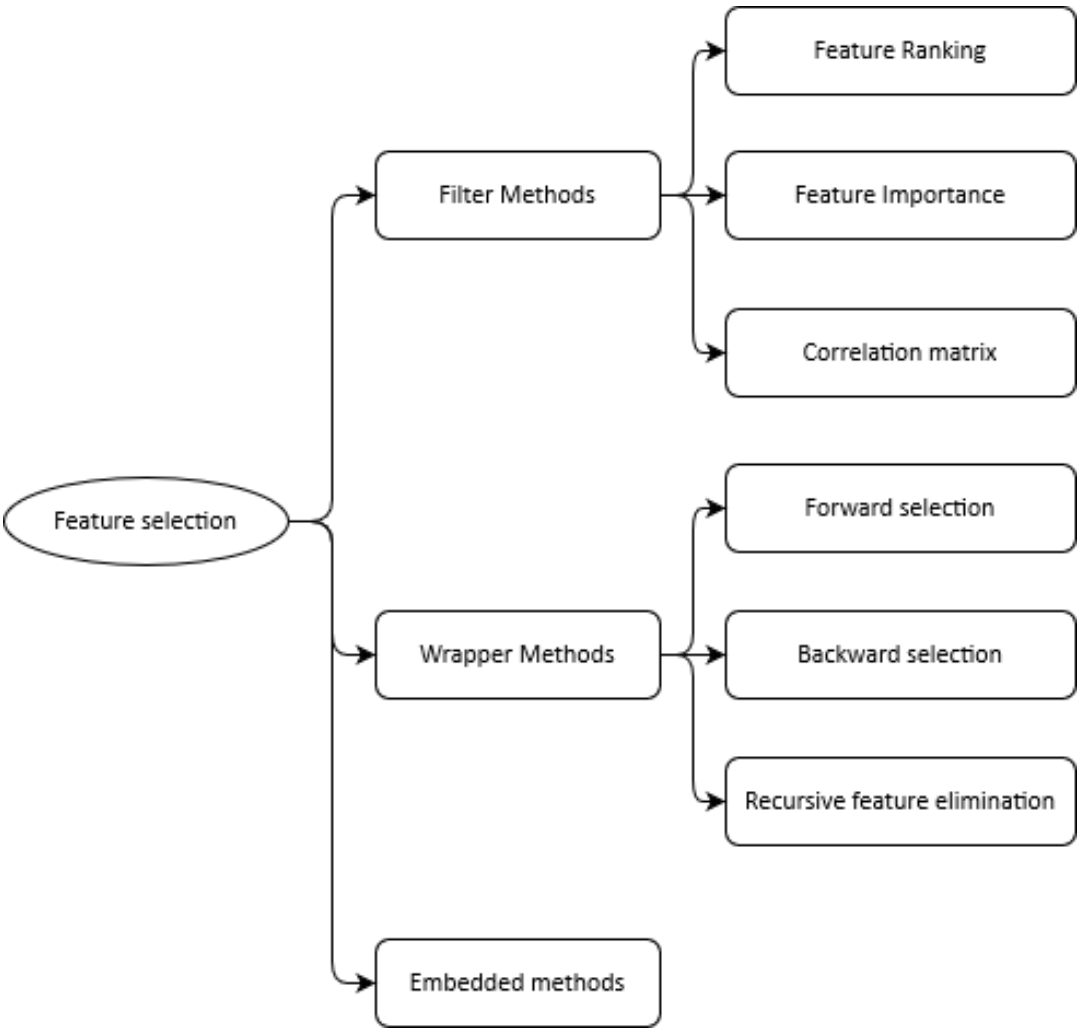

Figure 2.6 Feature Selection Methods classification

2.2.3 Random Forest

Random forest is machine learning technique belonging to ensemble methods. Ensemble are the methods of combining results of several different models working on same problem to get more flexible (less bias) and less sensitive (less variance) outcomes. The widely used ensemble methods are boosting and bagging [37].

Bagging works by training models in a parallel fashion where each tree or model uses a separate set of features to predict or classify the target variable. The outcome is the aggregate of all models used.

Boosting is rather a sequential process where the consequence of one model works as input to the next one which helps learning from the mistakes made at earlier stage.

The Figure 2.7 summarizes the ensemble methods.

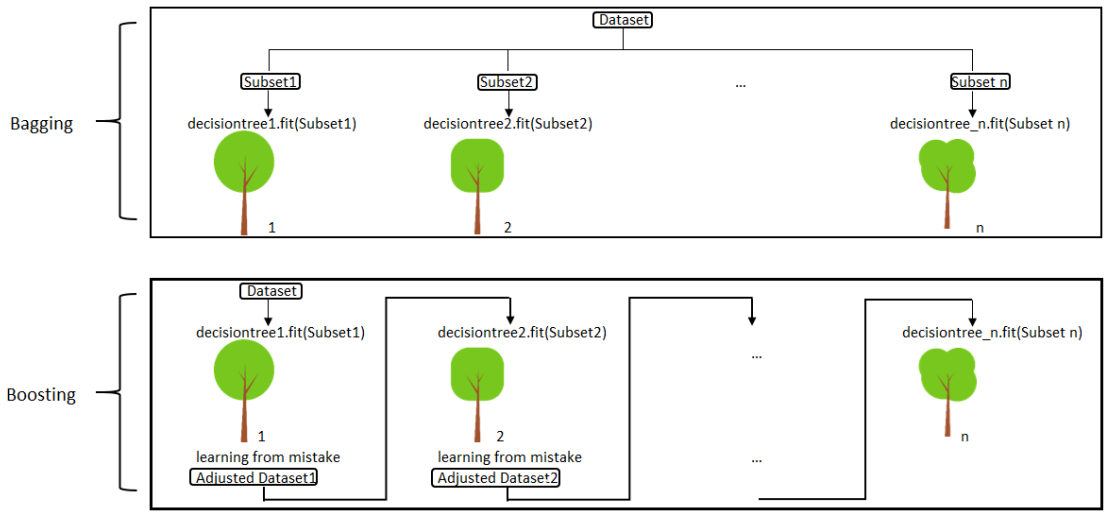

Figure 2.7 Bagging and Boosting techniques of Ensemble methods [37]

Random forest is a bagging type of ensemble learning methods. The binary unit of the random forest is decision tree. Decision tree is also an easy to implement approach of prediction but it results in high variance when the number of predictors (features) are more. Then random forest come into the picture to help with the problem of high variance. One of the advantages of random forest is its ability to handle with large number of features [38].

CHAPTER 3

METHODOLOGY

As the main task of this work is to forecast the energy services using machine learning. For this purpose, Python is used. Python is used on platform called “Anaconda” which is an open source and free software package containing python and R language. The advantage of using anaconda over python is that it helps the user to work with all libraries in python regardless of the python version used. In other words, it makes libraries compatible with all versions. Anaconda 3 is used here which by default has Python 3.7 in it but it can be changed as per requirement. Anaconda navigator is the management window of this platform, which allows user to launch available programs without writing the command lines in command prompt windows. Jupyter Notebook was used to launch python, which opens in any installed web browser.

The methodology is explained in following steps;

1. Data acquisition
2. Pre-processing data
3. Feature creation
4. Feature selection
5. setting up model for training
6. Hyper-parameter tuning
7. Testing with test dataset
8. Error calculation

3.1 Data acquisition

The data to be used for analysis in this work is taken for a central building in Instituto Superior Tecnico (IST), Lisbon. This building is in the heart if IST as shown in Figure 3.1. This central building has offices, classrooms, lecture halls, conference rooms and a library. The data is collected from the website[1] that is a public platform and it provide the weather data for mainland Portugal. This weather station is located on the rooftop of South Tower of campus at an altitude of 135 meters, with coordinates 38.736°N, 9.138°S. This station provides the following main weather elements with every five minutes.

- Temperature
- Humidity
- Atmospheric pressure
- Wind speed and direction
- Precipitation
- Total solar radiation on horizontal plane

The weather data for 2017 and 2018 is used in this work. There were some gaps in the data collected from this platform. The reason can be the fault in system or the absence of operator in public holidays. Almost of 90 days of data was missing across the time span of two years at different times of the year.

[1] http://meteo.tecnico.ulisboa.pt

This missing data was then obtained from another public platform called CAMS[2] (Copernicus Atmosphere Monitoring Service), which provides the information related to air pollution and health, solar energy, greenhouse gases and climate everywhere in the world. The combined data from both platforms has been presented in Table 3.1.

Table 3.1 Description of raw data acquired for model

	mean	std	min	max
Temperature (Celsius)	16.28	5.21	2.54	42.03
Relative Humidity (%)	60.06	17.40	8.90	100.70
Wind Speed (m/s)	1.78	2.69	0.00	17.69
Wind Gust (m/s)	2.29	3.41	0.00	23.40
Pressure (mbar)	1018.97	6.63	979.00	1040.00
Solar Radiations (W/m^2)	207.52	292.37	0.00	1380.00
Rain (mm/h)	0.05	0.41	0.00	17.90
Daily Rain (mm)	0.60	2.54	0.00	34.50

As the task of this work is to forecast the energy services hourly so all data must be in hourly fashion. The data was averaged to hourly values so that we have 8760 values for each year.

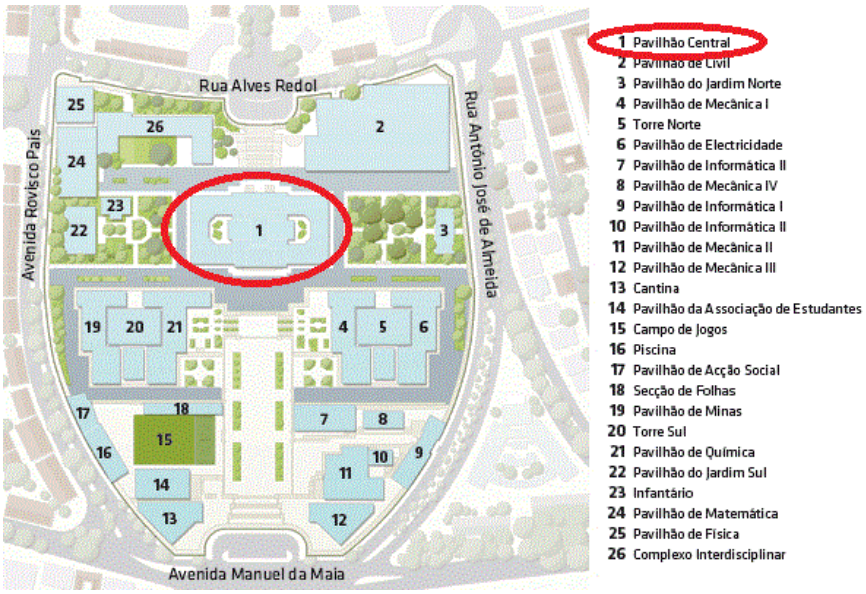

Figure 3.1 Location of building chose in campus

3.1.1 Energy Plus

Energy Plus is an open source building energy simulation software for modelling building heating, cooling, lighting, ventilating, and other energy flows. It was first released in 2001 having the best features of the BLAST (Building Loads Analysis and System Thermodynamics) and DOE–2 (Department of Energy), which are supposed to be the parent simulation software for Energy Plus. Energy Plus takes the user's description of the building's physical make up, installed equipment and mechanical systems

[2] https://atmosphere.copernicus.eu

along with the weather data, which is already in the software, to simulate the heating load, cooling load, equipment and lighting energy consumption.

In order to simulate the energy consumption, Energy Plus requires the two main files. One being the input file which contains the building's the physical description, installed equipment, lightings and the usage pattern of all those energy consuming devices based on the real time scenarios of usage of those installed systems. This may require the knowledge of the time period about which equipment is being used for what time period around the year and the frequency of occupants in different areas of building throughout the year. The whole building is modelled by writing all the necessary information discussed earlier to get the input file for Energy Plus. As modelling the building was not main task of this work and this file is being provided in form of *idf extension which can be directly used in the software for this work, so it will not be discussed in detail here. The EP launch window is shown in Figure 3.2 which tells how to upload the building model and weather file.

The second file needed for simulation is the weather file. Usually this file is already in the software. It has the average of all weather entities in that file for last 30 years, which means that providing an input file it will give the energy simulation results valid for every year. But for this work, we needed the two energy consumption simulations with respect to real weather data for the specific year. So, the default weather file is changed with the real weather file for both years 2017 and 2018. Here are the main elements of weather required for energy plus weather file;

- Temperature (dry bulb and dew point)
- Solar radiation components (beam and diffused)
- Relative humidity and atmospheric pressure
- Wind speed and direction
- Illuminance and precipitation
- Sky cloud cover and visibility

Unfortunately, not all the required data was available for the building location and the only the global solar radiation was known. The horizontal and beam components of hourly solar radiation were computed using the Equations 3.1 and 3.2 discussed in book [39]. The concept of clearness index has been used to calculate the radiation components.

$$K_t = \frac{G}{G_0} = \frac{I}{I_0} \tag{3.1}$$

$$\frac{I_d}{I} = \begin{cases} 1.0 - 0.09k_t & for\ k_t \leq 0.22 \\ 0.9511 - 0.1604k_t + 4.388k_t^2 & for\ 0.22 < k_t \leq 0.80 \\ -16.638k_t^3 + 12.336k_t^4 & \\ 0.165 & for\ k_t > 0.8 \end{cases} \tag{3.2}$$

Where,

G= Global horizontal radiation in watt per square meter

G_0= Extra-terrestrial radiation in watt per square meter

I= Hourly global horizontal radiation in watt hour per square meter

I_0= Hourly extra-terrestrial radiation in watt hour per square meter

I_d= Hourly diffused horizontal radiation in watt hour per square meter

I_b= Hourly beam horizontal radiation in watt hour per square meter

K_t= clearness index

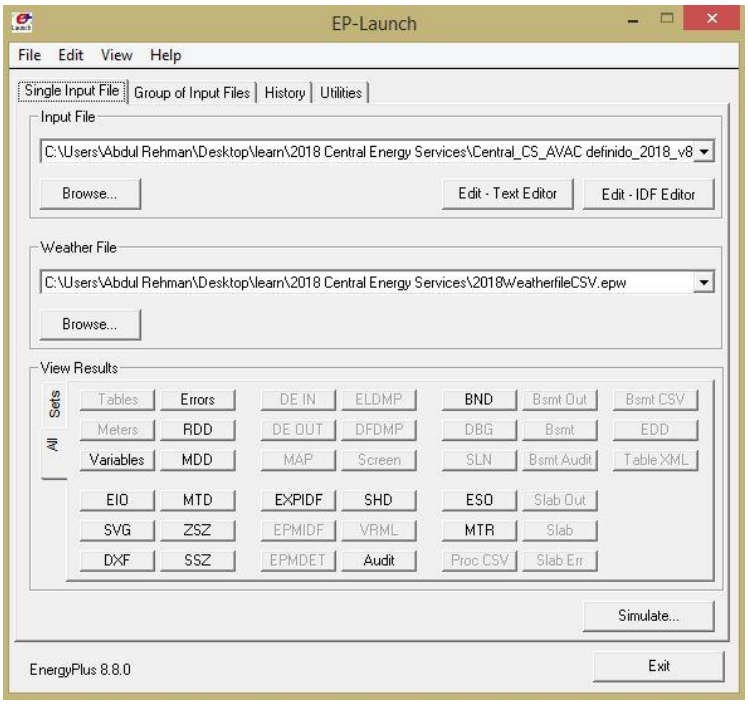

Figure 3.2 Energy Plus launch window

The default values of the radiation are replaced with the real ones in the weather file and this weather is converted into *EPW format by using the "weather Statistics and Conversions" option of the software. IDF editor as shown in Figure 3.3 allows the user to change and select the output variables required and their frequency under output meter tab. Eleven output variables have been selected with hourly frequency.

After changing the default weather file with real weather files for both years, the simulation is run on the software. The output of Energy Plus simulation has many files but the ones concerned for this work are in CSV (Comma Separated Variables) format.

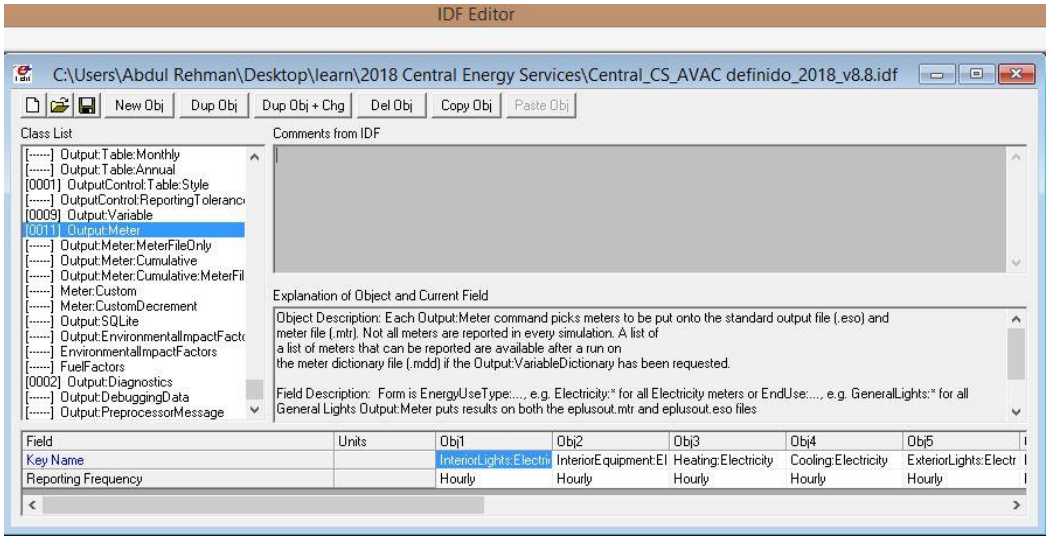

Figure 3.3 IDF Editor window of EP software

The Table 3.2 summarizes the output of simulations, which contains the hourly energy consumption of building in form of eleven entities. Count here shows total number of data points.

Table 3.2 Output of EP simulations showing the predicted Energy Services for 2018

	Count	Mean	Std	Min	Max
InteriorLights:Electricity [J](Hourly)	8760	5.654876e+07	6.700272e+07	2.511900e+06	1.936994e+08
InteriorEquipment:Electricity [J](Hourly)	8760	4.541372e+08	1.011802e+08	3.945575e+08	7.802335e+08
Heating:Electricity [J](Hourly)	8760	2.047681e+07	5.222656e+07	0.000000e+00	2.827290e+08
Cooling:Electricity [J](Hourly)	8760	7.234744e+07	2.727761e+07	6.192732e+07	3.970723e+08
ExteriorLights:Electricity [J](Hourly)	8760	5.543538e+06	5.556433e+06	0.000000e+00	1.124640e+07
Pumps:Electricity [J](Hourly)	8760	7.355005e+06	1.002980e-06	7.355005e+06	7.355005e+06
Electricity:HVAC [J](Hourly)	8760	6.238219e+07	5.891135e+07	3.046316e+07	3.618438e+08
Electricity: Plant [J](Hourly)	8760	6.953237e+07	1.229180e+06	6.928232e+07	8.549321e+07
Fans:Electricity [J](Hourly)	8760	3.173530e+07	2.610587e+06	3.046316e+07	4.182226e+07
Electricity:Building [J](Hourly)	8760	5.106860e+08	1.109730e+08	3.971810e+08	7.952908e+08
Electricity:Facility [J](Hourly)	8760	6.481441e+08	1.507395e+08	4.969265e+08	1.169746e+09

3.2 Pre-processing data

In-built python libraries are imported to the working notebook to pre-process the data. The main libraries used here are Pandas and NumPy. Pandas here helped us to read the data saved on computer in CSV format. Pre-processing of data involves finding any anomalies or missing values in data. We have two CSV files two, one having real weather data for year 2017 and other having output of energy plus simulations for 2017. Following steps summarize the pre-processing of data for this work;

- Converting temperature units from Celsius to kelvin to avoid any errors created by negative temperatures
- Dropping all the columns of Energy Services 2017 file except the one target variable (Facility Energy consumption)
- Converting Facility energy consumption from Joules to Kilowatts to easily understand the data
- Bringing both data sets in python time series format
- Merging data sets on time columns using pandas library

Following main libraries in Pythons have been used in this work.

3.2.1 Pandas

Pandas library is fundamental to any work done in python machine learning.[40] Here is the list of few things that it helps with and are used in this work;

- It helps to read and save the data in CSV format.
- It extracts the data from the CSV file stored in system to a data Frame easy to work.
- It calculates the main statistics of data like mean, median, maximum, minimum, etc.
- Provides the distribution of the whole data or a column in data.
- Data resampling and handling with missing data
- Merging two datasets and taking a subset of a data set
- Rearranging the dataset in pandas time-series format

3.2.2 NumPy

NumPy is another integral library in Python. It performs scientific computing operations in the data analysis.[41] It contains;

- Useful linear algebra, Fourier transform, and random number capabilities
- A powerful N-dimensional array object

3.2.3 Matplotlib

This is the plotting library of python. It helps visualizing data and comparing two data sates or entities in graphical from. It contains simple pyplot, bar-chart, histogram, correlation matrix, scatter plot to name a few.[42]

3.2.4 Scikit-Learn

This is main machine learning library in python.[43] It contains the function;

- Classification and Regression techniques
- Feature selection methods and scoring functions for feature selection
- Splitting data into train and test data sets
- Metrics to calculate the errors and accuracy of the models used

3.3 Feature creation

Two sets of features have been used for the model training. First set has seven features is provided by real weather data.

- Temperature
- Relative Humidity
- Atmospheric pressure
- Wind speed
- Wind gust
- Precipitation (hourly and daily)
- Solar radiations on horizontal plane (beam, diffused and global)

The second set contains constructed features using engineering knowledge and exploratory data analysis in python. It contains six features;

- Hour of day
- Day of week
- Month of the year
- Consumption in previous hour
- Average consumption of previous three hours
- Type of the day

The hour of day provides important information for example the consumption will be at peak in the afternoon when almost all the classrooms are being utilized and it will be less in the evening and morning hour as shown in figure below for a day in 2017.

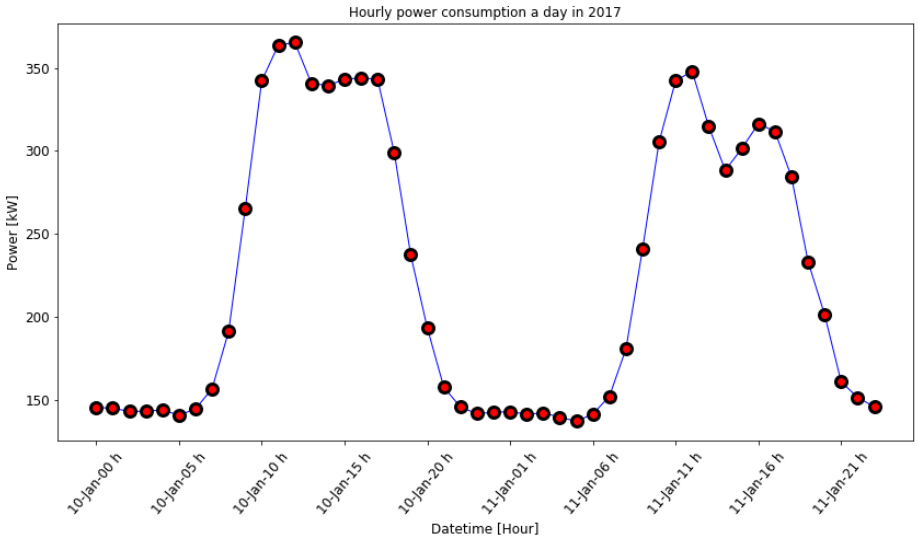

Figure 3.4 Power consumption for a day in 2017

The data analysis also tells how consumption varies around the year. Following figure shows that power consumption is less in the month of August as this is vacation month in campus.

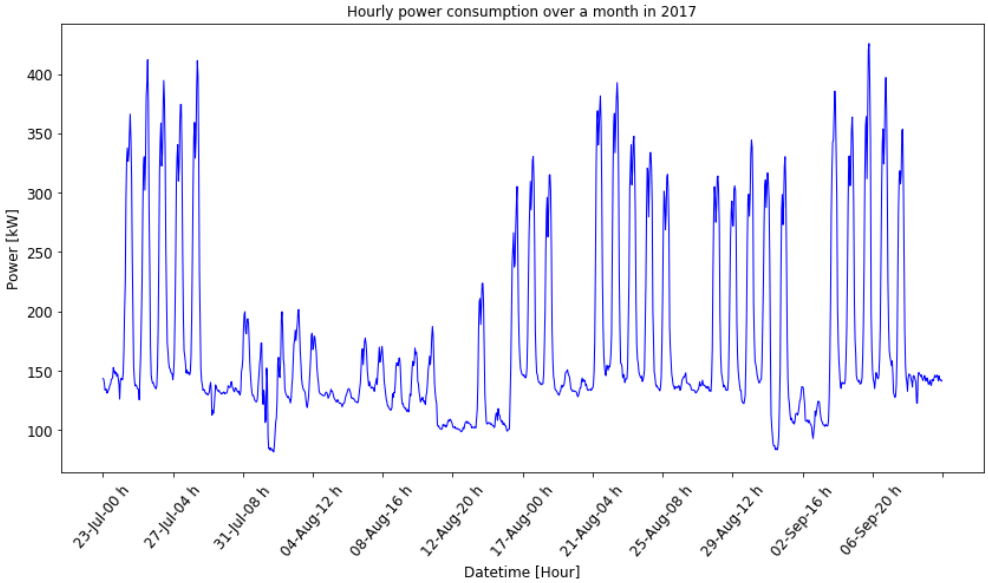

Figure 3.5 Power consumption for the month of August

Similarly the day of week tells if it is weekend or a normal working day. The type day has values 0 and 1 in the data. 0 means it is holiday and 1 means it a working day. This accounts for the public holidays also in addition to weekends. The consumption in previous hour and average of three hours also help in learning of model to better understand the pattern. This means the historical data is of importance for this model.

3.4 Feature Selection

None of the single method is efficient for feature selection, so both filter and wrapper methods have been used for feature selection and the top features in all methods have been used for the model training.

Three of the filter methods used are namely feature scores, feature importance and correlation matrix.

Feature scores uses *SelectkBest* method to provide the scores of the features. *SelectkBest* uses a statistical scoring function to calculate the scores of the features. Here the scoring function used is chi square. The mathematical form of Chi square is in Equation 3.3 below.

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{3.3}$$

Where

O_i = number of observations in class i

E_i = number of expected observations of class if there was no relationship between the feature and the target

This method calculates the χ^2score between each feature and the target variable and returns the K best features. A value of 10 is selected for K to find the 10 best features. The output of *SelectKBest* is as shown in Figure 3.6.

	Specs	Score
8	Global Horizontal Radiation (Wh/m2)	1.661637e+06
6	Direct Normal Radiation (Wh/m2)	1.270826e+06
7	Diffuse Horizontal Radiation (Wh/m2)	4.718567e+05
10	Power_KW	8.525313e+04
15	Power_KW-1	6.906341e+04
12	hour	1.003095e+04
0	HR	8.591672e+03
14	weekday	5.824537e+03
5	rain_day (mm)	4.209710e+03
4	rain_mm/h	1.428827e+03

Figure 3.6 Feature scored by Chi Square

The feature importance uses a model *ExtraTreeClassifier* which is inbuilt function of python. It provides with the scores of each feature. The higher score means the relevancy of feature to the target value. Again 10 best features are printed using *nlargest* command from pandas. The 10 best features are shown in Figure 3.7. The main difference is the choice of month and temperature instead of rain and rain_day.

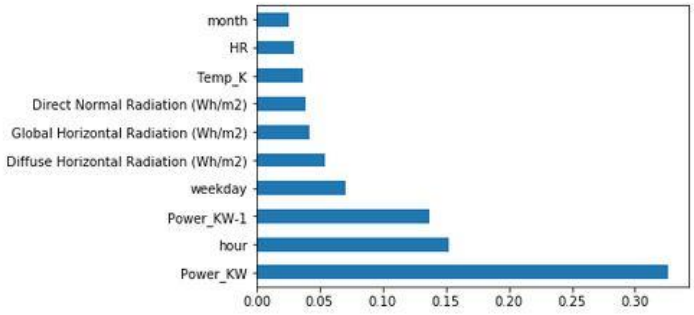

Figure 3.7 Feature importance defined by ExtraTreeClassifier

Correlation matrix provides the relation between all the features and the target variable. Correlation can either be positive or negative, a positive value indicates that target value increases as features value increases while in negative correlation the increasing feature value decreases the target value. The *heatmap* is created to better visualize the data using seaborn library as shown in Figure 3.8.

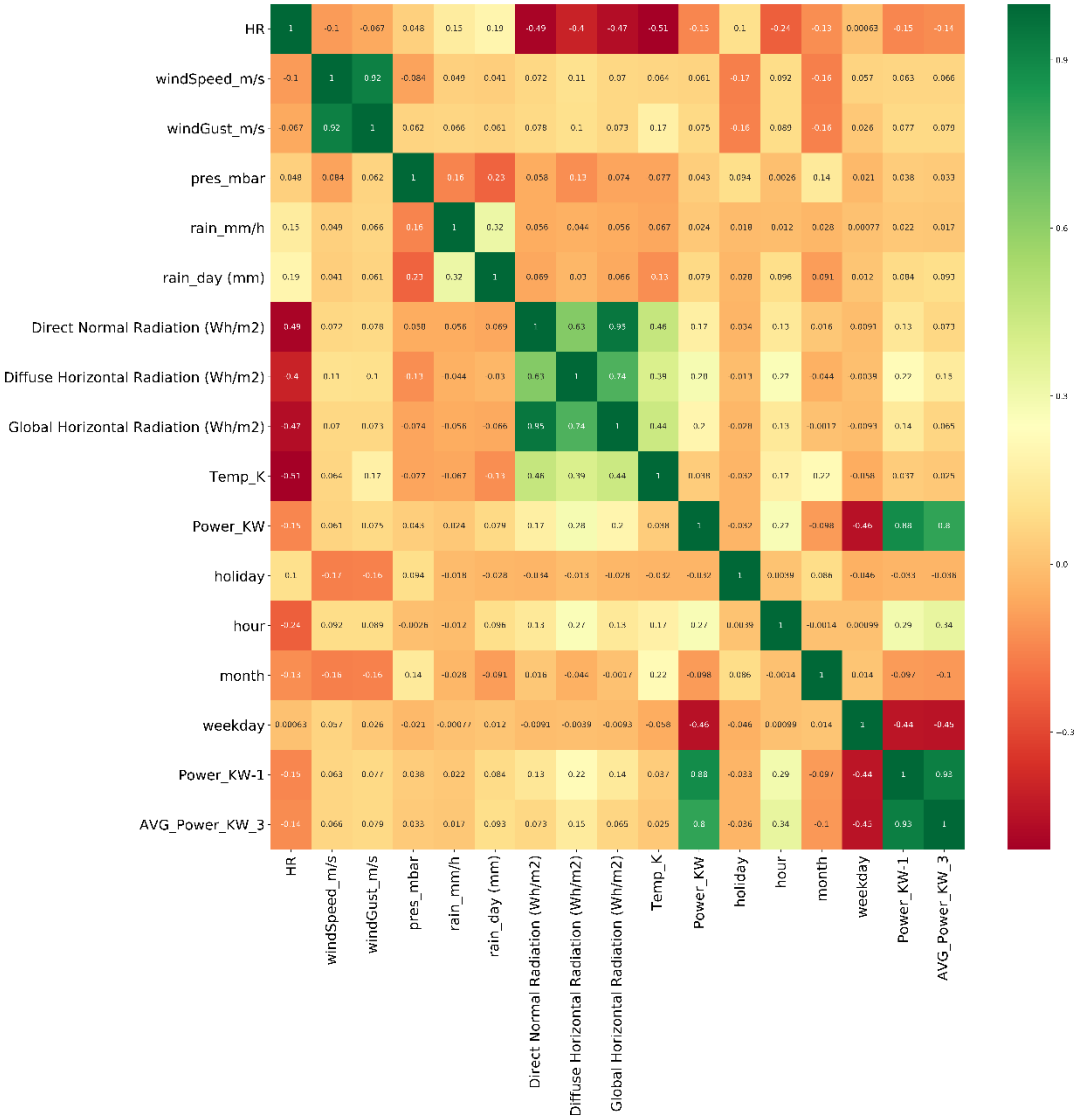

Figure 3.8 Heatmap showing correlation between features

The top 10 features selected from this correlation matrix are as following.

1. Direct normal radiation
2. Diffused Horizontal radiation
3. Global Horizontal radiation
4. Temperature
5. Consumption in previous hour
6. Average consumption in last three hours
7. Hour
8. Rain
9. Wind speed
10. Pressure

Recursive Feature elimination is used here from wrapper methods for feature selection. The estimator used here is *LinearRegression* and 10 best features are required to be calculated. RFE provides the ranking of features. The 10 best features are as follows;

1. Direct normal radiation
2. Diffused Horizontal radiation
3. Global Horizontal radiation
4. Temperature
5. Holiday
6. Hour
7. Relative humidity
8. Previous hour consumption
9. Average of previous three hours consumption
10. Rain

Finally, 8 features, which are common in all above methods, have been selected for the model training.

3.5 Setting up model for training

Setting up the model starts by splitting the data into training and testing sets. There can be any ratio between these two but more training data means more learning and that is what the task is. There is inbuilt function of *train_test_split* in Scikit learn library of python. Following lines show the commands used for splitting the data set.

```
X_train, X_test, Y_train, Y_test= train_test_split (X, Y, shuffle=True, test_size=0.2)
```

Here X specifies the features array and Y represents the target value. This command automatically splits the data into 75 percent for training and 25 percent for testing if the default parameters are used inside the brackets. The important parameters are explained as follows;

Test_size = the percentage of test data in input data

Shuffle= whether to shuffle or not the data while splitting. True means yes.

After we have split the data, we introduce the model to be used which is random forest here. We import the *RandomForestRegressor* from ensemble methods in Scikit library. Random forest requires few parameters to be set. But for the first run the default values of parameters is used. The model is fitted using the fit command as shown in following command lines.

```
rf = RandomForestRegressor (n_estimators=1000, random_state=42)
rf.fit(X_train, y_train)
```

After the simulation, the train and test error has been calculated using metrics function in Scikit library.

3.6 Hyper-parameter tuning

Random forest regressor has several parameters which are needed to be set for the better results and there is no specific value of each parameter to be used instead it differs with the type of data set. Few important parameters are listed in with their description.

Table 3.3 Description of parameters to be tuned [44]

max_features	The number of features to consider when looking for the best split • If int, then consider max_features features at each split. • If float, then max_features is a fraction and int(max_features * n_features) features are considered at each split. • If "auto", then max_features=sqrt(n_features). • If "sqrt", then max_features=sqrt(n_features) (same as "auto"). • If "log2", then max_features=log2(n_features). • If None, then max_features=n_features.
Max_depth	The maximum depth of the tree. If None, then nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples
Min_sample_leaf	The minimum number of samples required to be at a leaf node. • If int, then consider min_samples_leaf as the minimum number. • If float, then min_samples_leaf is a fraction and ceil(min_samples_leaf * n_samples) are the minimum number of samples for each node.
Min_samples_split	The minimum number of samples required to split an internal node. • If int, then consider min_samples_split as the minimum number. • If float, then min_samples_split is a fraction and ceil(min_samples_split * n_samples) are the minimum number of samples for each split.
Bootstrap	Whether bootstrap samples are used when building trees. If False, the whole dataset is used to build each tree.
N_estimators	The number of trees in the forest.

Hyperparameter tuning means running and fitting the model with many different values of parameters to find the optimum value of each of them which provides the best results of model.

Here a technique cross validation CV helps optimizing these parameters. Cross validation further divides the training set into K number of subsets (we provide the K value). The model uses K-1 subsets for training the model and the K^{th} set for testing which is called cross validation technique. The model

is fitted iteratively K times every time using new K^{th} subset for validation. CV technique is best explained in Figure 3.9 with 5-fold cross validation.

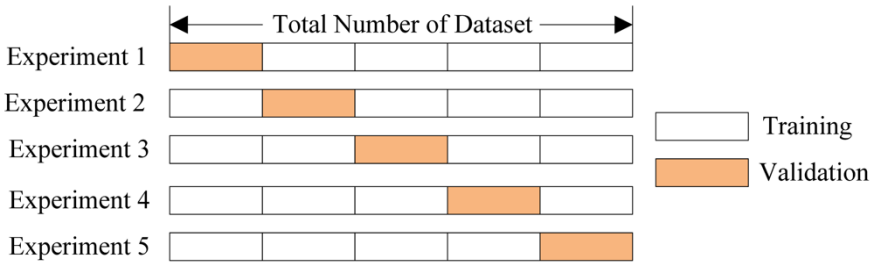

Figure 3.9 5 fold Cross Validation explained [45]

In this way, the model must try for each combination of the parameters and if a wide range of parameters is provided along with a high value of CV, the computation time increases rapidly. But Scikit library has a solution to this problem in form of *RandomizedSearchCV* command where the model does not try every combination of parameters but only a specific number of parameters set by the user. This information is passed under *n_iter* parameter of *RandomizedSearchCV.* A value of 100 is used for this work along with 10-fold CV which means the models is run and fit with 1000 random combinations and returns the best combination. The range of parameter searched is shown in Figure 3.10.

```
{'bootstrap': [True, False],
 'max_depth': [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, None],
 'max_features': ['auto', 'sqrt'],
 'min_samples_leaf': [1, 2, 4],
 'min_samples_split': [2, 5, 10],
 'n_estimators': [200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000]}
```

Figure 3.10 Parameter sets to be optimized for mode

With the data having hourly values of one year this took approximately two hours using four processors of system and the optimized set of parameters is viewed by using *best_params_* command which are shown in figure.

```
{'n_estimators': 1400,
 'min_samples_split': 2,
 'min_samples_leaf': 4,
 'max_features': 'sqrt',
 'max_depth': 80,
 'bootstrap': False}
```

Figure 3.11 Best Parameters to be applied on model

The model is trained again with the optimized parameters using the whole set of training values as got in *train_test_split* step.

3.7 Testing the model

Once the model has been trained with best parameters obtained by Hyperparameter tuning, the model is applied to the test data set which is new to the model. The test set here is the data for year 2018 which includes the weather features and synthetic features as created for year 2017 test data. The same 8 features are used which came out to be the best in training data by applying all feature selection techniques. The prediction is made by using *predict* command in python.

3.8 Error Calculation

Prediction performance is evaluated by using three metrics which are mean absolute error (MAE), root mean square error (RMSE), and coefficient of variance of root mean square error (CV-RMSE). First two are scale dependent while the last one is independent [46].

Mean absolute error (MAE) measures the absolute value of the difference between the actual and predicted value. It averages the values of error in a data set and it is independent of the direction of error. It gives the error value in the units of actual or predicted values.

Root mean square error (RMSE) is the square root of the variance of the residuals. It indicated how closely predicted values fit the actual values. Like MAE it also gives the error in the units of data.

Coefficient of variance of root mean square error (CV-RMSE) measures the percentage error and gives the clear view of the performance of the model. It is preferred to the MAE for the reason that denominator is mean of actual power consumption over a time period which cannot be zero as in case of MAE where denominator is actual power consumption (can be zero in some case).

The three metrics are mathematically expressed by equations;

$$MAE = \frac{\sum_{k=1}^{n} |y_a - y_p|}{n} \tag{3.4}$$

$$RMSE = \sqrt{\frac{\sum_{k=1}^{n} (y_a - y_p)^2}{n}} \tag{3.5}$$

$$CV - RMSE = \frac{\sqrt{\frac{\sum_{k=1}^{n} (y_a - y_p)^2}{n}}}{\frac{\sum_{k=1}^{n} y_a}{n}} \tag{3.6}$$

Where y_a means actual value and y_p means predicted values and n is total number of values in data set.

There is a threshold for CV-RMSE set by researchers and international bodies for a model to be a good-fit explained in Table 3.4. Three main bodies are;

- American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) guidelines 14 [47]
- International Performance Measurements and Verification protocol (IPMVP) [48]
- M&V guidelines for FEMP [49]

Table 3.4 Limits set for a good fit model

Standard/guideline	Monthly criteria CV-RMSE (%)	Hourly criteria CV-RMSE (%)
ASHRAE Guideline 14	15	30
IPMVP	-	30
FEMP	15	30

CHAPTER 4

RESULTS AND DISCUSSION

This section has been subdivided into two. The first section is about the simulation results from EP software while the second section discusses the ML model performance.

4.1 Simulation Results

As discussed in the earlier section Energy Plus software uses the default weather file and it had to be replaced with the real weather data file for both the years 2017 and 2018. The simulation results have shown that it improves the output by providing values closer to the actual power consumption.

It has been found out that using default weather file for EP oversimplifies the model and gives the results which are far away from the actual power consumption. So, simulations have been run twice using default and real weather files and the simulation results compared to the actual values. The graphical analysis has shown that the real data provides the values closer to the actual ones specifically for the months of March, April, November and December. Figure 4.1 and Figure 4.2 show that concerned part of graphs zoomed in.

This proves that the time and effort invested in getting the real weather data will help in better machine learning model in the later stage.

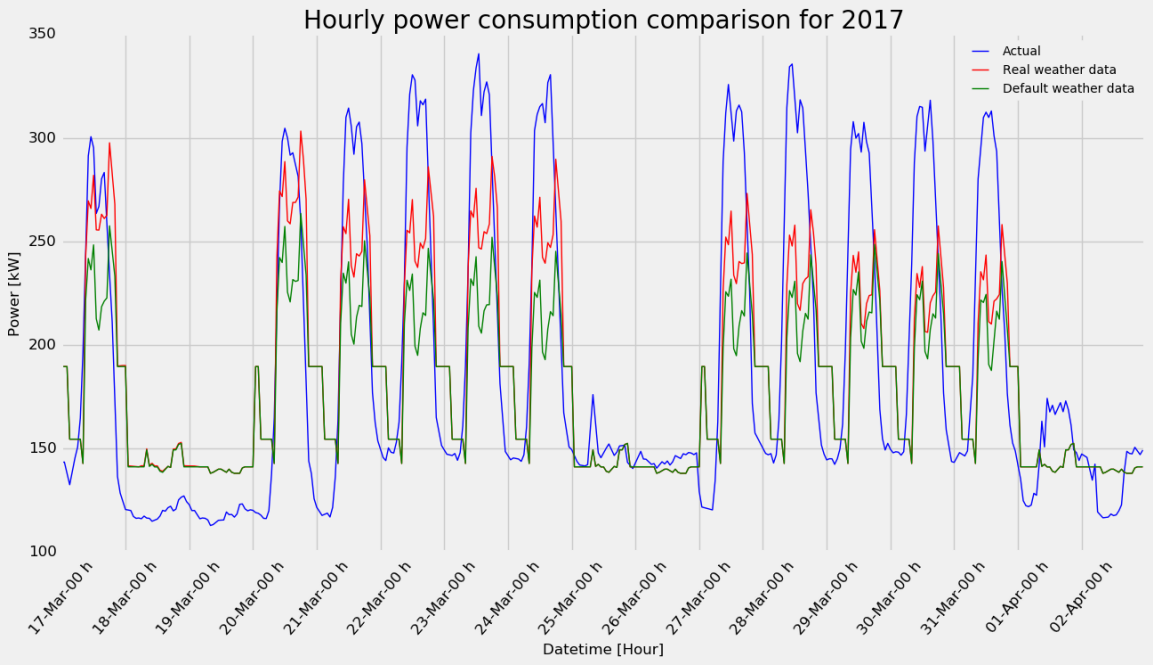

Figure 4.1 EP Simulation results comparison for 2017 data

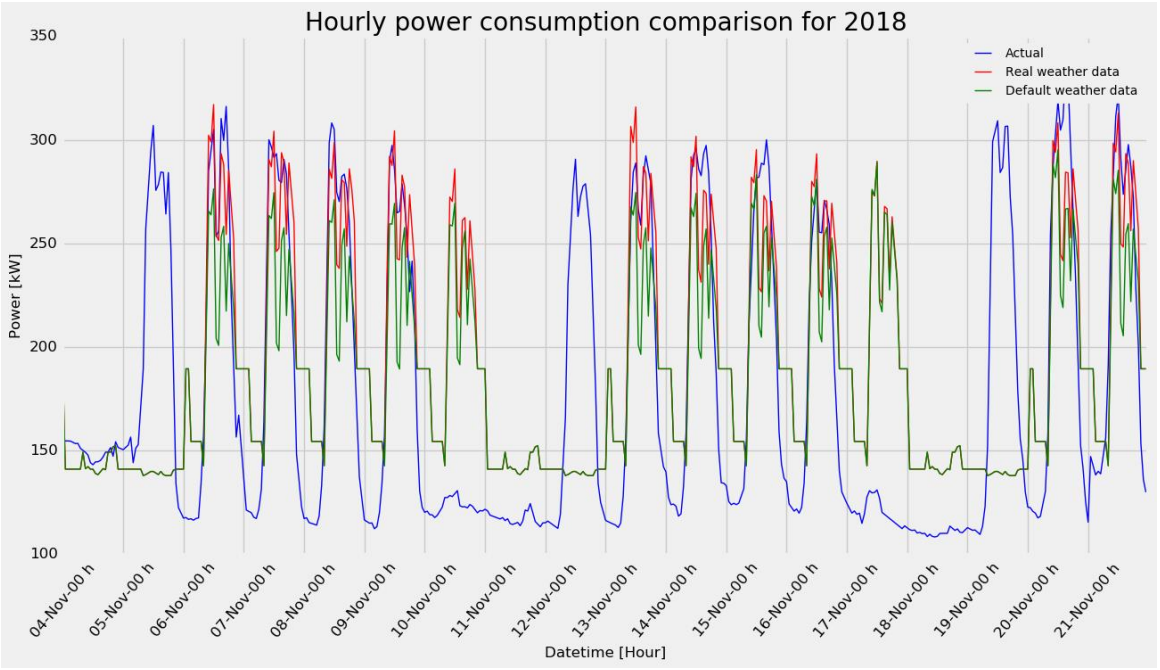

Figure 4.2 EP Simulation results comparison for 2018 data

4.2 Forecasting results

Power consumption is forecasted for four types of consumption.

- Facility
- Building
- HVAC
- Exterior lights

The relation between them is shown by chart in figure.

4.2.1 Facility

Here the input to the model is the power consumption for year 2017 simulated by EP. The forecast has been done for year 2018 and that forecast is compared against the EP simulated power consumption and with the actual power consumption.

Figure 4.3 shows only a small portion of the graph comparison for the month of February. The predicted power consumption follows the EP simulated consumption for 2018 which has been used here to validate the model. Prediction shows the values slightly higher than the highest values of the day and slightly lower than the lowest in the day.

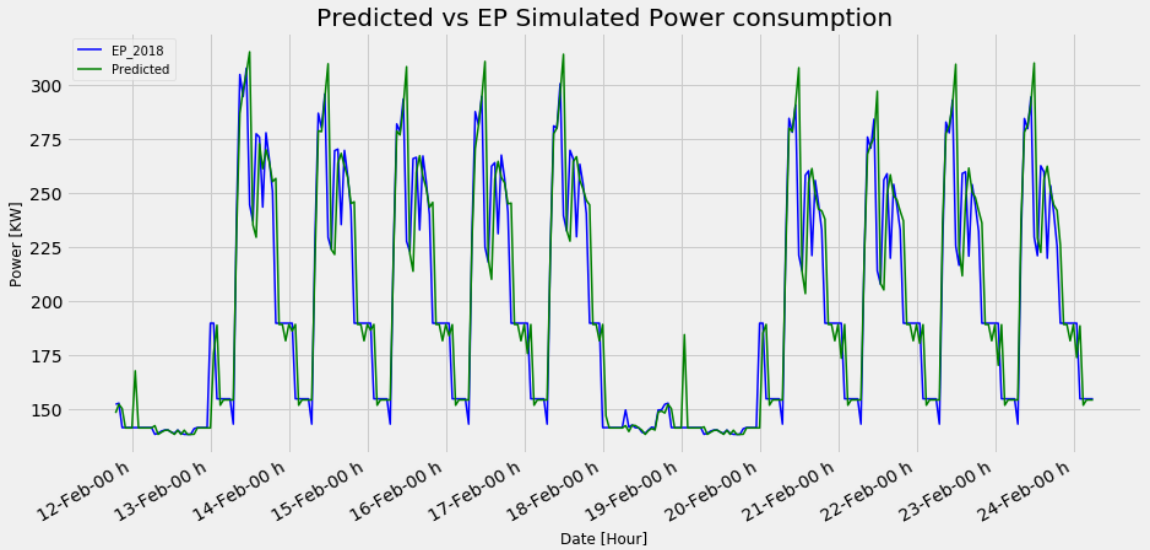

Figure 4.3 RF model prediction compared against EP simulated for 2018

The validation metrics used here are MAE, RMSE and CV-RMSE which gives the values 8.6 KW, 16.7 KW and 9.2% respectively. MAE shows mean of absolute error between the forecast and actual values. Here a value of 8.6 means that on average any predicted value of the dataset is 8.6 kilowatts away from the EP simulated values irrespective of the direction i.e. can be less or more than EP values. RMSE shows how far is the farthest values of prediction to the regression line. For this case all the predicted values fall in the range of 16.7 kilowatts around the regression line. The facility power consumption values range from 138 KW to 324 KW which gives the actual range of 186 KW. If we compare the values of MAE and RMSE to the actual range of power consumption, they are just 10 percent which seems to be an acceptable range. CV-RMSE is scale independent indicator of model performance. For this case, CV-RMSE has a value of 9.2 percent which is in the acceptable range according to all three main international bodies discussed in previous chapter.

The next step was to compare the prediction to the actual power consumption and it is shown graphically in Figure 4.4 and Figure 4.5 which shows the predicted values are rarely near the actual values over a period of one month. This fact is represented by the values of MAE, RMSE and CV-RMSE which are 52 KW, 69 KW and 38% respectively. These high numbers indicate a faulty model for prediction but this hypothesis is proven wrong by running the model with actual power consumption as input and comparing it with actual power consumption for 2018. The values of MAE, RMSE and CV-RMSE comes out to be 7.2 KW, 11.4 KW and 6.3% respectively which are quite reasonable and well in range. This is graphically represented in Figure 4.6.

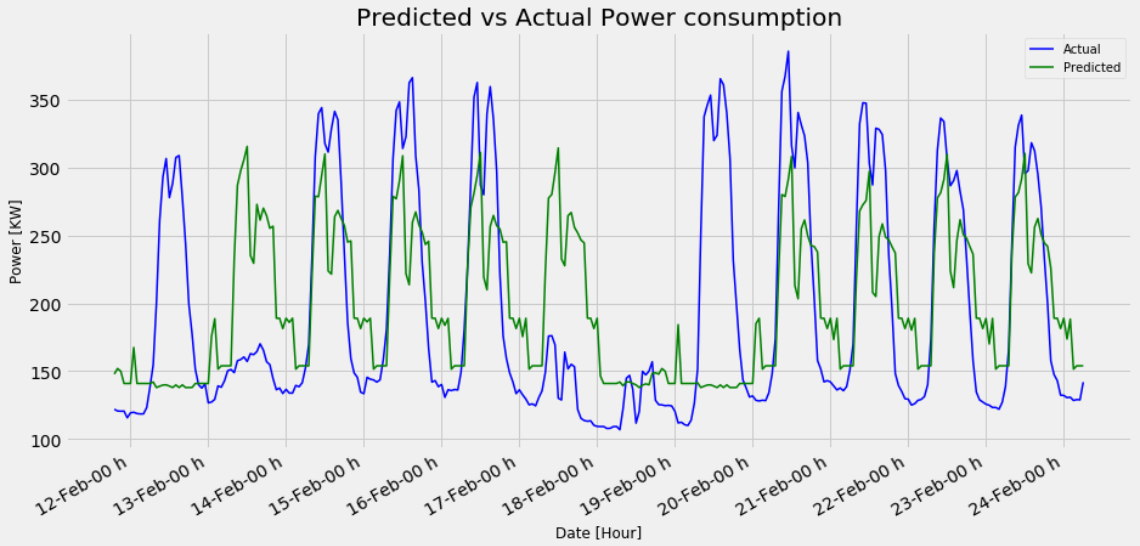

Figure 4.4 RF model prediction compared against actual power consumption for 2018

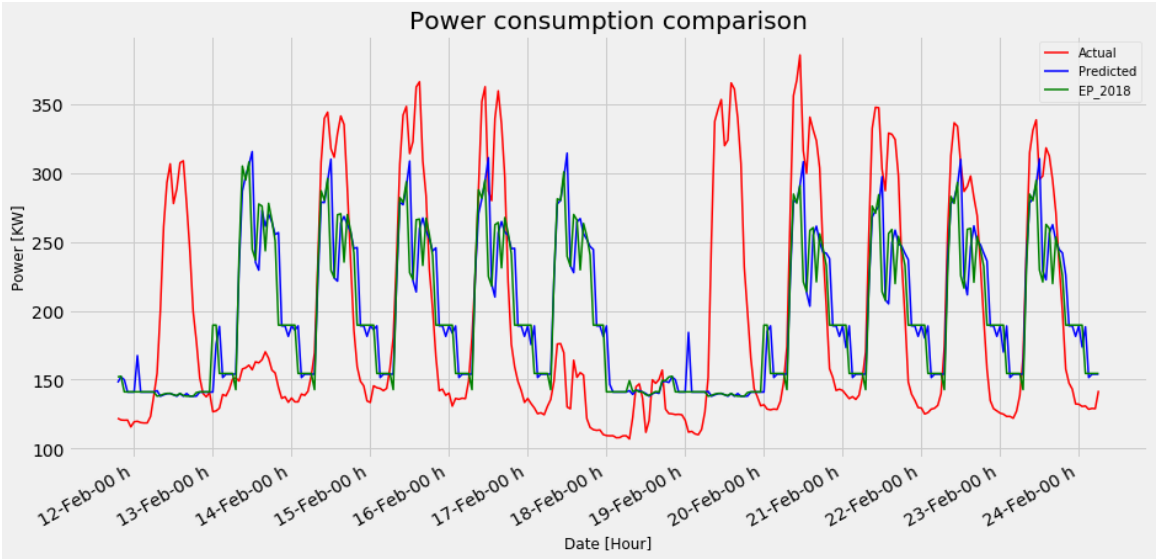

Figure 4.5 RF model compared against EP simulation and actual power consumption for 2018

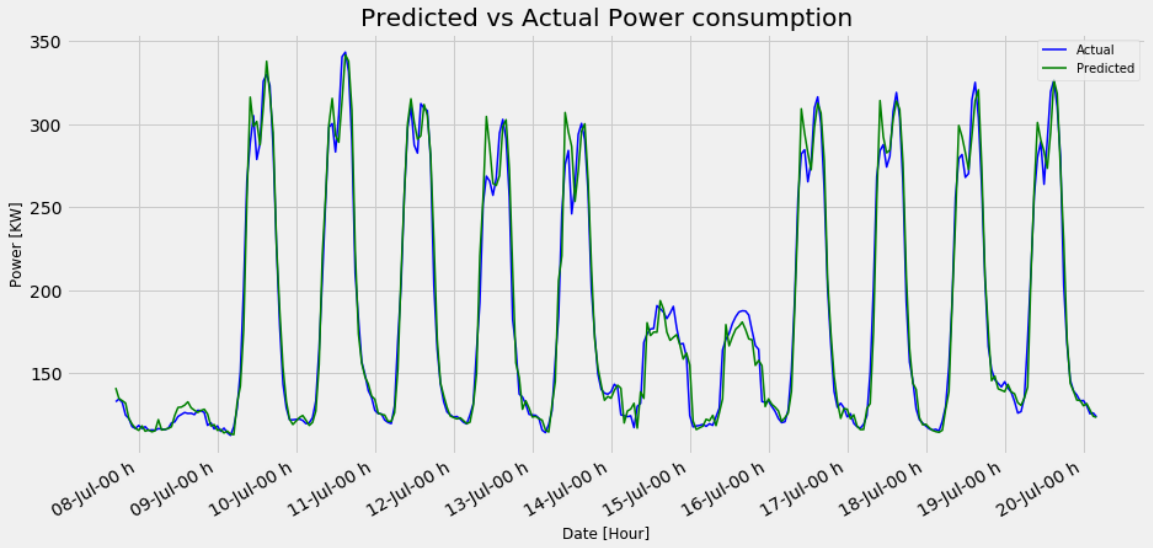

Figure 4.6 RF model performance using real data of power consumption

4.2.2 Building

Building power consumption comprises of interior equipment and interior lights. The model is reconstructed to forecast building power consumption. It includes all the feature selection methods and optimizing the parameters of *RandomForestRegressor* with 10 folds cross validation. The optimized parameters are as shown in Table 4.1.

Table 4.1 Optimized (tuned) parameters for building model

Parameter	**value**
n_estimators	400
min_samples_split	10
min_samples_leaf	4
max_features	'auto'
max_depth	70
bootstrap	True

The prediction is graphically compared to the EP simulated values shown in Figure 4.7 which shows a good fit. The metric MAE, RMSE and CV-RMSE has the values 6.5 KW, 13.4 KW and 9.5% respectively. EP simulated power consumption values (used for validation) range from 110 KW to 220 KW and scale dependent metrics has the values under 10 percent of the range. Similarly, the CV-RMSE has a value of around 10 percent which is accepted by all international bodies' criteria for a good fit.

Unfortunately, the actual building power consumption was not available to compare the model against it.

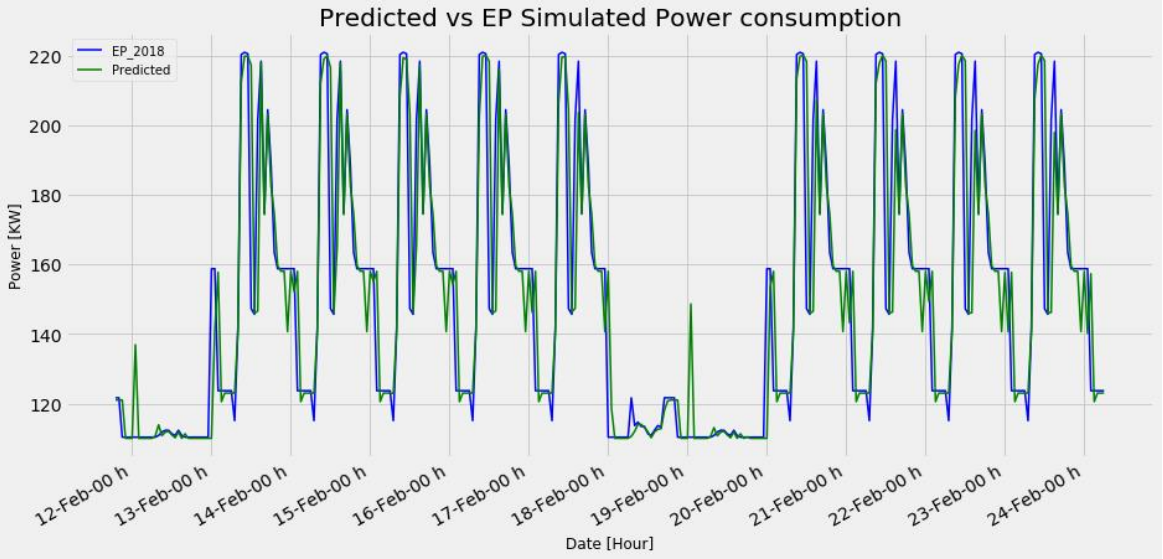

Figure 4.7 RF model prediction compared against EP simulations for 2018

4.2.3 HVAC

HVAC consumption in EP consists of the heating and fans power consumption. The model is trained using the features which are slightly different from the features used for facility and building power consumption. They include relative humidity, wind speed, global solar radiations, temperature, hour and type of the day, month of year and consumption in previous hour. These features are the results of application of all features selection methods. Again, the hyperparameter tuning resulted in different optimized parameters for *RandomForestRegressor* shown in Table 4.2.

Table 4.2 Optimized (tuned) parameters for HVAC model

Parameter	value
n_estimators	1600
min_samples_split	5
min_samples_leaf	1
max_features	'auto'
max_depth	10
bootstrap	True

The forecasting results are compared to the EP simulated results and are shown in Figure 4.8. The graphs show that prediction varies from EP simulated values only on weekends where simulations show less consumption while prediction show significant high values. Moreover, the metrics MAE, RMSE and CV-RMSE comes out to be 4 KW, 8.8 KW and 37% respectively. When compared to the range of values (8 KW to 92 KW), the scale dependent metrics lie below 15 percent. While CV-RMSE has the value, which is more than internationally accepted criteria of good fit but it can be justified by looking at the month of August when EP simulations has almost zero values and model predicts the significant values.

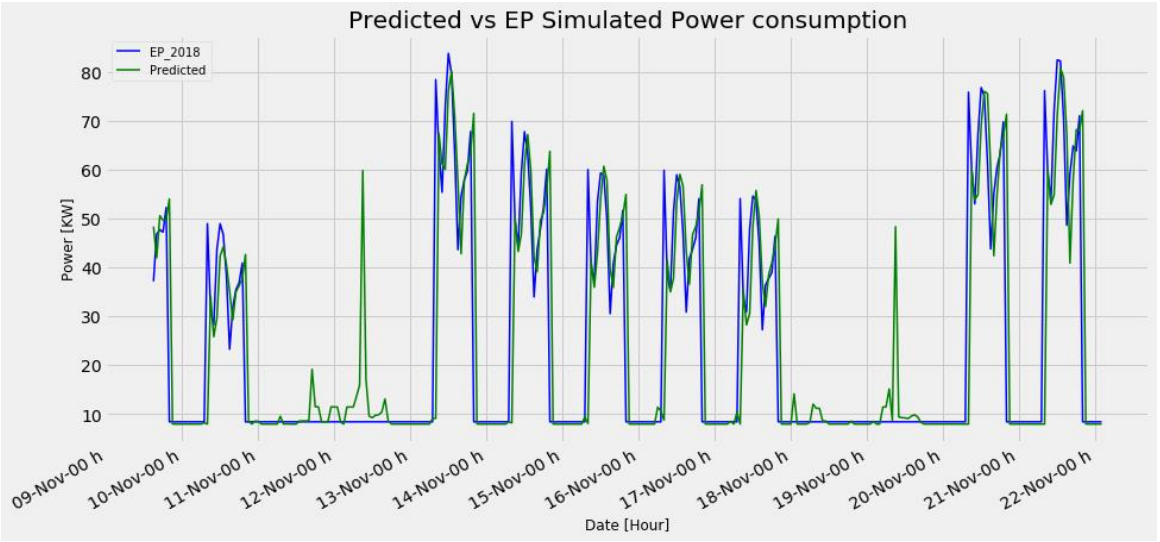

Figure 4.8 RF model prediction compared against EP simulations for 2018

Figure 4.9 shows the comparison of the prediction with EP simulated and actual power consumed by HVAC system. It indicates that EP model hugely overestimates the HVAC consumption.

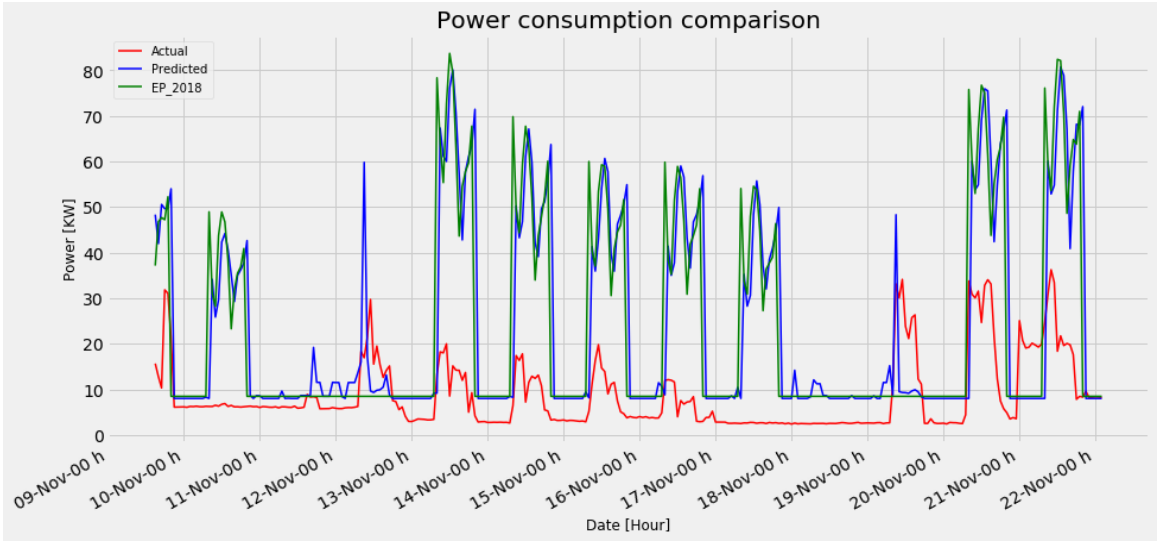

Figure 4.9 RF model performance compared against actual HVAC consumption for 2018

The next step was to train the model with the real HVAC consumption in 2017 and test it against the real HVAC consumption in 2018. The metric MAE, RMSE and CV-RMSE has the values 3.5 KW, 6.4 KW and 32% respectively which are better than the model tested on simulated data. The trend is compared in the Figure 4.10 which shows the better fit compared to the simulated data and proves the accuracy of the model.

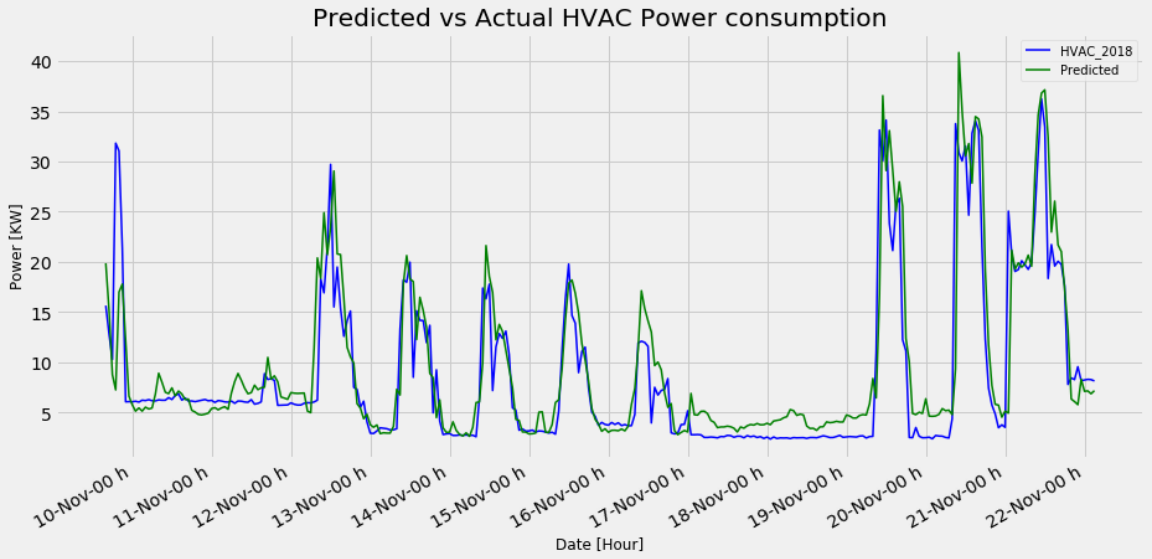

Figure 4.10 RF model performance with real HVAC consumption data

4.2.4 Exterior lights

It provides the power consumed by the light's exterior to the building and they are used only during night and their consumption is significantly small compared to the other energy services. The model trained using the optimized parameters shown in Table 4.3.

Table 4.3 Optimized (tuned) parameters for Exterior Lights model

Parameter	value
n_estimators	1200
min_samples_split	8
min_samples_leaf	3
max_features	'sqrt'
max_depth	6
bootstrap	True

The comparison with the EP simulation results is shown in Figure 4.11 which shows the good fit. The predicted values are slightly lower than the simulated values for most of the days.

The metrics MAE, RSME and CV-RMSE has values of 0.29 KW, 0.81 KW and 28% respectively. EP simulated values has a range of almost 4 KW which means the scale dependent metrics are below 20 percent of the range. While CV-RMSE has the value 28 percent which is quite high but the reason for this is found by looking at the month of August. August is the vacation month in campus and model predicts the power consumed by lights which is much more than the EP simulated power consumption. EP model takes August as almost zero consumption (especially for lights).

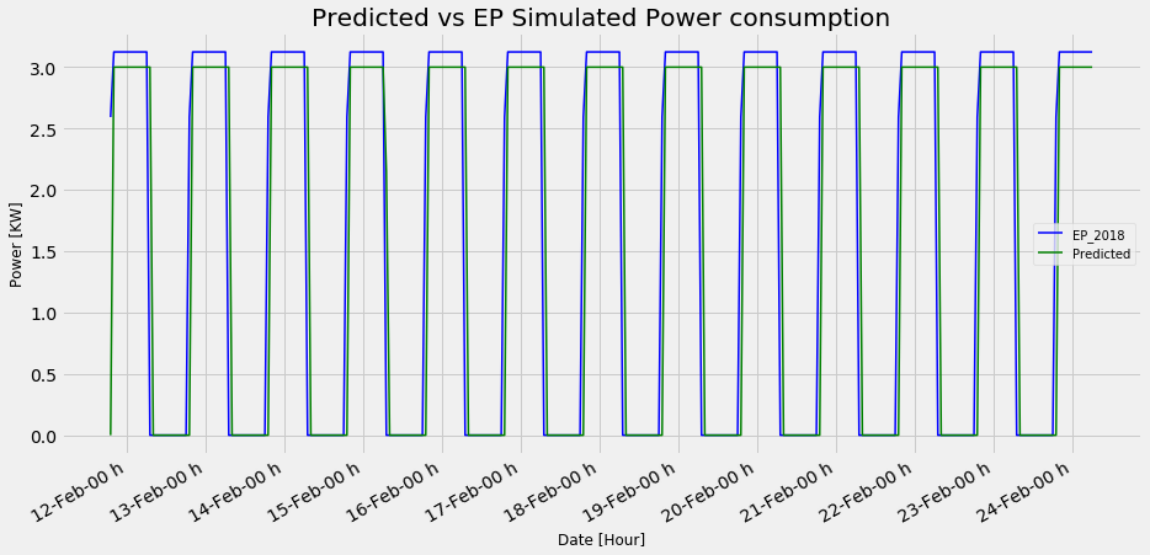

Figure 4.11 RF model performance compared against EP simulations for 2018

4.2.5 Summary

Overall the model performed well. There were some errors exceeding the limit but they are justified by the vacation month (August) false prediction especially for HVAC prediction.Table 4.4 summarizes the errors in accordance with the validation limits. Here RF model errors are comparing prediction with simulation results.

Table 4.4 All four RF model errors compared to international standards

	CV-RMSE [%] for hourly prediction			
	RF model	**IPMVP**	**FEMP**	**ASHRAE**
Facility	9.2	20	30	30
Building	9.8			
HVAC	32			
Exterior Lights	28			

CHAPTER 5

CONCLUSIONS

This work combined the numerical and predictive ML methods of forecasting energy services in the buildings. It was found out that putting an effort to collect and prepare the real weather data for EP simulations helped to get better prediction results. The predicted consumption with real weather was closer to the real values for first and third quarter of the year.

RandomForestRegressor ML model was applied to forecast the energy services based on EP simulated data. The other input to the model were metrological data for the first year and some features based on historical and engineering knowledge. The results showed that the historical data like consumption in previous hour has a close relation to the output. Feature Selection step resulted in more accurate results and saved the computational time.

Hyperparameter tuning step with using *RandomizedSearchCV* and 10-fold cross validation was implemented to obtain optimized parameters for *RandomForestRegressor* model. Although it consumed a significant amount of time but it helped in shaping the model performance.

The ML model was validated using three types of errors; MAE, RMSE (scale dependent) and CV-RMSE (scale independent). The percentage error (CV-RMSE) was compared against the limits defined by the internationally accepted organizations for the hourly building energy prediction.

The model performed significantly well with the prediction error of 15 percent of less in case of Exterior lights, Facility and Building energy consumption using the EP simulated data. Using the available actual data for total energy consumption resulted in approximately 10 percent error (even less).

The model performed somewhat strange in case of HVAC consumption prediction. The percentage error was more than the limits defined internationally (37 % for EP simulated and 32 % for actual data).

5.1 Limitations and future work

The limitations found were in the outdated models of EP which underestimated the total consumption and overestimated the HVAC consumption. EP models probably have been constructed a long time ago which therefore do not account for the changes made in equipment or the energy utilization policy of the building.

Moreover, the models were developed using an old version (v8.2) of EP software which is no more available on website. The models were updated to the latest version (v8.8) of software using a special command offered by the later versions to deal with such problems. These conversions took a considerable amount of time and may have affected the performance.

The weather data was not easily available for the whole span of experiments. Almost 15 % weather data was approximated or constructed based on the trends in data. Some entities of weather have been taken from another metrological station which affected the performance.

The suggestions for the future work are to use state of the art models for the simulations and use the latest version of software to avoid the unnecessary conversions and save the time. The other ML models, which are out of the scope of this work, can be used to compare the model performance and get better forecasting results.

REFERENCES

[1] "Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energyperformance of buildings," 2010.

[2] M. W. Ahmad, M. Mourshed, and Y. Rezgui, "Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption," *Energy and Buildings*, vol. 147, pp. 77–89, Jul. 2017.

[3] G. F. Fan, L. L. Peng, W. C. Hong, and F. Sun, "Electric load forecasting by the SVR model with differential empirical mode decomposition and auto regression," *Neurocomputing*, vol. 173, pp. 958–970, Jan. 2016.

[4] M. W. Ahmad, M. Mourshed, B. Yuce, and Y. Rezgui, "Computational intelligence techniques for HVAC systems: A review," *Building Simulation*, vol. 9, no. 4, pp. 359–398, Aug. 2016.

[5] A. Yang, W. Li, and X. Yang, "Short-term electricity load forecasting based on feature selection and Least Squares Support Vector Machines," *Knowledge-Based Systems*, vol. 163, pp. 159–173, Jan. 2019.

[6] Y. Aslan, S. Yavasca, and C. Yasar, "Long term electric peak load forecasting of Kutahya using different approaches," *International Journal on Technical and Physical Problems of Engineering*, vol. 3, no. 7, pp. 87–91, 2011.

[7] H. M. Al-Hamadi and S. A. Soliman, "Long-term/mid-term electric load forecasting based on short-term correlation and annual growth," *Electric Power Systems Research*, vol. 74, no. 3, pp. 353–361, Jun. 2005.

[8] T. Al-Saba and I. El-Amin, "Artificial neural networks as applied to long-term demand forecasting," *Artificial Intelligence in Engineering*, vol. 13, no. 2, pp. 189–197, Apr. 1999.

[9] R. E. Edwards, J. New, and L. E. Parker, "Predicting future hourly residential electrical consumption: A machine learning case study," *Energy and Buildings*, vol. 49, pp. 591–603, 2012.

[10] S. I. Hill, F. Desobry, E. W. Garnsey, and Y.-F. Chong, "The impact on energy consumption of daylight saving clock changes," *Energy Policy*, vol. 38, no. 9, pp. 4955–4965, 2010.

[11] S. S. Pappas, L. Ekonomou, D. C. Karamousantas, G. E. Chatzarakis, S. K. Katsikas, and P. Liatsis, "Electricity demand loads modeling using AutoRegressive Moving Average (ARMA) models," *Energy*, vol. 33, no. 9, pp. 1353–1360, Sep. 2008.

[12] A. Veit, C. Goebel, R. Tidke, C. Doblander, and H. A. Jacobsen, "Household electricity demand forecasting - Benchmarking state-of-the-art methods," *e-Energy 2014 - Proceedings of the 5th ACM International Conference on Future Energy Systems*, pp. 233–234, 2014.

[13] H. Takeda, Y. Tamura, and S. Sato, "Using the ensemble Kalman filter for electricity load forecasting and analysis," *Energy*, vol. 104, pp. 184–198, Jun. 2016.

[14] W. Christiaanse, "Short-Term Load Forecasting Using General Exponential Smoothing," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-90, no. 2, pp. 900–911, Mar. 1971.

[15] Y. Chakhchoukh, P. Panciatici, and L. Mili, "Electric Load Forecasting Based on Statistical Robust Methods," *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 982–991, Aug. 2011.

[16] D. J. C. MacKay, "Bayesian Non-Linear Modeling for the Prediction Competition," in *Maximum*

Entropy and Bayesian Methods, Dordrecht: Springer Netherlands, 1996, pp. 221–234.

[17] Y. T. Chae, R. Horesh, Y. Hwang, and Y. M. Lee, "Artificial neural network model for forecasting sub-hourly electricity usage in commercial buildings," *Energy and Buildings*, vol. 111, pp. 184–194, Jan. 2016.

[18] Y. Fu, Z. Li, H. Zhang, and P. Xu, "Using Support Vector Machine to Predict Next Day Electricity Load of Public Buildings with Sub-metering Devices," in *Procedia Engineering*, 2015, vol. 121, pp. 1016–1022.

[19] O. Pauly, "Random Forests for Medical Applications," *Basic and Applied Ecology*, vol. 4, no. 5, pp. 441–451, 2003.

[20] A. Troncoso, S. Salcedo-Sanz, C. Casanova-Mateo, J. C. Riquelme, and L. Prieto, "Local models-based regression trees for very short-term wind speed prediction," *Renewable Energy*, vol. 81, pp. 589–598, Sep. 2015.

[21] A. Lahouar and J. Ben Hadj Slama, "Day-ahead load forecast using random forest and expert input selection," *Energy Conversion and Management*, vol. 103, pp. 1040–1051, Oct. 2015.

[22] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[23] Y. H. Huang, "Video advertisement mining for predicting revenue using random forest," Purdue University, West Lafayette, Indiana, 2015.

[24] O. Mutanga, E. Adam, and M. A. Cho, "High density biomass estimation for wetland vegetation using WorldView-2 imagery and random forest regression algorithm," *International Journal of Applied Earth Observation and Geoinformation*, vol. 18, pp. 399–406, Aug. 2012.

[25] P. Serras, G. Ibarra-Berastegi, J. Sáenz, and A. Ulazia, "Combining random forests and physics-based models to forecast the electricity generated by ocean waves: A case study of the Mutriku wave farm," *Ocean Engineering*, vol. 189, p. 106314, Oct. 2019.

[26] L. Benali, G. Notton, A. Fouilloy, C. Voyant, and R. Dizene, "Solar radiation forecasting using artificial neural network and random forest methods: Application to normal beam, horizontal diffuse and global components," *Renewable Energy*, vol. 132, pp. 871–884, Mar. 2019.

[27] R. Feng *et al.*, "Recurrent Neural Network and random forest for analysis and accurate forecast of atmospheric pollutants: A case study in Hangzhou, China," *Journal of Cleaner Production*, vol. 231, pp. 1005–1015, Sep. 2019.

[28] A. Lahouar and J. Ben Hadj Slama, "Hour-ahead wind power forecast based on random forests," *Renewable Energy*, vol. 109, pp. 529–541, Aug. 2017.

[29] A. Lahouar and J. Ben Hadj Slama, "Random forests model for one day ahead load forecasting," *2015 6th International Renewable Energy Congress, IREC 2015*, pp. 1–6, 2015.

[30] G. Dudek, "Short-Term Load Forecasting Using Random Forests," in *Intelligent Systems'2014: Proceedings of the 7th IEEE International Conference Intelligent Systems IS'2014*, Volume 2., Warsaw, Poland: Springer, Cham, 2015, pp. 821–828.

[31] David Fumo, "Types of Machine Learning Algorithms You Should Know," *towardsdatascience.com*, 2017. [Online]. Available: https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861. [Accessed: 24-Oct-2019].

[32] Raheel Shaikh, "Feature Selection Techniques in Machine Learning with Python," *towardsdatascience.com*, 2018. [Online]. Available: https://towardsdatascience.com/feature-

selection-techniques-in-machine-learning-with-python-f24e7da3f36e. [Accessed: 24-Oct-2019].

[33] S. Salcedo-Sanz, L. Cornejo-Bueno, L. Prieto, D. Paredes, and R. García-Herrera, “Feature selection in machine learning prediction systems for renewable energy applications,” *Renewable and Sustainable Energy Reviews*, vol. 90, pp. 728–741, Jul. 2018.

[34] S. Jurado, À. Nebot, F. Mugica, and N. Avellana, “Hybrid methodologies for electricity load forecasting: Entropy-based feature selection with machine learning and soft computing techniques,” *Energy*, vol. 86, pp. 276–291, Jun. 2015.

[35] B. Remeseiro and V. Bolon-Canedo, “A review of feature selection methods in medical applications,” *Computers in Biology and Medicine*, vol. 112, p. 103375, Sep. 2019.

[36] N. AlNuaimi, M. M. Masud, M. A. Serhani, and N. Zaki, “Streaming feature selection algorithms for big data: A survey,” *Applied Computing and Informatics*, Jan. 2019.

[37] Lujing Chen, “Basic Ensemble Learning (Random Forest, AdaBoost, Gradient Boosting)- Step by Step Explained,” *towardsdatascience.com*, 2019. [Online]. Available: https://towardsdatascience.com/basic-ensemble-learning-random-forest-adaboost-gradient-boosting-step-by-step-explained-95d49d1e2725. [Accessed: 24-Oct-2019].

[38] X. Chen and H. Ishwaran, “Random forests for genomic data analysis,” *Genomics*, vol. 99, no. 6, pp. 323–329, Jun. 2012.

[39] J. A. Duffie and W. A. Beckman, *Solar engineering of thermal processes*, 4th editio. Wiley, 2013.

[40] Wes McKinney& PyData Development Team, “pandas: powerful Python data analysis toolkit — pandas 0.25.2 documentation,” *pandas.pydata.org*, 2019. [Online]. Available: https://pandas.pydata.org/pandas-docs/version/0.25/. [Accessed: 24-Oct-2019].

[41] NumPy Community, “NumPy Reference,” 2016.

[42] matplotlib.org, “Matplotlib: Python plotting — Matplotlib 3.1.1 documentation,” 2017. [Online]. Available: https://matplotlib.org/. [Accessed: 24-Oct-2019].

[43] scikit-learn.org, “An introduction to machine learning with scikit-learn — scikit-learn 0.21.3 documentation,” 2017. [Online]. Available: https://scikit-learn.org/stable/tutorial/basic/tutorial.html. [Accessed: 24-Oct-2019].

[44] scikit-learn.org, “3.2.4.3.2. sklearn.ensemble.RandomForestRegressor — scikit-learn 0.21.3 documentation,” 2017. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html. [Accessed: 24-Oct-2019].

[45] stackoverflow.com, “Python - how to implement walk forward testing in sklearn,” 2015. [Online]. Available: https://stackoverflow.com/questions/31947183/how-to-implement-walk-forward-testing-in-sklearn. [Accessed: 24-Oct-2019].

[46] C. Fan, F. Xiao, and Y. Zhao, “A short-term building cooling load prediction method using deep learning algorithms,” *Applied Energy*, vol. 195, pp. 222–233, Jun. 2017.

[47] K. L. Gillespie *et al.*, “Measurement of Energy and Demand Savings. Guide 14. ASHRAE,” vol. 8400, p. 170, 2002.

[48] IPMVP New Construction Subcommittee, “International Performance Measurement & Verification Protocol: Concepts and Options for Determining Energy Savings in New

Construction," *International Performance Measurement & Verification Protocol*, vol. III, no. April, pp. 1–249, 2003.

[49] DOE US, "M&V guidelines: measurement and verification for performance-based contracts - version 4.0," *Federal Energy Management Program*, vol. 3, no. November, pp. 1–108, 2015.