MACHINE LEARNING-BASED APPROACH FOR EV CHARGING BEHAVIOR

by

Sakib Shahriar

A Thesis presented to the Faculty of the
American University of Sharjah
College of Engineering
In Partial Fulfillment
of the Requirements
for the Degree of

Master of Science in
Computer Engineering

Sharjah, United Arab Emirates

April 2021

Declaration of Authorship

I declare that this thesis is my own work and, to the best of my knowledge and belief, it does not contain material published or written by a third party, except where permission has been obtained and/or appropriately cited through full and accurate referencing.

Signed: Sakib Shahriar

Date: 22 April 2021

Approval Signatures

We, the undersigned, approve the Master's Thesis of Sakib Shahriar

Thesis Title: Machine Learning-Based Approach for EV Charging Behavior

Date of Defense: 6 April 2021

Name, Title and Affiliation	Signature
Dr. Abdulrahman Al-Ali Professor, Department of Computer Science and Engineering Thesis Advisor	
Dr. Ahmed Osman-Ahmed Professor, Department of Electrical Engineering Thesis Co-Advisor	
Dr. Raafat Aburukba Assistant Professor, Department of Computer Science and Engineering Thesis Committee Member	
Dr. Hasan Mir Professor, Department of Electrical Engineering Thesis Committee Member	
Dr. Fadi Aloul Head Department of Computer Science and Engineering	
Dr. Lotfi Romdhane Associate Dean for Graduate Affairs and Research College of Engineering	
Dr. Sameer Al-Asheh Interim Dean College of Engineering	
Dr. Mohamed El-Tarhuni Vice Provost for Graduate Studies Office of Graduate Studies	

Acknowledgement

بسم الله الرحمن الرحيم الحمد لله رب العالمين والصلاة والسلام على أشرف المرسلين سيدنا محمد ﷺ

I would like to thank my parents, family, and friends for all their help and encouragement during my graduate program.

I would like to extend my gratitude to my advisors Dr. Abdulrahman Al-Ali and Dr. Ahmed Osman for their constant support, motivation, and valuable feedback. I truly appreciate their contribution.

I would like to thank the American University of Sharjah and in particular the department of Computer Science and Engineering for this valuable opportunity and for providing me with a partial graduate teaching assistantship. Last but not least, I would like to acknowledge Dr. Edmo Campos for providing me with the research assistantship.

Dedication

To my loved ones and to the future of Electric Vehicles

Abstract

As smart city applications are moving from conceptual models to the development phase, smart transportation, of smart cities' applications, is gaining ground nowadays. Electric vehicles (EVs) are considered to be one of the major pillars of smart transportation. EVs are ever-growing in popularity due to their potential contribution in reducing dependency on fossil fuels and greenhouse gas emissions. However, large-scale deployment of EV charging stations poses multiple challenges to the power grid and public infrastructure. The solution to this problem lies in the utilization of scheduling algorithms to better manage the growing public charging demand. Modeling EV charging behavior using data-driven tools and machine learning algorithms can improve scheduling algorithms. Researchers have focused on using historical charging data for predictions of behaviors such as departure time and energy needs. However, variables such as weather, traffic, and nearby events, which have been neglected to a large extent, can perhaps add meaningful representations, and provide more accurate predictions. Therefore, in this thesis we propose the usage of historical charging data in conjunction with the weather, traffic, and events data to predict EV departure time and energy consumption. Several popular machine learning algorithms including random forest, support vector machine, XGBoost, and deep neural networks are investigated. The best predictive performance is achieved by an ensemble-learning model, which improves upon the existing works in the literature with SMAPES of 9.9% and 11.6% for session duration and energy consumptions, respectively. In both predictions, we demonstrate a significant improvement compared to previous work on the same dataset and we highlight the importance of traffic and weather information for charging behavior predictions.

Keywords: ***Electric vehicles (EVs); charging behavior; machine learning; smart city; smart transportation.***

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACN	Adaptive Charging Network
ANN	Artificial Neural Networks
API	Application Programming Interface
ARI	Adjusted Rand Index
ARIMA	AutoRegressive Integrated Moving Average
BEV	Battery Electric Vehicles
BMM	Beta Mixture Model
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DKDE	Diffusion-based Kernel Density Estimator
DT	Decision Tree
EV	Electric Vehicle
GKDE	Gaussian Kernel Density Estimator
GMM	Gaussian Mixture Model
ICE	Internal Combustion Engine
KDE	Kernel Density Estimator
K-NN	K-Nearest Neighbor
LR	Linear Regression
PDF	Probability Density Function
PHEV	Plug-in Hybrid Electric Vehicles
PSF	Pattern Sequence-based Forecasting
ReLU	Rectified Linear Units
RF	Random Forest

RMSE	Root Mean Square Error
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MED	Mean Estimation Deviation
ML	Machine Learning
MLP	Multilayer Perceptron
NB	Naive Bayes
SMAPE	Symmetric Mean Absolute Percentage Error
SOC	State Of Charge
SVM	Support Vector Machine
SVR	Support Vector Regression
TOU	Time Of Use
UCLA	University of California, Los Angeles
UN	United Nations

Chapter 1. Introduction

In this chapter, we provide a short introduction about electric vehicles (EVs) and the growth in their popularity as well as the problems related to their charging. We also summarize the recent works utilizing data-driven approaches for solving the EV charging problem and discuss some of their limitations. Then, we present the problem investigated in this study in addition to the thesis contribution. Finally, the general organization of the thesis report is presented.

1.1. Overview

In recent years, climate change has become a growing concern. As such, the United Nations (UN) has placed combatting climate change under one of the sustainable development goals, with plans to jointly raise one hundred billion dollars by 2020 to fight the crisis [1]. Additionally, thirty-three countries jointly declared a climate emergency as of January 2021 further adding to the significance of the global crisis [2]. The transportation sector is responsible for over a quarter of the world's energy consumption [3]. According to the UN, two-thirds of the world population is projected to live in urban areas by 2050 [4], which would inherently increase the demand for vehicles to provide urban mobility, and in turn, increase fossil fuel consumption and greenhouse emissions. According to a Chinese study, electric vehicles (EVs) could potentially provide a 45% reduction in carbon emissions compared to conventional internal combustion engine (ICE) vehicles after considering the energy cost of production, assembly, transportation and usage [5]. Therefore, EVs are considered to be the frontrunners in providing a clean source of transportation.

Within the smart cities context, the massive growth in EV popularity [6] can be attributed to the rapid improvements in battery technology. The latest EVs can travel between 300-500 km per full charge, unlike the older models which would often last less than 200 km per full charge [7]. The improvement in battery technology has made EVs far more usable, not only for commuting short distances but also for inter-city travel. Consequently, the number of EV charging stations have grown, allowing greater flexibility for drivers to plan their drives. Moreover, the reliability of EVs has improved considerably since the earlier days, thereby offering greater consumer trust and satisfaction. For example, according to a case study to investigate EV user satisfaction in South Korea, it was found that cost-savings played a major role in EV satisfaction [8]. The study also concluded that user satisfaction is statistically significant for existing EV users to repurchase EVs as well as recommend the purchase of EVs to others. The overall technological improvement can also be attributed to the market competitiveness with many private and public companies taking the initiative to produce commercial EVs. In the last decade, the growth in the number of EVs worldwide has skyrocketed as shown in Figure 1.1. It provides a comparison of the number of plug-in hybrid electric vehicles (PHEV), battery electric vehicles (BEV) across China, Europe, the US and globally.

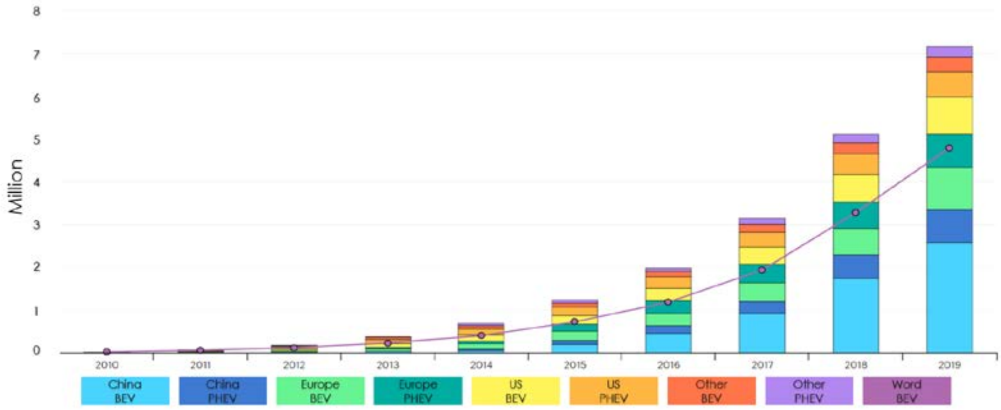


Figure 1.1: The significant increase in EV sales across the globe in the last decade [9]

Despite the promising signs, there remain a few challenges. Firstly, most EVs take a long time to charge therefore causing great inconvenience. In addition to this, many EV owners cannot charge their cars at home but must rely on public charging stations. The power requirements of EV charging is very high and often unpredictable. Munkhammar *et al.* [10] presented a case study where the integration of household-level EV charging increased the electricity use by 14-61% and the standard deviation of power consumption by 100-900%. Therefore, integrating EVs on a massive scale will place huge constraints on the power distribution grid [11]. Un-coordinated EV charging behavior is likely to cause further degradation and instability in power distribution networks. The implications of the power constraints indicate that it is virtually impossible to increase the charging station capacity to meet the growing changing needs. Unlike gas stations for ICE vehicles, where the vehicles can get refueled in minutes, EVs often require hours to recharge. The simple solution of increasing the public charging capacity by deploying more stations is not suitable because, in addition to the power needs, there also exists a physical space limitation. Therefore, the number of charging stations can only be increased to a limited number. Rather, the optimal solution is to better manage the scheduling of charging stations.

Several works have focused on scheduling to efficiently manage the charging load using optimization problems [12], [13]. In [12], multiobjective optimization was used to determine the charging station that can result in minimum charging cost as well as quickest charging and travel time. The authors in [13] on the other hand, maximized the total revenue for a parking lot and the number of EVs fulfilling their charging requirement. Such approaches can provide useful guidance for determining EV charging operations and management strategies, in terms of factors such as maximum capacity and revenue. However, day-to-day management of the charging load requires a more sophisticated approach.

Studies have shown that energy management with regards to EV charging greatly impacts the wholesale electricity market [14] as well as the overall CO_2 emissions, adding further significance to the need for understanding charging behavior. Researchers have investigated the psychological dynamics that influence charging behavior [15], [16]. The authors in [15] concluded that EV users found the charging process to be convenient overall and that most users exhibit a pattern of charging their vehicles three times every week and driving about thirty-seven kilometers every day. According to [16], most users start their charges with 40-50% state of charge (SOC) whereas about 8% of the users engage in risky charging behavior with less than 20 miles of range remaining in their vehicles. They also conclude that EV drivers are less likely to charge their vehicles in the morning and adjusting the time of use (TOU) pricing based on this could help to distribute the charging load better. Spoelstra [17] used charging transactions data and interviews with EV drivers to provide an analysis of EV charging behavior and factors that influence such behavior. For instance, it was noted that charging behavior is generally based on habits and convenience and that drivers usually connect their vehicles to the same charging ports that are known to them. Furthermore, the battery level does not influence the EV drivers' charging decisions and the charging behavior is consistent between urban and rural areas of the Netherlands, where the study was conducted. Although the outcome of these studies provides a high-level understanding of EV charging behavior, it is important to quantify the results for scheduling and management.

Bi *et al*. [18] performed an agent-based traffic simulation, where the charging stations are deployed at existing petrol stations and residential parking locations in Singapore. They concluded that charging behavior significantly impacts the simulation outcomes. In [19], a stochastic simulation is proposed to study the charging behavior over longer periods, without the need for a sophisticated transportation dataset. Analysis of charging behavior using such simulations contain assumptions that might not hold in real-world scenarios.

An estimation of EV behavior derived from ICE vehicle driving data is presented in [20] using car travel data in Sweden. The authors in [21] developed a framework for generating synthetic data for EVs using probabilistic models and nine months of GPS data collected from EVs in Ireland. Although useful for specific case studies, these datasets may not reflect the unpredictable nature of charging behavior in everyday scenario. However, synthetically generated data can be beneficial in certain situations, such as when real-world datasets are not available and when the size of the datasets are significantly smaller. Other strategies such as multi-location charging, whereby employees are encouraged to charge at home as well as the workplace, to control the charging load have shown promising results [22]. However, these approaches are only suitable in theory as it is not easy to control or enforce user charging behavior.

With the emergence of big data analytics and machine learning (ML), which have revolutionized fields of natural language processing, audio, and video recognition, the focus has shifted towards utilizing data-driven approaches [23] to solve the EV charging problem. Using historical data of charging load and user behavior, ML algorithms can be applied to learn the various trends and patterns from the data. After the training phase, accurate predictions can be obtained. Such predictions can then be utilized, either independently or in conjunction with other algorithms, to enhance EV charging scheduling strategies. Studies have shown that ML algorithms are capable of providing good forecasts for time series data [24], and can thus be used for charging behavior predictions which are time-dependent. However, the performance of an ML algorithm depends, to a large extent, on the quality of the dataset as well as the selection of input features used to train the model. Therefore, it is important to formulate the right problem and provide the models with distinguishable input features for them to provide accurate predictions.

1.2. Thesis Objectives

Driven by the recent success of ML algorithms in various domains and the great challenges in EV charging, the focus of this work will be on predicting two of the most important aspects of EV charging behavior pertaining to scheduling, namely, EV session duration and energy consumption. Accurate predictions of these behaviors can not only lead to better management of the charging demand but also provide great benefits to EV charging operators by minimizing the operating costs. In this work, we will train various supervised ML algorithms using weather, traffic, and local events data, which were not considered by previous works in the literature, along with historical charging data to improve charging behavior predictions. Although previous works have shown ML models to be effective in predicting the charging behavior, we believe that with the addition of the new variables, we can improve upon the results of the existing works.

1.3. Research Contribution

The contributions of this thesis can be summarized as follows:

- It presents a novel approach in EV charging behavior prediction that take advantage of weather, traffic, and local events data in addition to the historical charging records.
- It trains several machine learning algorithms including random forest, support vector machine, artificial neural networks and ensemble learning models for predictions of session duration and energy consumption on the adaptive charging network (ACN) dataset.
- It provides an empirical analysis with regards to the impact of using the additional features on the accuracy of predictions and highlights the improvements upon previous works.

1.4. Thesis Organization

The rest of the thesis is organized as follows: Chapter 2 provides background information about EV charging as well as the building blocks of ML and the variations of ML algorithms. Then, various regression ML models and their evaluation metrics are clearly defined. Moreover, we organize related works into three categories in Chapter 2. The first two contains ML applications for charging behavior using clustering approaches and various other charging behaviors besides session duration and energy consumption, respectively. The final part of the literature review compares the existing works utilizing ML for the prediction of session duration and energy consumption, which is the focus of this thesis. The overall methodology is clearly outlined in Chapter 3. This includes a definition of the charging behavior as well as the relevant datasets and the merging process. Furthermore, preprocessing steps including data cleaning and handling outliers is also described in this chapter. Chapter 4 presents an exploratory analysis of the dataset which illustrates the relationships between the different variables. Feature engineering steps such as encoding cyclic features are also clearly outlined in this chapter. The experimental setup is likewise explained in this chapter and further supported by a graphical representation of the proposed framework. The results obtained across various models and their comparisons are presented in Chapter 5. This chapter also provides an analysis of the results and a comparison with previous works in the literature. Finally, Chapter 6 concludes the thesis and outlines future work that could be used for further improvements.

Chapter 2. Background and Literature Review

In this chapter, we discuss various levels of EV charging and their distinctions. Then, we present an overview of ML algorithms including supervised, unsupervised, and deep learning. We also discuss the popular ML algorithms used in this work as well as some of the existing works and provide the common metrics for the evaluation of regression-based ML algorithms. Next, we provide a comprehensive literature review of the existing works that have utilized ML for the analysis and prediction of EV charging behavior. Finally, we highlight the problem statement and discuss some of the motivations behind this work.

2.1. EV Charging Standards and Categories

EV charging is primarily divided into three charging levels as defined by the Electric Power Research Institute. Level 1 charging provides the slowest charging rate, operating at standard 120 V/15A [25]. The charging equipment, in this case, is installed on the EV and the power is transferred to the vehicle using a plug and cord set. This category was popular in the earlier days, when EVs were first introduced, but are less common today. Level 2 charging, on the other hand, uses 240 VAC and has been utilized for both private and public facilities. It provides faster charging as compared to Level 1 but requires the installation of dedicated charging equipment. Level 3, also referred to as ‘fast charging’, utilizes 480 VAC and is typically deployed in commercial or public spaces to provide ‘grab and go’ service that is similar to gas stations for ICE vehicles. Level 3 provides the quickest charging rate which allows vehicles to recharge in less than thirty minutes. This is considered to be the future charging technology for EVs along with wireless charging. Figure 2.1 illustrates the three charging techniques.

Despite the emergence of fast EV charging, the number of publicly accessible fast chargers is less than half compared to slow chargers. Figure 2.2 illustrates a comparison between the two charging rates across various countries. According to [9], publicly accessible chargers make up 12% of global light-duty vehicle chargers in 2019. However, the number of both slow and fast publicly accessible chargers have grown by 60% in 2019 and the trend is likely to continue.

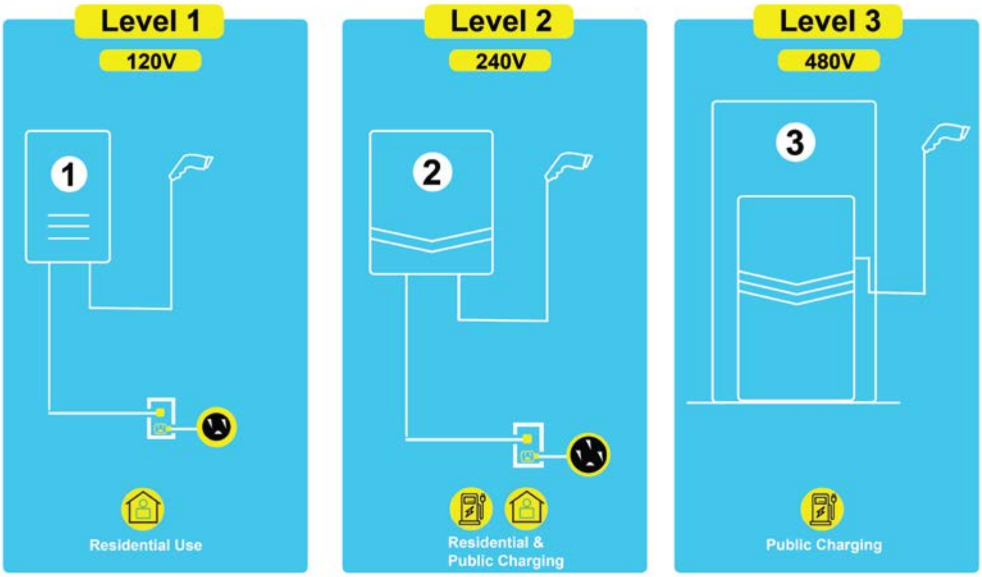


Figure 2.1: The three levels of EV charging [26]

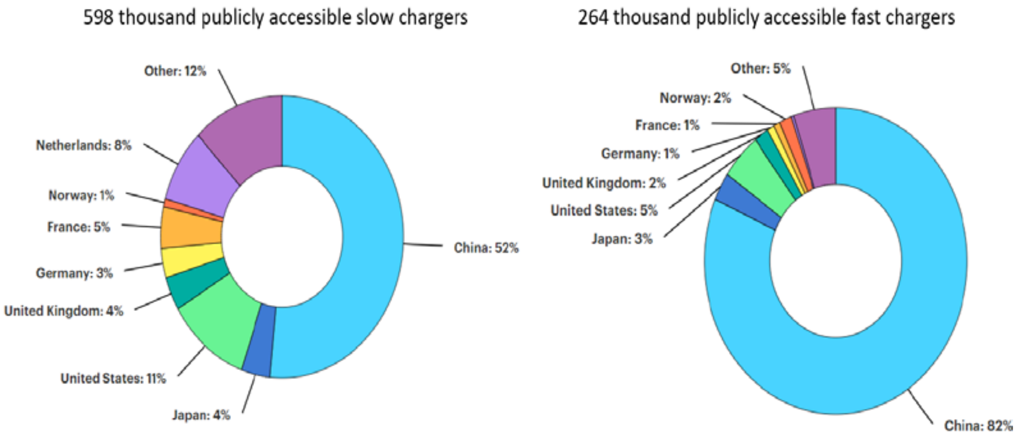


Figure 2.2: A comparison of slow (left) and fast (right) publicly accessible chargers in 2019 [9]

EV charging can also be categorized into residential charging and non-residential (commercial) charging. Typically, Level 1 and Level 2 chargers are deployed for residential purposes. Residential charging behavior is more predictable, and scheduling is therefore considered easier. In most cases, users leave their vehicles to charge overnight or arrange charging sessions depending on their working hours. The number of vehicles using residential charging is also predictable because usually, people who own EVs in a given area is likely to utilize the stations within that residential area. As a result of user flexibility, a quicker rate of charging is often not required because the charging can take place whenever the car is parked. This allows the operators in residential charging to distribute the charging load evenly based on the TOU. In contrast, a non-residential charging station is unpredictable and depends on a lot of factors. There is also an expectation for a faster charging rate as users may not necessarily spend longer hours outdoors. Also, take for instance the example of a public charging facility near a shopping mall. The charging traffic in this case will depend on a lot of factors including weather, day of the week and mall offers. Therefore, it is more significant to understand the charging behavior of non-residential charging facilities, which are dynamic in nature, to provide more precise scheduling.

2.2. Machine Learning and Predictive Analytics

Machine learning (ML) provides computer systems with the ability to learn from experience without the need for explicit programming. The experience in this context is the dataset that the algorithms used to train themselves on. With time, the models can discover the underlying trends and patterns in the dataset. Therefore, the quality of the dataset in terms of the data being clean and the input features being relevant is highly significant. Upon successful learning, these models can make accurate predictions and therefore provide predictive analytics. ML algorithms are typically categorized into supervised and unsupervised learning. Further categorization can be done depending on the type of the variable to be predicted, also known as the response variable. If the response variable is continuous, the problem being solved is called a regression problem. Conversely, if the response variable is categorical, the problem is called a classification problem. Figure 2.3 illustrates the difference between regression and classification in the context of EV charging. The figure on the left portrays the prediction of energy consumption based on charging session duration. This is a regression problem because the response variable, energy, is a continuous value. In contrast, the figure on the right portrays the distinction of EV drivers who prefer to charge their vehicles during nighttime against those who prefer to charge their vehicles during the day. In this case, it is a classification problem because the variable of interest is categorical.

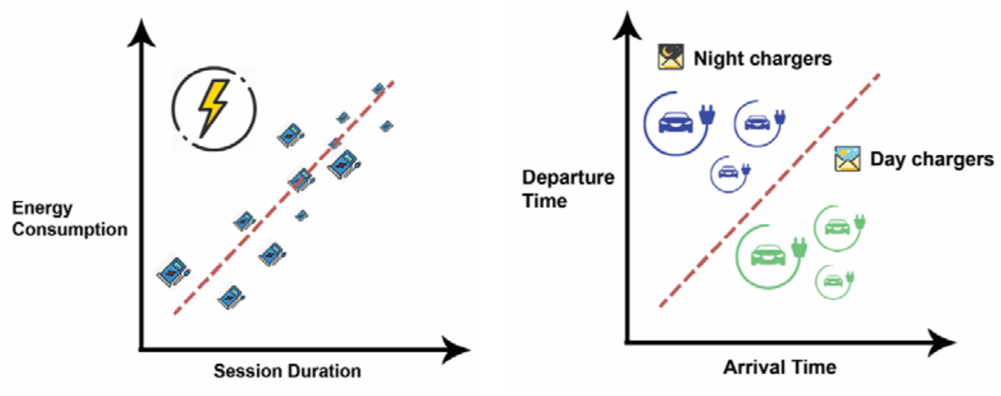


Figure 2.3: Illustration of regression problem (left) and classification problem (right) [26]

2.2.1. Supervised learning. In supervised machine learning, models are trained from the labelled training dataset. As such, the dataset contains both the input variables and the corresponding response variable, often called the target variable. The model iteratively learns the mapping between the input and the response variables by optimizing a given objective function. A simple example in the context of EV charging is a dataset containing the arrival time of the vehicle, the name of the city, and the departure time of the vehicle. If the goal is to predict the departure time, the ML model will learn the relationship between the arrival time, the name of the city (input variables) and the departure time (response variable). A discussion of all the supervised learning algorithms is beyond the scope of this work. However, the algorithms used in this work as well as some of the ones used in the related works for the prediction of EV charging behavior is provided next.

2.2.1.1. *Linear Regression.* Linear regression (LR) can be used to model the mathematical relationship between the output variable and one or multiple input variables (multiple LR). In LR, it is assumed that there is a linear relationship between the response variable and the input features. LR can be represented by Equation 1:

$$y = b_o + b_1 x_1 + b_2 x_2 + \cdots + b_n x_n \tag{1}$$

where y represents the target variable, b_o represents the y-intercept, $[x_1, x_2 \cdots x_n]$ represents the input features and $[b_1, b_2 \cdots b_n]$ represents the regression coefficients. Gradient descent [27] method is often used to find the coefficients by minimizing the sum of the squared error iteratively after starting with random values for the coefficients. As the name suggests, LR is used for regression problems and is particularly useful when the dataset is linearly separable. Moreover, the algorithm itself is very simple to implement and explain. Overfitting [28] is a major challenge in training most ML algorithms. Overfitting occurs when a given model performs exceptionally well during the training phase, often by using unnecessary input features, but fails to make generalized predictions. The performance of LR can be impacted by overfitting as well as the presence of outliers. Another drawback of LR is that it assumes the data to be independent of each other and this is not always true. Therefore, LR is generally used only as a baseline model for complex problems.

2.2.1.2. *Random forest.* The term 'forest' comes from the fact that this algorithm is composed of multiple trees. A decision tree (DT) can be used for both classification and regression problems. Similar to a flow chart, DTs separate complex decisions into a combination of simpler decisions using split points from the input features. The points where decisions take place are called decision nodes. The points where no further split is made are called the leaf nodes. For regression problems, the average value of all the items in the leaf node is taken for prediction. For classification problems, the leaf nodes are the set of classes being predicted, and the class obtained in a leaf node is taken as the final prediction. DTs are simple to explain, usually by plotting a tree diagram which can help in understanding the prediction making process. However, a single DT often fails to provide good predictions and is prone to overfitting. In a random forest (RF) algorithm, predictions are made by aggregating multiple decision trees. Bagging method is used in this case where the trees are created from the various bootstrap sample, i.e., sample with replacement. For regression, the average predictions of the trees are taken and for classification majority vote across the trees are taken [29] as final predictions. RF is an example of ensemble ML, where individual ML models are first evaluated and then integrated into a single model that can often produce superior predictive performance than the individual models. The motivation behind such an approach is similar to asking multiple experts about an opinion, and then taking their votes to make the final decision [30]. An RF model greatly reduces overfitting as compared to a simple DT model. Moreover, both DT as well as RF can be used for classification and regression problems, unlike LR. However, significantly more computational resources are needed for RFs as compared to DTs because a large number of DTs are needed for building an RF. As a result, the training time is also considerably longer. Furthermore, unlike both LR and DT, the results obtained using RF is difficult to interpret because it is difficult to visualize the decision-making process due to the model complexity.

2.2.1.3. ***Gradient boosting.*** Similar to an RF, a gradient boosting algorithm [31] uses multiple DTs. However, in this algorithm, each tree is built sequentially and as a result, the errors made by previous trees are taken into consideration, which often leads to superior performance. In contrast, RF is an ensemble of trees that are considered independent of each other. XGBoost [32] is a more recent variation of the gradient boosting algorithm. XGBoost has gained popularity over the last few years due to its success in machine learning competitions, mainly because of its effectiveness in dealing with the bias-variance tradeoff [33]. This means that the algorithm can avoid overfitting on the training data while at the same time, maintaining enough complexity to obtain meaningful representations. As the trees are built sequentially, the training time for gradient boosting algorithms are usually longer than RFs. Also, gradient boosting algorithms are generally more prone to the presence of outliers and can overfit more compared to RFs.

2.2.1.4. ***Support vector machine.*** A support vector machine (SVM) [34] is mainly used for classification problems, but can also be used for regression in which case, they are often referred to as support vector regression (SVR) [35]. SVM uses the maximum margin algorithm concept, where the goal is to find the perfect hyperplane that can maximize the margin between the respective classes. Using kernels such as linear, polynomial, and radial basis function (RBF), the inputs can be mapped to high dimensional feature spaces where they can be linearly separable. The use of different kernels allows the model to find solutions to a wide range of complex problems. Also, the use of a regularization parameter ensures greater generalization compared to other models. However, SVMs are not suitable for larger datasets due to their long training time. It is also difficult to interpret the results obtained using SVMs.

2.2.1.5. ***K-Nearest Neighbor.*** Although k-nearest neighbor (K-NN) [36] algorithms can be used for both regression and classification, they are more popular for classification problems. For K-NN, a dedicated training phase is not required, and it is also known as a form of lazy learning. For predicting a new data point, a distance measure, typically Euclidean distance, is used to find its *k* nearest neighbors. Then it is assigned to the class that constitute a majority of the neighbors. Figure 2.4 illustrates this process where *k* is set to three and therefore can be also called a 3-NN algorithm.

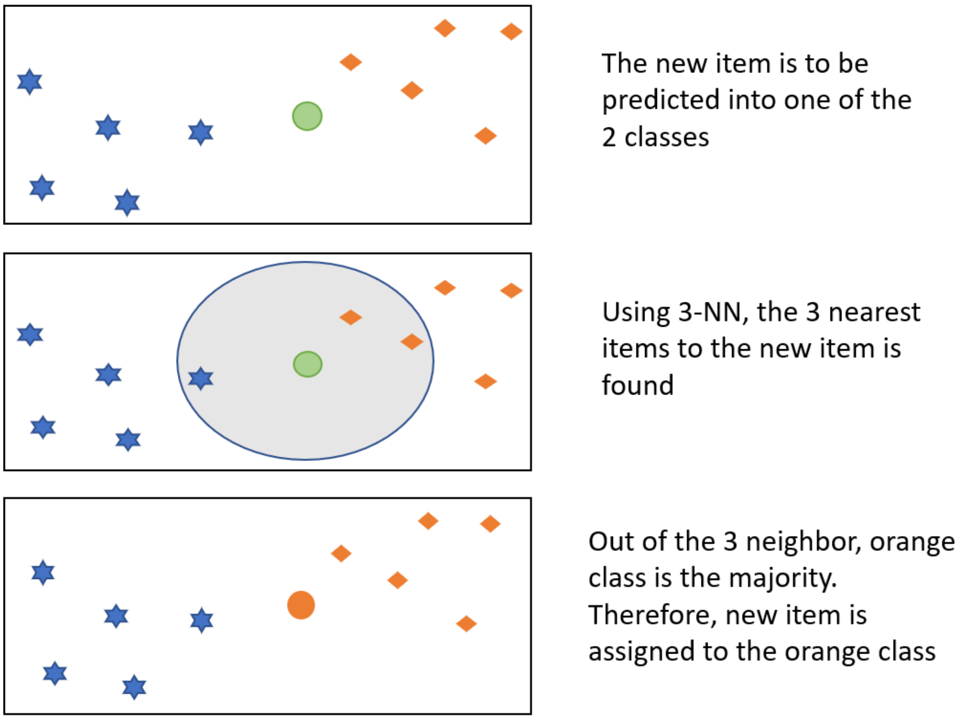


Figure 2.4: Illustration of new member assignment in the k-NN algorithm using three neighbors [26]

In the above example, assuming two initial classes, orange and blue, the task is to classify the new item into one of the classes. Assuming 3-NN, Euclidean distance can be used to find the three nearest neighbors to the new item. The three neighbors of the new point are the two orange points and one blue point. Therefore, the new item will be assigned to the orange class as it constitutes the majority. In regression, the average value of the neighboring points is taken as the prediction value. K-NN has numerous advantages as compared to the aforementioned algorithms mainly due to it not requiring dedicated training. As a result, it is faster to implement and new data points can easily be added without retraining the model. However, K-NN is usually not recommended for larger datasets due to the growing cost of calculating the distance between points. Moreover, the algorithm is highly sensitive to noisy data and outliers and requires feature scaling such as normalization.

2.2.2. Ensemble learning. In ensemble learning, a set of individually trained classifiers are combined and then used to predict new instances, often providing more accurate predictive performance than the individual classifiers [37]. Figure 2.5 illustrates the concept of ensemble learning. Both RF and XGBoost are examples of ensemble learning, where individual models (in these cases DTs) are first evaluated and then integrated into a single model. The motivation behind such an approach is similar to asking multiple experts about an opinion, and then taking their votes to make the final decision [30]. The main advantage is that ensemble learning models generally do not suffer from overfitting because they can greatly reduce the model variances. However, ensemble learning is computationally very expensive and can be limited by memory constraints and long training times. This problem is often solved by utilizing parallel computation, whereby the base models are trained independently across various machines to speed up the training process. Furthermore, ensemble learning models suffer from a lack of interpretability as it is difficult to explain the predictive process when many base models are involved.

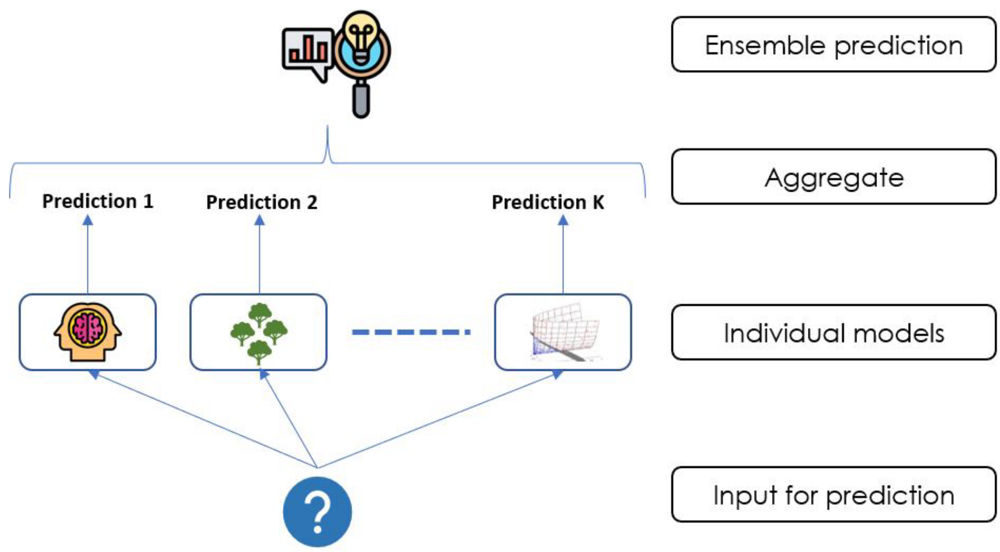


Figure 2.5: Illustration of ensemble learning

2.2.3. Unsupervised learning. In unsupervised learning, the training dataset is comprised of only input variables, without labeled output variables. The goal of the ML model is to find structure within the dataset. Cluster analysis is a common example of unsupervised learning, whereby the objective of the ML model is to find clusters of items that have some common elements between them. Unsupervised learning can be utilized to find clusters of EV behavioral patterns. Figure 2.6 provides an illustration of clustering in the context of EV charging. Based on the arrival time and day of the week, we notice three distinct clusters or groups of charging behavior. The main advantage of unsupervised learning is that labeling the data is often costly and requires manual work, whereas unsupervised learning does not require labeling of the response variable. Furthermore, interesting patterns can be automatically discovered using this approach as the model is only trained to find structure within the dataset and not necessarily make a specific prediction. However, not being able to make specific prediction can be a disadvantage in various applications which require precision. Finally, the learning process is slower in many cases because the algorithm calculates various possibilities of organizing different points in the dataset.

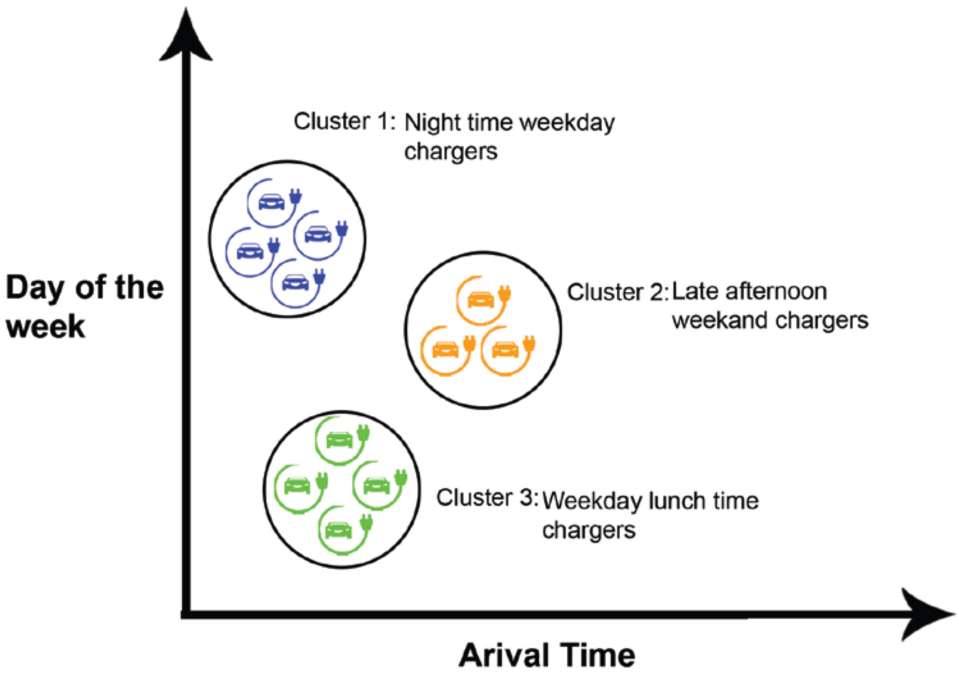


Figure 2.6: A Simple illustration of clustering in the context of EV charging behavior [26]

2.2.3.1. ***Clustering algorithms.*** In k-means clustering, individual data points form k clusters, with each point being assigned at the beginning to k center points in a random manner. Next, the data points are reassigned to the closest center based on new center calculations. The number of clusters must be known beforehand or can be calculated best on the elbow method [38]. K-means is a simple clustering algorithm that is sensitive to outliers and initial assignment. K-means is among the most popular clustering algorithms along with Density-based spatial clustering of applications with noise (DBSCAN) and hierarchical clustering.

2.2.3.2. ***Gaussian mixture model.*** Gaussian mixture model (GMM) [39] is a probabilistic learning model that can represent normally distributed subpopulations by considering multiple normal distributions of the dataset. Although GMMs are mainly used as unsupervised learning, variations of them exist for supervised learning. For the unsupervised model, prior knowledge of the subpopulations is not required. Based on the distributions of the dataset, such as binomial, Poisson, and exponential, various forms of mixture models can be derived. Other common mixture models include the beta mixture model (BMM), where the beta probability distribution is considered. Unlike k-means, GMMs can produce non-convex clusters which are usually more accurate. GMMs also allow mixed membership of the data points, i.e., a point in the dataset can belong to more than one cluster, which is not possible in other clustering algorithms. This can be useful in specific applications depending on the requirements.

2.2.3.3. ***Kernel density estimator.*** The shape of the probability density function (PDF) must be assumed in parametric estimation methods. When this is not possible, a nonparametric estimation can be used to estimate the PDF of a continuous random variable using kernel functions. Kernel functions must be symmetrical, nonnegative, and the area under the function curve must be equal to one. Popular kernels for KDE include normal or gaussian KDE (GKDE) and diffusion-based KDE (DKDE) [40].

2.2.4. Deep learning. Deep learning is a subset of ML that utilizes artificial neural networks (ANNs). Deep learning models, unlike ML models, contain a large amount of composition of learned functions. More specifically, using a layered hierarchy of concepts, complex concepts are defined in terms of simpler concepts and more abstract representations are gathered using less abstract ones [41]. Although deep learning is an emerging technology, it dates back to the 1940s. It was known by various names, such as connectionism and cybernetics, during earlier days. The recent success of deep learning-based models can be attributed to two main factors: 1) Availability of larger datasets to train DL models. 2) Availability of powerful computers to build and train complex models to achieve groundbreaking results [41]. Currently, deep learning-based models provide cutting-edge solutions to various areas in natural language processing, audio classification and computer vision.

Among the most common deep learning methods is the multilayer perceptron (MLP), which is often simply referred to as ANN. MLP uses non-linear approximation given a set of input features and MLPs can be applied for both regression and classification problems. An MLP consists of an input layer that takes in the set of features, the hidden layers that learn the representations, and the output layer that computes the final predictions. As the number of hidden layer increases, the depth of the network grows and consequently the overall complexity. Figure 2.7 shows an MLP with three hidden layers for binary classification, where the response variable belongs to either one of the two classes.

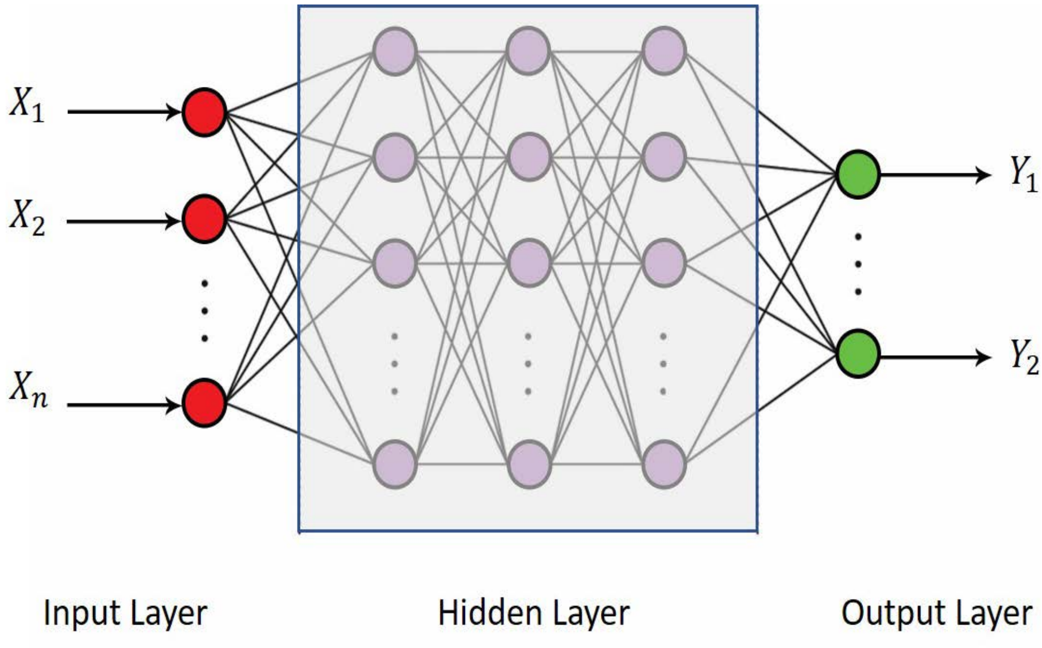


Figure 2.7: ANN with three hidden layers [26]

2.2.5. Evaluations of regression models. To assess the performance of ML models and perform comparative studies, it is important to define quantitative metrics for model evaluation. In this work, regression models will be considered due to the nature of the problem being solved and therefore we present four common metrics used for the evaluation of regression models. Assuming that the original value is represented by y and the predicted value is represented by $\bar{y}$, the average of the actual values is represented by μ and n represents the groups of values in the dataset, then the following methods (Equations 2-5) are commonly used to evaluate the performance of regression models:

Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}{n}} \tag{2}$$

Mean Absolute Error (MAE):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \bar{y}_i| \tag{3}$$

Coefficient of determination or R^2:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}{\sum_{i=1}^{n}(y_i - \mu)^2} \tag{4}$$

Symmetric Mean Absolute Percentage Error (SMAPE):

$$SMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \bar{y}_i|}{(|y_i| - |\bar{y}_i|)/2} * 100 \tag{5}$$

If the predicted value, $\bar{y}$ is very different from the actual value y, the result of all of these metrics except R^2 will be high. Generally, lower scores of RMSE, MAE and SMAPE indicate accurate predictions, and this occurs when the predicted value, $\bar{y}$ is very close to the actual value y. The R^2 value is a measure of goodness of fit for regression and is usually a score between 0 and 1. A score of 1 indicates perfect predictions and generally, a higher value represents better performance. We do not consider mean absolute percentage error (MAPE) because it is inconvenient when the actual value y is close to zero, therefore creating a bias. Rather we consider SMAPE which is more suitable for EV charging prediction since both the original and the predicted values are in the denominator [42]. In RMSE, large errors are given higher weights and therefore it is suitable for applications where large errors are specifically undesirable. MAE, on the other hand, is much easier to interpret.

2.2.6. EV charging datasets.

2.2.6. EV charging datasets. The success of a good predictive ML model depends on the quality of the dataset. This section will go over the commonly used datasets for studying EV charging behavior. Two EV charging datasets were presented in [43], one of them containing about 8500 residential charging sessions and the other dataset containing more than one million sessions from a public charging facility in the Netherlands. The residential dataset contains charging data spanning for a year (March 2012-March 2013) along with the trip details of EVs using a GPS logger. This provides a good platform for studying residential EV driving and charging behavior. The non-residential dataset spans from January 2011 until December 2015 and was collected by ElaadNL. My Electric Avenue [44] consists of the driving and charging behavior of UK drivers from January 2014 to November 2015. Residential water and energy data collected by Pecan Street's water and electricity research is hosted by Dataport [45]. Although this dataset is available to the public for research, it is limited to residential EV charging behavior. ACN-Data [46] is among the most recently released public dataset on EV charging, containing more than thirty thousand charging sessions collected from two non-residential charging sites in California. Additional user data such as estimated departure time and requested energy is collected through user mobile application by scanning a QR code. When a user does not use the mobile application, default values are generated for these fields, without attaching user identifier for such sessions.

2.3. Related Work

This section provides an overview of the existing works utilizing ML for analysis and prediction of EV charging behavior. The focus of this work is on predicting two specific charging behavior, namely, the session duration and energy consumption due to their usefulness for EV scheduling. These fall under the category of supervised machine learning. Nonetheless, we will present a brief review of unsupervised learning approaches as well as other types of EV charging behavior, such as charging rate and fast charge usage.

2.3.1. Unsupervised learning approaches. GMM was used in [47] to find thirteen distinct clusters of charging behavior for non-residential charging. Charging sessions containing information about start time, connection duration, the distance between two sessions and hours between sessions were considered as features. The distinction between daytime and overnight charging was found to be the largest distinction between the types of charging sessions. The clustering result was evaluated using adjusted Rand index (ARI) for clusters 7-13, with all except one (ARI of 0.54) ARI value being below 0.6, and therefore indicating good agreement in general. GMM was also used in [48] to create EV profiles that captured charging behavior by considering several charging events, start time and SOC. The EV charging profiles generated were then validated with average charging demand. Flammini *et al*. [49] used charging transaction data and developed a BMM to represent the multi-modal probability distributions of variables such as connection time and idle time. The proposed model showed a good fit when compared to the empirical data graphically. Additionally, the following conclusions were made after analysis: 25% of the total energy is supplied in the weekend, significant differences were noticed for plug-in and plug-out profiles among weekdays and weekends, 50% of the recharges last for less than four hours, and the idle time on average lasts for four hours. While these results provide good insight into charging behavior, the proposed model was not validated for predictive performance. DBSCAN clustering was used in [50] to find three clusters of EV charging behavior based on arrival and departure times. The first cluster, named charge near home, contained sessions with most arrival times during afternoon and evening and most departures the next morning. The second cluster, named charge near work, contained sessions with arrivals in the morning and departures in the evening. The final cluster, named park to charge, contained sessions scattered throughout the day with short idle time (i.e., they charge quickly and leave). The authors also provided qualitative and graphical analysis using violin plots to explain charging behavior but failed to provide a performance evaluation of clustering. The work in [51] used k-means clustering with Euclidian distance cost function to categorize user charging behavior into four groups, using mean and standard deviation of arrival and departure times as well as the Pearson correlation coefficient between stay duration and energy consumption. The authors did not provide an evaluation for clustering, but the labels generated by clustering were then used by ANN to classify user behavior. A similar approach was presented in [52] where k-means clustering was used to find three clusters of charging behavior using Euclidean distance measure. The cluster evaluation was not performed, but the results were used by the K-NN algorithm for classification and the accuracy of classification was 97.9 with the area under a ROC curve value of 0.994.

The authors in [53] used spectral clustering with Euclidian distance with a penalty term to cluster charging tails, which are the start and end times for EV battery absorption stage portion of the charging curve. The number of clusters was selected to be 6 after comparing the silhouette scores. The clusters were then transformed into two-dimensional space using the t-distributed stochastic neighbor embedding for visualization. Gerossier *et al.* [54] used hierarchical clustering with the Ward linkage method to understand the EV charging behavior of forty-six residential EVs. The clustering result indicated four groups of behaviors which were night and morning chargers (which makes up more than 50% of the sample), evening chargers when people usually return from work, charging sessions scattered throughout the day, and late evening chargers. The performance of clustering was not evaluated, but the results were used to forecast the charging load using an RF model that achieved comparable results (MAE of 4.9 kW) to the benchmark gradient boosting method. An expectation maximization algorithm was used in [55] to find four clusters of charging behavior. Then, a mixture model was used to predict EV behavior and simulation results showed that as prediction error increases, the cost reduction and savings decreases. In [56], k-means clustering was used to find patterns in EV charging profiles of three UK counties. To find the optimum number of clusters, Davies–Bouldin evaluation criterion was used. Although plots were used to display cluster centroids and provide a graphical analysis of daily charging demand, the paper did not provide a performance evaluation of the clustering. The results obtained from clustering of charging behavior can be helpful in the operation and management of charging station. For instance, they can provide a comparison of the various charging groups and help the operators determine the peak and non-peak charging hours. Consequently, the operators can utilize the TOU to cut down on electricity costs and maximize profit. However, the results obtained from these approaches cannot directly be integrated into the scheduling framework, which is crucial for managing the charging load as discussed previously.

2.3.2. Other charging behaviors. Although we are considering the predictions of session duration and energy consumptions in this work, it is worth pointing out some of the other examples of charging behavior that can be predicted using ML. In [57], the authors used ensemble models including RF, naive bayes (NB) and ANNs to predict whether or not the EVs will be charged the next day in a household. The hours of the day in which the EVs will be charged the next day is also predicted. Among the input features used for the predictive models included charging consumption of the previous day and charging occurrence time of the previous day. The combination of RF, NB, AdaBoost, and gradient boosting algorithms provided the best performance achieving a true positive rate of 0.996 for predicting whether the EVs will be charged the next day. They also obtained an accuracy of 0.724 for predicting the hours of the next day when the EVs will be charged. Binary logistic regression was used in [58] to classify whether or not the driver will make use of fast charging in a given day. Features such as travel time duration, driving speed, temperature, and whether the driver's last trajectory included fast charging was used to develop the predictive model. The proposed model achieved superior performance compared to an LR model with an overall prediction accuracy of 0.894. Additionally, the following conclusions were drawn; drivers are more likely to fast charge with increased travel duration and travel distance, and drivers who exhibit fast charging habits are more likely to use fast charging on their next day trajectories. Venticinque *et al*. [59] used SVR with RBF kernel to predict the time to the next plug. Using residential charging data, the best performing model achieved an MAE of 0.124 minutes and RMSE of 0.158 minutes. In [60], a dataset consisting of charging processes, i.e. time-series data of charging power, was used from a workplace to predict charge profiles. The best performance was achieved by XGBoost model with an MAE of 126 W outperforming ANN and LR models. The result of this approach when integrated to form schedules resulted in up to a 21% increase in energy charge for the EV. Mies *et al*. [61] used LR to model charging speed by considering variables such as temperature, connection time, and SOC. Some variables such as temperature were found to impact charging speed. For instance, it was noted that an increase in one degree Celsius resulted in a charging speed increase of 3.7 W. The model was analyzed graphically and statistically, and while the charging speed is very relevant in terms of predicting the departure time, the study failed to consider the predictive performance of the model. The authors in [62] used an RF model to predict charging capacity and the daily charging times. The proposed model outperformed SVR on the training set and achieved MAPE of 9.76% on prediction of the charging load for the next fifteen minutes for a single station. For a group of charging stations, the proposed RF model once again outperformed both SVR and DT models with a MAPE of 12.8%. Feature importance analysis showed that previous day's charge was the most important predictor. A time series-based forecasting method, autoregressive integrated moving average (ARIMA), was used in [63] to predict the parking lot charging demand using expected arrival and departure times. The proposed model, which decouples the daily charging demand of EV parking area from the seasonally changing load profile, outperformed regular ARIMA achieving MAPE of 1.44%. Time series-based forecasting methods rely solely on the historical data of the response variable and do not consider other features for prediction. The predictions are made by interpolating the general trend of the past values. For a detailed review of all aspects of EV charging behavior, the readers are encouraged to refer to the survey paper in [26]. Although useful for various applications, these types of charging behavior cannot directly be used by a scheduler to manage the charging load. Next, we will provide a detailed review of the related works that focused on the prediction of session duration and energy consumption using supervised ML.

2.3.3. Supervised learning for predictions of session duration and energy consumption. As will be defined in the following sections, session duration is directly related to the departure time. It is the departure time minus the arrival time, which is a known variable. Therefore, one can assume the prediction of either the session duration or the departure time to have the same application. Lee *et al*. [46] introduced a novel dataset for non-residential EV charging consisting of over thirty-thousand charging sessions. They used GMM to predict session duration and energy needs by considering the distribution of the known arrival times. The testing dataset included the month of December 2018 and the reported SMAPEs were 14.4% and 15.9% for the session duration and energy consumption, respectively. Result comparisons show that the GMM predictive model achieved significantly greater predictions when compared to user inputs, where the users were asked to estimate their departure times and energy needs. In this work, only historical charging data was considered for obtaining the predictions. The paper also presented other useful application such as integrating solar generation into workplace charging. In [64], the authors utilized an SVM model for the prediction of arrival and departure time for EV commuters. The dataset used for training consisted of three years (2012-2014) of charging data of commuters using EVs in the University of California, San Diego campus. Using historical arrival and departure times and temporal features i.e., week, day, and hour, the reported MAPEs were 2.9% and 3.7% for arrival and departure times, respectively. The proposed model demonstrated superior performance against a simple persistence reference forecast. The paper failed to address SVM hyperparameter tuning which can often enhance predictions [65]. Frendo *et al*. [66] predicted the departure time of EVs using regression models. The models were trained on historical data containing over one-hundred thousand charging sessions spanning over three years. Eight input features were used including, car ID, car type, weekday, charging point, car park location, parking floor, and arrival time. For prediction, three regression models were trained namely, linear regression, XGBoost, and ANN. XGBoost achieved the best results with an MAE of eighty-two minutes. The predictions made by the ML models had a significant impact on scheduling quality. In [42], several ML models, including DT, K-NN, and RF was utilized to predict session duration and energy consumption from two charging datasets. The first dataset contained charging sessions from the University of California, Los Angeles (UCLA) campus, thus representing non-residential charging behavior. The second dataset represented residential charging data from EV drivers in the UK. For session duration, SVR performed the best (SMAPE 10.54%) followed by LR (SMAPE 11.05%). As for energy consumption, RF performed the best (SMAPE 8.65%) with DKDE a close second (SMAPE 8.73%). Based on the preliminary results obtained by various models, the authors selected SVR, RF, and DKDE to form an ensemble model. The proposed ensemble model outperformed the individual models in both predictions. The SMAPE for charging duration was 10.4% and the SMAPE for energy consumption was 7.5%. The results from the proposed model when applied to a scheduling algorithm not only reduced peak load by 27% but also reduced charging cost by 4%.

Xiong *et al*. [67] predicted the start time and session duration using mean estimation. Session duration was then used to obtain energy consumption predictions using LR. The charging behavior predictions were integrated to flatten the charging load profile and stabilize the power grid. However, the predictions were not evaluated quantitatively and therefore does not indicate how well the proposed LR model functions. In [68], several regression models were used to predict the energy requirements using public charging stations data for the US state of Nebraska. Besides historical charging data, parameters such as season, weekday, location type, and charging fees were used as input features. On the test set, XGBoost outperformed LR, RF, and SVM obtaining an R^2 value of 0.52 and an MAE of 4.6 kWh. The authors in [69] used K-NN to predict energy consumption at a charging outlet using data from a university campus. The problem was formulated as a time-series forecast, whereby energy consumption prediction for the next day (next 24 hours) was made using energy consumption of previous days. The highest SMAPE was 15.3% using a *K* value of 1 (1-NN) and a time-weighted dot product dissimilarity measure. The predictive model was integrated into a cell phone application that can predict the end time of charging and the available energy in about one second. Similarly, Majidpour *et al*. [70] also predicted the next day energy needs of a charging station based on previous days energy consumption using various algorithms including SVM and RF. This work also experimented with pattern sequence-based forecasting (PSF) [71], where clustering is first applied to classify the days and predictions are consequently made for that day. In this approach, clustering ensures that the predictive algorithm takes into account similar data points and consequently learn the patterns withing each cluster. PSF and SVR methods achieved the best performance on the UCLA dataset, with the current hour and previous hour energy being the most significant input variables. However, it must be noted that the SVR took a significantly longer time for hyperparameter tuning. The PSF-based approach provided the most accurate results with average SMAPE value of 14.1%. Khaki *et al*. [72] used a non-parametric approach to predict session duration and energy consumption. They used historical charging data from the UCLA campus which was collected over twenty months. Using a graphical plot of mean estimation deviation (MED) for comparison, the proposed DKDE method was superior compared to GKDE. The prediction results minimized load variance and charging cost. However, the prediction results were not quantified and therefore cannot be used to assess the quality of the predictive model. A Hybrid estimator that uses both GKDE and DKDE was proposed in [73] to predict charging session duration and energy consumption. A combination of the UCLA charging dataset and UK driver's charging data from My Electric Avenue was used for training. Comparison of MED shows that the accuracy of prediction is better using the hybrid model as compared to the individual models, with the reported MED being 0.75 hours for stay duration and 1.68 kWh for energy consumption. Table 2.1 provides a summary of the related works in the literature, specifically in the context of session duration and energy consumption predictions using ML algorithms.

Table 2.1: Supervised learning for session duration and energy consumption

Source	Prediction	Model	Input Features	Results
[46]	Session length, energy consumption	GMM	Historical charging data	SMAPE: 14.4% duration, 15.9% consumption
[64]	Arrival time, departure time	SVM	Historical charging data	MAPE: 2.9% arrival, 3.7% departure
[66]	Departure time	XGBoost	Historical charging data, vehicle type, charging location	MAE: 82 minutes
[42]	Session length, energy consumption	Ensemble model of SVM, RF & DKDE	Historical charging data	SMAPE: 10.4% duration, 7.5% consumption
[67]	Start time, session length, energy consumption	Linear regression	Historical charging data	-
[68]	Energy requirements	XGBoost	Historical charging data, season, weekday, location type, charging fees	R^2: 0.52, MAE: 4.6 kWh
[69]	Energy consumption	k-NN	Last few days energy consumption	SMAPE: 15.3%
[70]	Energy consumption	PSF	Last few days energy consumption	SMAPE: 14.1%
[72]	Session length, energy consumption	DKDE & GKDE	Historical charging data	-
[73]	Session length, energy consumption	Hybrid KDE	Historical charging data (combination of US & UK datasets)	MED: 0.75 hours duration, 1.68 kWh consumption

2.4. Motivation and Problem Statement

Although the above works from the literature have successfully applied ML for the prediction of session duration and energy consumption, they have mainly focused on utilizing historical charging data. In some cases, additional features such as vehicle information, charging location information, and seasonal information were used. This has motivated us in this work to investigate the use of additional input features including weather, traffic, and local events and consequently observe its impact on the accuracy of charging behavior predictions.

The motivation for improving the performance of predictive models, with regards to EV charging behavior, results from the significant impact these models can have on EV charging operations. In [42], the predictions of stay duration and energy consumption obtained using the ensemble model is utilized by an optimal scheduling algorithm which reduces charging cost and minimizes the load variance. During high TOU, the electricity price is higher, so the algorithm performs peak load shaving and conversely during low TOU, the algorithm performs valley filling to take advantage of the low electricity cost. Numerical simulation applied to real-world data showed that by using predictions made by ML, peak load was reduced by 27% and charging cost was reduced by 4%. The load and cost reduction was in comparison to uncoordinated EV charging, in which case EV load profiles were not considered with the TOU. Moreover, coordinated charging using ML predictions and real data both resulted in similar cost reductions, indicating the quality of predictions to be accurate. Similarly, [55] considered the wholesale electricity price to calculate energy savings and cost reduction. The authors compared the impact of prediction error on cost savings and concluded that as prediction error increases, the cost reduction and saving decreases. They also performed a simulation to study the impact of ML prediction and cost savings for EV charging. Using the simulation results from [55], we have produced the plots shown in Figure 2.8 that highlight the significance of accurate predictions made by ML models. Therefore, a key motivation of this work is to develop ML models that can improve upon existing works and consequently lead to EV charging cost savings.

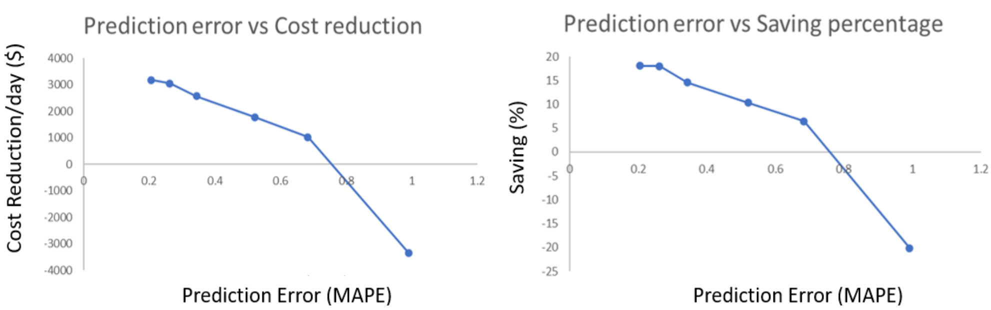


Figure 2.8: Impact of ML prediction accuracy on EV charging cost reductions (left) and savings (right)

The ML framework presented in [60] utilized XGBoost algorithm for predictions of charge profiles. The study performed further simulations to show a 21% increase in charging efficiency when the results of the ML model is integrated as opposed to not obtaining predictions of user charging behavior. The time-series model presented in [63] reported a potential saving of $770k annually for the 6-bus system and $240M for the IEEE-24 bus system if the proposed method is used to forecast charging load profiles. Given the promising impacts of ML models in benefiting EV charging operations and the fact that even a small improvement in predictions can have a large impact on cost reductions and savings, we were further incentivized to utilize additional input features that could potentially lead to more accurate charging behavior predictions. Therefore, in this work, our objective is to consider variables that have largely been ignored by existing works including weather data, traffic data, and daily events near the charging station to obtain improved predictions

Chapter 3. Methodology

In this chapter, we first define the EV charging behavior considered for prediction. We then introduce the charging dataset as well as the additional weather, traffic, and campus events data. Finally, we outline the necessary steps used for pre-processing the dataset.

3.1. EV Charging Behavior

Assuming t_{con} represents the connection time when the car first plugs in, t_{discon} represents the disconnection time when the car plugs out and leaves the station, and e represents the energy delivered to the car during the session, we consider the session charging behavior $B_{session}$ as following:

$$B_{session} \triangleq (t_{con}, t_{discon}, e) \tag{6}$$

Based on the above, we can define the length of the charging session or the session duration, S_{dur}, as follows:

$$S_{dur} = t_{discon} - t_{con} \tag{7}$$

In this work, we predict both the session duration and the session energy consumption of an individual charging record and assume that the connection time is known.

3.2. Dataset Description

Besides the charging dataset, we also make use of weather, traffic, and local events data to predict the charging behavior. First, we describe the datasets used in this work and the approaches in data collection. Additionally, the attributes of the datasets are highlighted.

3.2.1. Charging dataset. Scheduling of EV charging is more significant in public charging structures due to the unpredictable nature of the charging behavior, especially in public spaces like shopping malls. The ACN [46] dataset is among the few publicly available datasets for non-residential EV charging and thus will be utilized in this work. The dataset contains charging records from two stations on the Caltech university campus, namely JPL and Caltech. Unlike the Caltech station which is open to the public, the JPL station is only accessible to employees. Therefore, the JPL station data will not be considered in this work. More than eighty charging ports are available in the Caltech facility with a power capacity of 300 kWh that can support forty-two ports at a time. In addition to the automated data collection related to charging sessions such as the connection time and energy delivered, users may manually enter further details such as their estimated departure and energy requirements, by scanning a QR code using a mobile application. The dataset can be accessed from [74] by either a web portal or a python application programming interface (API).

3.2.2. Weather data. Although there is a small weather station located at the Caltech campus [75], which is close to the ACN charging facility, we did not consider the data from this station in this work due to the presence of missing values and irregular interval recordings for the wind variable. This could potentially make predictive models less reliable. Moreover, this station did not record variables such as rainfall and snowfall which could potentially impact charging behavior. We, therefore, used the weather data from NASA's Modern-Era Retrospective analysis for Research and Applications, Version 2 (MERRA-2) [76] which provides data for the precise location of the charging station. The accuracy of satellite weather data in comparison to ground stations has been compared in [77]. Although it has been shown that given a specific location some weather parameters may be more accurately detected using ground stations, for the purpose of this work we do not require a high level of accuracy. Rather, a more general perception of the impact of weather on the charging behavior is required. For example, we are interested in observing the impact of charging behavior during heavy rainfall as opposed to drier conditions. The weather variables used, and their respective descriptions and units are summarized in Table 3.1.

Table 3.1: Weather variable and their descriptions

Weather Parameter	Description	Unit
Temperature	Temperature at 2m above ground level	Kelvin
Relative humidity	Relative humidity at 2m above ground level	Percentage
Wind speed	Wind speed at 10m above ground level	Meter per second
Rainfall	The depth of rainfall	Millimeters
Snowfall	The density of snowfall	Kilograms per meter squared
Snow depth	The depth of snowfall	Meters
Short-wave irradiation	Surface incoming shortwave irradiation (broadband)	Watts per meter squared

3.2.3. Traffic data. Obtaining historical traffic data for specific roads and regions is challenging. Conventional traffic collection methods include intrusive approaches such as road tubes and piezoelectric sensors as well as non-intrusive approaches like microwave radar and video image detection [78]. With most of these approaches, scalability is a concern. Also, in most cases, specific roads that are required for a study are not covered. For instance, the city of Pasadena (where the charging data originates from) provides an open data site [79] for the traffic count around the city. However, for most roads in the city, it only contains traffic count for some time. Therefore, it is not usable in our case where we require regular interval data. Additionally, not all roads and streets are covered. As a result, we decided to use traffic data from google maps, which has also been used in previous ML applications [80]. The data is collected by recording the location data from the commuter's mobile devices, provided they use the application and have agreed to share their location. The data collected from individuals are anonymized and aggregated to address any privacy concerns [81]. The google maps distance matrix API can be used to retrieve the data. Given source and destination coordinates, the travel distance and the time taken for the trip is provided for a given departure time. We retrieved historical trip time for nine of the closest roads and streets which one must take to access the charging station.

3.2.4. Events data. Since the charging station is located on the Caltech university campus, we decided to include campus events and find out if the number of events has an impact on the charging behavior. The number of events in an hour was obtained from the Caltech university website calendar [82]. Certain events that were listed on the calendar did not take place at the Caltech campus. For instance, many sports events took place at other venues and seminars held at different universities were also listed on the calendar. We utilized the description provided with the event to find out whether the event took place at the Caltech campus. Events on both weekdays and weekends as well as holidays were considered. Furthermore, we opted not to include the holidays themselves as events although it is listed as such on the calendar. For simplification, we decided to round the minutes to the nearest hour. Therefore, if an event started at 10.20 am, it was counted as an event starting at 10 am. Figure 3.1 displays some of the campus events on January 11, 2019.

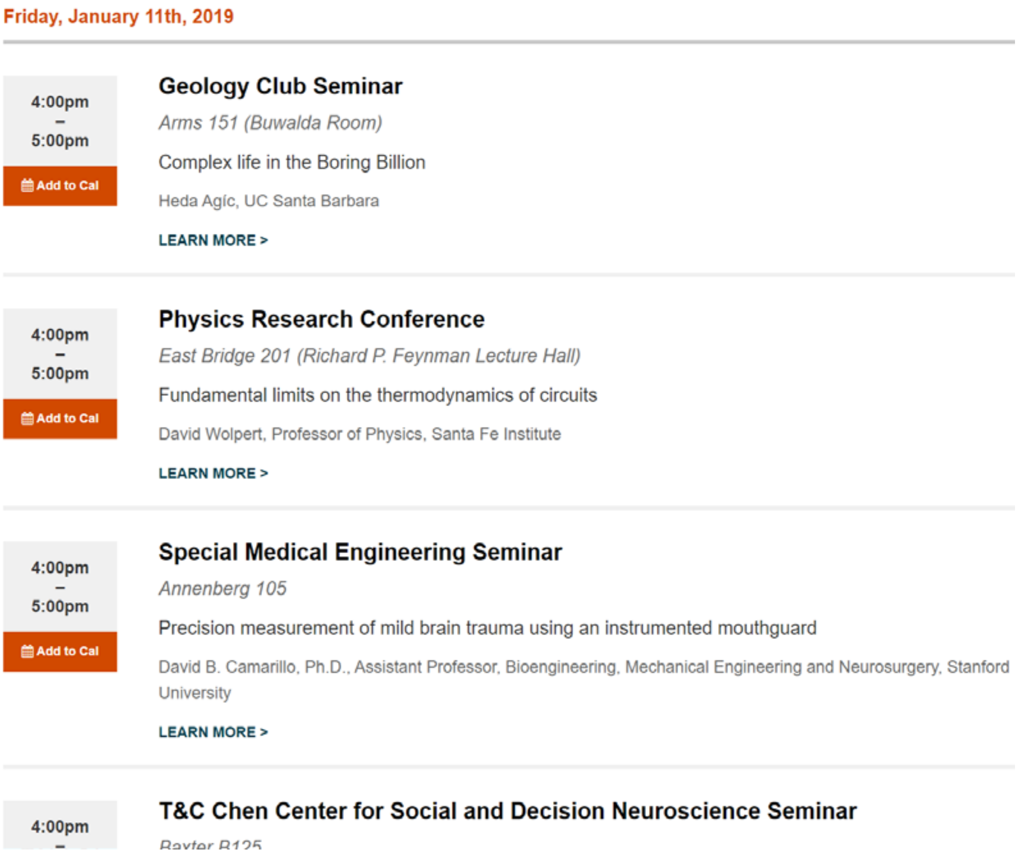


Figure 3.1: A sample of the campus events taken from the Caltech university calendar [82]

3.3. Data Preprocessing

Cleaning and preprocessing the dataset is vital to ensuring the quality of the predictive models. Factors such as the presence of outliers and missing values can negatively impact the performance of many models, as discussed in Section 2.2. The preprocessing steps include removing faulty records and handling outliers.

The presence of outliers can negatively impact model performance. A common technique of graphically detecting outliers is boxplots [83]. The boxplots for both target variables contained outliers, as illustrated in Figure 3.2. We noticed that the outliers for both variables are not consistent, i.e., we have far too many outlier points for energy consumption as compared to the session duration. Certain vehicles may consume a far greater amount of energy even if the session duration is not too long. Therefore, counting these values as outliers is not appropriate as it would result in discarding valuable data points.

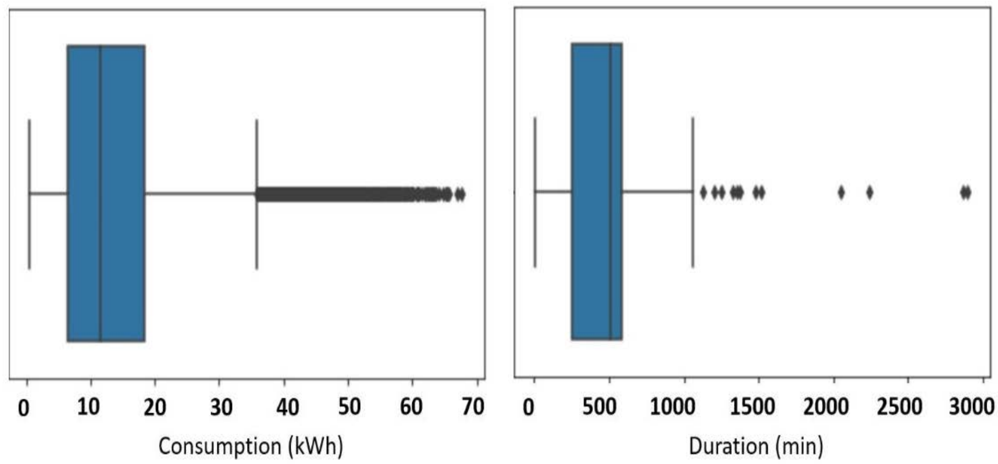


Figure 3.2: Boxplots of energy consumption (left), session duration (right)

Consequently, we opted to perform multivariate outlier detection using the isolation forest algorithm. The algorithm constructs an ensemble of iTrees for a given data set. The outliers are those instances that have short average path lengths on the iTrees [84]. By randomly selecting a variable and a split value between the minimum and maximum of the selected variable, the observations are 'isolated'. Partitioning of observations is repeated recursively until all of them have been isolated. After the partitioning, observations that have shorter path lengths for some particular points are likely to be the outliers. Figure 3.3 illustrates the outlier detection process of the target variables. The region within the red line contains the inliers, points that are considered normal, and those outside the red line are outliers. The x and the y axis have been normalized for both variables. A total of 697 outliers were detected which accounts for 4% of the total observations.

For the charging data, we only considered charging records that were registered, i.e., contained user IDs, and this accounted for 97% of the records. For the weather data, the time of recording was in universal time and we used the *pytz* [85] library in python to convert the time zone to be the same as that of the charging records (i.e. California local time). We also converted the temperature units from kelvin to degrees Celsius. Then, for each given hour, we computed the average of the previous seven hours of weather as well as the average of the next ten hours. This would allow us to understand the manner in which the previous weather and the weather after arrival impacts the charging decision. For instance, heavy snowfall in the previous hours may account for shorter charging duration and so on. With regards to the traffic data, we also converted the time zone from coordinated universal time into California local time to make the merging process convenient. We then aggregated the traffic for each hour across the nine selected roads and streets. It must be noted that we considered the average trip time as well as the maximum trip time as estimated by google maps. Finally, we aggregated the total events on the campus for each hour.

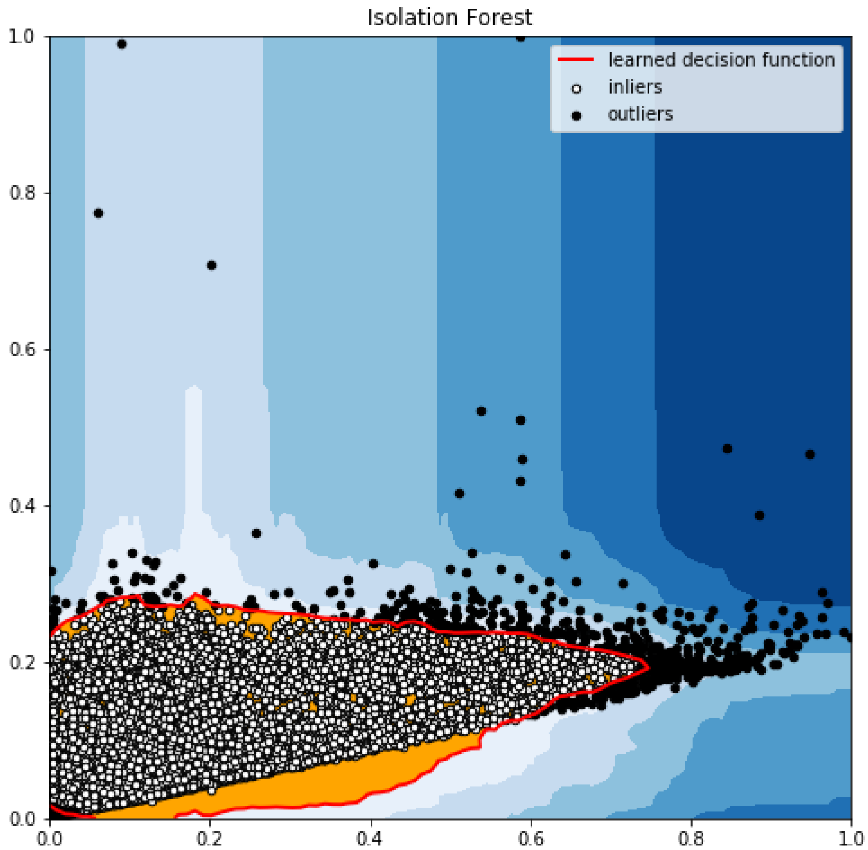


Figure 3.3: Outlier detection using isolation forest

To merge the various data, the time-series fields were converted to date-time objects using *pandas* [86] library. Then to obtain weather, traffic, and events for a particular charging record, we first obtained the nearest hour that the connection time belongs to. For example, the connection time of 22:11 belongs to 10 pm. This allows us to easily extract the other information because they were recorded in hourly intervals. Instead of simply selecting the traffic level for a given time, we selected the total traffic after arrival until the end of the day. If a vehicle arrived at 2 pm, for instance, we accumulated the traffic from 2 pm until the end of that day. This would allow the model to learn the manner in which the traffic level impact the charging behavior. Similarly, we considered the total events after arrival until the end of the day.

Chapter 4. Data Analysis and Experimental Setup

In this chapter, we first present an exploratory data analysis to identify various trends and patterns. Next, we describe the feature engineering methods to transform several variables into more meaningful representations. Finally, we summarize our experimental setup and present a graphical illustration of the framework.

4.1. Exploratory Data Analysis

Data visualization allows us to discover the relationships between different variables of the dataset and this can prove useful in feature engineering and selection. For visualization, *ggplot* [87] library from the R programming language was used.

Figure 4.1 presents the total number of charging sessions recorded by each day of the week, i.e., the number of charging sessions aggregated for a given day of the week.

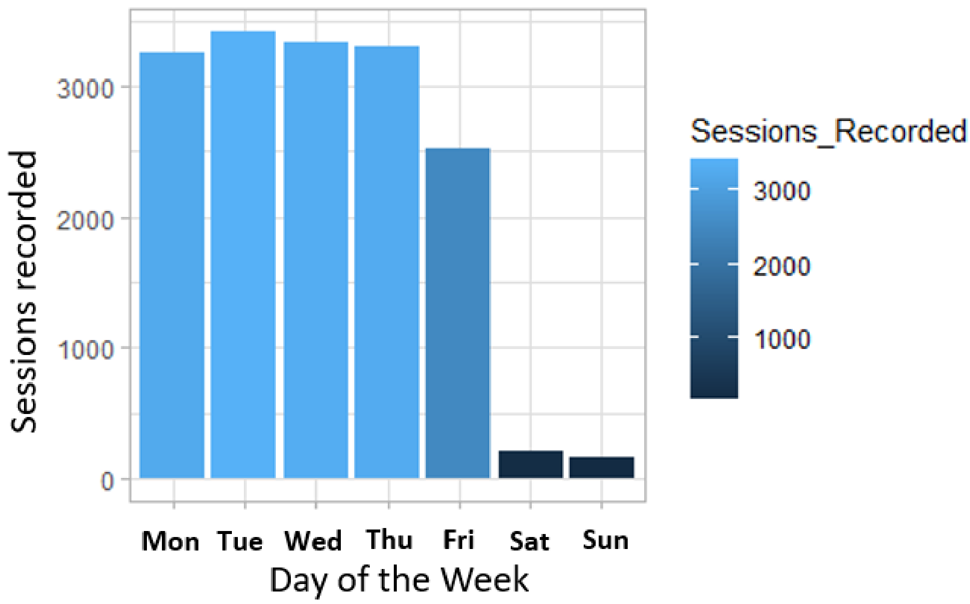


Figure 4.1: Number of sessions recorded by day of the week

It is quite clear that during weekdays about ten times more charging sessions occur compared to weekends (Saturday and Sunday). A possible explanation for this is that due to the charging station being located at a university campus, it is likely that most users who utilize this charging facility are the students and staff of the university.

Therefore, naturally, the number of vehicles on weekdays, during which the classes are ongoing, will be higher. We also note that Fridays record a significantly lower number of sessions than other weekdays. This is probably because it is the last day of the week and some users may avoid charging during this day. Next, we display the various graphs pertaining to session duration before presenting the graphs related to energy consumption.

4.1.1. Visualization for session duration. Figure 4.2 shows the data distribution of the total session duration. It highlights the total number of records for the various ranges of session durations. This can help us understand how long most users spend charging their EVs.

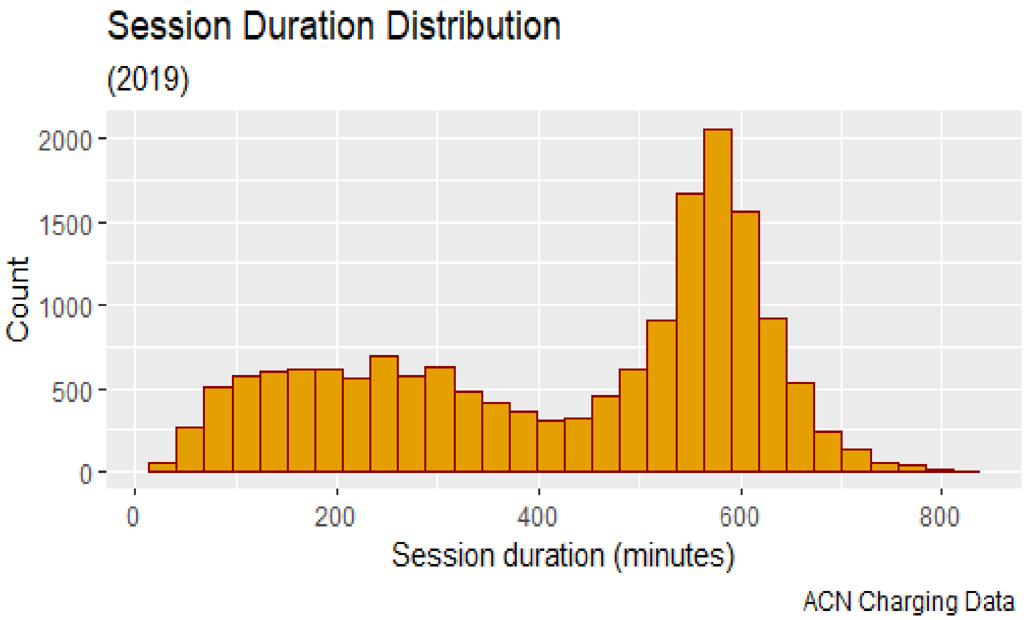


Figure 4.2: Distribution of session duration

We notice from the session duration distribution that on average most EVs tend to spend between 550 and 650 minutes (about 9-11 hours) in a session. In very rare cases, some sessions lasted for more than 800 minutes (about 13 hours). There is also a large number of shorter sessions that are between 20 minutes and 400 minutes. Figure 4.3 plots session duration by month and Figure 4.4 illustrates session duration by days of the week.

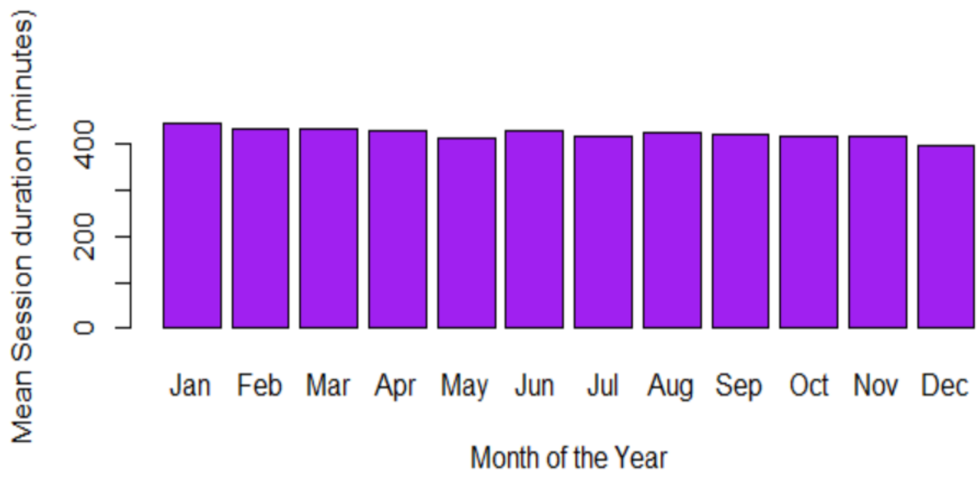


Figure 4.3: Session duration by month

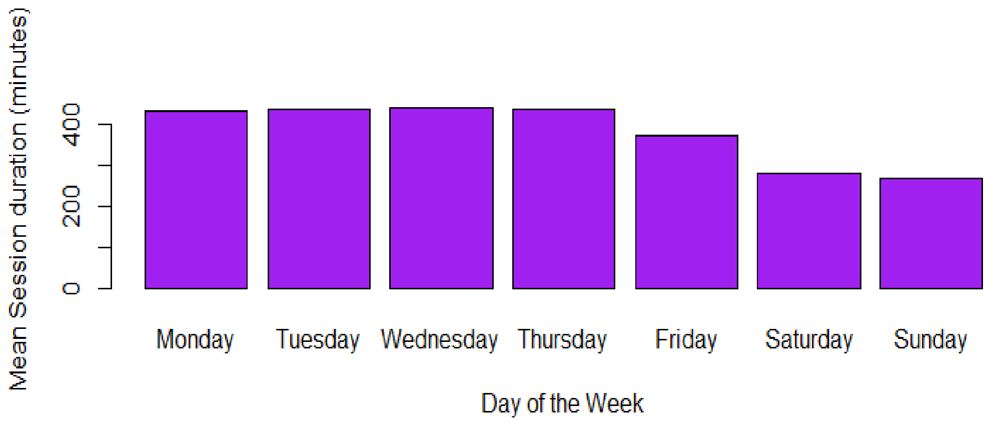


Figure 4.4: Session duration by days of the week

There are no significant differences among the month in terms of session duration, but January records the longest durations on average. From the day of the week plot, we notice that during weekends the session duration is significantly shorter as users tend to stay for shorter durations and get on with their weekend activities. Conversely, during weekdays many prefer to stay around their working hours resulting in longer charging sessions. We also notice shorter durations on Fridays compared to other weekdays. This is most probably because it is the last day of the week and many users may prefer to leave the campus earlier during this day. We will next present some of the meaningful visualizations of weather variables with regards to session duration. In Figure 4.5, we illustrate the temperature against the session duration.

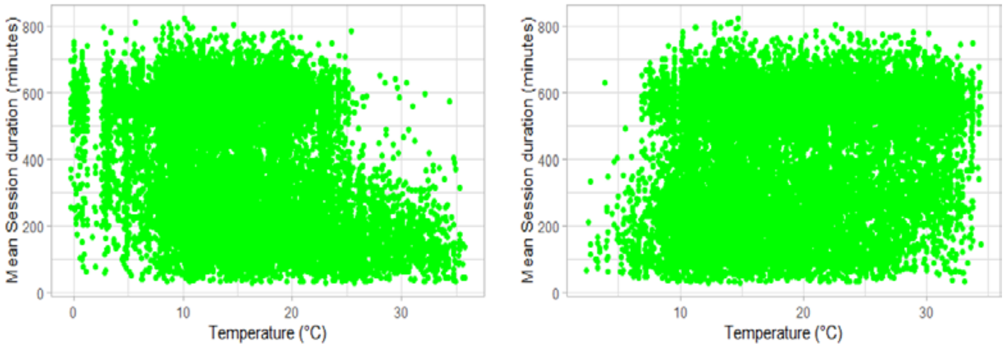


Figure 4.5: Previous temperature (left) and next temperature (right) against session duration

We notice a general downward trend for the previous average temperature, i.e., as the temperature before the arrival increases from 20 to 30 degrees, session length decreases. The density of the plot decreases as the temperature increases, suggesting a smaller number of sessions during warmer temperatures. We do not see any significant trend for the temperature of the next few hours after the arrival. Since relative humidity generally has an inverse relationship to temperature, we observe the opposite trend for the humidity against session duration plots, which are presented in Appendix A1. Figure 4.6 displays the trend between rainfall and session duration.

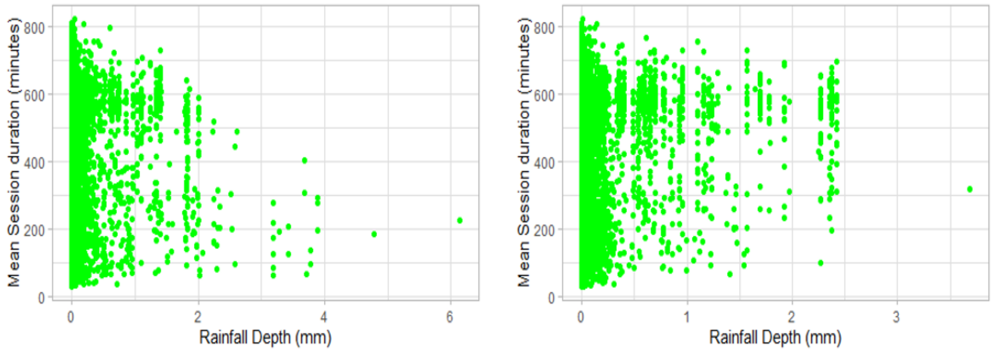


Figure 4.6: Previous rainfall (left) and next rainfall (right) against session duration

A slight downward trend is noticeable in the previous rainfall curve which indicates that generally session duration tends to be lower as the amount of rainfall prior to the EV arrival increases. For the rainfall amount after arrival, we do not observe any significant trend. Both snowfall and windspeed plots did not display any important trends with regards to session duration, and they are presented in Appendix A2 and A3, respectively. In Figure 4.7, we plot the irradiation against session duration.

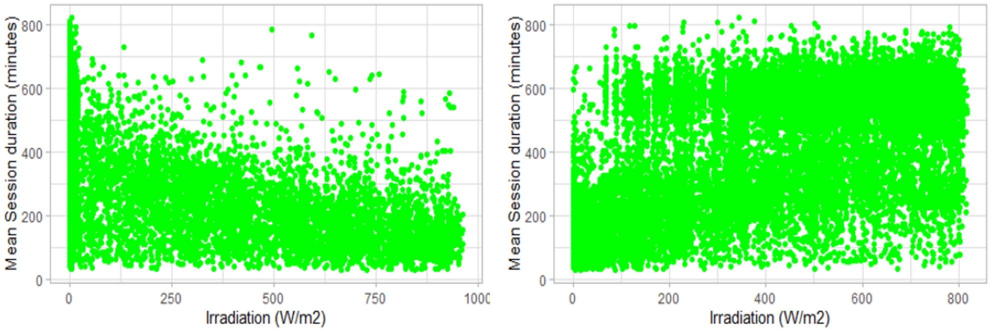


Figure 4.7: Previous irradiation (left) and next irradiation (right) against session duration

From the left curve in Figure 4.7, we observe that as the average of the previous hours' irradiation increases the session length tends to be shorter. As for the irradiation after arrival, no significant trend is noticed. We next illustrate the traffic and events after arrival against session duration in Figure 4.8.

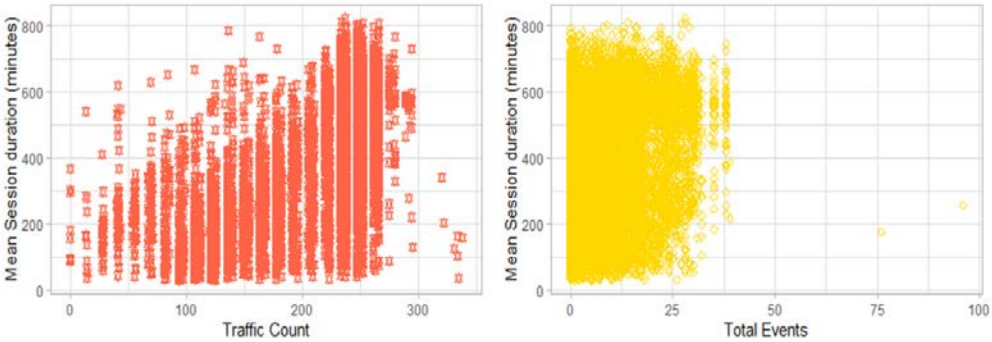


Figure 4.8: Traffic count level (left) and total campus events (right) against session duration

We notice an upward trend for traffic level with regards to session duration. This implies that as traffic level increases, session duration tends to be longer. For the total number of events after arrival, no significant relationship can be observed with regards to session duration. Next, we will analyze the graphs for the second response variable, energy consumption.

4.1.2. Visualization for energy consumption. Figure 4.9 displays the distribution of energy consumption. This can help us identify the most common energy consumption ranges as well as the extreme ones.

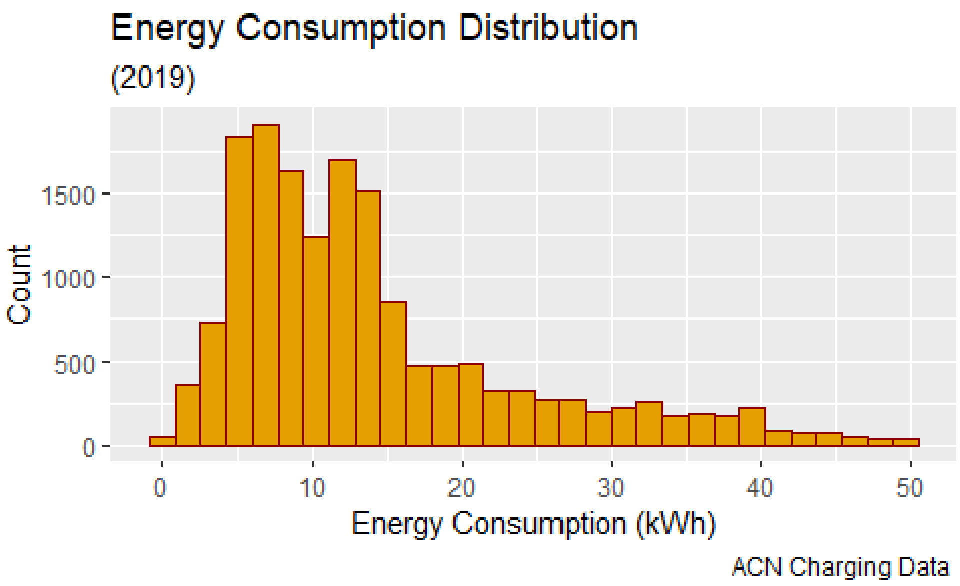


Figure 4.9: Distribution of energy consumption

From the energy consumption distribution, we see that most sessions fall under 5-15 kWh, with a few sessions exceeding 30 kWh. Figure 4.10 plots energy consumption by month and Figure 4.11 shows consumption by days of the week.

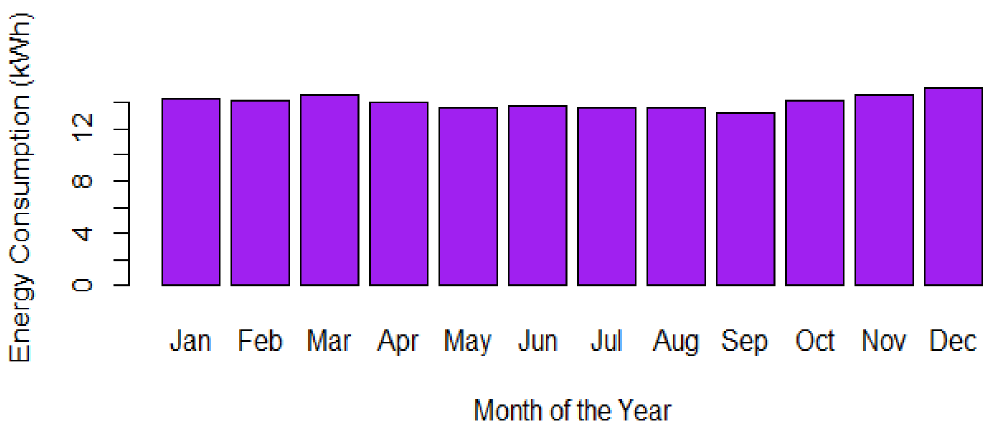


Figure 4.10: Energy consumption by month

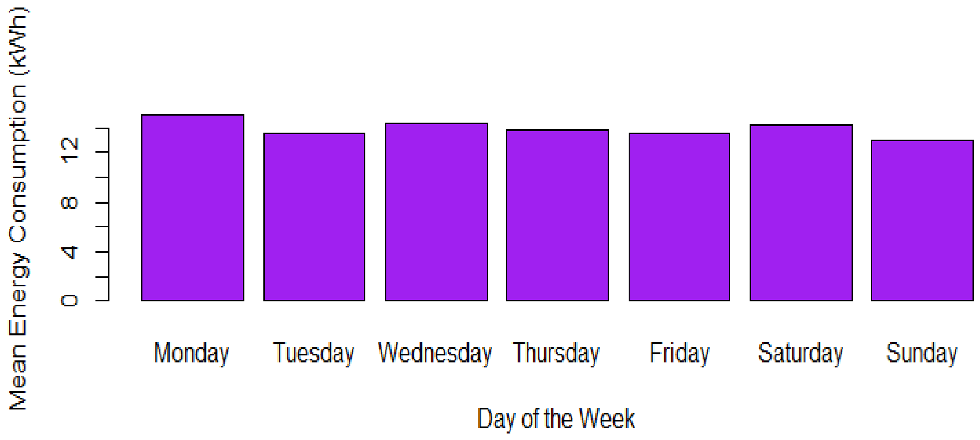


Figure 4.11: Energy consumption by days of the week

There are no significant differences among the month in terms of average energy consumption, but we observe a slight increase towards the end of the year, i.e., from October to December. It is also evident that Mondays record the highest consumption on average from Figure 4.11. No other significant difference in energy consumption was noticed between weekdays and weekends. Next, we illustrate the energy consumption by federal holidays in Figure 4.12.

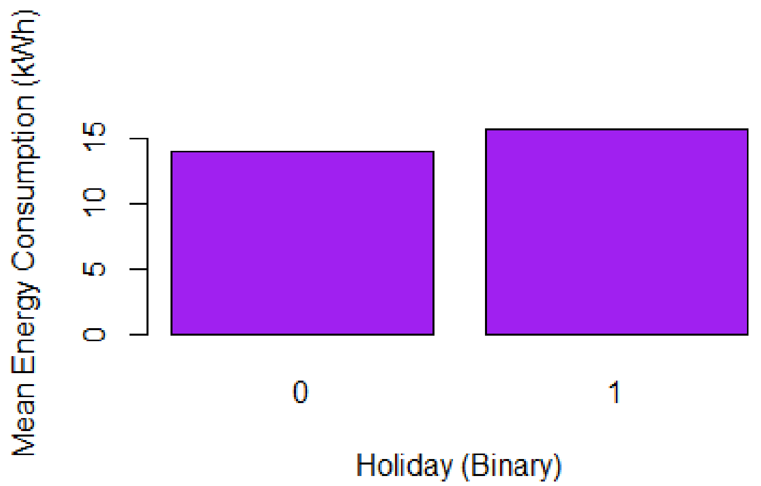


Figure 4.12: Energy consumption by US federal holidays, 0 indicates non-holidays and 1 indicates holidays

From the above plot, we notice that during federal holidays, energy consumption on average is slightly higher. We next present some of the noteworthy visualizations of weather variables with regards to energy consumption. In Figure 4.13, we plot temperature against energy consumption.

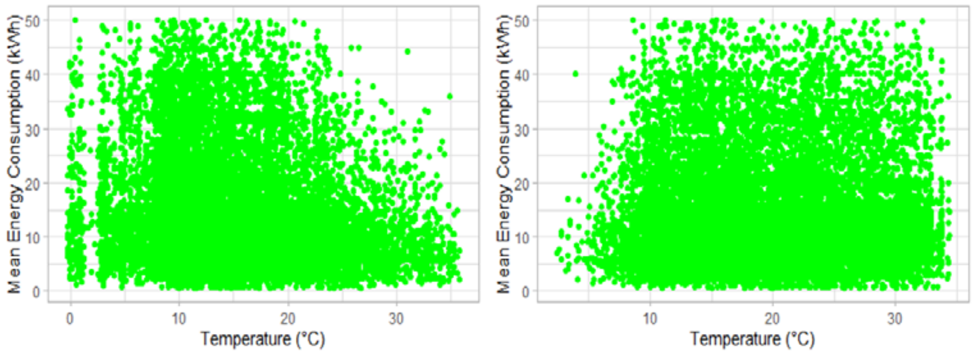


Figure 4.13: Previous temperature (left) and next temperature (right) against energy consumption

We notice a general downward trend for energy consumption, i.e., as temperature before EV arrival increases from 20 to 30 degrees, consumption decreases. This conclusion is consistent with the analysis for the session duration. Likewise, we do not see any significant trend for the temperature of the next few hours after arrival and the energy consumption. The humidity against session duration, presented in Appendix A4, displayed the opposite trend to that of the temperature. Figure 4.14 illustrates the rainfall and energy consumption trend.

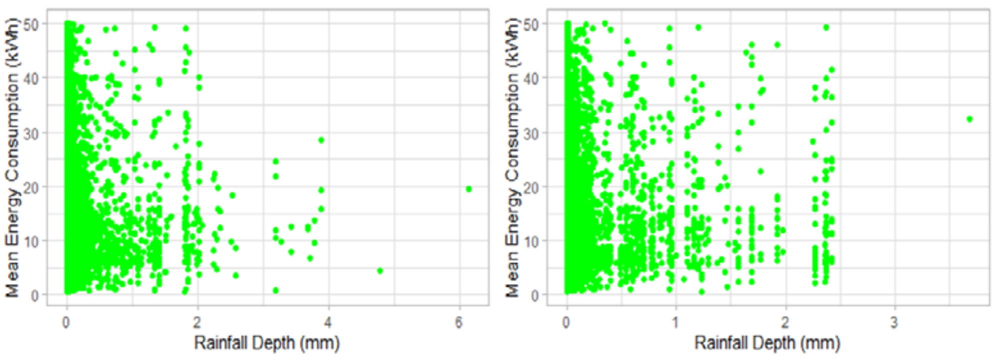


Figure 4.14: Previous rainfall (left) and next rainfall (right) against energy consumption

A slight downward trend is noticeable in the previous average rainfall curve, which signifies that for increasing rainfall prior to EV arrival, energy consumption is slightly lower. For the rainfall amount after arrival, we do not observe any significant trend. The snowfall and windspeed plots are presented in Appendix A5 and A6, respectively. We did not observe any significant trends for both of these plots. Figure 4.15 displays irradiation against energy consumption.

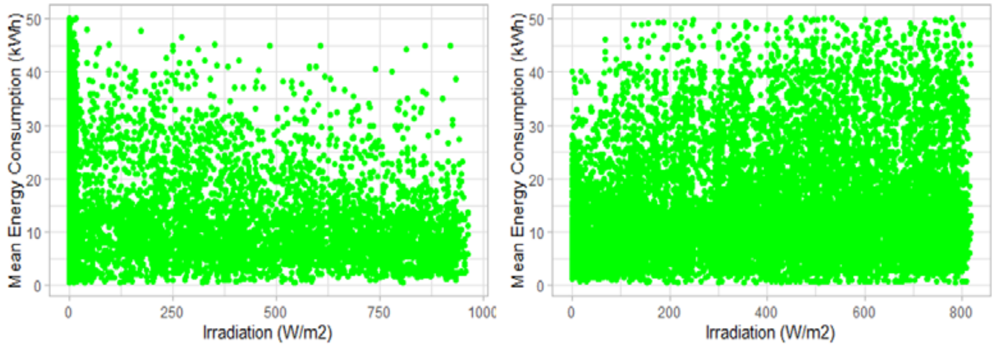


Figure 4.15: Previous irradiation (left) and next irradiation (right) against energy consumption

We observe that as the average of the previous hours' irradiation increases, the consumption tends to be lower, but no significant trend is noticed for irradiation after arrival. We next present the traffic and events after arrival with regards to energy consumption in Figure 4.16.

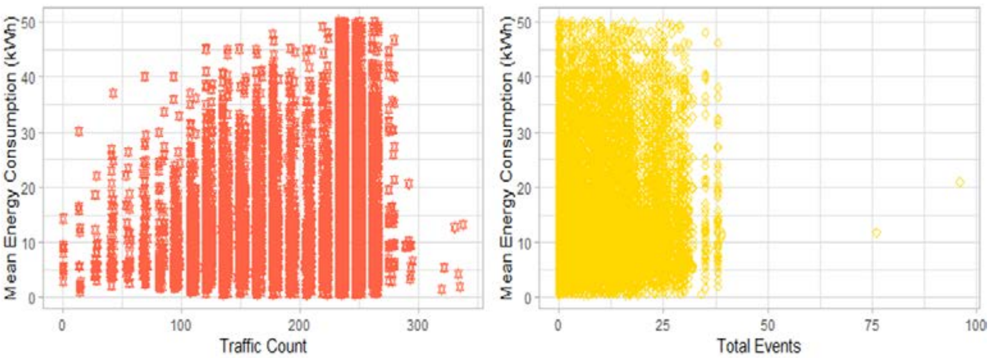


Figure 4.16: Traffic count level (left) and total campus events (right) against energy consumption

Similar to the session duration analysis, an upward trend can be noticed for the traffic count, indicating a higher consumption as the traffic level increases. No significant conclusion can be drawn for the number of events after arrival. In the next section, we describe the feature engineering steps.

4.2. Feature Engineering

Feature engineering refers to the transformation of data into meaningful representation using human knowledge. This process is labor intensive but important nonetheless because this is a weakness of the learning algorithms. Feature engineering relies on human ingenuity and prior knowledge to compensate for the inability of the algorithms to extract and organize discriminative information from the data [88]. For instance, a date value as input to the model is perhaps better represented as multiple features of the day, month, and year. We discuss future engineering steps next.

Firstly, we converted the time fields that will be used by the models into the numeric format by simply dividing the minute by sixty and adding to the hour. This would convert the time 5:39, for instance, into 5.65. Next, for each charging record, we identified the corresponding user and found out their average departure time, session duration, and energy consumption. This was done by finding out the user ID of the charging record and aggregating his previous records. We used the arrival time as a numeric feature as this is assumed to be a known variable. However, the arrival time also has other components such as the date information. Using this, we extracted the hour of the day, day of the month, the month of the year, day of the week, whether the day is a weekend and whether the day falls in a US federal holiday. However, temporal information such as day, hour, and month are cyclic ordinal features. This is because the hour value of 23 corresponding to 11 pm, for example, is close to the hour value of 0 which corresponds to 12 am. To represent the proximity of these values, trigonometric transformation was performed as follows:

$$f_x = \sin\left(\frac{2\pi f}{max(f)}\right) \tag{8}$$

$$f_y = cos\left(\frac{2\pi f}{max(f)}\right) \tag{9}$$

where f represents the cyclic feature to be transformed, f_x and f_y represents the first and second components of the cyclic feature, respectively. To transform other categorical variables, one-hot encoding was used. In this approach, a single variable with *n* points and *k* distinct classes is transformed into *k* binary variables with *n* points each.

Table 4.1: List of input features and their descriptions

Feature	Description
session_length	Length of charging duration, the target variable
kWh_delivered	Session energy consumption, the target variable
time_con	Numerical representation of the connection time (arrival time)
day_of_week	Day of the week, one-hot encoded
is_weekend	Binary variable indicating whether the session took place on a weekend
holiday	Binary variable indicating whether the session took place on a US federal holiday
hr_x	Sine components of the hour
hr_y	Cosine component of the hour
day_x	Sine components of the day
day_y	Cosine components of the day
mnth_x	Sine component of the month
mnth_y	Cosine component of the month
mean_d_time	Historical average departure time
mean_con	Historical average consumption
mean_dur	Historical average session length
traffic_aft_arvl	Average traffic level after arrival
max_traffic_aft_arvl	Maximum traffic level after arrival
events_after_arrival	Total campus events after arrival
avg_temp_prv	The average temperature of the last 7 hours
avg_temp_nxt	The average temperature of the next 10 hours
avg_hum_prv	The average humidity of the last 7 hours
avg_hum_nxt	The average humidity of the next 10 hours
avg_win_prv	The average wind speed of the last 7 hours
avg_win_nxt	The average wind speed of the next 10 hours
avg_rain_prv	The average rainfall of the last 7 hours
avg_rain_nxt	The average rainfall of the next 10 hours
avg_snwfall_prv	The average snowfall of the last 7 hours
avg_snwfall_nxt	The average snowfall of next 10 hours
avg_snwdpth_prv	The average snow depth of the last 7 hours
avg_snwdpth_nxt	The average snow depth of the next 10 hours
avg_irradiation_prv	The average irradiation of the last 7 hours
avg_irradiation_nxt	The average irradiation of the next 10 hours

For a suitable representation of numeric variables, feature scaling is a common transformation, where the goal is to normalize the range of the numeric features. Feature scaling is important because it results in faster training convergence for various models. Moreover, for certain models, feature scaling ensures that the performance is not affected by extreme values. In other cases, feature scaling can speed up the model training process. There are various scaling techniques, including scaling by domain where all the features are scaled to a specific range such as [0, 1] and scaling to minmax where the features are scaled to the range [0, R], in which case the minimum of the maximum value of the feature in all directions is assigned as the radius of the sphere R [89]. However, in this work, we used standardization which ensures the values of each feature have zero mean and unit variance. The transformations were performed using the *preprocessing* package of the *Scikit-learn* [90] library. Table 4.1 lists all the features used for model training in this work along with their descriptions. In total, we identified thirty input features for model training, which is significantly large when compared to the existing works.

4.3. Model Selection and Experimental Setup

We selected all charging sessions from the ACN dataset that belonged to the 2019 calendar year, which ensures we consider the seasonal factors during training. Due to the COVID-19 pandemic, we opted not to select the charging records belonging to 2020 because the lockdown measures around the globe will not represent the usual charging behavior. The dataset was split such that 80% of the records were used for model training and 20% for model evaluation. During the training phase, we performed K-fold cross-validation, where the algorithms are repeatedly trained K times with a fraction of 1/K training examples left out for testing [91]. In this case, we selected the common K value of 10. This procedure was chosen to find out the optimal sets of parameters. To configure model hyperparameters, we utilized a grid search method that determines the optimal set of parameters from a given list by trying out all possible values of the specified parameters [92]. After finding out the best parameters, the model was retrained on the entire training set using these parameters. Following this, the model was evaluated using the aforementioned regression metrics on the data it has never seen before, i.e., the test set.

Inspired by the success of ensemble learning methods in some of the previous works in the literature, we also decided to experiment with ensemble learning. We used two variants of ensemble models, namely voting regressor and stacking regressor, using the *ensemble* package of the *Scikit-learn* library. In a voting regressor, several base regressors are trained on the entire training set, and the average of the predictions made by the base models are treated as the final prediction. The stacking regressor is based on the concept of stacked generalization, where predictions made by the base models are used as inputs to a final estimator, which is trained using cross-validation, to generate predictions [93]. Figure 4.17 provides a graphical representation of the proposed framework in this work.

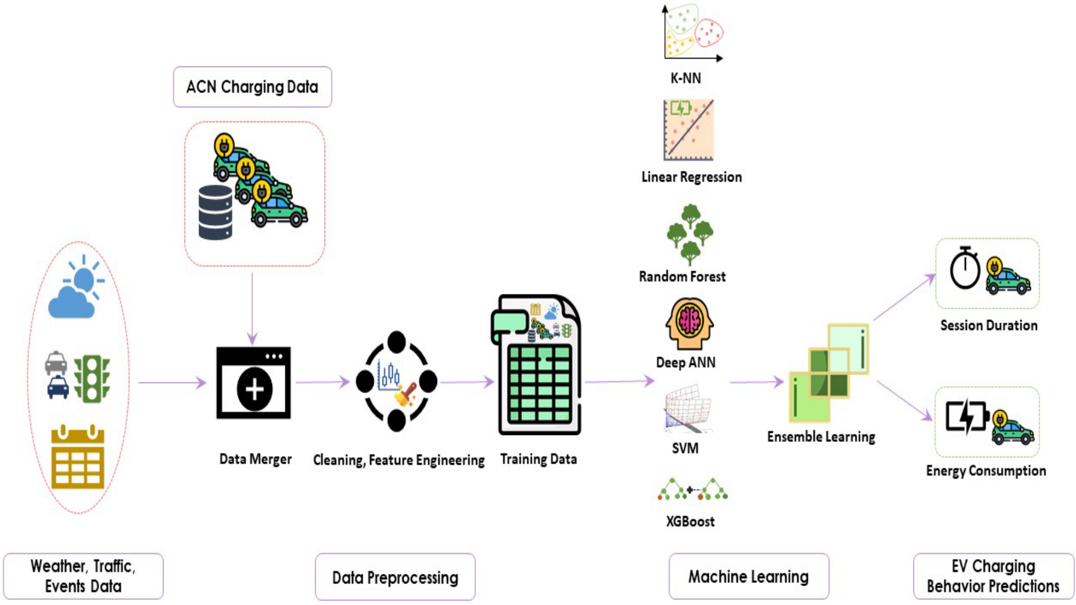


Figure 4.17: Visual representation of the proposed framework

After successfully merging the weather, traffic, and events data with the ACN charging data using the approach mentioned in Section 3.3, we perform the necessary preprocessing and feature engineering steps as described earlier. We then split the dataset into training and test set in an 80-20 split ratio. Next, we use the training dataset to train six popular ML algorithms, namely, K-NN, LR, RF, deep ANN, SVM, and XGBoost. To determine the best sets of hyperparameters for all the models, we utilize 10-fold cross-validation. Next, using these models we develop the voting and stacking ensemble regressor. It must be noted that not all the individual ML models are necessarily going to be used to construct the ensemble models. Rather, depending on the initial results in the training set, the best performing models will be utilized accordingly for constructing the ensemble models. Finally, we assess all the base models as well as the ensemble learning models on the test set, which the models have not encountered before. For evaluation, we use the four metrics defined earlier in Section 2.2.5. This enables us to evaluate how well the models have generalized to data it has not come across before and also allows us to determine whether or not overfitting has taken place during the training phase. The same process for model training is repeated for both the predictions of session duration and energy consumption. However, the data cleaning and preprocessing steps are not repeated, i.e., it is a common step for both predictions. Finally, in order to compare our results to those in the previous works, we select the result of the model that achieved the highest accuracy in the test set to be the representative.

Chapter 5. Results and Analysis

In this chapter, we first present the feature selection approach as well as a visualization of the feature importance plots. The results obtained from the model training phase is presented next, showing a comparison across various ML models and the development of the ensemble models. We then present the results on the test set for all the models and also provide an analysis of the performance. Finally, we compare our work with the existing works in the literature.

5.1. Feature Selection

We begin the experiment with an RF algorithm that can be used to visualize the variable importance [29], whereby features that contribute most to accurate predictions can be identified. This is a method for feature selection, where certain variables, that are not important and can often hinder performance, are removed. Variables that have almost no relationship to the response variable can often confuse the model and produce inferior performance. For our case, the inclusion of the least important variables had a very insignificant performance gain and hence we decided to include them in model training. In other words, removing the least important variables made the performance slightly worse.

Moreover, variables can be ranked in terms of their relative importance to each other. This is determined by each feature's contribution in producing the most effective splits in an RF algorithm. The feature that contributes the most to the decision-making process will be ranked higher. In Figure 5.1, we illustrate the top ten important variables for session duration predictions. The size of the horizontal bar represents the relative importance.

The two most important predictors of session duration are the maximum traffic after arrival and the time of connection. This indicates the effectiveness of including the traffic information for the prediction of session duration, which has not been considered in previous related works. Following this, we have the historical average session duration, departure time, and energy consumption that contributed significantly. Based on this we can conclude that the historical charging behavior of EV is important to consider for predictions, and therefore, methods such as using a mobile phone application that allows historical data collection is important. Finally, we have five of the weather parameters that are represented in the top ten features and hence this highlights the usefulness of including the weather information. Irradiation, temperature, wind, and humidity are found to be more useful than rainfall and snowfall. However, it must be noted that the campus events variable was not part of the top ten features. The two least important features were federal holidays and snowfall. Next, the feature importance plot for energy consumption is presented in Figure 5.2.

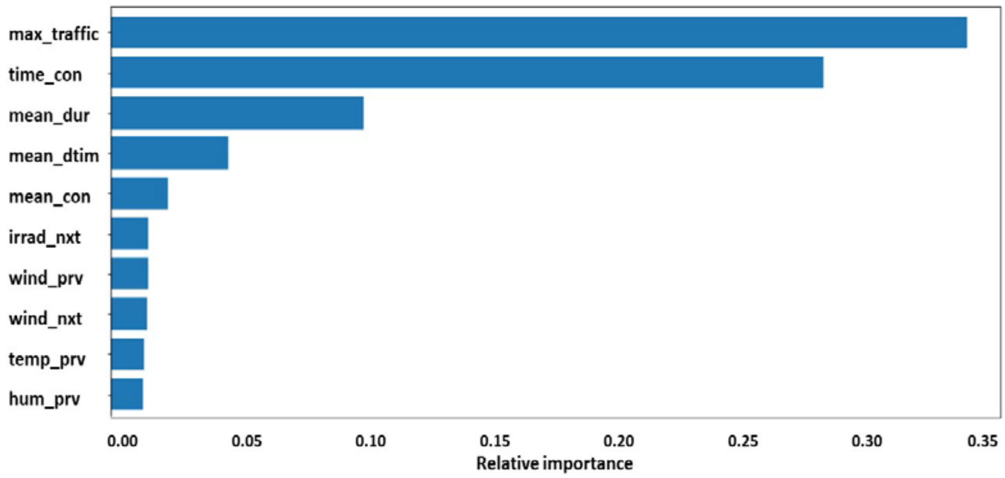


Figure 5.1: Top ten features for session duration

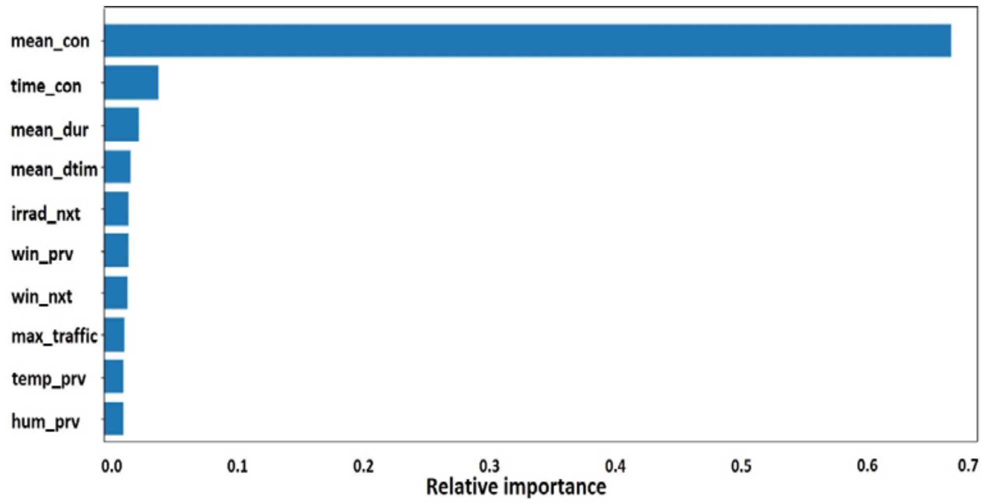


Figure 5.2: Top ten features for energy consumption

In the case of energy consumption predictions, the historical average energy consumption is by far the most significant of the predictors. This is because a specific EV will consume similar energy if its session duration is consistent. This yet again highlights the importance of obtaining historical user data. The arrival or connection time is the second most significant, followed by the historical average session duration and departure time. For both session duration and energy consumption, the EV arrival time, therefore, had a significant impact on the charging behavior. The remaining top ten features were made up of several weather variables along with the maximum traffic after arrival. This further demonstrates the value of including these variables as additional information. Similar to the session duration, irradiation, wind, temperature, and humidity were found to be more important than rainfall and snowfall. The three least significant predictors were the days of the week, federal holidays, and snowfall. Next, we present the results obtained from training the ML models for session duration predictions.

5.2. Session Duration Predictions

As mentioned earlier, the hyperparameters for the models were determined using the grid search approach and 10-fold cross-validation on the training data set. For the deep ANN training, we experimentally determined an architecture with three hidden layers of 64, 32 and 16 nodes, respectively to be the most suitable using 10-fold cross-validation. Rectified Linear Units (ReLU) [94] was used as the activation function for all hidden layers and the output layer contained a linear activation as the prediction is expected to be a continuous numeric value. The learning rate value was set to 0.001 and Adam [95] algorithm was utilized for model optimization. The training batch size was selected to be 32 and the model was allowed to train for a total of fifteen iterations. Figure 5.3 displays the curve for determining the number of neighbors for K-NN as well as the training loss curve for deep ANN.

For determining the optimal *K* value for K-NN, we decided to use a portion of the training data as validation. This means we trained the model a total of fifty times for *k* values ranging from 1 to 50 and evaluated the models on a validation set. As can be seen from the left figure in Figure 5.3, the performance does not increase after increasing the number of neighbors to more than ten. As a result, we have selected the *K* parameter to be ten and the model was consequently a 10-NN one. We then combined the validation set with the training set and retrained the model using the optimal parameter. From the ANN loss curve in the right of Figure 5.3, it is clear that as the training and validation losses are almost identical, there was no overfitting of the model. The training process was relatively quick and did not require additional graphical processing power. Table 5.1 summarizes the scores for all ML models on the training set, which is the average across 10-folds as mentioned earlier.

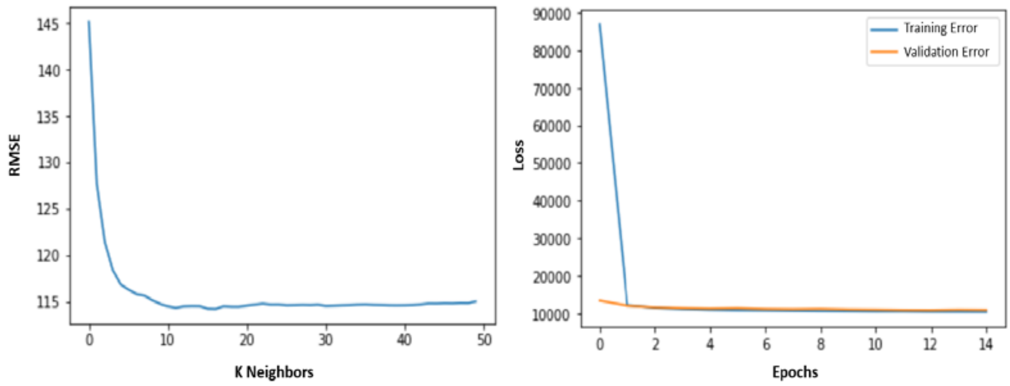


Figure 5.3: Determining K value for K-NN (left) and ANN training loss curve (right) for session duration

Table 5.1: Results from the training set across all ML models for prediction of session duration

Metrics/Model	**RMSE (minutes)**	**MAE (minutes)**	$\mathbf{R^2}$	**SMAPE (%)**
LR	109	78.5	0.67	12.0
K-NN	111	79.1	0.65	11.6
RF	100	68.7	0.72	10.1
SVM	102	68.2	0.71	10.2
XGBoost	100	69.3	0.72	10.3
Deep ANN	103	73.1	0.71	10.7
Voting Ensemble	99.5	67.4	0.72	10.0
Stacking Ensemble	99.5	68.0	0.72	10.1

The three best performing models on the training set were RF, XGBoost, and SVM. Deep ANN comes close to these performances in terms of RMSE and R^2 values. However, if we only consider MAE and SMAPE, deep ANN performs worse than the top three. Moreover, both LR and K-NN scores are far worse than the rest. As a result, we aggregated the three best performing model in the training phase into two ensemble models, which resulted in an improved training score, as can be seen from Table 5.1. Next, we present the results on the test set, not encountered by the predictive models before. For reference, we also selected the user estimates of their departures as a prediction. This value was collected through a smartphone application where users were asked to enter their estimates of their departure time and consumption upon arrival. We summarize the results on the test set in Table 5.2.

Table 5.2: Results from the test set across all ML models for prediction of session duration

Metrics/Model	**RMSE (minutes)**	**MAE (minutes)**	**R^2**	**SMAPE (%)**
LR	107	77.3	0.53	12.0
K-NN	110	78.2	0.53	11.5
RF	98.7	68.0	0.63	10.1
SVM	101	67.4	0.64	10.1
XGBoost	97.9	68.0	0.63	10.1
Deep ANN	101	73.7	0.57	10.9
Voting Ensemble	97.7	**66.5**	**0.73**	**9.92**
Stacking Ensemble	**97.5**	67.1	**0.73**	9.95
User predictions	430	394	-4.20	69.9

As highlighted, the best results were obtained using the ensemble learning approach, which is consistent with previous works [42], [57]. The voting regressor performs best on MAE and SMAPE and the stacking regressor performs the best in terms of RMSE, whereas they both achieve the same R^2 score. The results are consistent with the training performance with RF, SVM and XGBoost resulting in similar performance and deep ANN slightly worse. Both LR and K-NN performs much worse compared to the other models. Predictions made by the users about their session length is also inaccurate compared to their actual session length. This indicates that relying on users to provide an estimate of their own departure time is perhaps not suitable. Further analysis of the results is presented in Section 5.4, in comparison to the energy consumption predictions, which are presented next.

5.3. Energy Consumption Predictions

A similar approach to the session duration prediction was also used for energy consumption predictions. However, we selected a different deep ANN architecture after experimentation, which in this case contained two hidden layers with 64 and 16 nodes, respectively. Likewise, ReLU activation function for hidden layers and Adam optimization was used. The training batch size was 64 and the number of epochs was set to 20. Figure 5.4 displays the plot for determining the number of K-NN neighbors as well as the training loss curve for deep ANN training.

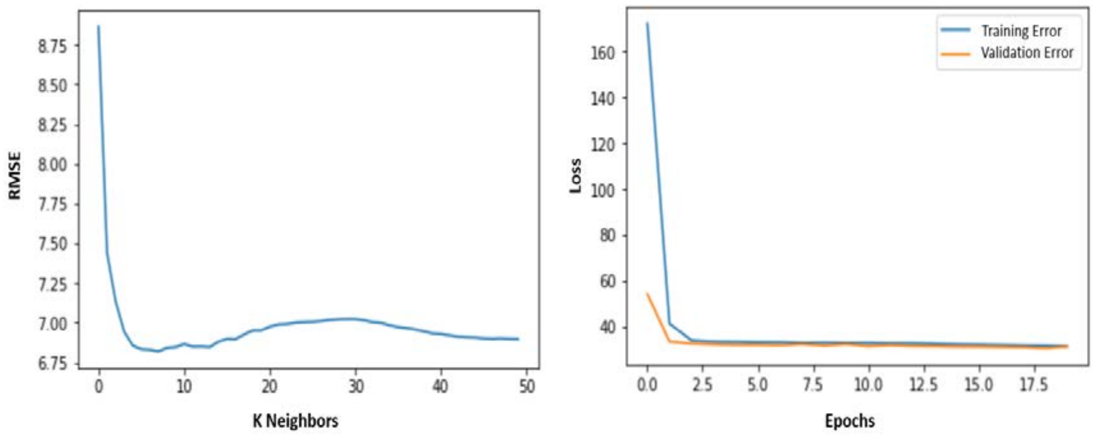


Figure 5.4: Determining K value for K-NN (left) and ANN training loss curve (right) for energy consumption

We selected a portion of the training data as validation data for finding out the optimal *K* value for K-NN model. Next, we trained the model a total of fifty times for *K* values ranging from one to fifty and evaluated all the models on the validation set. The RMSE values fluctuate after any *K* values of more than ten, as can be seen on the left of Figure 5.4. The lowest value of *K* was seven, which was selected to develop a 7-NN model. We then combined the validation set with the training set and retrained the model using the optimal parameter. The ANN training loss curve, as shown in the right of Figure 5.4, displays no overfitting as the training and validation losses are almost identical. Table 5.3 summarizes the scores obtained on the training set, which are the average scores from the 10-fold cross-validation.

Table 5.3: Results from the training set across all ML models for prediction of energy consumption

Metrics/Model	RMSE (minutes)	MAE (minutes)	R^2	SMAPE (%)
LR	5.78	3.75	0.65	14.1
K-NN	6.71	4.72	0.53	18.0
RF	5.49	3.40	0.69	11.9
SVM	5.65	3.53	0.67	12.6
XGBoost	5.56	3.49	0.68	12.4
Deep ANN	5.61	3.60	0.67	12.9
Voting Ensemble	5.50	3.42	0.69	12.0
Stacking Ensemble	5.48	3.40	0.69	11.9

RF obtained the best training scores whereas XGBoost, SVM and deep ANN achieved similar scores in terms of SMAPE. Consistent with the session duration predictions, both LR and K-NN models performed the worst. We selected the top three models, i.e., RF, SVM, and XGBoost to form the two ensemble models. In this case, the ensemble models did not improve upon the best performing RF model, but rather achieved similar results on training. The results from the test set, which the models have not seen before, are presented in Table 5.4. We also compare the results with user predictions about their respective consumptions.

Table 5.4: Results from the test set across all ML models for prediction of energy consumption

Metrics/Model	RMSE (minutes)	MAE (minutes)	R^2	SMAPE (%)
LR	5.83	3.74	0.46	13.9
K-NN	6.68	4.71	0.05	17.7
RF	5.50	3.39	0.54	11.7
SVM	5.69	3.54	0.51	12.4
XGBoost	5.61	3.48	0.51	12.1
Deep ANN	5.65	3.55	0.55	12.5
Voting Ensemble	5.54	3.41	0.69	11.8
Stacking Ensemble	**5.50**	**3.38**	**0.70**	**11.6**
User predictions	20.6	11.8	0.04	55.0

The best results as highlighted were obtained using the stacking ensemble model. The improvement using ensemble learning for energy consumption prediction was not as significant when we compare it with the session duration predictions. K-NN performance is by far the worst with an R^2 value of 0.05. The user predictions about their energy consumptions are not accurate in this case as well. In the next section, we further discuss the above results in comparison to session duration predictions and provide additional justifications.

5.4. Comparison and Discussion

When we compare across both the predictions, looking at the overall R^2 and the SMAPE, it appears that the prediction of energy consumption is perhaps more challenging. In general, better values of R^2 and SMAPE are obtained in session duration predictions. This is consistent with the previous work on the ACN data [46]. However, in another case the opposite was observed [42], i.e., the prediction of energy consumption was simpler.

We also noticed earlier from the feature importance plot in Section 5.1 that the energy consumption was greatly influenced by one single predictor, i.e., the historical energy consumption. In contrast, the session duration was influenced by multiple predictors including traffic, arrival time and historical data. Therefore, predictive models for energy consumption in situations where no historical data is available is likely to produce significantly worse performance.

Moreover, in both scenarios, it was also noticed that the user predictions about their behavior are very different from their actual behavior, which further emphasizes the need for predictive analytics. The users' predictions in terms of their energy consumption are slightly more accurate when compared to their predictions of session duration, as indicated by better R^2 and SMAPE values. The inaccuracy in users' predictions can be attributed to the users' lack of interest in entering their estimates every time they decide to charge their vehicles. A possible solution is to incentivize users to provide more accurate estimates which could potentially lead to the users being more responsible in entering their estimates. This could in turn provide the ML models with an additional predictor.

When comparing across the various ML models, both K-NN and LR models were by far the two worst performing models in both predictions. It is likely that the results for K-NN were not very accurate due to it being more suited for classification tasks. In regression problems, taking the average of all the neighboring points is not ideal because we are assuming that similar users have a common average departure time or energy consumption. Also, LR assumes a simple linear relationship between the different variables. However, in a complex problem such as this one, a simple linear model cannot provide an accurate fit of the dataset. We also noticed that deep ANN performed slightly worse than the other three traditional ML models. Deep learning models are proven to be superior in dealing with images and audio data, where feature extraction is generally not performed. However, in applications such as this where we perform feature extraction, traditional ML models can often lead to better performance. This conclusion is also consistent with other ML applications with regards to energy management, where traditional machine learning outperformed the deep learning-based model [96].

Furthermore, predictions obtained using ensemble learning were more accurate in comparison to those obtained by individual ML models in both scenarios, although the impact was more significant for session duration prediction. This is most likely because, in the first scenario, the top three performing models had similar training performance and combining their predictions resulted in an improvement. However, in the latter scenario, one model outperformed all the other models in training and hence the improvement using ensemble learning was not as significant.

Looking at the previous works in the literature, the results obtained in this work outperformed all the previous works that reported similar evaluation metrics ([46], [66], [68], [69], [70]). We summarize the results from the previous works in comparison to the one achieved in this work in Table 5.5.

In comparison to [42], the results obtained in this work for session duration is more accurate although we do not improve upon their results for energy consumption. This is most likely because the authors in [42] utilized both residential and non-residential data for their predictions, and residential charging behavior in most cases are more consistent and simpler to predict. Conversely, in this thesis, only non-residential charging data was used for predictions. Therefore, it is possible that the overall accuracy for energy consumption prediction in [42] is superior because it was greatly influenced by the more accurate predictions in residential charging sessions.

Table 5.5: Performance comparison with previous related works in the literature

Source	Session Duration	Energy Consumption	Dataset Used
[46]	SMAPE: 14.4%	SMAPE: 15.9%	ACN (historical charging)
[66]	MAE: 82 minutes	Not considered	German charging data (historical charging, vehicle & location info)
[42]	SMAPE: 10.4%	SMAPE: 7.5%	UCLA campus (historical charging) and Residential charging data from the UK
[68]	Not considered	R2: 0.52	Nebraska public charging (historical charging, temporal & location)
[69]	Not considered	SMAPE: 15.3%	UCLA campus (historical energy)
[70]	Not considered	SMAPE: 14.1%	UCLA campus (historical energy)
Our work	**SMAPE: 9.92%, MAE: 66.5 minutes**	**SMAPE: 11.6%, R2: 0.7**	**ACN, weather, traffic, and events data**

However, it must be noted that all previous works except [46] used a different dataset compared to this work and therefore a comparison is perhaps not suitable. Therefore, keeping the comparison across the same dataset, we can conclude that the utilization of the additional weather, traffic, and events data resulted in an improvement in the EV charging behavior predictions. The proposed approach can be implemented for the other datasets as part of future research work.

We summarize the comparison of session duration in Figure 5.5 and Figure 5.6 based on SMAPE and MAE. Similarly, the comparison of energy consumption is presented in Figure 5.7 and Figure 5.8 based on SMAPE and R^2. In Figure 5.5, Figure 5.6, and Figure 5.7 a lower vertical size of the bar indicates better performance because we are comparing the mean absolute errors of the predictive models. However, in Figure 5.8, a higher vertical size of the bar indicates better performance because we are comparing the R^2, which provides an indication about the accuracy of the regression model rather than the error.

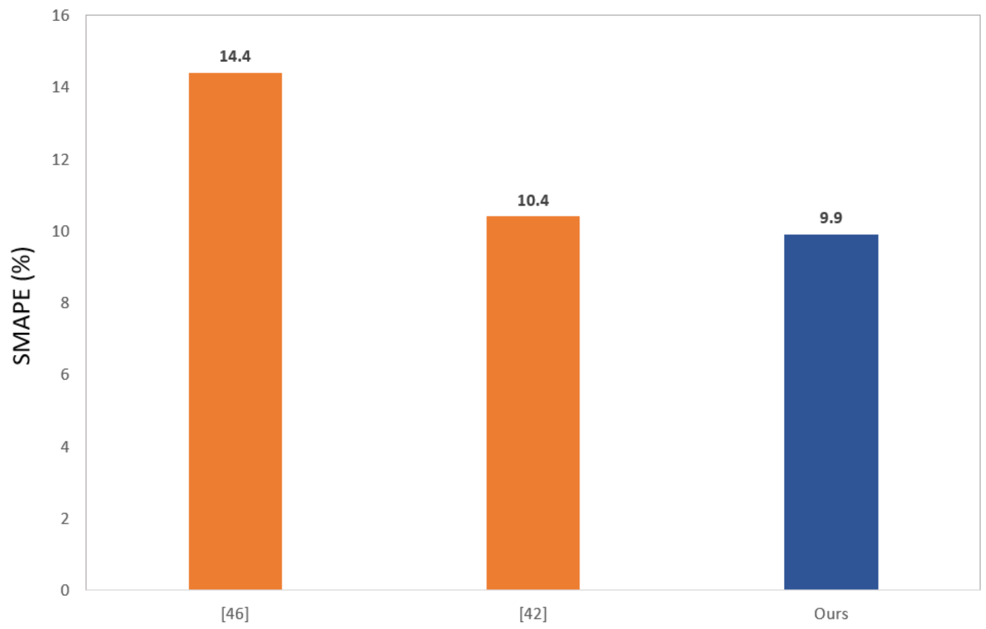


Figure 5.5: Performance comparison of session duration with previous works using SMAPE (lower SMAPE indicates better performance)

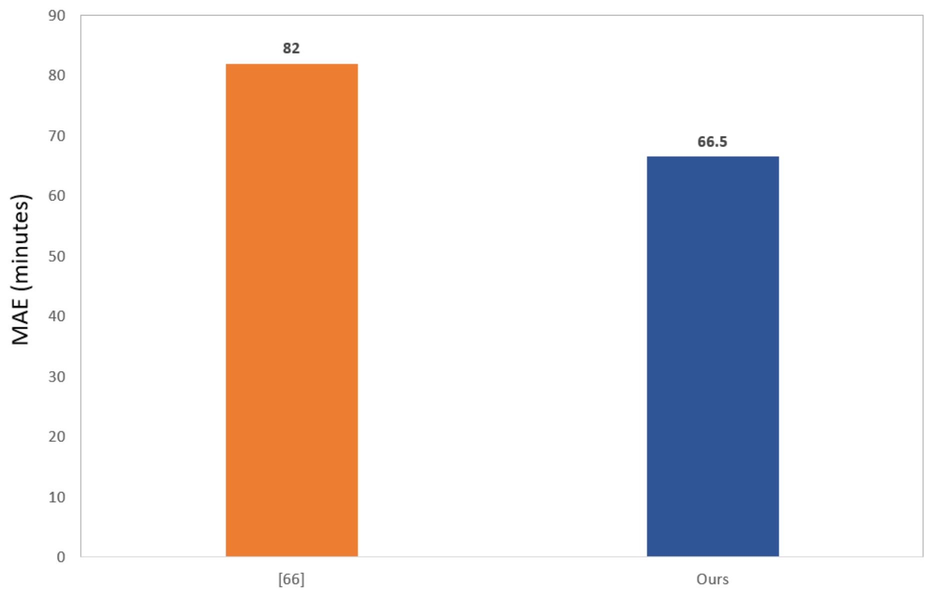


Figure 5.6: Performance comparison of session duration with previous works using MAE (lower MAE indicates better performance)

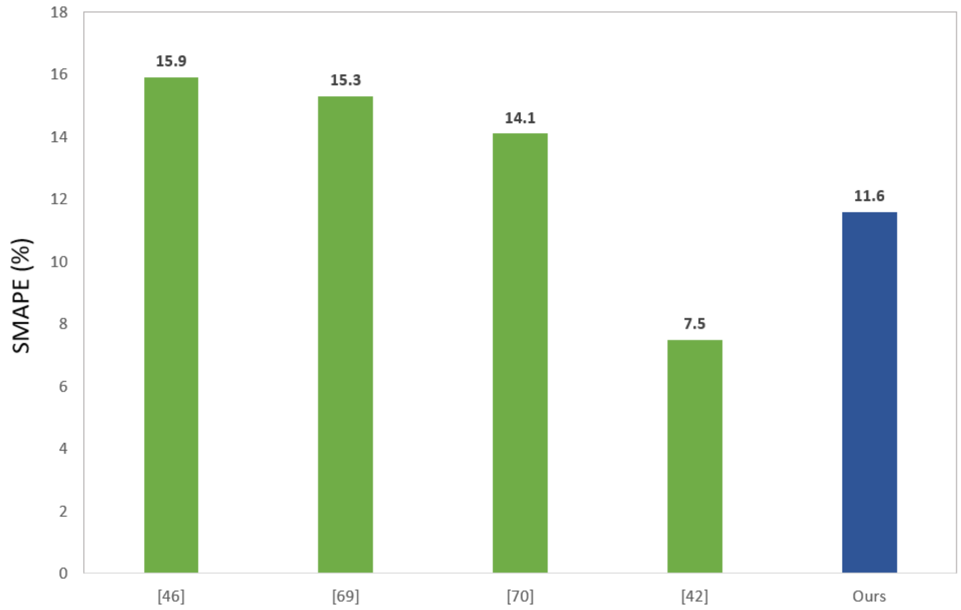


Figure 5.7: Performance comparison of energy consumption with previous works using SMAPE (lower SMAPE indicates better performance)

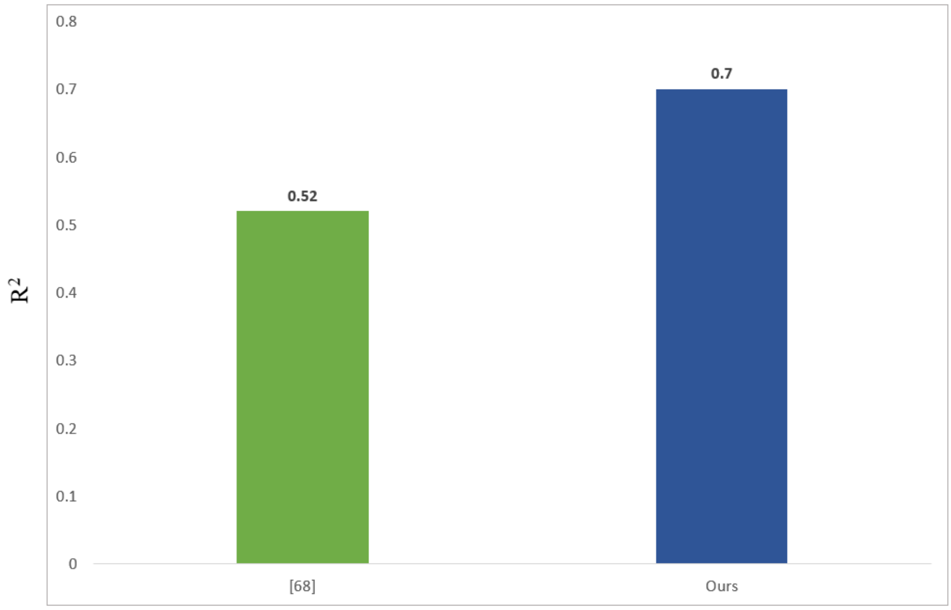


Figure 5.8: Performance comparison of energy consumption with previous works using R2 (higher R2 indicates better performance)

Scheduling remains an important tool for the effective management of EV charging. While we have highlighted the improvement in scheduling and the reduction in charging operating costs by increasing ML prediction accuracies, it would be interesting to determine how scheduling performance would be impacted using the additional features proposed in this study. A conceptual framework of such a system that integrates ML with scheduling is shown in Figure 5.9.

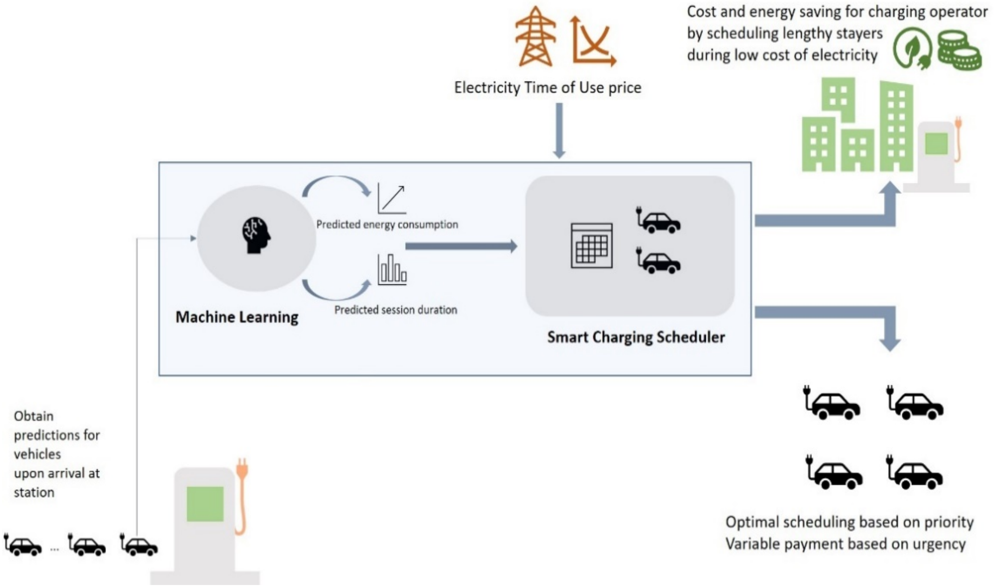


Figure 5.9: Application of the proposed ML algorithm on scheduling for public charging infrastructure

Upon the EV's arrival at the charging station, a pre-trained ML model will compute the predictions regarding the charging behavior of the corresponding vehicle. Specifically, the prediction of the vehicle's expected departure time and energy needs will be obtained. The predictions made by the ML framework can then be fed into the smart charging scheduler for producing optimal scheduling. Public charging facilities usually have more parking spots and charging ports than they can accommodate. For instance, the charging facility in [46] had over 80 charging ports but only with a power capacity of 300 kWh that can support 42 ports at a given time. In peak time and definitely in the future with the increasing number of EVs, we will be presented with a scenario where the charging facility is packed with EVs. In this case, if the behavior of EVs is known beforehand such as their expected stay duration, we can prioritize charging vehicles that will leave the station earlier. One possible way to obtain knowledge of these behaviors is to request the drivers for their estimated departure time and other requirements. However, this method is not realistic because of human bias and the inaccuracy of human predictions was highlighted in Chapter 5. Similarly, if the EV is predicted to be consuming more energy during the charging session than average, it would be more efficient to distribute the charging overtime. As discussed in section 2.4, the scheduler can utilize the predictions made by ML models to reduce overall cost and energy for the charging operator. By using real-time electricity price rate and TOU, the scheduler can reschedule low priority EVs, as determined by the ML to have latest departure time. Therefore, if the current time is high TOU, these low priority vehicles may be rescheduled to have their charges delivered during low TOU. Similarly, EVs with high energy needs and flexible stay duration can be scheduled during low TOU to take advantage of the lower electricity rates. In addition, when the parking space is in high demand, if the ML algorithm predicts the EV to stay for longer hours, the operator can inform the user of additional charges if the vehicle is parked even after it has finished charging. Therefore, by considering dynamic charging load demand through real-time TOU data and electricity price, the EV charging operators can maximize on their profits using the charging behavior predictions from the predictive models.

Chapter 6. Conclusion and Future Work

In this thesis, we presented a framework for the prediction of two of the most important EV charging behaviors with regards to scheduling, namely session duration and energy consumption. Unlike previous work, we took advantage of weather, traffic, and events data along with historical EV charging data. We presented a framework for successfully integrating the additional data with historical charging records. Moreover, we provided detailed preprocessing steps for data formatting and handling outliers. We utilized various feature engineering techniques including trigonometric transformation and feature scaling. Furthermore, we trained six popular ML models along with two ensemble learning algorithms for the prediction of the two charging behaviors. The results obtained in terms of prediction performance was demonstrated to be superior to the results in the previous works. We obtained a SMAPE score of 9.92% in session duration predictions and a SMAPE score of 11.6% on energy consumption predictions. We have also provided a significant improvement in charging behavior prediction on the ACN dataset and demonstrated the potential of utilizing traffic and weather information in charging behavior prediction.

We have quantitatively shown in the previous chapter that the traffic and weather variables are important predictors in EV charging behavior, particularly in the case of session duration. Although the use of local events data (campus events in this case) had an insignificant impact in terms of performance gain, it cannot be ruled out for future work. In this work, we obtained all campus events from the Caltech university calendar. However, perhaps only the major events that generally draw more crowds to the campus should be taken into consideration. It is also possible that events data may not impact EV charging behavior in a university campus significantly. However, for other public spaces such as shopping malls, for example, events such as the end of the year sale could be important predictors. Therefore, similar experiments on other public charging spaces should be carried out to determine the impact of local events.

Social media can also be explored as a means to obtain information about local events as well as EV driving behavior. For example, a larger social media activity around the charging facility compared to usual could indicate a special event or occasion. It may also be possible to use social media data directly for predicting user behavior if the data can be associated with the user ID. However, this will require the user to register with the charging facility mobile application using a social media account and can raise privacy concerns. The use of social media nonetheless has potential for charging behavior prediction. For instance, social media is an effective tool for estimating human behavior [97] and is also a significant predictor of truck drivers' travel time [98].

Moreover, it is likely that the use of vehicle information such as the vehicle model and vehicle type can improve predictions, especially in terms of energy consumption. This is because a particular EV model will have a fixed charging rate and battery capacity. This is also accurate for the type of vehicle being charged. An electric bus, for instance, will probably require a greater charging time than a light EV. The use of vehicle information has been supported by previous research works. For example, one of the previous works utilized vehicle information [66] but not in conjunction with the weather, traffic, and events data. Therefore, this work can be extended to include vehicle information by requesting the user to enter their vehicle details when registering with the mobile application. This user input is likely to be accurate because it is only requested once during registration.

Furthermore, it is worth investigating the use of clustering algorithms and utilizing the results obtained from those algorithms as an input feature into the supervised learning model. This can be implemented using a PSF-based approach such as the one proposed in [71] and other similar approaches. The use of clustering will help by first narrowing down similar groups of users to a specific category. The ML predictions can then be obtained within that particular category. Consequently, the category the user belongs to can also be used as an additional input feature to the model. It is worth mentioning that this approach does not add any new information, but rather serves as an alternative feature representation for the models.

Moreover, in this work, we considered nine of the closest roads and streets in terms of geographical proximity to the charging facility. However, given the importance of traffic data, it is worth looking at a greater number of roads and streets across the city to further enhance the performance. An in-depth study could be carried out to determine the optimal number of roads and streets to consider for a particular area.

Prediction of idle time can also be considered as an additional charging behavior. This is the time between the departure time and the time when the vehicle has been fully charged. Many drivers may decide to stay connected even after their vehicle has been fully charged. If ML algorithms can be trained to predict the idle time, we can prioritize charging vehicles that have shorter idle times historically and control the charging rate of vehicles with long idle times. This prediction can be easily integrated into the conceptual framework shown in Figure 5.9.

Furthermore, we have only considered charging sessions that were recorded by user ID. Although in this case, such records represented an overwhelming majority of the dataset, we cannot ignore the predictions of sessions that have no ID associated. These sessions could belong to non-regular campus visitors who do not have their vehicles registered with the mobile application. Therefore, a separate predictive model can be trained to predict the charging behavior of sessions without any historical data.

In this work, we considered six popular ML models including one deep learning model. It is also worth investigating with time-series models such as ARIMA as well as probabilistic models and perhaps integrating their predictions into the ensemble learning models. Moreover, recurrent neural networks that consider historical information by using memory blocks can prove to be useful in this application. Although less significant than public charging, the proposed approach could be used to determine the impacts of weather, traffic, and events on residential charging behavior. Finally, we only considered data for the year 2019 in this study. Therefore, this work can be further extended for 2020 and 2021 to study the EV charging behavior during the COVID-19 pandemic.

References

[1] United Nations Statistical Commission and others, “Global indicator framework for the Sustainable Development Goals and targets of the 2030 Agenda for Sustainable Development,” UN Resolution A/RES/71/313, 2017. Accessed: Apr. 22, 2021. [Online]. Available: https://unstats.un.org/sdgs/indicators/Global%20Indicator%20Framework%20after%20refinement_Eng.pdf

[2] “Climate Emergency Declaration and Mobilisation In Action,” *CEDAMIA*. https://www.cedamia.org/global/ (accessed Jan. 17, 2021).

[3] “Key World Energy Statistics 2018 – Analysis,” IEA. https://www.iea.org/reports/key-world-energy-statistics-2019 (accessed Jun. 01, 2020).

[4] “68% of the world population projected to live in urban areas by 2050, says UN,” UN DESA | United Nations Department of Economic and Social Affairs, May 16, 2018. https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html (accessed Jun. 01, 2020).

[5] X. Zhang, F. Gao, X. Gong, Z. Wang, and Y. Liu, “Comparison of Climate Change Impact Between Power System of Electric Vehicles and Internal Combustion Engine Vehicles,” in *Advances in Energy and Environmental Materials*, Singapore, 2018, pp. 739–747.

[6] “Global EV Outlook 2019 – Analysis,” IEA. https://www.iea.org/reports/global-ev-outlook-2019 (accessed Jun. 01, 2020).

[7] T. Franke, N. Rauh, and J. F. Krems, “Individual differences in BEV drivers’ range stress during first encounter of a critical range situation,” *Applied Ergonomics*, vol. 57, pp. 28–35, 2016, doi: https://doi.org/10.1016/j.apergo.2015.09.010.

[8] Y. Kwon, S. Son, and K. Jang, “User satisfaction with battery electric vehicles in South Korea,” *Transportation Research Part D: Transport and Environment*, vol. 82, p. 102306, 2020, doi: https://doi.org/10.1016/j.trd.2020.102306.

[9] “Global EV Outlook 2020 – Analysis,” IEA. https://www.iea.org/reports/global-ev-outlook-2020 (accessed Mar. 21, 2021).

[10] J. Munkhammar, J. D. K. Bishop, J. J. Sarralde, W. Tian, and R. Choudhary, “Household electricity use, electric vehicle home-charging and distributed photovoltaic power production in the city of Westminster,” *Energy and Buildings*, vol. 86, pp. 439–448, 2015, doi: https://doi.org/10.1016/j.enbuild.2014.10.006.

[11] K. Clement-Nyns, E. Haesen, and J. Driesen, “The Impact of Charging Plug-In Hybrid Electric Vehicles on a Residential Distribution Grid,” *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 371–380, 2010.

[12] Z. Moghaddam, I. Ahmad, D. Habibi, and Q. V. Phung, “Smart Charging Strategy for Electric Vehicle Charging Stations,” *IEEE Transactions on Transportation Electrification*, vol. 4, no. 1, pp. 76–88, 2018.

[13] M. Ş. Kuran, A. C. Viana, L. Iannone, D. Kofman, G. Mermoud, and J. P. Vasseur, “A Smart Parking Lot Management System for Scheduling the Recharging of Electric Vehicles,” *IEEE Transactions on Smart Grid*, vol. 6, no. 6, pp. 2942–2953, 2015.

[14] A. Foley, B. Tyther, P. Calnan, and B. [Ó Gallachóir, “Impacts of Electric Vehicle charging under electricity market operations,” *Applied Energy*, vol. 101, pp. 93–102, 2013, doi: https://doi.org/10.1016/j.apenergy.2012.06.052.
[15] T. Franke and J. F. Krems, “Understanding charging behaviour of electric vehicle users,” *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 21, pp. 75–89, 2013, doi: https://doi.org/10.1016/j.trf.2013.09.002.
[16] L. Hu, J. Dong, and Z. Lin, “Modeling charging behavior of battery electric vehicle drivers: A cumulative prospect theory based approach,” *Transportation Research Part C: Emerging Technologies*, vol. 102, pp. 474–489, 2019, doi: https://doi.org/10.1016/j.trc.2019.03.027.
[17] J. Spoelstra, “Charging behaviour of Dutch EV drivers,” Master’s Thesis, Utrecht University, Utrecht, Netherlands, 2014.
[18] R. Bi, J. Xiao, V. Viswanathan, and A. Knoll, “Influence of charging behaviour given charging station placement at existing petrol stations and residential car park locations in Singapore,” *Procedia Computer Science*, vol. 80, pp. 335–344, 2016.
[19] P. Liu and J. Yu, “Charging behavior characteristic simulation of plug-in electric vehicles for demand response,” in *2016 UKACC 11th International Conference on Control (CONTROL)*, Belfast, UK, 2016, pp. 1–6.
[20] P. Grahn, K. Alvehag, and L. Söder, “PHEV Utilization Model Considering Type-of-Trip and Recharging Flexibility,” *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 139–148, 2014.
[21] J. Brady and M. O’Mahony, “Modelling charging profiles of electric vehicles based on real-world electric vehicle charging data,” *Sustainable Cities and Society*, vol. 26, pp. 203–216, 2016, doi: https://doi.org/10.1016/j.scs.2016.06.014.
[22] O. Frendo, S. Karnouskos, N. Gaertner, O. Kipouridis, K. Rehman, and N. Verzano, “Charging Strategies and Implications for Corporate Electric Vehicle Fleets,” in *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, Porto, Portugal, 2018, pp. 466–471.
[23] Y. Cao et al., “Mobile Edge Computing for Big-Data-Enabled Electric Vehicle Charging,” *IEEE Communications Magazine*, vol. 56, no. 3, pp. 150–156, 2018.
[24] N. K. Ahmed, A. F. Atiya, N. E. Gayar, and H. El-Shishiny, “An empirical comparison of machine learning models for time series forecasting,” *Econometric Reviews*, vol. 29, no. 5–6, pp. 594–621, 2010.
[25] K. Morrow, D. Karner, and J. Francfort, “Plug-in hybrid electric vehicle charging infrastructure review,” *US Department of Energy-Vehicle Technologies Program Final Rep. 58517*, 2008.
[26] S. Shahriar, A. R. Al-Ali, A. H. Osman, S. Dhou, and M. Nijim, “Machine Learning Approaches for EV Charging Behavior: A Review,” *IEEE Access*, vol. 8, pp. 168980–168993, 2020, doi: 10.1109/ACCESS.2020.3023388.
[27] S. Ruder, “An overview of gradient descent optimization algorithms,” *arXiv:1609.04747*, 2016.
[28] D. M. Hawkins, “The Problem of Overfitting,” *J. Chem. Inf. Comput. Sci.*, vol. 44, no. 1, pp. 1–12, Jan. 2004, doi: 10.1021/ci0342472.
[29] A. Liaw, M. Wiener, and others, “Classification and regression by randomForest,” *R news*, vol. 2, no. 3, pp. 18–22, 2002.

[30] R. Polikar, “Ensemble based systems in decision making,” *IEEE Circuits and Systems Magazine*, vol. 6, no. 3, pp. 21–45, 2006.

[31] J. H. Friedman, “Greedy Function Approximation: A Gradient Boosting Machine,” *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[32] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, USA, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[33] D. Nielsen, “Tree boosting with xgboost-why does xgboost win" every" machine learning competition?,” Master’s Thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2016.

[34] C. J. C. Burges, “A Tutorial on Support Vector Machines for Pattern Recognition,” *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, Jun. 1998, doi: 10.1023/A:1009715923555.

[35] D. Basak, S. Pal and D. C. Patranabis, "Support vector regression", *Neural Inf. Process. Lett. Rev.*, vol. 11, no. 10, pp. 203-224, Oct. 2007.

[36] T. Cover and P. Hart, “Nearest neighbor pattern classification,” *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[37] D. Opitz and R. Maclin, “Popular ensemble methods: An empirical study,” *Journal of artificial intelligence research*, vol. 11, pp. 169–198, 1999.

[38] T. M. Kodinariya and P. R. Makwana, “Review on determining number of Cluster in K-Means Clustering,” *International Journal*, vol. 1, no. 6, pp. 90–95, 2013.

[39] C. E. Rasmussen, “The infinite Gaussian mixture model.,” in *Advances in Neural Information Processing Systems*, 2000, vol. 12, pp. 554–560.

[40] Z. I. Botev, J. F. Grotowski, D. P. Kroese, and others, “Kernel density estimation via diffusion,” *The annals of Statistics*, vol. 38, no. 5, pp. 2916–2957, 2010.

[41] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. The MIT Press, 2016.

[42] Y.-W. Chung, B. Khaki, T. Li, C. Chu, and R. Gadh, “Ensemble machine learning-based algorithm for electric vehicle user behavior prediction,” *Applied Energy*, vol. 254, p. 113732, 2019, doi: https://doi.org/10.1016/j.apenergy.2019.113732.

[43] C. Develder, N. Sadeghianpourhamami, M. Strobbe, and N. Refa, “Quantifying flexibility in EV charging as DR potential: Analysis of two real-world data sets,” in *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Sydney, NSW, Australia, 2016, pp. 600–605, doi: 10.1109/SmartGridComm.2016.7778827.

[44] J. D. Cross and R. Hartshorn, "My Electric Avenue: Integrating electric vehicles into the electrical networks", *6th Hybrid and Electric Vehicles Conference (HEVC 2016)*, London, UK, pp. 1-6, 2016.

[45] “Pecan Street Data,” Pecan Street Inc. https://www.pecanstreet.org/dataport/about/ (accessed Jun. 12, 2020).

[46] Z. J. Lee, T. Li, and S. H. Low, “ACN-Data: Analysis and Applications of an Open EV Charging Dataset,” in *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, New York, USA, 2019, pp. 139–149, doi: 10.1145/3307772.3328313.

[47] J. R. Helmus, M. H. Lees, and R. [van den Hoed, “A data driven typology of electric vehicle user types and charging sessions,” *Transportation Research Part C: Emerging Technologies*, vol. 115, p. 102637, 2020, doi: https://doi.org/10.1016/j.trc.2020.102637.
[48] J. Quirós-Tortós, A. N.- Espinosa, L. F. Ochoa, and T. Butler, “Statistical Representation of EV Charging: Real Data Analysis and Applications,” in *2018 Power Systems Computation Conference (PSCC)*, Dublin, Ireland, 2018, pp. 1–7, doi: 10.23919/PSCC.2018.8442988.
[49] M. G. Flammini, G. Prettico, A. Julea, G. Fulli, A. Mazza, and G. Chicco, “Statistical characterisation of the real transaction data gathered from electric vehicle charging stations,” *Electric Power Systems Research*, vol. 166, pp. 136–150, 2019, doi: https://doi.org/10.1016/j.epsr.2018.09.022.
[50] N. Sadeghianpourhamami, N. Refa, M. Strobbe, and C. Develder, “Quantitive analysis of electric vehicle flexibility: A data-driven approach,” *International Journal of Electrical Power & Energy Systems*, vol. 95, pp. 451–462, 2018, doi: https://doi.org/10.1016/j.ijepes.2017.09.007.
[51] Y. Xiong, B. Wang, C. Chu, and R. Gadh, “Electric Vehicle Driver Clustering using Statistical Model and Machine Learning,” in *2018 IEEE Power Energy Society General Meeting (PESGM)*, Portland, OR, USA, 2018, pp. 1–5.
[52] Y. Shen, W. Fang, F. Ye, and M. Kadoch, “EV Charging Behavior Analysis Using Hybrid Intelligence for 5G Smart Grid,” *Electronics*, vol. 9, no. 1, p. 80, 2020.
[53] C. Sun, T. Li, S. H. Low, and V. O. K. Li, “Classification of electric vehicle charging time series with selective clustering,” *Electric Power Systems Research*, vol. 189, p. 106695, 2020, doi: https://doi.org/10.1016/j.epsr.2020.106695.
[54] A. Gerossier, R. Girard, and G. Kariniotakis, “Modeling and Forecasting Electric Vehicle Consumption Profiles,” *Energies*, vol. 12, no. 7, 2019, doi: 10.3390/en12071341.
[55] Y. Xiong, B. Wang, C. Chu, and R. Gadh, “Vehicle grid integration for demand response with mixture user model and decentralized optimization,” *Applied Energy*, vol. 231, pp. 481–493, 2018, doi: https://doi.org/10.1016/j.apenergy.2018.09.139.
[56] E. Xydas, C. Marmaras, L. M. Cipcigan, N. Jenkins, S. Carroll, and M. Barker, “A data-driven approach for characterising the charging demand of electric vehicles: A UK case study,” *Applied Energy*, vol. 162, pp. 763–771, 2016, doi: https://doi.org/10.1016/j.apenergy.2015.10.151.
[57] S. Ai, A. Chakravorty, and C. Rong, “Household EV Charging Demand Prediction Using Machine and Ensemble Learning,” in *2018 IEEE International Conference on Energy Internet (ICEI)*, Beijing, China, 2018, pp. 163–168, doi: 10.1109/ICEI.2018.00037.
[58] Y. Yang, Z. Tan, and Y. Ren, “Research on Factors That Influence the Fast Charging Behavior of Private Battery Electric Vehicles,” *Sustainability*, vol. 12, no. 8, 2020, doi: 10.3390/su12083439.
[59] S. Venticinque and S. Nacchia, “Learning and Prediction of E-Car Charging Requirements for Flexible Loads Shifting,” in *Internet and Distributed Computing Systems*, Naples, Italy, 2019, pp. 284–293.

[60] O. Frendo, J. Graf, N. Gaertner, and H. Stuckenschmidt, “Data-driven smart charging for heterogeneous electric vehicle fleets,” *Energy and AI*, vol. 1, p. 100007, 2020, doi: https://doi.org/10.1016/j.egyai.2020.100007.

[61] J. J. Mies, J. R. Helmus, and R. Van den Hoed, “Estimating the Charging Profile of Individual Charge Sessions of Electric Vehicles in The Netherlands,” *World Electric Vehicle Journal*, vol. 9, no. 2, 2018, doi: 10.3390/wevj9020017.

[62] Y. Lu et al., “The Application of Improved Random Forest Algorithm on the Prediction of Electric Vehicle Charging Load,” *Energies*, vol. 11, no. 11, 2018, doi: 10.3390/en11113207.

[63] M. H. Amini, A. Kargarian, and O. Karabasoglu, “ARIMA-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation,” *Electric Power Systems Research*, vol. 140, pp. 378–390, 2016, doi: https://doi.org/10.1016/j.epsr.2016.06.003.

[64] Z. Xu, “Forecasting electric vehicle arrival & departure time on UCSD campus using support vector machines,” PhD Thesis, UC San Diego, San Diego, CA, USA, 2017.

[65] K. Duan, S. S. Keerthi, and A. N. Poo, “Evaluation of simple performance measures for tuning SVM hyperparameters,” *Neurocomputing*, vol. 51, pp. 41–59, 2003, doi: https://doi.org/10.1016/S0925-2312(02)00601-X.

[66] O. Frendo, N. Gaertner, and H. Stuckenschmidt, “Improving Smart Charging Prioritization by Predicting Electric Vehicle Departure Time,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–8, 2020, doi: 10.1109/TITS.2020.2988648.

[67] Y. Xiong, C. Chu, R. Gadh, and B. Wang, “Distributed optimal vehicle grid integration strategy with user behavior prediction,” in *2017 IEEE Power Energy Society General Meeting*, Chicago, IL, USA, 2017, pp. 1–5, doi: 10.1109/PESGM.2017.8274327.

[68] A. Almaghrebi, F. Aljuheshi, M. Rafaie, K. James, and M. Alahmad, “Data-Driven Charging Demand Prediction at Public Charging Stations Using Supervised Machine Learning Regression Methods,” *Energies*, vol. 13, no. 16, p. 4231, 2020.

[69] M. Majidpour, C. Qiu, P. Chu, R. Gadh, and H. R. Pota, “Fast Prediction for Sparse Time Series: Demand Forecast of EV Charging Stations for Cell Phone Applications,” *IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 242–250, 2015, doi: 10.1109/TII.2014.2374993.

[70] M. Majidpour, C. Qiu, P. Chu, R. Gadh, and H. R. Pota, “A novel forecasting algorithm for electric vehicle charging stations,” in *2014 International Conference on Connected Vehicles and Expo (ICCVE)*, Vienna, Austria, 2014, pp. 1035–1040, doi: 10.1109/ICCVE.2014.7297504.

[71] N. Bokde, M. W. Beck, F. M. Álvarez, and K. Kulat, “A novel imputation methodology for time series based on pattern sequence forecasting,” *Pattern Recognition Letters*, vol. 116, pp. 88–96, 2018, doi: https://doi.org/10.1016/j.patrec.2018.09.020.

[72] B. Khaki, Y. Chung, C. Chu, and R. Gadh, “Nonparametric User Behavior Prediction for Distributed EV Charging Scheduling,” in *2018 IEEE Power Energy Society General Meeting (PESGM)*, Portland, OR, USA, 2018, pp. 1–5.

[73] Y. Chung, B. Khaki, C. Chu, and R. Gadh, “Electric Vehicle User Behavior Prediction Using Hybrid Kernel Density Estimator,” in *2018 IEEE*

International Conference on Probabilistic Methods Applied to Power Systems (PMAPS), Boise, ID, USA, 2018, pp. 1–6.

[74] “ACN-Data -- A Public EV Charging Dataset.” https://ev.caltech.edu/dataset (accessed Mar. 03, 2021).

[75] “TCCON Weather.” http://tccon-weather.caltech.edu/ (accessed Mar. 03, 2021).

[76] R. Gelaro et al., “The Modern-Era Retrospective Analysis for Research and Applications, Version 2 (MERRA-2),” *Journal of Climate*, vol. 30, no. 14, pp. 5419–5454, Jul. 2017, doi: 10.1175/JCLI-D-16-0758.1.

[77] R. Mendelsohn, P. Kurukulasuriya, A. Basist, F. Kogan, and C. Williams, “Climate analysis with satellite versus weather station data,” *Climatic Change*, vol. 81, no. 1, pp. 71–83, Mar. 2007, doi: 10.1007/s10584-006-9139-x.

[78] G. Leduc and others, “Road traffic data: Collection methods and applications,” *Working Papers on Energy, Transport and Climate Change*, vol. 1, no. 55, pp. 1–55, 2008.

[79] “Pasadena Traffic Count Website.” https://data.cityofpasadena.net/datasets/eaaffc1269994f0e8966e2024647cc56 (accessed Jan. 21, 2021).

[80] F. Goudarzi, “Travel Time Prediction: Comparison of Machine Learning Algorithms in a Case Study,” in *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, Exeter, UK, 2018, pp. 1404–1407, doi: 10.1109/HPCC/SmartCity/DSS.2018.00232.

[81] D. Barth, The Bright Side of Sitting in Traffic: Crowdsourcing Road Congestion Data, 2009, [online] Available: http://googleblog.blogspot.ca/2009/08/bright-side-of-sitting-in-traffic.html.

[82] “Calendar | www.caltech.edu.” https://www.caltech.edu/campus-life-events/master-calendar (accessed Mar. 03, 2021).

[83] J. W. Tukey, Exploratory Data Analysis, Boston, MA, USA:Addison-Wesley, 1977.

[84] F. T. Liu, K. M. Ting, and Z. Zhou, “Isolation Forest,” in *2008 Eighth IEEE International Conference on Data Mining*, Pisa, Italy, 2008, pp. 413–422, doi: 10.1109/ICDM.2008.17.

[85] S. Bishop, Pytz–world timezone definitions for Python. Online][Cited: May 25, 2012.] http://pytz. sourceforge. net, 2016.

[86] W. McKinney and others, “pandas: a foundational Python library for data analysis and statistics,” *Python for High Performance and Scientific Computing*, vol. 14, no. 9, pp. 1–9, 2011.

[87] H. Wickham et al., “ggplot2: create elegant data visualisations using the grammar of graphics,” *R package version*, vol. 2, no. 1, 2016.

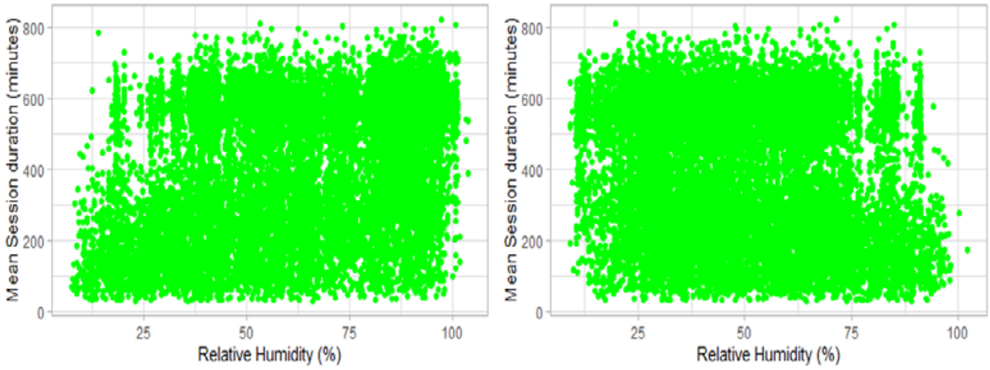
[88] Y. Bengio, A. Courville, and P. Vincent, “Representation Learning: A Review and New Perspectives,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013, doi: 10.1109/TPAMI.2013.50.

[89] P. Juszczak, D. Tax and R. P. W. Duin, "Feature scaling in support vector data description", in *ASCI 2002 8th Annual Conf. of the Advanced School for Computing and Imaging*, Lochem, Netherlands, 2002, pp. 95-102.

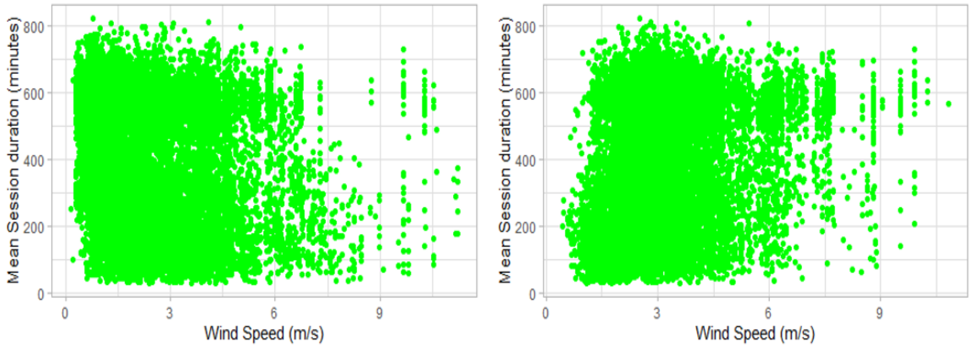
[90] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.
[91] Y. Bengio and Y. Grandvalet, “No unbiased estimator of the variance of k-fold cross-validation,” *Journal of machine learning research*, vol. 5, no. Sep, pp. 1089–1105, 2004.
[92] Qiujun Huang, Jingli Mao, and Yong Liu, “An improved grid search algorithm of SVR parameters optimization,” in *2012 IEEE 14th International Conference on Communication Technology*, Chengdu, China, 2012, pp. 1022–1026, doi: 10.1109/ICCT.2012.6511415.
[93] D. H. Wolpert, “Stacked generalization,” *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992, doi: https://doi.org/10.1016/S0893-6080(05)80023-1.
[94] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines", *Proc. 27th Int. Conf. Mach. Learn.*, Haifa, Israel, pp. 807-814, 2010.
[95] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv:1412.6980*, 2014.
[96] B. Huchuk, S. Sanner, and W. O’Brien, “Comparison of machine learning models for occupancy prediction in residential buildings using connected thermostat data,” *Building and Environment*, vol. 160, p. 106177, 2019, doi: https://doi.org/10.1016/j.buildenv.2019.106177.
[97] M.-A. Abbasi, S.-K. Chai, H. Liu, and K. Sagoo, “Real-world behavior analysis through a social media lens,” in *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*, Maryland, USA, 2012, pp. 18–26.
[98] D. Yuniar, L. Djakfar, A. Wicaksono, and A. Efendi, “Truck driver behavior and travel time effectiveness using smart GPS,” *Civil Engineering Journal*, vol. 6, no. 4, pp. 724–732, 2020.

Appendix A

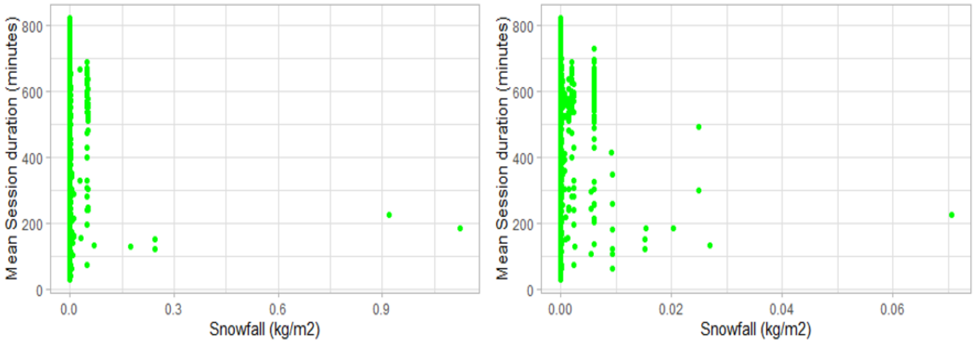
In this section, we present the figures related to the exploratory data analysis where no significant relationship between the variables were observed. For both session duration and energy consumption, the respective figures for the humidity, wind speed, and snowfall are presented from Appendix 1 to Appendix 6.



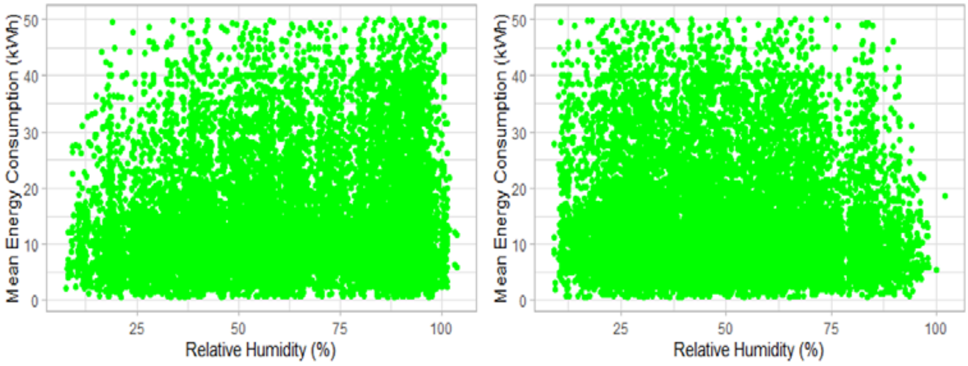
Appendix 1 Previous humidity (left) and next humidity (right) against session duration



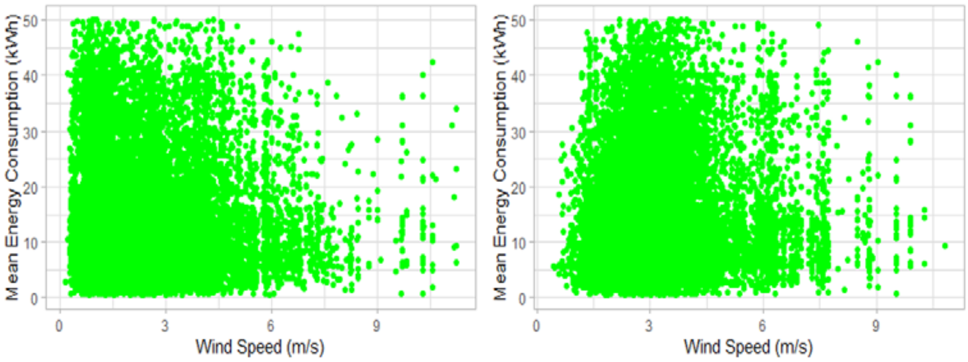
Appendix 2 Previous wind speed (left) and next wind speed (right) against session duration



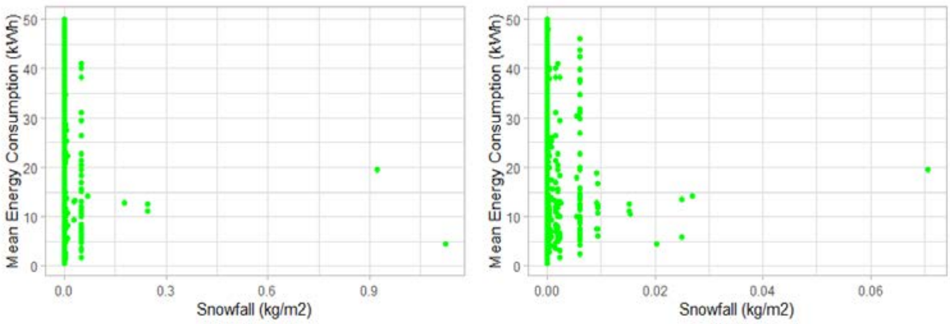
Appendix 3 Previous snowfall (left) and next snowfall (right) against session duration



Appendix 4 Previous humidity (left) and next humidity (right) against energy consumption



Appendix 5 Previous wind speed (left) and next wind speed (right) against energy consumption



Appendix 6 Previous snowfall (left) and next snowfall (right) against energy consumption

Vita

Sakib Shahriar received his Bachelors of Science in Computer Engineering from American University of Sharjah, U.A.E in December 2018. During his undergraduate program, he specialized in internet of things (IoT) and unsupervised machine learning.

In January 2019, he joined the Computer Engineering master's program in the American University of Sharjah as a graduate teaching assistant. He was also employed as a graduate research assistant from September 2019. During this time, he authored several journal and conference papers in machine learning and deep learning.