Electric Vehicle to Grid (eV2G) Price Forecasting

Timothy Rampiaray   
Department of ECE UCF  
tarampia@knights.ucf.edu  
Assignment4

*Abstract*—The demand for power within grid system’s due to the rise in battery electric vehicles (BEVs), plug-in battery electric vehicles (PHEVs), and extended range electric vehicles (EREVs). This paper looks at the dynamic state of a 33 Bus network over a 24-hour period of 1800 vehicles. A control algorithm will be hypothesized to dynamically control price at certain areas of the bus network during high load times.

# Introduction

In a liberalized market, players will need to change their strategies and how the act upon the new energy demands that will be required in a grid. In Power Systems (PS) there are many resources available that will have to be managed appropriately. These resources include Demand Response (DR), Renewable Energy (RE), Distributed Generation (DG) and systems of storage have been gaining traction.

A prominent replacement for the Internal Combustion Engine is Electric Vehicles (EV). The advantage of EVs is there ability to reduce overall CO2 emissions. In the Vehicle-to-Grid (V2G) concept EVs can be used. The optimization of a grid with some many different factors and the multitude of energy resources can turn into a large combinatorial problem.

For the given simulated network bus an algorithm will be developed to simulate the dynamic price change at specific bus network points for EVs to charge

.

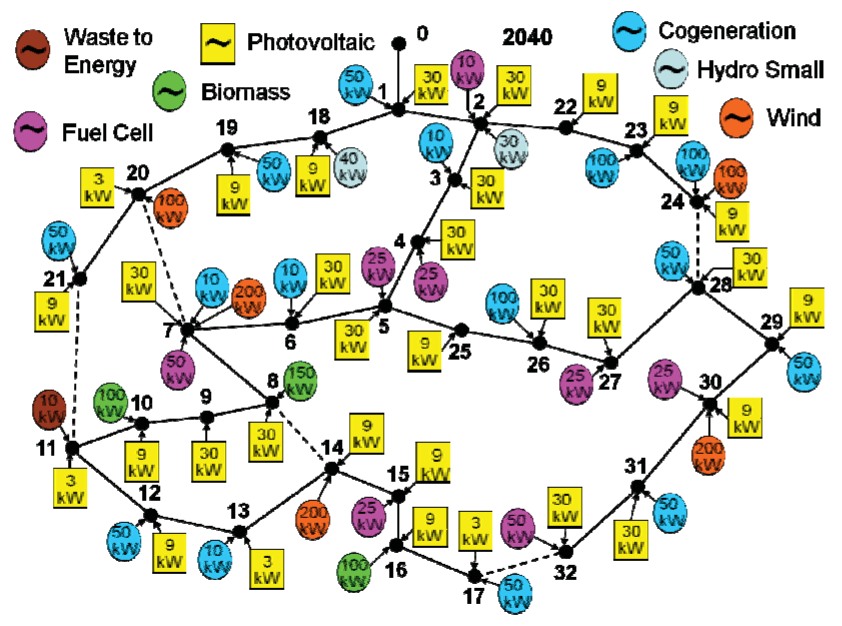
# Case study

## 33-Bus Network

The 33-Bus Network [1], will have elements such as Waste to Energy, Photovoltaic, Biomass, Fuel Cell, Hydro, Wind and Cogeneration. This replicates a configuration of a network that may be seen in the year 2040 (*Figure 1.*). The peak load for this 33-Bus MV sized network is 4.36 MVA.

VEHICLE 
CLASS 
DESCRIPTION 
• wneets. with a rnaximum 
mass Of 400kg or 550kg for a 
vehicle (not irau&ng the of 
the batteries in an electrically gx:nvered 
vehicle) and a maximum Mt 
whatever of or of 
15kW 
vehicle. four wheels and up 8 
Seats in addition to the driver'S Wat_ 
vehicle. a 
maximum laden mass o' 
vehicle. a 
mass between 3,5Wkg and 
12,000kg. 
5: vehicle classes 

*Table 1. Vehicles used in simulation tests*



*Figure 1. Configuration of network in 2040*

## Data

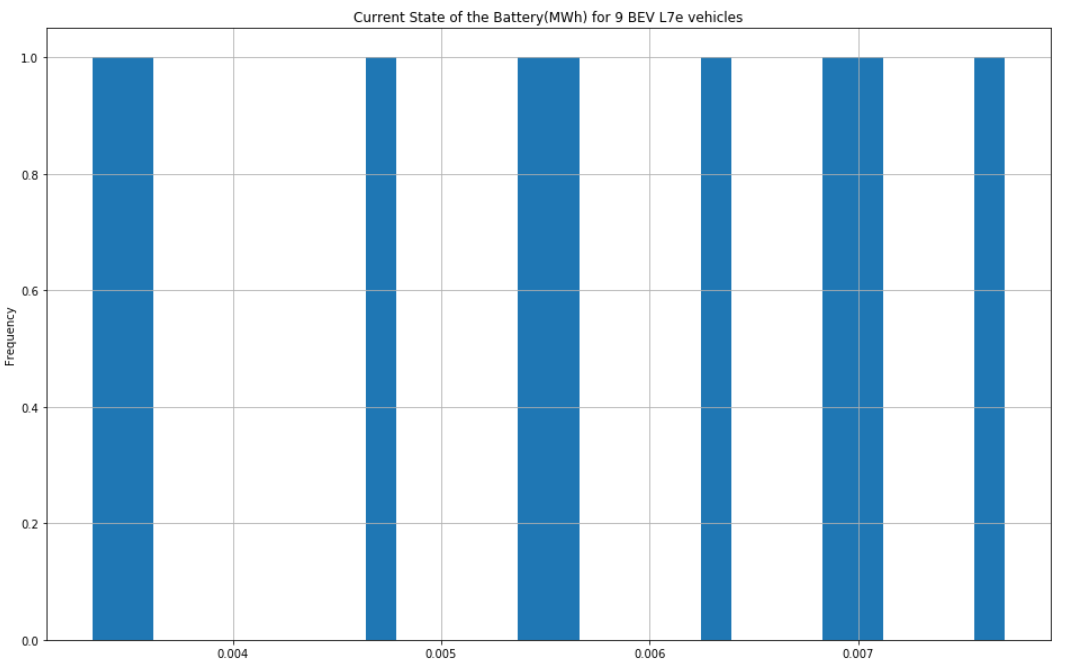
The data consists of 1800 real world vehicles[2] (*Table 1.*) that would will be entering and leaving different locations on the bus network over a 24 hour period. The eights types of vehicle specs can be seen in the table below. (*Table 2.*) The simulated data for the 24-Hour period also contains the initial state of the battery capacity for all 1800 vehicles.

Table 1. EV battery specifications 
Battery capacity (kWh) 
Charging rates (kW) 
Vehicle class 
Ml 
BEV 
L7e 
Ml 
PHEV 
Ml 
EREV 
Max 
120 
15 
13.6 
13.6 
226 
226 
Mean 
29 
8.7 
8.2 
17 
17 
Min 
10 
51 
2.2 
12 
12 
Slow charge rate 
2-8.8 
1.3-3.3 
10 
1-3 
3-5.3 
3-5.3 
Fast charge rate 
3-240 
10-45 
35-60 
3—7.5 
11 
11 

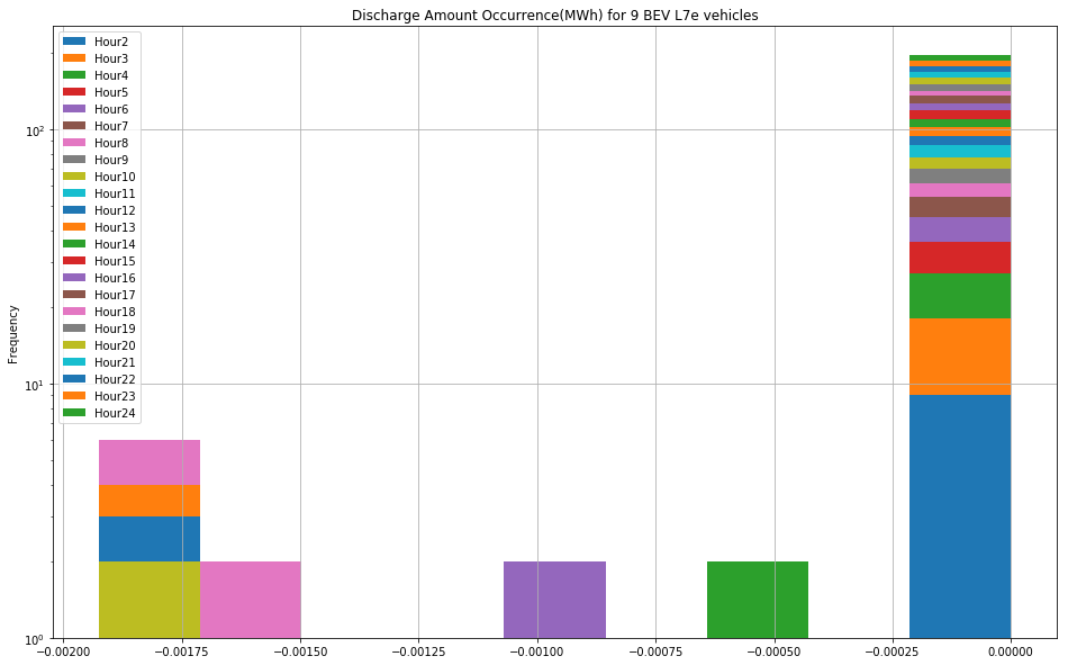
*Table 1. Battery Capacity of vehicles used*

# Data preparation and visualization

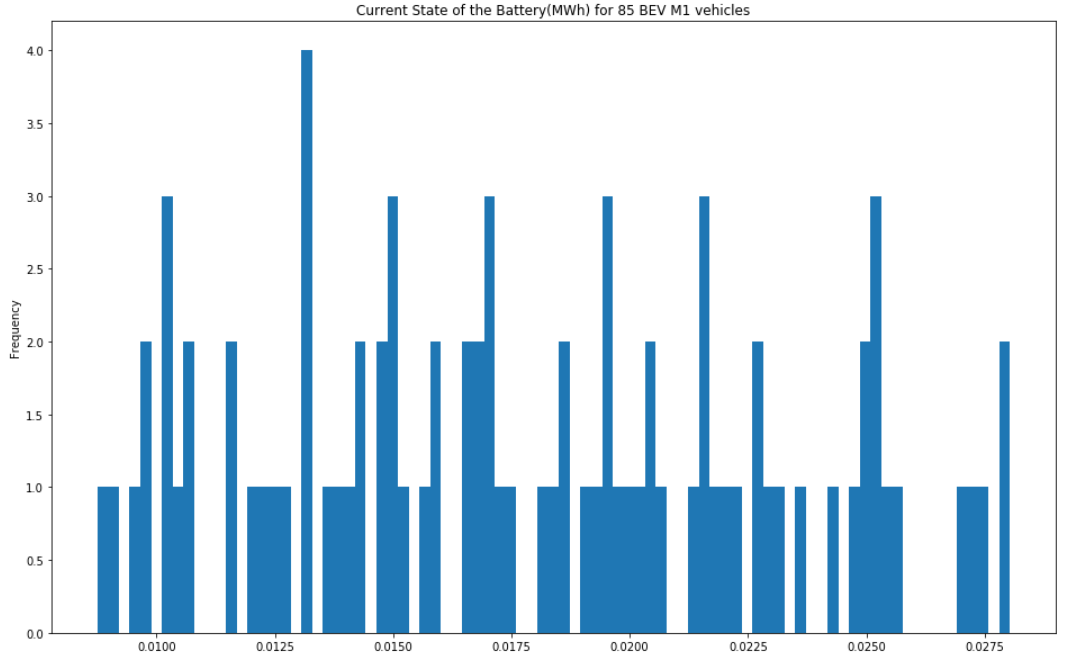
The data for the vehicles within the network can be visualized for the 24-Hour period. The following figures represent the data for the four types of electric vehicles. Figures (2,4,6,8) represent the Current State of the Battery for specific EVs. The Figures (3,5,7,9) represent the various frequency of power discharges over the course of the 24-Hour period for the corresponding EVs.



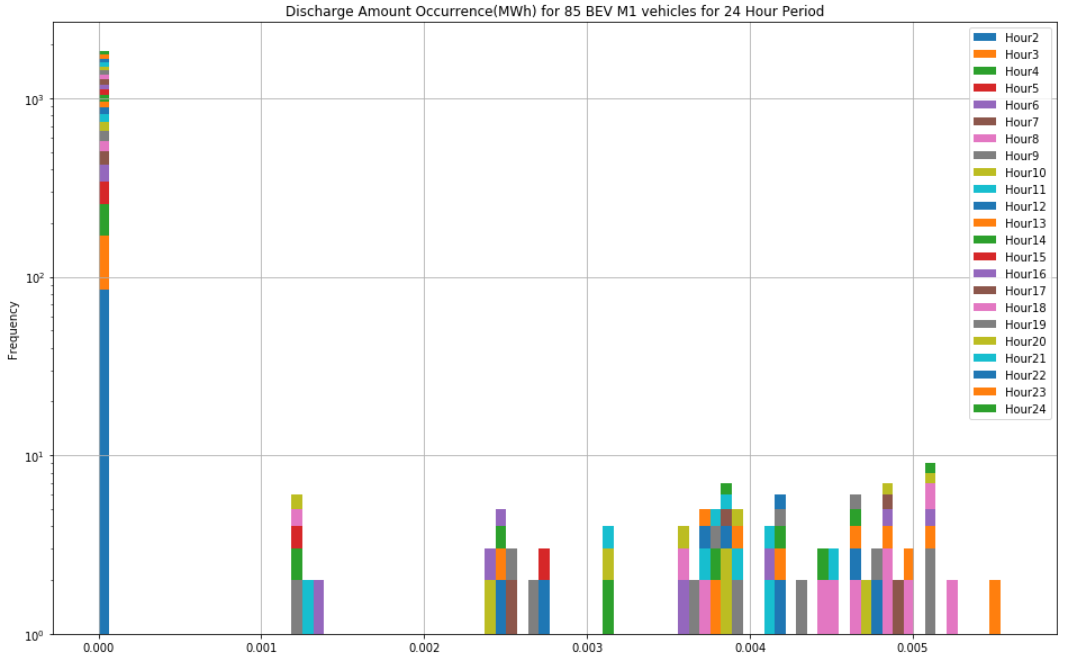
*Figure 2. Current State of Battery (MWh) for 9 BEV L7e*

**

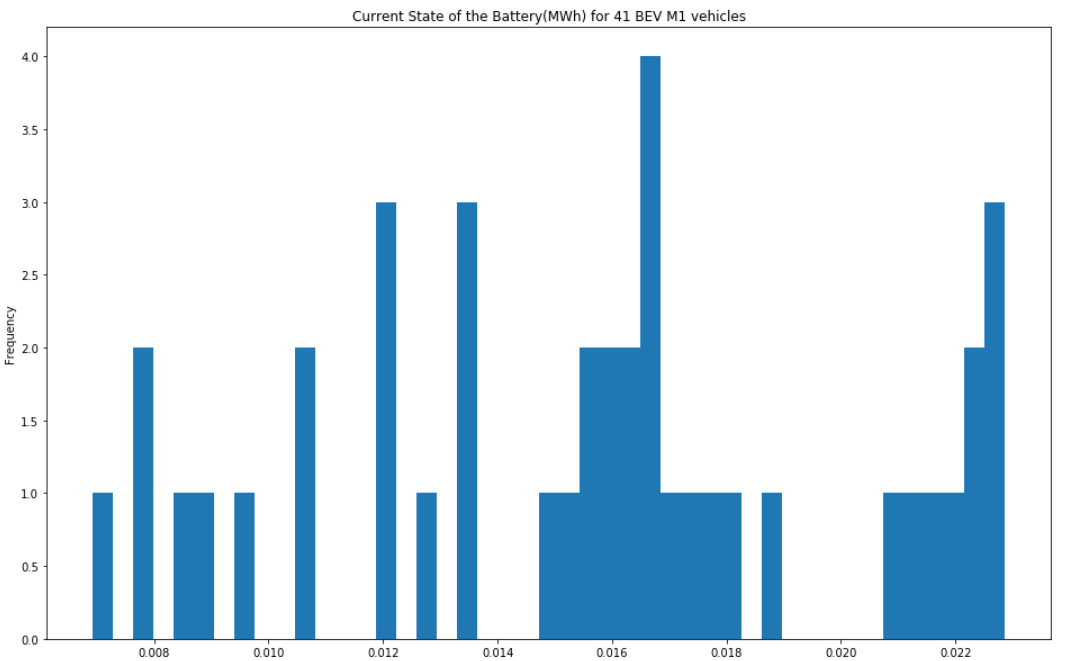
*Figure 3. Battery (MWh) Discharge Amount Frequency for 9 BEV L7e*

**

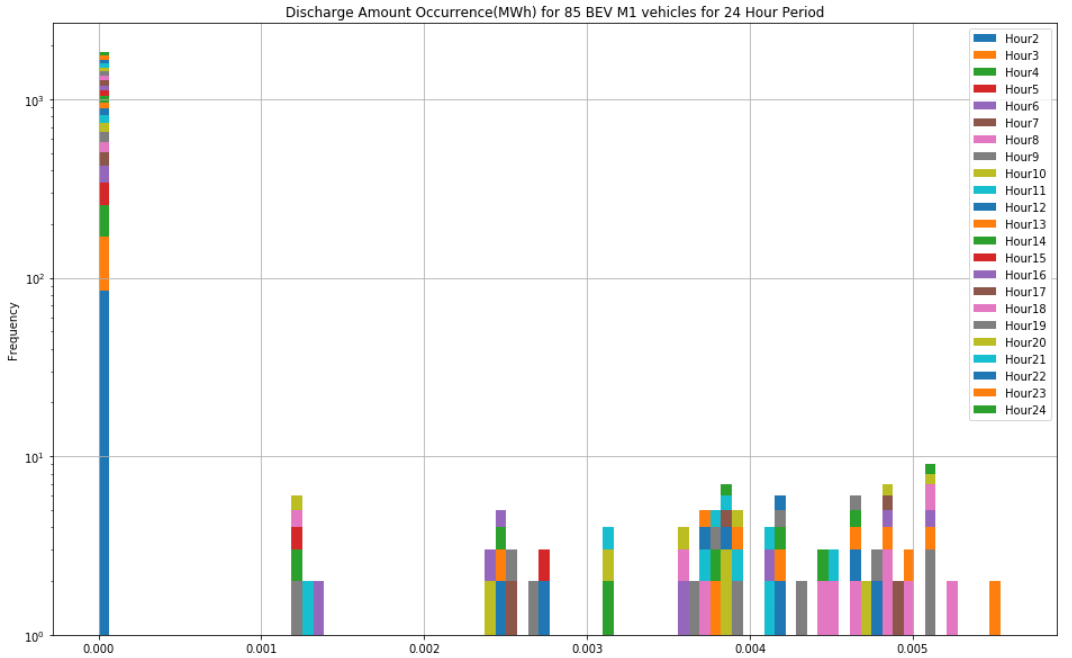
*Figure 4. Current State of Battery (MWh) for 85 BEV M1*

**

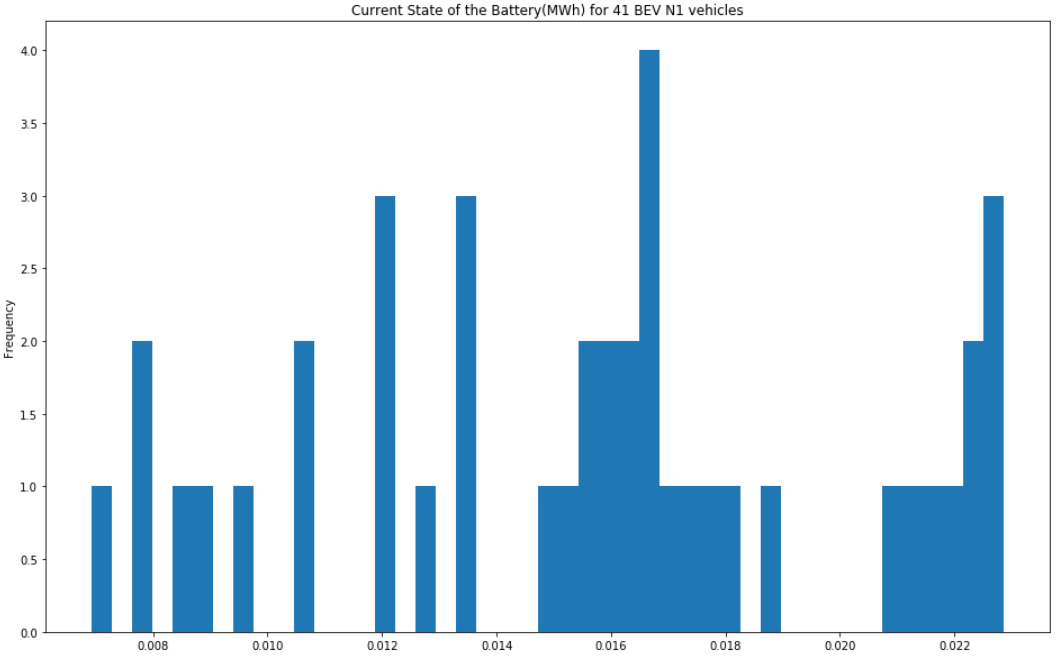
*Figure 5. Battery (MWh) Discharge Amount Frequency for 85 BEV M1*

**

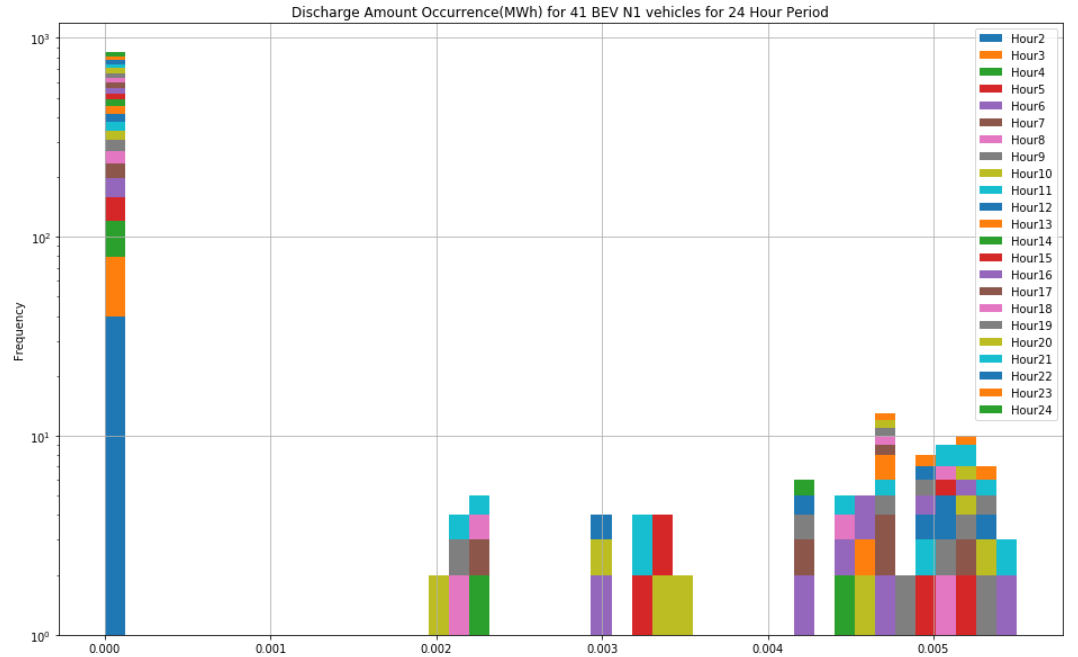
*Figure 6. Current State of Battery (MWh) for 41 BEV M1*

**

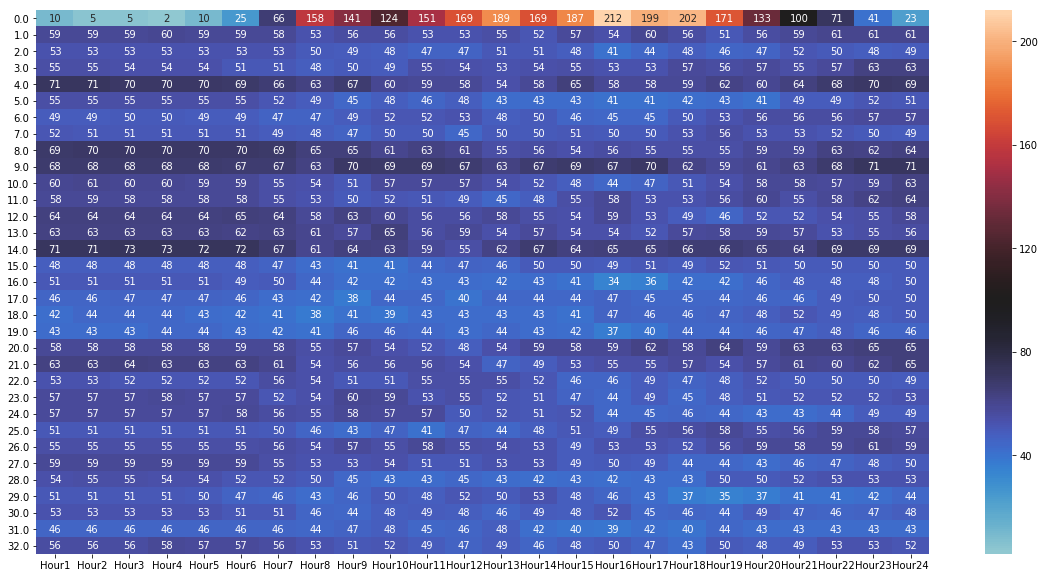
*Figure 7. Battery (MWh) Discharge Amount Frequency for 85 BEV M1*

**

*Figure 8. Current State of Battery (MWh) for 41 BEV N1*

**

*Figure 9. Battery (MWh) Discharge Amount Frequency for 41 BEV N1*

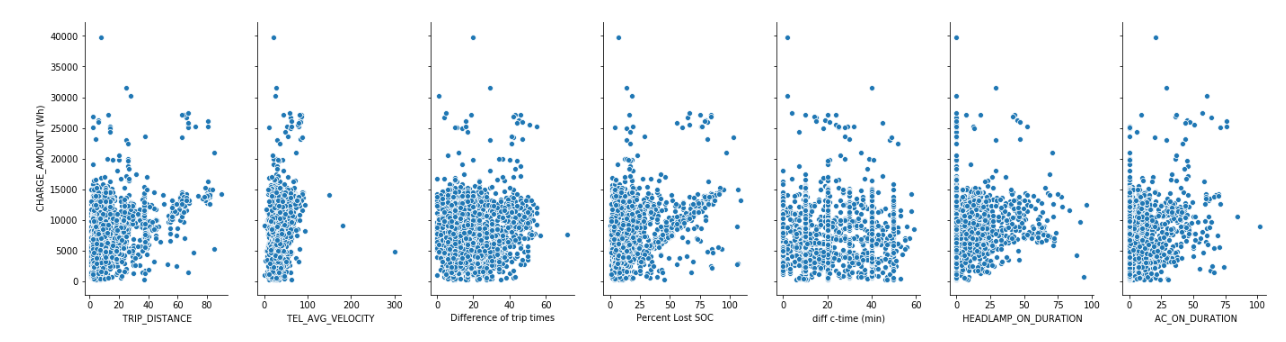
**

*Figure 10. The frequency of vehicle discharges at a time period and point on the Bus.*

For example, as seen in *Figure 6.* The most common battery state for the BEV N1 vehicles was ~0.017MWh. Also, the most common discharge amount for the same vehicle was ~0.0045 MWh *Figure 7.* The heat map (*Figure 10.*) is seen as the most useful chart because it depicts where the most and least vehicles are drawing power from. For example, Bus 14 from Hour 1-6 shows high vehicle traffic and is thus an area to concentrate on for future algorithm development.

# Smart-Grid Smart-City Electric Vehicle Trial Data

A new data set will be introduced for the goal of the project, to predict pricing based on Charge amount (Wh) needed. The data comes from the Department of the Environment and Energy with the Australian Government. It was part of the Smart Grid Smart City (SGSC) project. In the train a fleet of 20 2010 Mitsubishi iMiEV cars were used by businesses and households. The data encompassed individual trips including appliances within the vehicle such as air-conditioning, lights, etc. The data was collected for the period August 2011 to May 2013. A description of this data can be seen in (*Figure 11.*)



*Figure 11.* Features vs Charge Amount (Wh)

# Regression

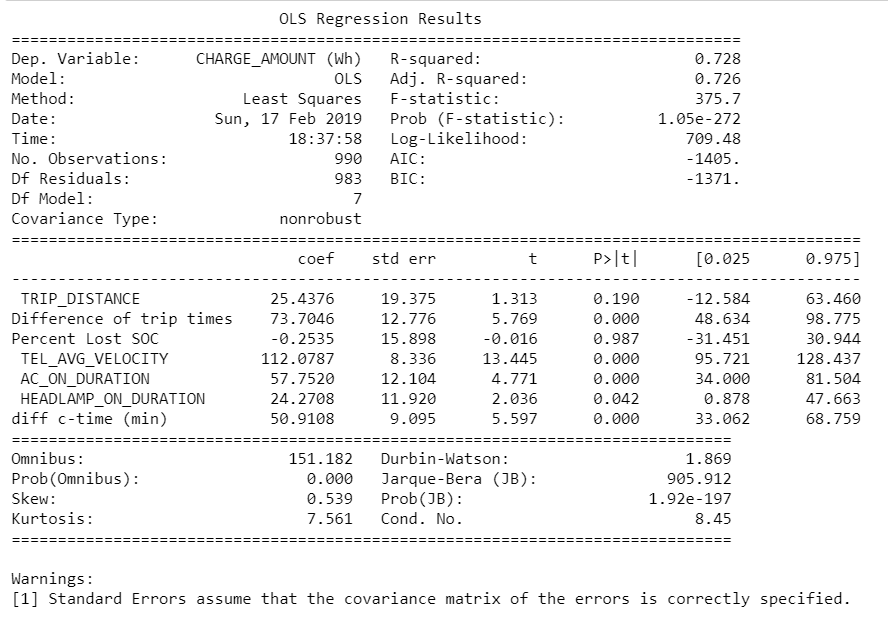
In order to accurately estimate Charge Amount (Wh) after an individual trip, the original data must be account for. The first step is to locate the correct parameters from the original dataset. The features that will be considered are Trip Distance, Percentage of SOC lost, Average Velocity, air-conditioning on-time, headlamp on-time, trip time and charging time. Next, will be to normalize the data. For this project Linear Regression was compared with Ridge and Lasso Regression. The finding was compared by using the following Performance Metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). (*Table 2.*)

|  |  |
| --- | --- |
|  | Original |
|  | LinearRegression |
| mean absolute error (MAE) | 0.082486224 |
| mean squared error (MSE) | 0.011469584 |
| Root mean squared error (RMSE) | 0.107096145 |
|  | RidgeRegression |
| mean absolute error (MAE) | 0.089336625 |
| mean squared error (MSE) | 0.012822929 |
| Root mean squared error (RMSE) | 0.113238374 |
|  | LassoRegression |
| mean absolute error (MAE) | 0.089341991 |
| mean squared error (MSE) | 0.01282442 |
| Root mean squared error (RMSE) | 0.113244957 |

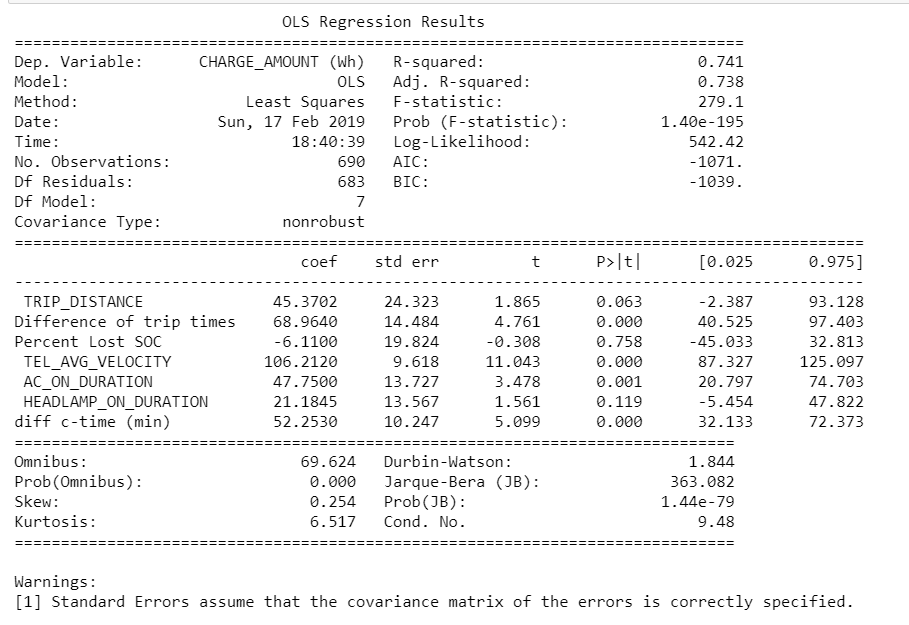
*Table 2. Results of the Original DataSet*

This result was then compared with dropping velocity and charge time. Then again with Dropping velocity and Trip-

time (*Table 5.*) Then the original (*Table 3.*) was taken back into consideration but training the data for 75% of the data set lead to a more accurate result (*Table 4.*)



*Table. 3 Original*

**

*Table 4. 75% trained data*

##### 

*Table 5.*

##### Feature Extraction/Selection

Feature Selection is used to filter irrelevant or repetitive features from the dataset. The difference between extraction and selection is that feature selection keeps a subset of the original features while extraction created brand new ones.

In the following figure we see the correlation matrix between the independent variables or features. In Table 6. We can see that a correlation matrix is used. This matrix determines if certain features pairs are highly correlated with one another. If two features are highly correlated it is possible that this will cause the model to over-fit the data, thus produce a worse result. From the correlation matrix below it makes sense to try the model without lost state of charge, trip distance, average velocity and charge amount.

We see the mean ranking in *Figure 12.* Each feature is ranked according to their meaning to the data. Stability selection via randomized lasso, recursive feature elimination (RFE), regression model feature ranking and random forest feature ranking were chosen to do the mean ranking. The top two most important features are the trip distance and the headlamp on time. The least are the odometer and the average velocity.

For purposes of the model, the correlation matrix and the mean ranking results will be tested to see what will give the best outcome. We start with testing all nine features in *Figure 13.* In this model the variance score was 92% for linear regression. In the following trial models *(Figures 14-16 and table 7.)* we can see the variance score was less than optimal. The model to be used going forward will use all 9 features.



*Table 6.*

##### 

*Figure 12.*

##### 

*Table 7.*

##### 

Figure 13.

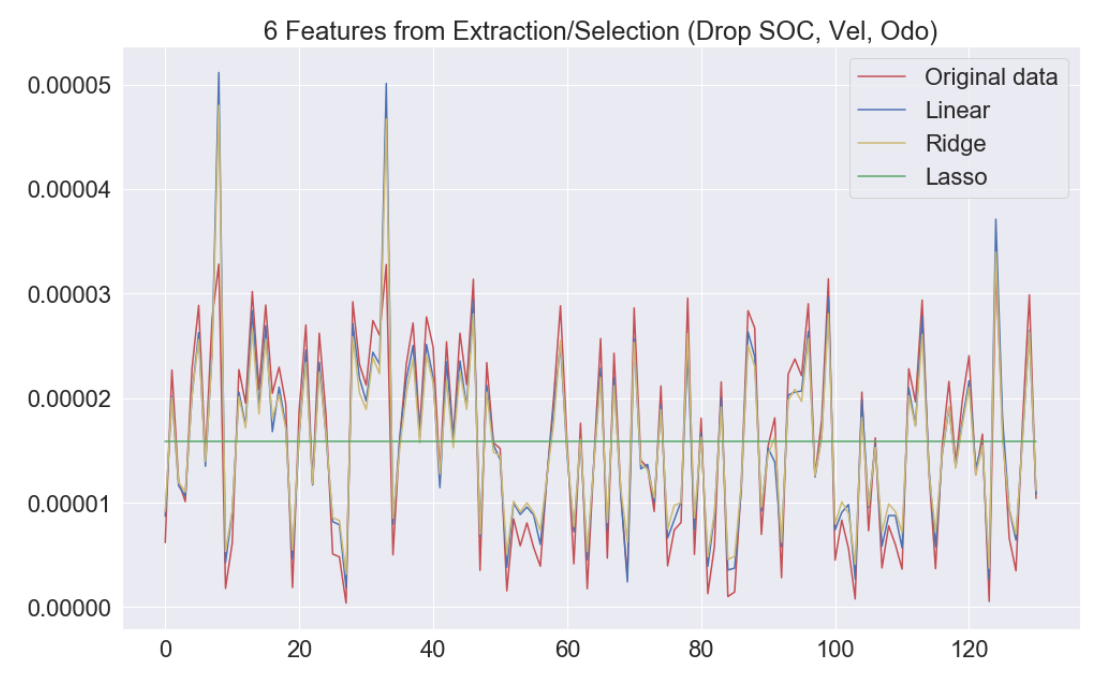


Figure 14.

##### 

Figure 15.

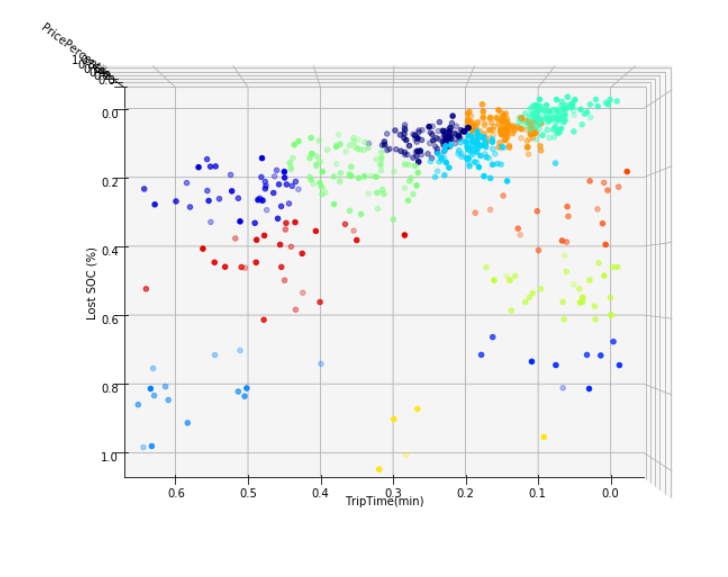
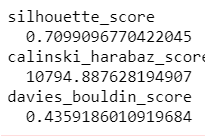
##### 

Figure 16.

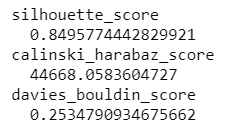
##### Clustering

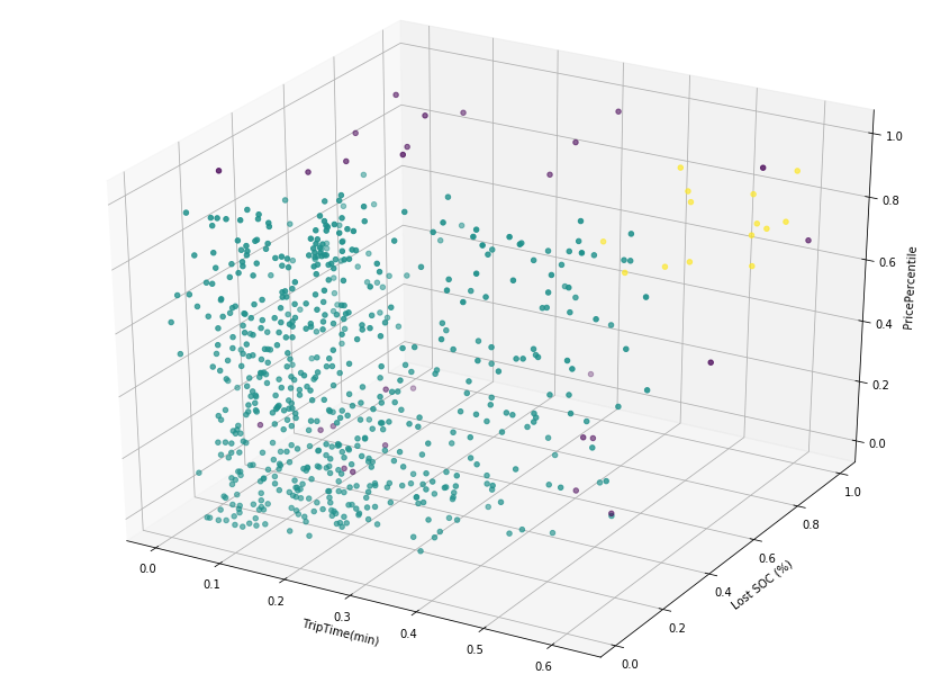
This is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, bioinformatics, data compression, and computer graphics\*. Kmeans clustering is unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. Hierarchical Clustering, specifically Agglomerative, is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy. DBSCAN (Density-based spatial clustering of applications with noise)

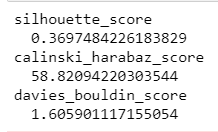
 It groups together points that are closely packed together (points with many nearby neighbors), marking as outliers’ points that lie alone in low-density regions (whose nearest neighbors are too far away).

##### 



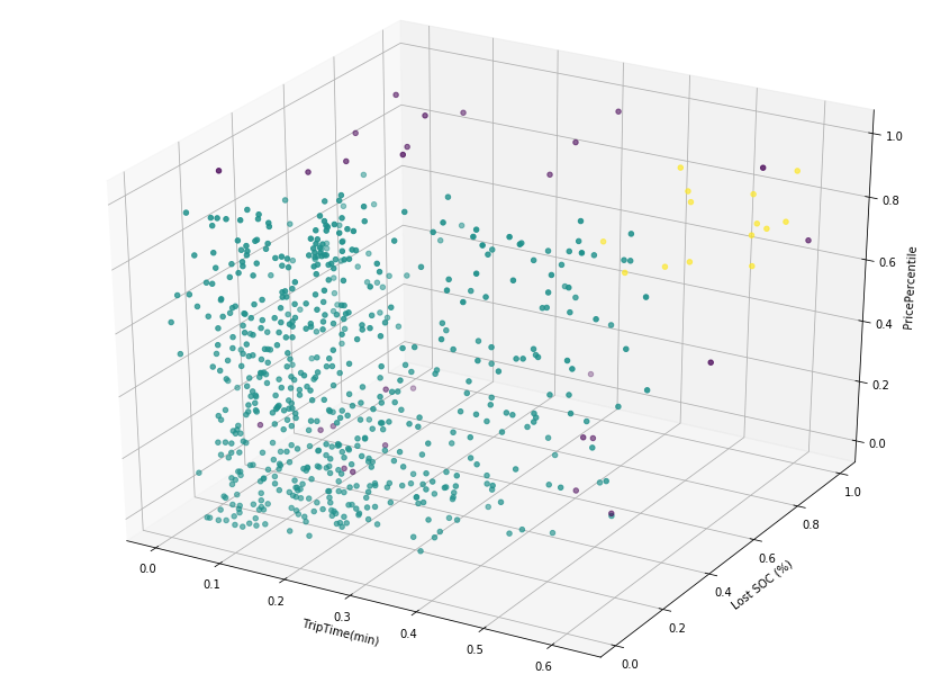


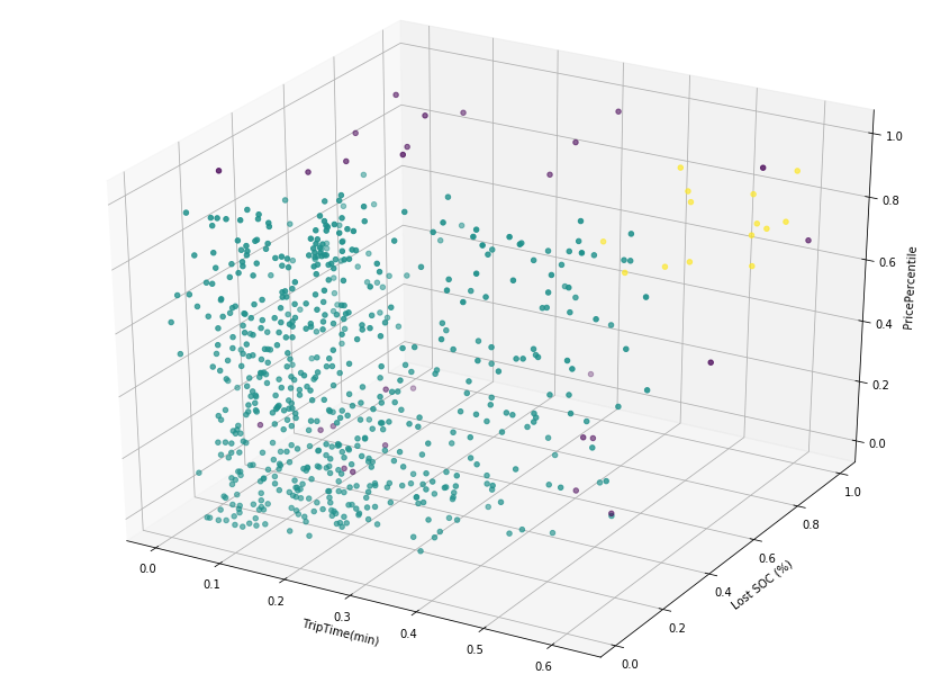


##### References

1. Soares, J., Sousa, T., Morais, H., Vale, Z., & Faria, P. (2011). An optimal scheduling problem in distribution networks considering V2G. *2011 IEEE Symposium on Computational Intelligence Applications In Smart Grid (CIASG)*. doi:10.1109/ciasg.2011.5953342
2. Faria, P., Vale, Z. and Baptista, J. (2015). Demand Response Programs Design and Use Considering Intensive Penetration of Distributed Generation. *Energies*, 8(6), pp.6230-6246.Soares, J., Canizes, B., Lobo, C., Vale, Z., & Morais, H. (2012). Electric Vehicle Scenario Simulator Tool for Smart Grid Operators. *Energies,5*(6), 1881-1899. doi:10.3390/en5061881
3. Data.gov.au. (2019). *Search*. [online] Available at: https://data.gov.au/dataset/ds-dga-87f276c3-5fba-4f31-9032-199793d6f4a7/details [Accessed 18 Feb. 2019].

\*Definition from Wikipedia





##### Alternate Methods for Improvement – assignment5

It is said that data analytics is an iterative process. In this section, we will go through different methods for improving upon the already exisiting model.

##### preprocessing

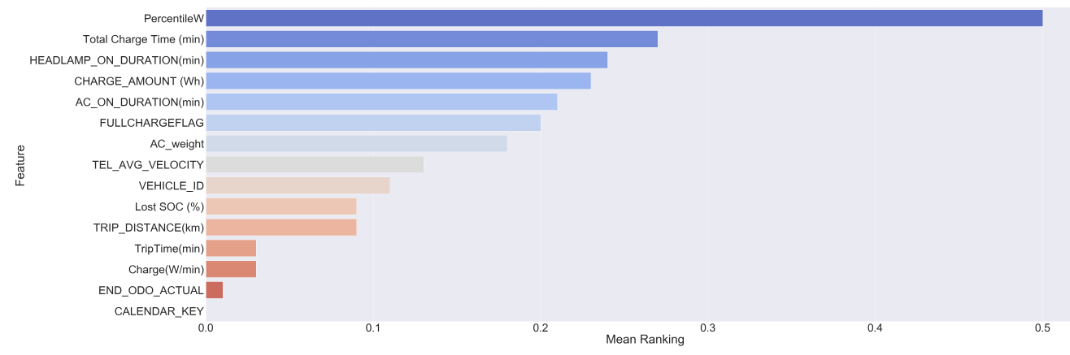
For preprocessing, we have a couple of differences. In this iteration we look to put more features into the model from the original data. For example, this time we will have FULLCHARGEFLAG, which is an indication if the vehicle was fully charged. Also, added was the vehicle identification and the calendar key. The calendar key is a hard coded numerical value which indicates the specific time and date of a particular trip.

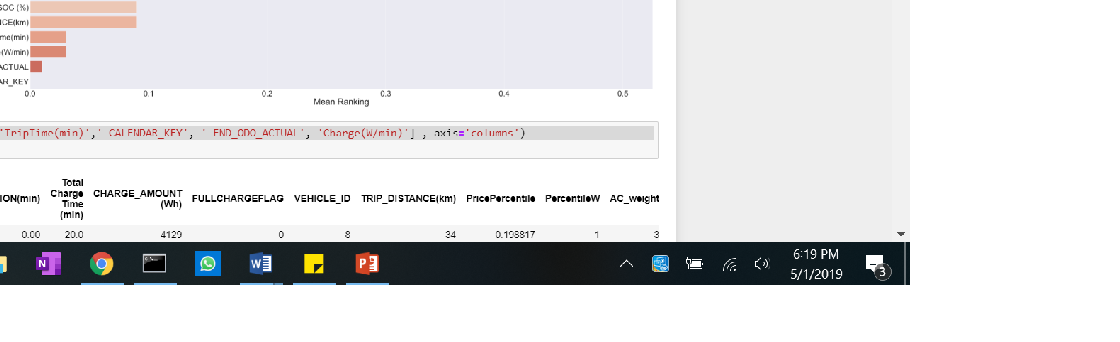
##### Feature Extraction

For the first iteration of the model, little was done in terms of feature extraction. For this iteration a number of features were created from the raw data. One example of a feature that was created is the charge speed. The charge speed is just the watts per minute that a particular vehicle took to charge their vehicle in the specific instance. Another feature that was extracted was the air-conditioning weight. The AC weight can be noted as a hard-coded value for when the AC on duration meets certain thresholds. For example if the AC on duration is above 21.6 minutes the hard coded value assigned will be 3. PercentileW also works similarly to the AC on time giving a higher weight for higher percentiles.

##### Feature Selection

For feature selection, mean ranking was chosen to decipher the importance of features. Randomized lasso and random forest regressor as well as linear, ridge and lasso regression coefficients were used to rank the 15 features. The mean ranking for all the features can be seen in Figure 17. The mean is essentially the average coefficient of each feature. It can be seen some features hold much more importance than other features. In this case the bottom four features will be dropped from the model.





*Figure 17.*

##### Clustering

For modeling purposes Kmeans clustering and Hierarchical clustering were chosen for the model. Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). It is a main task of exploratory data mining, and a common technique for statistical data analysis, used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, bioinformatics, data compression, and computer graphics\*. Kmeans clustering is unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. Hierarchical Clustering, specifically Agglomerative, is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy. For kmeans clustering 3 clusters shows the best results by trail and error. For hierarchical clustering 4 clusters shows the best results.

##### Regression/model prediction

For the portion of the model linear, ridge and lasso regression were chosen to predict the values with the model. The data was normalized before being processed. The error was calculated using mean absolute error, mean squared error and root mean squared error. The original data was then plotted against the three regression for a visual interpretation of the data.

##### Results

As seen from the chart below the performance has greatly increased by looking at individual clusters. The best clustering from the results data shows that using the Hierarchical clustering #0 will derive the best results for the model. The variance score is a near perfect. The error for this model is the smallest at a mean absolute error of 0.00614 for linear regression, along with the other associated errors. Looking at which specific regression creates the least error, lasso regression for hierarchical cluster #0 comes out as the best fit for the model. This is a vast improvement from the previous assignment results.



##### Appendix

END 0D0 ACTUAL 
HEADLAMP ON DURATION(min) 
TripTime(min) 
TRIP DISTANCE(km) 
0 
0 
0 
0 
O 
O 
O 
O 
0 
0 
0 
0 
0 
0 
0 
0 
0 
0 
0 
Total Charge Time (min) 
0 
아류k. 
•&녑투 
•승|류1` 

|  |
| --- |
| {'rlasso/Stability': {'TripTime(min)': 0.0, |
| 'Lost SOC (%)': 0.0, |
| ' TEL\_AVG\_VELOCITY': 0.0, |
| ' AC\_ON\_DURATION(min)': 0.0, |
| ' HEADLAMP\_ON\_DURATION(min)': 0.0, |
| 'Total Charge Time (min)': 0.0, |
| ' CHARGE\_AMOUNT (Wh)': 0.0, |
| ' END\_ODO\_ACTUAL': 0.0, |
| ' TRIP\_DISTANCE(km)': 0.0}, |
| 'RFE': {'TripTime(min)': 0.62, |
| 'Lost SOC (%)': 0.5, |
| ' TEL\_AVG\_VELOCITY': 0.12, |
| ' AC\_ON\_DURATION(min)': 0.38, |
| ' HEADLAMP\_ON\_DURATION(min)': 0.88, |
| 'Total Charge Time (min)': 0.75, |
| ' CHARGE\_AMOUNT (Wh)': 0.25, |
| ' END\_ODO\_ACTUAL': 0.0, |
| ' TRIP\_DISTANCE(km)': 1.0}, |
| 'LinReg': {'TripTime(min)': 0.08, |
| 'Lost SOC (%)': 0.83, |
| ' TEL\_AVG\_VELOCITY': 0.03, |
| ' AC\_ON\_DURATION(min)': 0.81, |
| ' HEADLAMP\_ON\_DURATION(min)': 0.49, |
| 'Total Charge Time (min)': 0.09, |
| ' CHARGE\_AMOUNT (Wh)': 0.04, |
| ' END\_ODO\_ACTUAL': 0.0, |
| ' TRIP\_DISTANCE(km)': 1.0}, |
| 'Ridge': {'TripTime(min)': 0.0, |
| 'Lost SOC (%)': 0.0, |
| ' TEL\_AVG\_VELOCITY': 0.0, |
| ' AC\_ON\_DURATION(min)': 0.0, |
| ' HEADLAMP\_ON\_DURATION(min)': 0.0, |
| 'Total Charge Time (min)': 0.0, |
| ' CHARGE\_AMOUNT (Wh)': 1.0, |
| ' END\_ODO\_ACTUAL': 0.04, |
| ' TRIP\_DISTANCE(km)': 0.0}, |
| 'Lasso': {'TripTime(min)': 0.0, |
| 'Lost SOC (%)': 0.0, |
| ' TEL\_AVG\_VELOCITY': 0.0, |
| ' AC\_ON\_DURATION(min)': 0.0, |
| ' HEADLAMP\_ON\_DURATION(min)': 0.0, |
| 'Total Charge Time (min)': 0.0, |
| ' CHARGE\_AMOUNT (Wh)': 0.0, |
| ' END\_ODO\_ACTUAL': 0.0, |
| ' TRIP\_DISTANCE(km)': 0.0}, |
| 'RF': {'TripTime(min)': 0.0, |
| 'Lost SOC (%)': 0.0, |
| ' TEL\_AVG\_VELOCITY': 0.0, |
| ' AC\_ON\_DURATION(min)': 0.0, |
| ' HEADLAMP\_ON\_DURATION(min)': 0.0, |
| 'Total Charge Time (min)': 0.0, |
| ' CHARGE\_AMOUNT (Wh)': 0.0, |
| ' END\_ODO\_ACTUAL': 0.0, |
| ' TRIP\_DISTANCE(km)': 0.0}, |
| 'Mean': {'TripTime(min)': 0.12, |
| 'Lost SOC (%)': 0.22, |
| ' TEL\_AVG\_VELOCITY': 0.02, |
| ' AC\_ON\_DURATION(min)': 0.2, |
| ' HEADLAMP\_ON\_DURATION(min)': 0.23, |
| 'Total Charge Time (min)': 0.14, |
| ' CHARGE\_AMOUNT (Wh)': 0.22, |
| ' END\_ODO\_ACTUAL': 0.01, |
| ' TRIP\_DISTANCE(km)': 0.33}} |