Course : CMPT 318 (Cybersecurity)
Topic : Anomaly Detection on the Power grid Data Set
Semester : Spring 2018
Group # : 4
Simon Fraser University

Authors: Zeeshaan Manji 301228629, Gurtej Singh Rai 301243505,
Carlson 301193089, Victor 301267266

Abstract:

This paper demonstrates and describes some methods related to concepts and techniques of cybersecurity to explore and analyze anomalies in a real data set. The data set we used to detect the anomalies is from some parts of the U.S electrical power grid. It monitors household power consumption with few features and collected time. We advocate solutions based on constructing methods and models to distinguish normal data and anomalous data. We also make plots to indicate the result precisely of each method or model and ferret the correlation among features and infer reasons causing these anomalies.

Table of Contents

Introduction

According to the increasing number of cyberattacks, it causes considerable damage not only for finance but also physics that influences our lives and business badly. It is getting a lot worse when it cause physical harm such as the critical infrastructures, which refers to water management systems, electric power grids, etc. The electrical power grid includes variety of power consumption and the household power consumption is a part of it.

According to Nguyen and Nahrstedt, because there have been accidental damages on the electric power grid and took a lot of time to be repaired. It shows that it is a very serious problem if those were attacks instead of the accidents. It would be hard to distinguish if the system is malfunctioning or there has been a virus [7]. Once they are being target, it even threaten human safety ultimately.

Generally in cybersecurity, analysis is based on finding abnormal data called anomaly. By given the household power consumption data file from the electrical power grid, it is crucial that to seek the abnormal power consumption at a specific time or in a duration defined as anomaly and analyze reasons causing these anomalies over different contexts such as peak period and accident and also spread warnings about until being fixed.

Background

Anomaly detection is a technique used for finding patterns in data that deviates from normal behavior. The most common term used for these data points that do not conform to these normal behavior is called anomalies or outliers. Different anomaly detection techniques have been developed based on specific application domains. Anomalies in data usually translate to significant and critical actionable information in different domains. Anomaly detection is used in a variety of domains such as finance or insurance where it is used for fraud detection, health where it is used for detecting health problem, intrusion detection for cybersecurity in companies and many other domains to detect anomalous behavior. Each domain has different data values and applying the wrong technique will lead to incorrect findings since there are contextual anomaly. Contextual anomalies are data instance that are anomalous in the specific context but not otherwise.

There are three categories of anomaly detection technique: Supervised anomaly detection, Semi-supervised anomaly detection and Unsupervised anomaly detection. Supervised anomaly detection requires a labeled data set that labels normal and abnormal data.. The labeled data set will train a classifier to detect anomalies in a different data set. It is difficult to get a label data set since it is costly and there is a much lower chance of anomalies in a data set. In semi-supervised anomaly detection, we build a model based on the normal behavior from the training data set that has been labeled. Once the model is built, test data is fed into the model and the likelihood will determine of the data is normal or anomalous. Unsupervised anomaly detection is similar to semi-supervised version but the model if built based on assuming that majority of the train data will be normal.

There are a couple of things such as Noise, Novelties and Outliers that has to be looked out for in a large data set since value may cause hindrance while analyzing the data. Noise is data that adds no value to data analysis such as random data that do not conform. Novelty are patterns in data that may look anomalous at first but after further analysis they are incorporated into the normal model. Outliers is a point in the data set that is distant from the rest of the data and these are commonly referred to as anomalies.

The Problem Description

The problem we will be focusing on for this project is detecting unknown attacks using Anomaly-based IDS such as point anomalies, and contextual anomalies in individual household electric power consumption based on the data set being provided, and being able to find patterns on data instances which don't conform with the normal behavior. We will also be looking for early detection and warning about suspicious and potentially harmful anomalies such as the anomalies which mitigate the impact of attacks. Other problems we will be looking at when we are analyzing our data set include missing data, labels to indicate if an specific instance is an "anomaly" or "normal".

Electric power grid is very vulnerable to the cyber-attacks, by working on the problems described we can reduce and produce warning about suspicious attacks to avoid physical damages which can affect human resources.

The Methodology used for solving the problem

1. Understanding the Problem Scope
2. Understanding the features in the data set.
3. Splitting the Data set(Into Windows)
4. Exploring the Data set (Min, Max, Standard Deviation, Mean, Median, Correlation)
5. Anomaly Detection :
   - 5.1 Moving Average Function to detect Point Anomalies.
   - 5.2 Finding outliers in the Data set to detect Point Anomaly Detection)
   - 5.3 Hidden Markov Model to detect Contextual Anomalies.

1.Understanding the Problem Scope

Before delving into the large data set, we needed to understand the problem at hand and seeing how our solution will help the problem. The electric power grid throughout North America is connected by regions as seen in **Figure 1**. It shows that the electric power grid is vulnerable because the communication between the industries are standardized across different countries which limits security [4]. Electric power grid is a critical infrastructure since power is delivered to a large region. If a perpetrator can cut off any connections, this can lead to blackouts across regions. The infrastructure in the power grid is using Supervisory Control and Data Acquisition (SCADA) systems which is the underlying architecture that is powering everything. If anything happens to the SCADA system, it can cause catastrophic issues affecting the power grid. It is not only the software which leads to these attacks, it can also be because of old systems and technologies which needs repairing to protect from malware. For this project we are focusing on finding a solution to detect and warn us about suspicious activity and anomalies in order to have the upper hand during a cyber-attack by coming up with countermeasure. By creating a model of existing data we can use this to detect potential anomalous behavior. We can also feed new data into our model to determine if there are suspicious and anomalous activities.

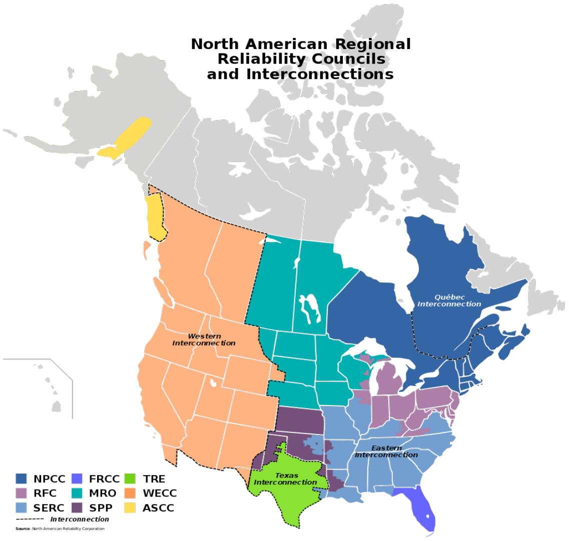


Figure 1 Shows how the electric power grid is connected by regions [3]

2. Understanding the Features

We began the project by first researching and understanding the features in the Data set that we were not familiar with. After conducting a few days of research we defined the following features:

Global active power: is the real power consumption by appliances not included in the sub meters. It is household global minute-averaged active power and its unit is kilowatt.[8]

Global reactive power :is the household global minute-averaged reactive power with the unit of kilowatt. It refers that the power moves forward or backward that is not used or leaked. Compared to active power, it is the imaginary power consumption and the active power is the real power consumption. [8]

Global intensity: is the power consumption in a amount level or strength. It is the household global minute-averaged current intensity and its unit is ampere. Sub metering also represents the power consumption and each sub metering defines the power consumption by a specific type of appliance with unit watt-hour of active energy. [8]

Voltage: is the minute average voltage that is used in the household. The higher the voltage means there is a high flow of electrical current.[8]

Sub Metering 1: refers that the power are consumed by appliances in the kitchen, such as microwave, oven dishwasher. [8]

Sub Metering 2: refers the power consumption of the type of appliances related to the laundry room, and for instance the power is consumed by washing-machine, tumble-drier. [8]

Sub metering 3: are the electric heating system as air-conditioner and water heater.[8]

3. Splitting the Data

We created functions to extract (Months, Days, Years, Minute and Hours) from the Dataset which helped us to split our date and time into new separate columns .This was the most challenging part, as was we needed to understand the different data types (Classes of the fields in R) ,and how to convert that to a datatype which can be used for extracting records from the data set. After adding extra columns which were defined by the new functions that we made, it made it easy to split the data. As a group we decided to split the data into seasons ,and two different time windows (Summer, Winter and Spring) which we defined as;

- Summer (May-August)

  Summer Evening (6:00PM- 11:59PM)

  Summer Day (6:00AM- 5:59PM)

- Winter (September-December)

  Winter Day (8:00AM- 5:59PM)

  Winter Evening (6:00PM-11:59PM)

- Spring(January-April)

  Spring Day (8:00 AM - 5:59PM)

  Spring Evening (6:00PM-11:59PM)

The authors of this project also explored different windows in the season such as Monday-Friday and Saturday-Sunday in the Summer, as the patterns of Monday- Friday were similar, and the pattern for Saturday and Sunday were slightly different.

4. Exploring the Data set

We read an article about power demand throughout a day in the New England Region on U.S. Energy Information Administration website [1], we determined to split the data based on the chart seen in **Figure 2**. From **Figure 2**, we can see that the peak demand was between 6:30 PM-7:00 PM so we made sure to include those data point when exploring the data. Since the peak demand was during the 6:30 PM time period, we decided to extract the after work hours to do further analysis. We choose after work hour as 6:00PM-11:59PM assuming that this period would lead to the highest power consumption with everyone being home after work.

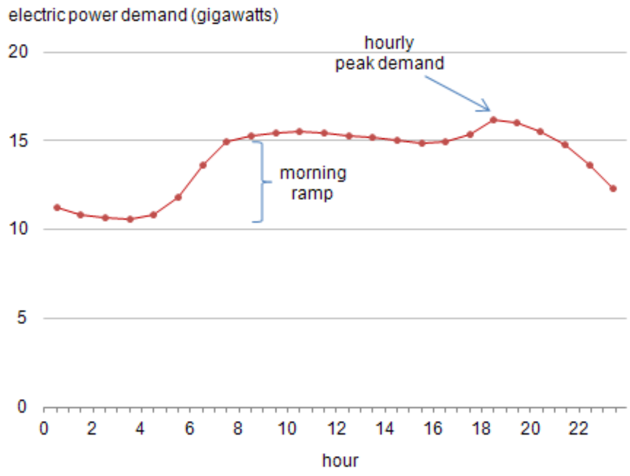


Figure 2 Power demand throughout the day in a U.S. region [1]

After determining the time frame we would use, we dug deeper by analyzing all the data between 6:00PM-11:59PM. We plotted the charts of Global Active Power (**Figure 3**), Global Reactive Power (**Figure 4**), Global Intensity (**Figure 5**),and Global Voltage (**Figure 6**) and split it into months and years to look for patterns. Each boxplot in the month consist of ~30 data points (# of days in the month), each data point is the average consumption between 6:00PM-11:59PM. We extracted Mean, Median, Standard Deviation, Min ,and Max from each featured to see if there are any decisive manner to choose one feature for further examination (**Figure 7**). We also extracted the same information from the test data set to see if there are any major differences compared to the training data set (**Figure 8**).

From the information gathered from **Figure 7**, it was obvious that Voltage was not a good feature to choose since the range was much smaller than the other 3 features. It was difficult to determine which of the remaining 3 features would be the best to use for detecting anomaly. In the end we decided to use Global Active Power since it is the true power being consumed [2]. We also saw some increases in Mean, Median and Min and decreases in standard deviation and Max when comparing the training data set to the test data set (**Figure 8**). We did not choose Global Reactive Power since it is an imaginary power consumption which would not provide us with accurate results. We did not choose Global Intensity since the values are dependent on the sub meters which would require more variables involved in data analysis. We also compared the pairs of features to look for correlations that could be used for multivariate HMM (**Figure 9**). We noticed that Global Active power and Global Intensity had the highest correlation of 0.7059081.

After narrowing down the features to Global Active Power and 6:00PM-11:59PM range, we still had a large amount of data to deal with. By looking at **Figure 3**, we see that there is a dip in Global Active Power during the summer months of May to August. We decided that we should do some further analysis in the summer months since the values are much lower during this time frame. We thought that it would be easier to detect anomalies if the values are lower than other seasons and months. We extracted the summer months and got the data from 2007-2009 as seen in **Figure 10**. We noticed some odd behaviors in some of the months so we used this subsetted data for our anomaly detection methods.

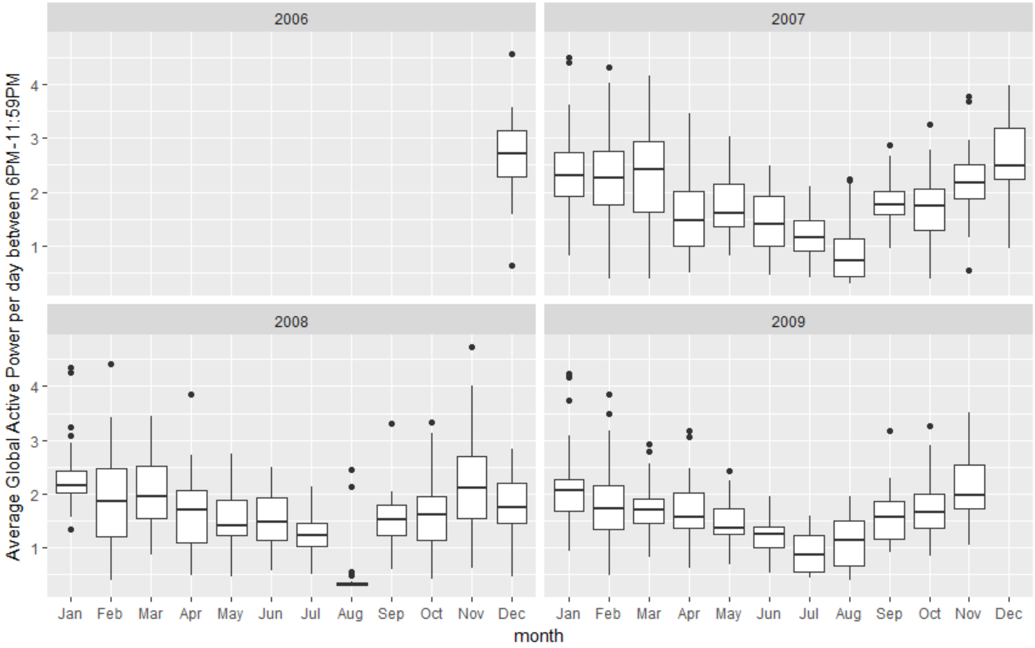


Figure 3 Average Global Active power between 6:00PM-11:59PM split into months and from Dec 2006 - Nov 2009

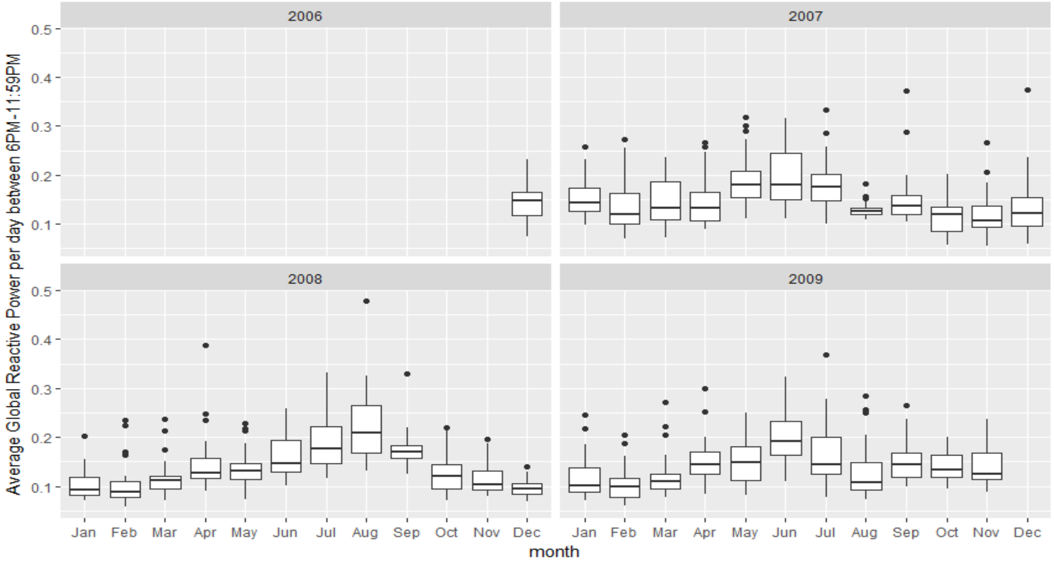


Figure 4 Average Global Reactive power between 6:00PM - 11:59PM split into months and from Dec 2006 - Nov 2009

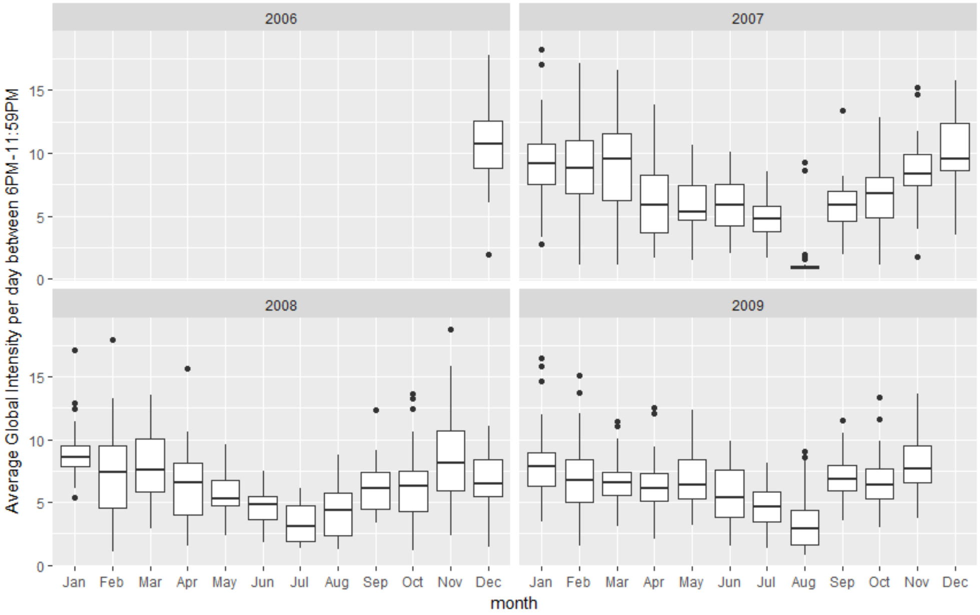


Figure 5 Average Global Intensity between 6:00PM - 11:59PM split into months and from Dec 2006 - Nov 2009

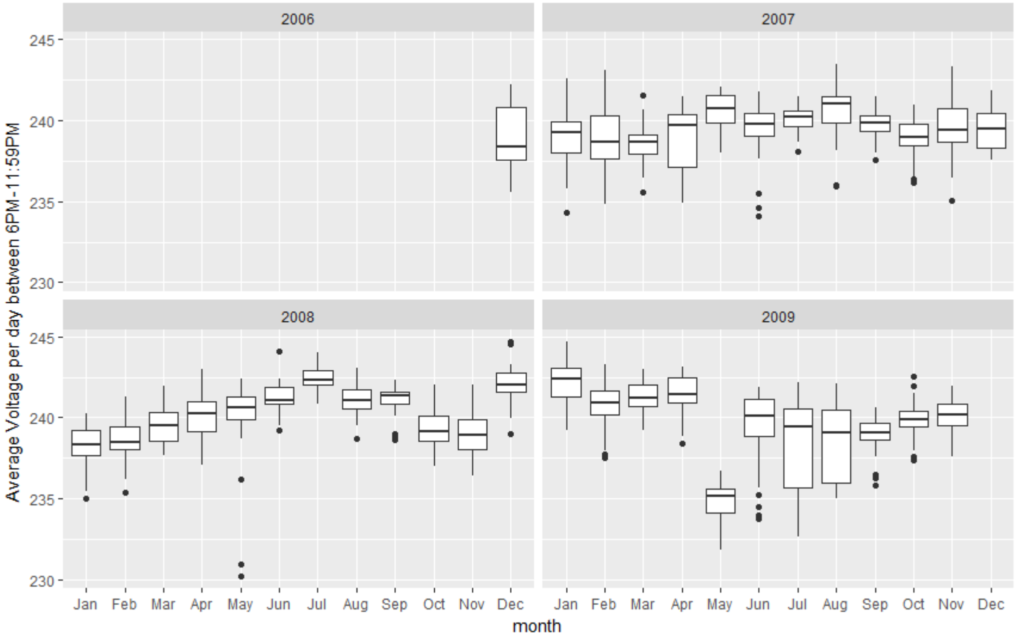


Figure 6 Average Voltage between 6:00PM-11:59PM split into months and from Dec 2006 - Nov
2009

	Global Active Power	Global Reactive Power	Global Intensity	Voltage
Mean	1.705203	0.1445405	6.590604	239.7323
Median	1.645	0.1331778	6.366667	239.9879
Standard Deviation	0.8080928	0.05457133	3.272637	2.082183
Max	4.738554	0.4791389	18.775	244.7102
Min	0.2937974	0.05486667	0.7588889	230.1702

Figure 7 General data from each features between 6:00PM -11:59PM and from Dec 2006 - Nov 2009

	Global Active Power	Global Reactive Power	Global Intensity	Voltage
Mean	1.972866 (+0.267663)	0.1434749 (-0.0010656)	6.043124 (-0.54748)	241.0027 (+1.2704)
Median	1.915633 (+0.270633)	0.1317 (-0.0014778)	5.862222 (-0.504445)	240.8458 (+0.8579)
Standard Deviation	0.6599563 (-0.1481365)	0.04766132 (-0.00691001)	2.730943 (-0.541694)	1.825263 (-0.25692)
Max	4.637836 (-0.100718)	0.3749 (-0.1042389)	16.12056 (-2.65444)	246.908 (+2.1978)
Min	0.7572279 (+0.4634305)	0.05734426 (+0.00247759)	1.273522 (+0.5146331)	235.584 (+5.4138)

Figure 8 General data from each features between 6:00PM -11:59PM and from Dec 2009 - Nov 2010 (From Test Data set). The numbers in bracket is the difference from Figure 7

Feature pairs	Correlation
Global Active Power and Global Reactive Power	0.1251776
Global Active Power and Global Intensity	0.7059081
Global Active Power and Voltage	-0.3705675
Global Reactive Power and Global Intensity	0.2668563
Global Reactive Power and Voltage	-0.1348928
Global Intensity and Voltage	-0.5360697

Figure 9 Correlation between 2 Features between 6:00PM-11:59PM and from Dec 2006 - Nov 2009 (From Training Data set)

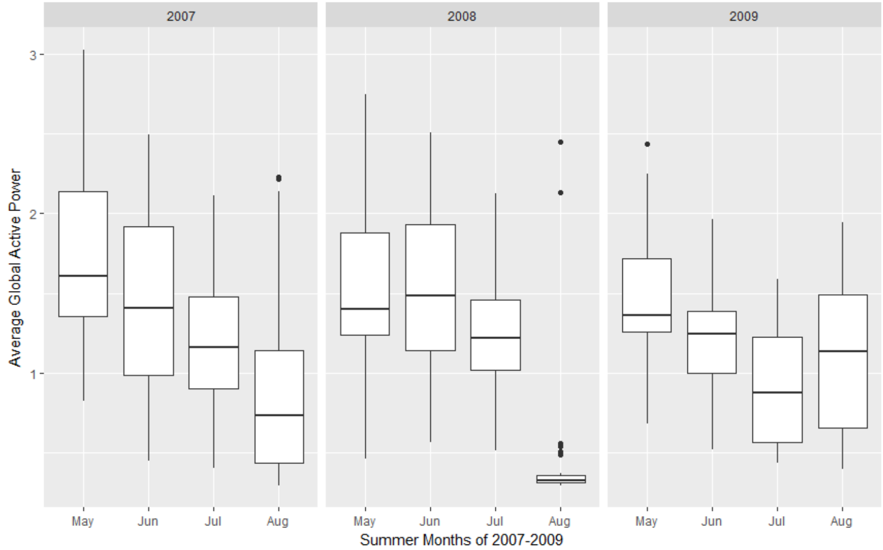


Figure 10 Average Global Active Power during the Summer Months of 2007-2009

5. Anomaly Detection

5.1 Moving Average:

The moving average method is that, it chooses a window with a duration as size and keep sliding the window by a specific step size until the window reach the last data point in the file. The method calculates average data in each window ,and use it to predict if each next data outside the window is normal or anomalous by comparison of the difference and a threshold.

The window size includes a number of data and it determines stability of each average in each window. We implement the moving average method firstly by determining a window size as 15 minutes and step size as 1 minute to travel the data. We also considered a circumstance that may cause error prediction, and that is the next data is the first data in the next day. It is not reasonable to use the last window in the former day to predict a data in a new day. Therefore we make each new start of prediction from the first 15 minutes of each day. The threshold value affects the quantity of normal data and anomalies significantly in this method. If the threshold is too small, the method will sensitively identify the normal or anomalous data. If the threshold is too large, the method is not able to rightly approach the correct detection, in other words, the method will determine a large number of normal data and small number of anomalies than exact. To decide the threshold value reasonably and convincingly, we use the statistical method to approach, which is “68-95-99.7” rule by setting the threshold dynamically based on each window. We set each threshold as three times of the standard deviation in the current window. “The "three-sigma rule of thumb" is related to a result also known as the **three-sigma rule,** which states that even for non-normally distributed variables, at least 88.8% of cases should fall within properly calculated three-sigma intervals”.[9]

5.2 Finding Outliers

To detect outlier in our data set, we determined the min and max value in our train set and applied that knowledge to our test set. If the values in the test data set are smaller than the min or bigger than the max then we classify these value as point anomalies. If we detect values that are not valid such as “NA” then we classify these as noise and the remaining data are considered normal. Since the detection based on corresponding time between training dataset and test dataset will be more accurately, we find the min and max of a specific time and detect the point of the corresponding time in the test dataset.

5.3 HMM Model

The implementation of the HMM model was quite challenging, and time consuming. We wanted to have about 5 different models, and we achieved this by selecting one weekend day in the winter and summer, and one weekday in the summer. We focused on three different time periods which were 6:00PM-11:59PM for summer and winter, and in the morning from 8:00AM-5:59PM for winter, and 6:00am-5:59pm in the Summer. The reasons on why we picked this time window was based on the pattern trends from the data exploration section of our study.

1. We first did a cross validation of our Train Data set and split it into two sets (Train and validation). We split the data with 70% Test data and 30% for Validation data. Furthermore, we took extra precaution when splitting this data to make sure we didn’t have one day of data being split between the two sets, as this is a time series data.

2. We created a Test Model Function which was left overnight to test ranging from 2 states to 17. It recorded the Log Likelihood, BIC of the Train set and Log Likelihood of the validation set. The summary gave us the BIC of each of the states for the Train Set.

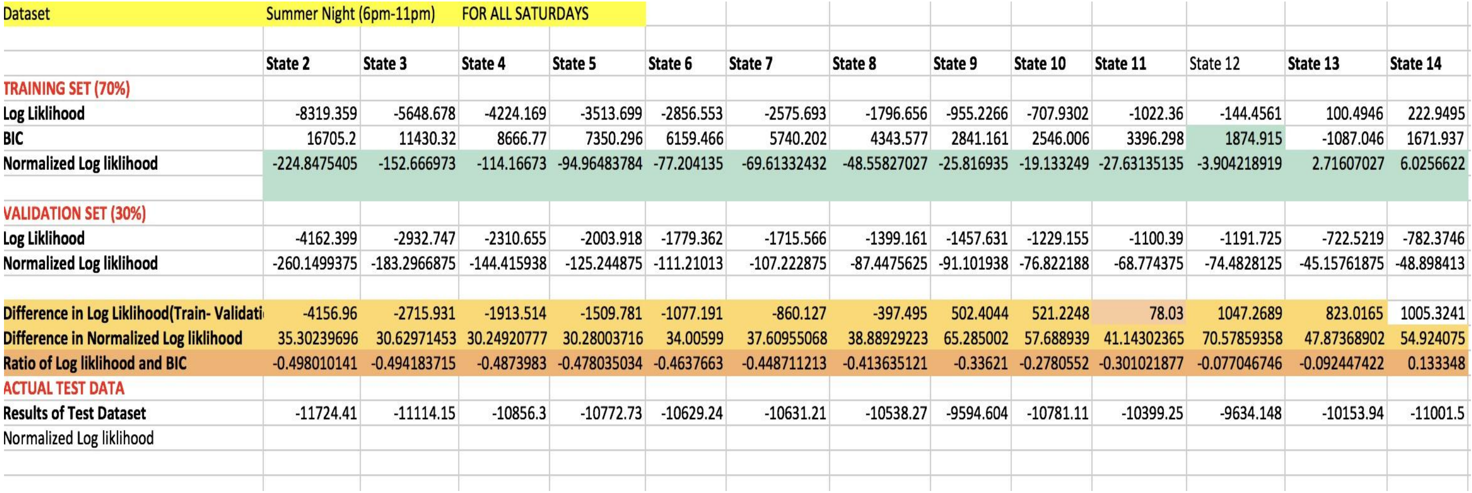
3. We put the numbers into an excel spreadsheet to start comparing our values and finding the best state, as seen in Figure 12 below.

4.To Normalize the Log Likelihood results we divided the result for the train and validation by the number of observations that were included in each result. The rationale behind normalizing the log likelihood is because the pattern length of each observation is different, and getting the average gives you a more accurate result when deciding the train and validation and also the train and test.

5. The way we achieved in finding the best state was by finding a high likelihood and low BIC. We compared each state Likelihood and BIC and made sure we avoided selecting a state which was overfitting our data.

6. After conducting this test we compared the Normalized log likelihood for the train and validation set .We used Microsoft Excel to record the difference between the two Log Likelihood (Train and Validation) to better assist in choosing a state, as seen in Figure 11 below.

7. After finding the best model with the best number of states we then used the “Train” data set been provided and compared the **Normalized Log Likelihood** to see if the train set had anomalies



Dataset	Summer Night (6pm-11pm)		FOR ALL SATURDAYS										
	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9	State 10	State 11	State 12	State 13	State 14
TRAINING SET (70%)													
Log Liklihood	-8319.359	-5648.678	-4224.169	-3513.699	-2856.553	-2575.693	-1796.656	-955.2266	-707.9302	-1022.36	-144.4561	100.4946	222.9495
BIC	16705.2	11430.32	8666.77	7350.296	6159.466	5740.202	4343.577	2841.161	2546.006	3396.298	1874.915	-1087.046	1671.937
Normalized Log liklihood	-224.8475405	-152.666973	-114.16673	-94.96483784	-77.204135	-69.61332432	-48.55827027	-25.816935	-19.133249	-27.63135135	-3.904218919	2.71607027	6.0256622
VALIDATION SET (30%)													
Log Liklihood	-4162.399	-2932.747	-2310.655	-2003.918	-1779.362	-1715.566	-1399.161	-1457.631	-1229.155	-1100.39	-1191.725	-722.5219	-782.3746
Normalized Log liklihood	-260.1499375	-183.2966875	-144.415938	-125.244875	-111.21013	-107.222875	-87.4475625	-91.101938	-76.822188	-68.774375	-74.4828125	-45.15761875	-48.898413
Difference in Log Liklihood(Train- Validati	-4156.96	-2715.931	-1913.514	-1509.781	-1077.191	-860.127	-397.495	502.4044	521.2248	78.03	1047.2689	823.0165	1005.3241
Difference in Normalized Log liklihood	35.30239696	30.62971453	30.24920777	30.28003716	34.00599	37.60955068	38.88929223	65.285002	57.688939	41.14302365	70.57859358	47.87368902	54.924075
Ratio of Log liklihood and BIC	-0.498010141	-0.494183715	-0.4873983	-0.478035034	-0.4637663	-0.448711213	-0.413635121	-0.33621	-0.2780552	-0.301021877	-0.077046746	-0.092447422	0.133348
ACTUAL TEST DATA													
Results of Test Dataset	-11724.41	-11114.15	-10856.3	-10772.73	-10629.24	-10631.21	-10538.27	-9594.604	-10781.11	-10399.25	-9634.148	-10153.94	-11001.5
Normalized Log liklihood													

Figure 11: Table Showing Our HMM Model Analysis for the Summer Saturday between (6:00pm-11:59pm)

Results for Anomaly Detection

In the sections below we will describe the solutions we got after using the Outlier Detection, Moving Average, and HMM model in R.

Point Anomalies Result

Min-Max (Outlier Detection)

The **Feature we picked for this was global active power** and our rationale behind this is because it represents the true actual power. We also found during the data exploration that the test dataset had a lower min compared to the train dataset hence, we have more anomalies that were less than the min therefore, the authors believed it would be a good idea to explore this.

Window	Min	Max
Monday-Friday	0.076	8.944
Saturday-Sunday	0.078	8.76
Monday-Friday (6:00AM-5:59PM)	0.078	8.76
Monday-Friday(6:00PM-11:59PM)	0.078	7.652
Saturday-Sunday (6:00AM-5:59PM)	0.078	8.944
Saturday-Sunday (6:00PM-11:59PM)	0.076	8.694

We found different min and max for the different windows to facilitate us to have more accurate results when finding anomalies in the test dataset. The above results are from the Train Dataset. We didn’t find any negative number in the training dataset for the above windows.

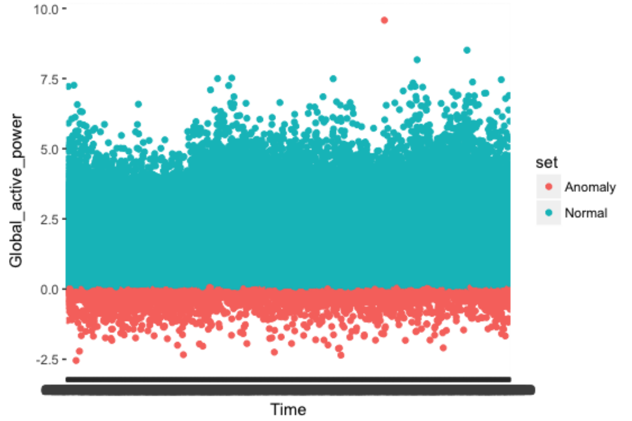


Figure 12 Summer Month (May-August) Monday-Friday

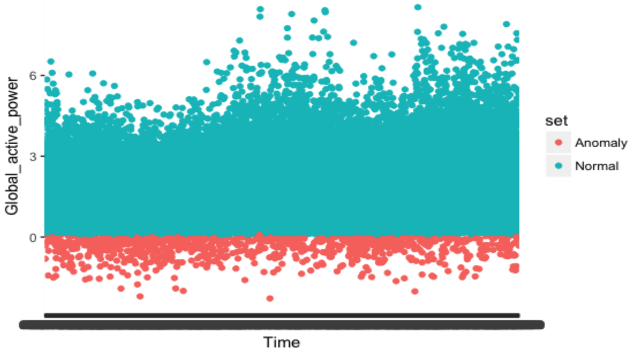


Figure 13 Summer Month (May- August) Saturday-Sunday

Window	Normal	Anomaly
Summer (Monday-Friday) based on the Train Dataset	97.81708595%	2.182914046%
Summer (Saturday-Sunday) Based on the Train Dataset	98.10491676%	1.895083237%

Figure 14: The following is the result for the Summer Test File for the given windows after using the Min-Max Algorithm to find anomalies. This is based on the Global Active Feature

Window	Normal	Anomaly
Winter (Monday-Friday) based on the Train Dataset	98.33592771%	1.664072293%
Winter(Saturday-Sunday) based on the Train Dataset	98.40060764%	1.599392361%

Figure 15 : The following is the result of the Winter Test for the given windows after using the Min-Max Algorithm to find anomalies. This is based on the Global Active Feature.

Moving Average

Using the 68-95-99.7 Threshold Statistical Rule("Three-Sigma Rule of Thumb")

The results below are what we got for using the moving average technique. We used the train set to help us find the anomalies as it had more data, and it also helped us analyze if our function was working correctly. **The feature we used for the Moving Average** was the global active power, as we found this the most useful feature of the dataset. We assumed it was fine to use the train data for this section only. **From Figure 14**, which illustrates Anomalies between Monday-Friday in the Summer, we can see that between 7:00AM-7:59PM there were over 200 anomalies, and between 8:00 PM-8:59 PM and 10:00PM-10:59 PM with about 150 anomalies. In Figure 15, we see the normal data is consistent between Monday-Friday. The Monday-Friday Dataset consisted of 125280 observations and the Saturday and Sunday consisted of 5180 observations.

Window	Normal	Anomaly
Summer Monday-Friday(Train Dataset)	97.70973668 %	2.290263319%
Summer Saturday-Sunday(Train Dataset)	97.67939815 %	2.320601852%

Figure 16: Percent of Normal and Anomalies based on 2008 Summer Year Train Dataset.

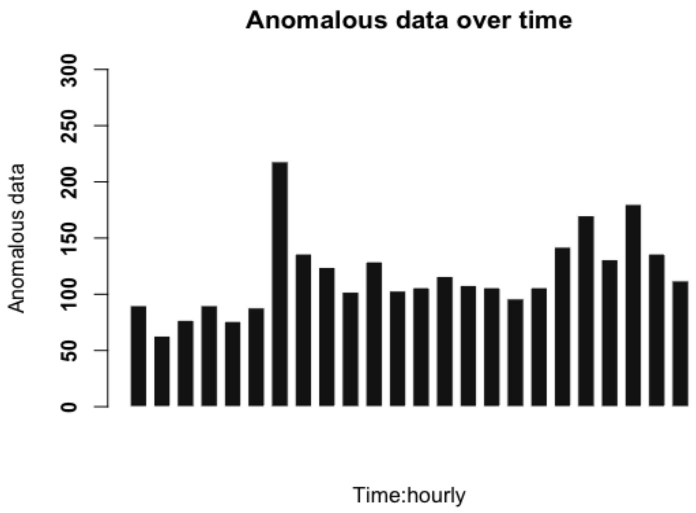


Figure 17 Moving Average Anomaly Detection (May-August) for Summer Monday-Friday Each bar represents an hour. The first bar represents (00:00-00:59) and so on

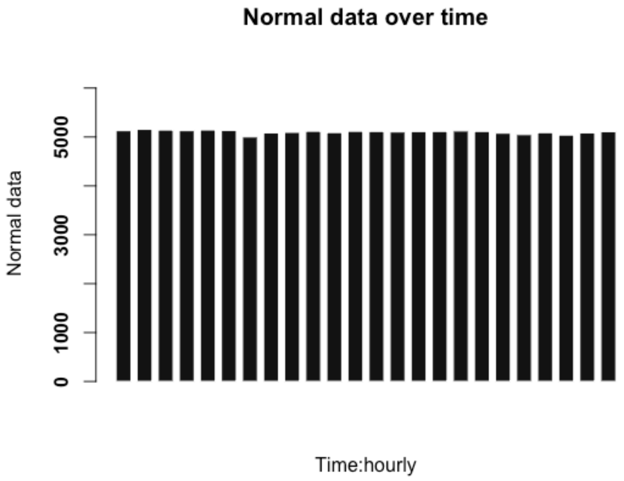


Figure 18(Left) Moving Average Result in Detecting the Normal Data in the Summer dataset(May-August) between Monday-Friday

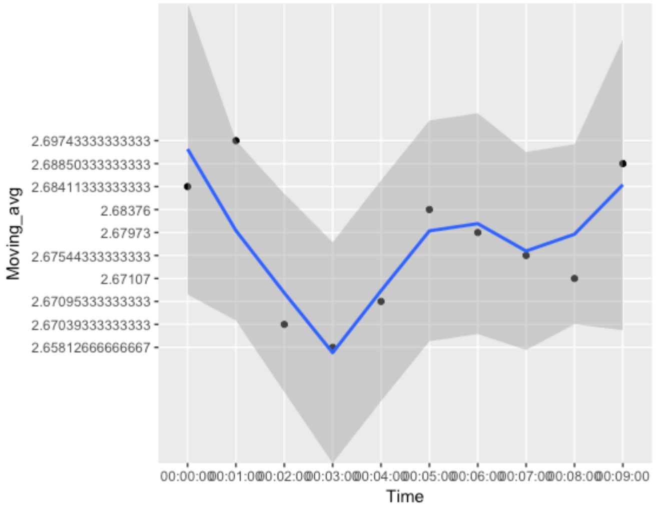


Figure 19: The graph on the left illustrates how our moving average function works. We got a graph that does the moving average for 10 windows which are each 15 minutes.

Static Threshold for Moving Average

Compared to the dynamic threshold method above ("Three-Sigma Rule of Thumb"), we also did a static threshold for 0.5,1,2 we selected these thresholds based on the mean, min and max. When the threshold was larger we had a smaller number of anomalies. Due to space constraints we have provided a link below to view the graph for Monday-Friday, and Saturday-Sunday in the Summer 2008.

Monday-Friday: https://drive.google.com/drive/folders/1IqIPbvvJeMbO8yxnb6DrEbBwivxvkvc7

Satarday-Sundayhttps://drive.google.com/drive/folders/1-0syee46KiQWJk-Ns0mo1R4JzDODIfYQ

Exact Value for each Window in the graph:

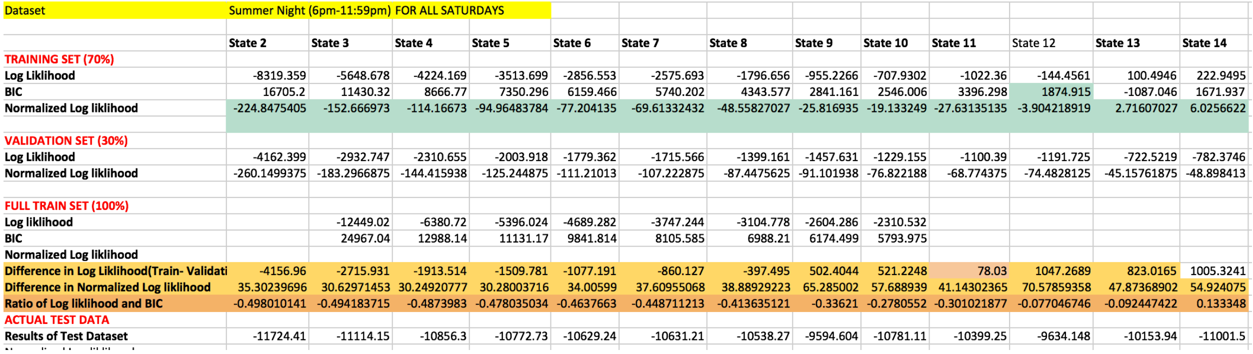
https://docs.google.com/spreadsheets/d/1imZjIwN8Dugq92yC-FPf73o-e2T7HjCWnNaMAxXEZ_Q/edit - gid=0

Contextual Anomalies using the HMM Model

In the models below we used the **global active power feature**, as mentioned above in the data exploration section of the report, as we found this to be the most useful feature in the data set because it represents the real power consumption. We decided to work with univariate HMM, as during the data exploration we noticed that this specific feature had patterns in the data, and we wanted to analyze this feature by finding more patterns that don't conform with the normal behavior. In this part we simplified our models by picking one day in the weekend, and one day in the weekday to find contextual anomalies. We did Summer and Winter Night for Saturday and Summer Evenings for Wednesday. We did an analysis for each model as seen in Figure **11** for the models before deciding the number of states it requires. We used the time's concept when building the HMM model, as each Saturday was counted as a different observation from the next Saturday. We defined each observation by the number of minutes. We used the same concept for Wednesday's when building the HMM.

<u>Model 1: Findings for Summer Evening Data set(May- August) 6:00PM-11:59PM for all Saturday's (Feature: Global Active Power)</u>

The Log Likelihood of our Train Data set was -1763.342 and the BIC was 4699.596 with 10 states. **Our rational** for picking 10 states was that State 11 was overfitting as we can see the BIC value increased, and State 13 and 14 had positive log likelihoods, and we weren't sure if this was overfitting our data. After normalizing the Log Likelihood by the number of observations we got a result of **-33.27060377** for the train data set. We then used the parameters of our train data set to compare it with the test data set been provided ,and the results we got for the Log Likelihood was -10788.54 and after normalizing this result we got a Log Likelihood of **-599.3633333.** There was a difference between the two log likelihoods, and this was a clear indication that there were contextual anomalies in the test data set.



Dataset	Summer Night (6pm-11:59pm) FOR ALL SATURDAYS												
	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9	State 10	State 11	State 12	State 13	State 14
TRAINING SET (70%)													
Log Liklihood	-8319.359	-5648.678	-4224.169	-3513.699	-2856.553	-2575.693	-1796.656	-955.2266	-707.9302	-1022.36	-144.4561	100.4946	222.9495
BIC	16705.2	11430.32	8666.77	7350.296	6159.466	5740.202	4343.577	2841.161	2546.006	3396.298	1874.915	-1087.046	1671.937
Normalized Log liklihood	-224.8475405	-152.666973	-114.16673	-94.96483784	-77.204135	-69.61332432	-48.55827027	-25.816935	-19.133249	-27.63135135	-3.904218919	2.71607027	6.0256622
VALIDATION SET (30%)													
Log Liklihood	-4162.399	-2932.747	-2310.655	-2003.918	-1779.362	-1715.566	-1399.161	-1457.631	-1229.155	-1100.39	-1191.725	-722.5219	-782.3746
Normalized Log liklihood	-260.1499375	-183.2966875	-144.415938	-125.244875	-111.21013	-107.222875	-87.4475625	-91.101938	-76.822188	-68.774375	-74.4828125	-45.15761875	-48.898413
FULL TRAIN SET (100%)													
Log liklihood		-12449.02	-6380.72	-5396.024	-4689.282	-3747.244	-3104.778	-2604.286	-2310.532				
BIC		24967.04	12988.14	11131.17	9841.814	8105.585	6988.21	6174.499	5793.975				
Normalized Log liklihood													
Difference in Log Liklihood(Train- Validati	-4156.96	-2715.931	-1913.514	-1509.781	-1077.191	-860.127	-397.495	502.4044	521.2248	78.03	1047.2689	823.0165	1005.3241
Difference in Normalized Log liklihood	35.30239696	30.62971453	30.24920777	30.28003716	34.00599	37.60955068	38.88929223	65.285002	57.688939	41.14302365	70.57859358	47.87368902	54.924075
Ratio of Log liklihood and BIC	-0.498010141	-0.494183715	-0.4873983	-0.478035034	-0.4637663	-0.448711213	-0.413635121	-0.33621	-0.2780552	-0.301021877	-0.077046746	-0.092447422	0.133348
ACTUAL TEST DATA													
Results of Test Dataset	-11724.41	-11114.15	-10856.3	-10772.73	-10629.24	-10631.21	-10538.27	-9594.604	-10781.11	-10399.25	-9634.148	-10153.94	-11001.5

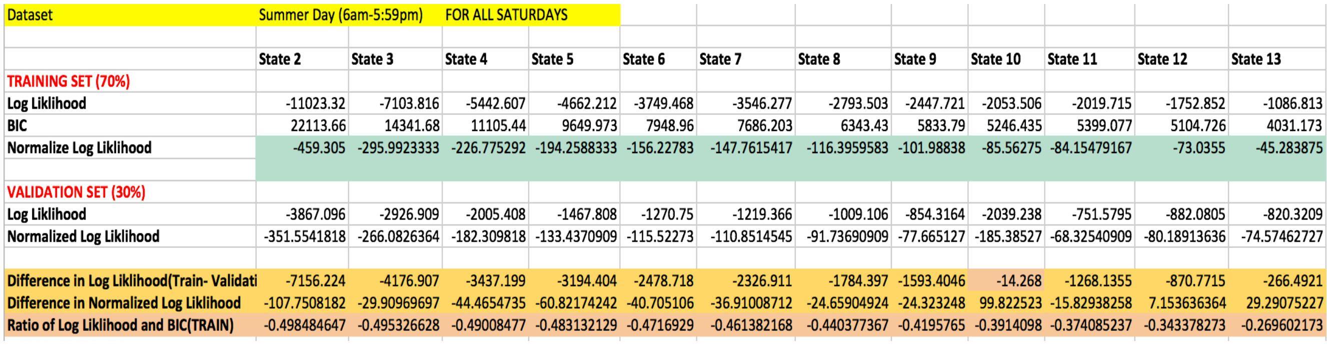
Figure 20: HMM Model Analysis for Summer Night between 6pm-11:59pm

Measure	Train Dataset (100%)	Test
Log Likelihood	-1763.342	-10788.54
Log Likelihood **after Normalizing**	**-33.27060377**	**-599.3633333**

Number of States	**10**	**10**

Model 2: Findings for Summer Day Data set(May-August) 6:00AM-5:59PM for all Saturdays

The Log Likelihood of our Train Data set was -1347.924, and the BIC was 4357.878 with 12 states. After Normalizing the result of our Train Log Likelihood I got a result of **-38.51211429.** We checked the Log Likelihood of our Test Data set and got a log likelihood of -16231.66, and after normalizing the result we got the log likelihood to be **-901.7588889.** Furthermore, we also ended up picking state 12 because the difference in the Log Likelihood between the Train and Validation set was only 7.15, and it had the best Log Likelihood and BIC given the difference. We couldn't go through more states because of time constraints and we noticed the data was overfitting between states 11, and used 12 .



Dataset	Summer Day (6am-5:59pm)		FOR ALL SATURDAYS									
	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9	State 10	State 11	State 12	State 13
TRAINING SET (70%)												
Log Liklihood	-11023.32	-7103.816	-5442.607	-4662.212	-3749.468	-3546.277	-2793.503	-2447.721	-2053.506	-2019.715	-1752.852	-1086.813
BIC	22113.66	14341.68	11105.44	9649.973	7948.96	7686.203	6343.43	5833.79	5246.435	5399.077	5104.726	4031.173
Normalize Log Liklihood	-459.305	-295.9923333	-226.775292	-194.2588333	-156.22783	-147.7615417	-116.3959583	-101.98838	-85.56275	-84.15479167	-73.0355	-45.283875
VALIDATION SET (30%)												
Log Liklihood	-3867.096	-2926.909	-2005.408	-1467.808	-1270.75	-1219.366	-1009.106	-854.3164	-2039.238	-751.5795	-882.0805	-820.3209
Normalized Log Liklihood	-351.5541818	-266.0826364	-182.309818	-133.4370909	-115.52273	-110.8514545	-91.73690909	-77.665127	-185.38527	-68.32540909	-80.18913636	-74.57462727
Difference in Log Liklihood(Train- Validati	-7156.224	-4176.907	-3437.199	-3194.404	-2478.718	-2326.911	-1784.397	-1593.4046	-14.268	-1268.1355	-870.7715	-266.4921
Difference in Normalized Log Liklihood	-107.7508182	-29.90969697	-44.4654735	-60.82174242	-40.705106	-36.91008712	-24.65904924	-24.323248	99.822523	-15.82938258	7.153636364	29.29075227
Ratio of Log Liklihood and BIC(TRAIN)	-0.498484647	-0.495326628	-0.49008477	-0.483132129	-0.4716929	-0.461382168	-0.440377367	-0.4195765	-0.3914098	-0.374085237	-0.343378273	-0.269602173

Figure 21: HMM model Analysis for Summer Day (6AM-5:59PM) for all Saturday's

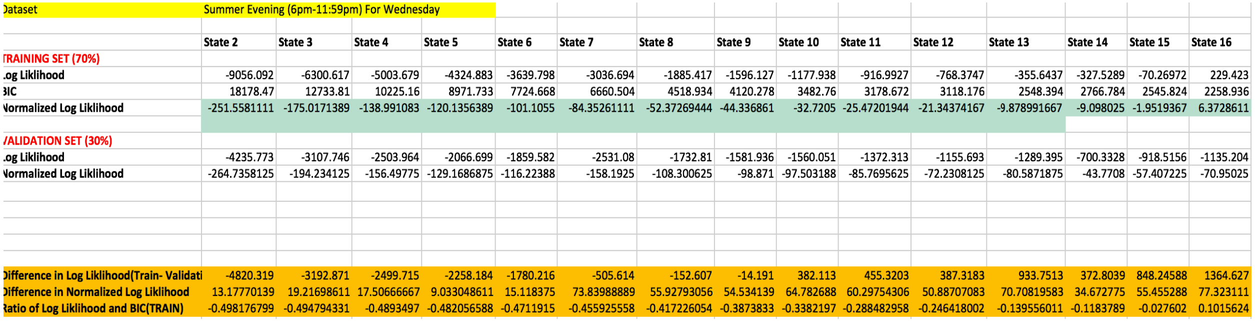
Measure	Train Dataset(100%)	Test
Log Likelihood	-1347.924	-16231.66

Log Likelihood **After Normalizing**	**-38.51211429**	**-901.7588889**
Number of States	**12**	**12**

The difference in the log likelihood shows that there were contextual anomalies in our Test Data set between this time window.

<u>Model 3: Finding Contextual anomalies for Summer (May-August) 6pm-11:59pm for all Wednesday's</u>

We decided to build a model for Wednesday's to detect anomalies in the evening hours after work. The reason why we picked Wednesday is because we felt that it's the midweek, and we wanted to explore the patterns between this time window. Our HMM Train model had a **log likelihood of -1391.432 and a BIC of 4976.593. The Rational** behind picking 14 states is that the difference between the Train and Validation Normalized Log likelihoods was only 34.672775, and state 14 had a high likelihood and low BIC, and state 15 had a higher difference between train and validation likelihood. We didn't pick state 5 even though it had a small difference between the train and validation log likelihood, and the reason behind this is because we got a better Log Likelihood and BIC in state 14 ,and this two units of measurement help us find the best model. **We can** see a difference between the Log Likelihoods of the Train and Test below, therefore this indicates there were contextual anomalies in this specific window



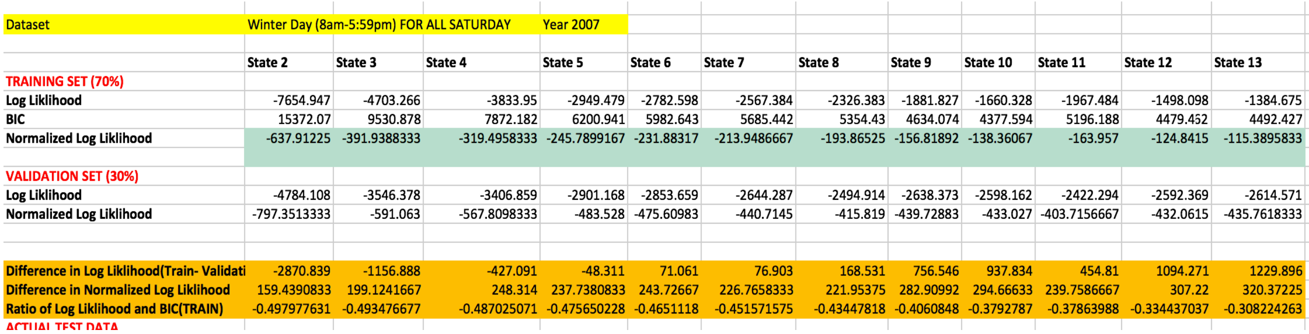
Dataset	Summer Evening (6pm-11:59pm) For Wednesday														
	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9	State 10	State 11	State 12	State 13	State 14	State 15	State 16
TRAINING SET (70%)															
Log Liklihood	-9056.092	-6300.617	-5003.679	-4324.883	-3639.798	-3036.694	-1885.417	-1596.127	-1177.938	-916.9927	-768.3747	-355.6437	-327.5289	-70.26972	229.423
BIC	18178.47	12733.81	10225.16	8971.733	7724.668	6660.504	4518.934	4120.278	3482.76	3178.672	3118.176	2548.394	2766.784	2545.824	2258.936
Normalized Log Liklihood	-251.5581111	-175.0171389	-138.991083	-120.1356389	-101.1055	-84.35261111	-52.37269444	-44.336861	-32.7205	-25.47201944	-21.34374167	-9.878991667	-9.098025	-1.9519367	6.3728611
VALIDATION SET (30%)															
Log Liklihood	-4235.773	-3107.746	-2503.964	-2066.699	-1859.582	-2531.08	-1732.81	-1581.936	-1560.051	-1372.313	-1155.693	-1289.395	-700.3328	-918.5156	-1135.204
Normalized Log Liklihood	-264.7358125	-194.234125	-156.49775	-129.1686875	-116.22388	-158.1925	-108.300625	-98.871	-97.503188	-85.7695625	-72.2308125	-80.5871875	-43.7708	-57.407225	-70.95025
Difference in Log Liklihood(Train- Validati	-4820.319	-3192.871	-2499.715	-2258.184	-1780.216	-505.614	-152.607	-14.191	382.113	455.3203	387.3183	933.7513	372.8039	848.24588	1364.627
Difference in Normalized Log Liklihood	13.17770139	19.21698611	17.50666667	9.033048611	15.118375	73.83988889	55.92793056	54.534139	64.782688	60.29754306	50.88707083	70.70819583	34.672775	55.455288	77.323111
Ratio of Log Liklihood and BIC(TRAIN)	-0.498176799	-0.494794331	-0.4893497	-0.482056588	-0.4711915	-0.455925558	-0.417226054	-0.3873833	-0.3382197	-0.288482958	-0.246418002	-0.139556011	-0.1183789	-0.027602	0.1015624

Figure 22: HMM model analysis for Summer Evening (6PM-11:59PM) for Wednesdays

Measure	Train Dataset(100%)	Test
Log Likelihood	-1391.432	-10128.64
Log Likelihood After Normalizing	**-26.75830769**	**-595.8023529**
Number of States	**14**	**14**

Model 4: Finding for Winter Day (September- December) 8:00AM-5:59PM for all Saturday's

The Log Likelihood of the train data set was -4073.709 and BIC was 9252.607 with 10 states. After normalizing the log likelihood we got -226.317167. We compared our Log Likelihood to the train data set been provided, and we got a Log Likelihood of -15405.97 and after normalizing this log likelihood we got -962.873125 therefore, indicating that our test data set for the given window had contextual anomalies.



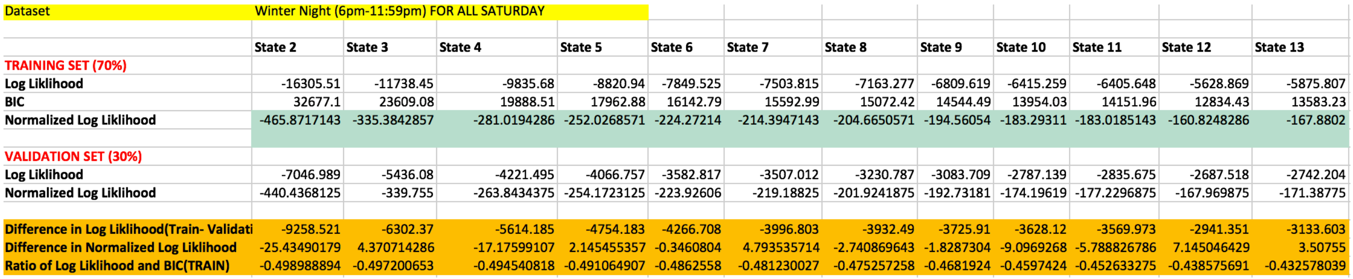
Dataset	Winter Day (8am-5:59pm) FOR ALL SATURDAY			Year 2007								
	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9	State 10	State 11	State 12	State 13
TRAINING SET (70%)												
Log Liklihood	-7654.947	-4703.266	-3833.95	-2949.479	-2782.598	-2567.384	-2326.383	-1881.827	-1660.328	-1967.484	-1498.098	-1384.675
BIC	15372.07	9530.878	7872.182	6200.941	5982.643	5685.442	5354.43	4634.074	4377.594	5196.188	4479.452	4492.427
Normalized Log Liklihood	-637.91225	-391.9388333	-319.4958333	-245.7899167	-231.88317	-213.9486667	-193.86525	-156.81892	-138.36067	-163.957	-124.8415	-115.3895833
VALIDATION SET (30%)												
Log Liklihood	-4784.108	-3546.378	-3406.859	-2901.168	-2853.659	-2644.287	-2494.914	-2638.373	-2598.162	-2422.294	-2592.369	-2614.571
Normalized Log Liklihood	-797.3513333	-591.063	-567.8098333	-483.528	-475.60983	-440.7145	-415.819	-439.72883	-433.027	-403.7156667	-432.0615	-435.7618333
Difference in Log Liklihood(Train- Validati	-2870.839	-1156.888	-427.091	-48.311	71.061	76.903	168.531	756.546	937.834	454.81	1094.271	1229.896
Difference in Normalized Log Liklihood	159.4390833	199.1241667	248.314	237.7380833	243.72667	226.7658333	221.95375	282.90992	294.66633	239.7586667	307.22	320.37225
Ratio of Log Liklihood and BIC(TRAIN)	-0.497977631	-0.493476677	-0.487025071	-0.475650228	-0.4651118	-0.451571575	-0.43447818	-0.4060848	-0.3792787	-0.37863988	-0.334437037	-0.308224263
ACTUAL TEST DATA												

Figure 23: HMM Model Analysis for Winter Day (8:00am-5:59pm) for all Saturday's

Measure	Train Dataset(100%)	Test
Log Likelihood	-4073.709	-15405.97
Log Likelihood after Normalizing	**-226.317167**	**-962.873125**
Number of States	**10**	**10**

Model 5: Finding for Winter Night (6:00PM-11:59PM) for all Saturday's

After training the model we picked the best model with 12 states as it was the one with the best Log Likelihood of -8152.175 and lowest BIC, and after Normalizing the Likelihood we got **-159.8465686.** For the Test Data set we got a Log likelihood of -8859.285 and after normalizing it we got **-553.7053125.** There was a small difference between the two values which indicates that they were few contextual anomalies in our test data set. This model had a few flaws, as there was some Missing Data in the dataset which may have affected the final result accuracy.



Dataset	Winter Night (6pm-11:59pm) FOR ALL SATURDAY											
	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9	State 10	State 11	State 12	State 13
TRAINING SET (70%)												
Log Liklihood	-16305.51	-11738.45	-9835.68	-8820.94	-7849.525	-7503.815	-7163.277	-6809.619	-6415.259	-6405.648	-5628.869	-5875.807
BIC	32677.1	23609.08	19888.51	17962.88	16142.79	15592.99	15072.42	14544.49	13954.03	14151.96	12834.43	13583.23
Normalized Log Liklihood	-465.8717143	-335.3842857	-281.0194286	-252.0268571	-224.27214	-214.3947143	-204.6650571	-194.56054	-183.29311	-183.0185143	-160.8248286	-167.8802
VALIDATION SET (30%)												
Log Liklihood	-7046.989	-5436.08	-4221.495	-4066.757	-3582.817	-3507.012	-3230.787	-3083.709	-2787.139	-2835.675	-2687.518	-2742.204
Normalized Log Liklihood	-440.4368125	-339.755	-263.8434375	-254.1723125	-223.92606	-219.18825	-201.9241875	-192.73181	-174.19619	-177.2296875	-167.969875	-171.38775
Difference in Log Liklihood(Train- Validati	-9258.521	-6302.37	-5614.185	-4754.183	-4266.708	-3996.803	-3932.49	-3725.91	-3628.12	-3569.973	-2941.351	-3133.603
Difference in Normalized Log Liklihood	-25.43490179	4.370714286	-17.17599107	2.145455357	-0.3460804	4.793535714	-2.740869643	-1.8287304	-9.0969268	-5.788826786	7.145046429	3.50755
Ratio of Log Liklihood and BIC(TRAIN)	-0.498988894	-0.497200653	-0.494540818	-0.491064907	-0.4862558	-0.481230027	-0.475257258	-0.4681924	-0.4597424	-0.452633275	-0.438575691	-0.432578039

Figure 24: HMM Model Analysis for Saturday's in the Winter Season

Measure	Train Dataset(100%)	Test
Log Likelihood	-8152.175	-8859.285
Log Likelihood after Normalizing	**-159.8465686**	**-553.7053125**
Number of States	12	12

Major problems encountered in the course of the project

We encountered several problems through the project.

1. Understanding the R language and the syntax. We had never worked with data frames before so understanding this concept was quite challenging.
2. Understanding the features of the Data set took us some time, as we needed to know which feature was most beneficial to us, and this required data exploration to check for average, standard deviation, correlation etc.
3. Splitting the data into seasons took us time. It required us to create extra columns in our data set to be able to extract certain data records from the frame. E.g. splitting the time into hour, minute and seconds.
4. Converting date format from factor class to numerical
5. Understanding the Moving Average concept, and how to use it to find point anomalies.
6. We first had issues deciding to pick a certain window frame. E.g. Seasons, Weekends, Fridays nights

7. Selecting the number of states for our HMM took use time, as we had to look for a high log likelihood, and low BIC which took us some time.
8. Splitting the test data into (test and validation set) for the HMM model.
9. How to train the model using the functions in R.
10. Positive Log Likelihood in HMM models, not enough research to show if positive log likelihoods doesn't lead to overfitting.

The Lessons learned.

1. Start early to learn R.
2. Review basic statistics terminology/methods.
3. Follow DRY principles and use existing packages.
4. Read the manual provided in the package to understand the uses.
5. Working with HMM models, learnt how to train data set more efficiently for better robustness.
6. Designing an HMM model requires you to recognize the patterns in the data to determine what window to use.

Conclusions

By using the moving average, min and max to detect outliers, and HMM Models, we were able to make the anomaly detection process easier. We managed to find anomalies in our test dataset using those methods. There were additional steps that were required to solve this problem. It could not have been done without understanding the problem scope, understanding the features in the data set, splitting the data and exploring the data set. The methodology we used was integral towards the success of our final results listed above.

We achieved in finding the point anomalies through two techniques min-max and moving average. We detected the outliers in the Summer between Monday-Friday and Saturday-Sunday , and as seen in **Figure 12** and **13**. The Figures illustrates anomalies over a period of 4 months, and each bar represents anomalies in a particular hour. From the graph above we observed more anomalies between the following time windows. 7:00AM-7:59AM , 8:00PM-8:59 PM , and 10:00PM-10:59 PM.

The contextual anomalies were found by building HMM models, as seen above in the methodology section. We managed to train five different models. We found contextual anomalies in our summer evening data between (6:00PM-11:59PM), our log likelihood of the train dataset was -33.27060377 after normalizing, and the test dataset had a log likelihood of -599.3633333 after normalizing. The log likelihood indicates that the patterns were different for the test dataset which indicates contextual anomalies. Furthermore, we managed to find more anomalies in the Summer day (6:00AM-5:59PM) where our log likelihood was -38.51211429 for the train dataset, and the test dataset had a log likelihood of -901.7588889. There was a significant difference between the two log likelihood which indicated contextual anomalies were present between this time window in the test dataset. During the weekday specifically Wednesday of the Summer between the hours (6:00 AM-5:59PM) our HMM model detected contextual anomalies with a normalized log likelihood of -26.75830769 and after running the train model on the test data we got a normalized log likelihood of -595.8023529.

Overall in this project we managed to find point anomalies, and contextual anomalies for the summer, and parts of the winter season. We have built functions that can detect anomalies for any given seasons.

References

[1] *Eia.gov*, 2018. [Online]. Available: https://www.eia.gov/todayinenergy/detail.php?id=830. [Accessed: 31- Mar- 2018].

[2] "True, Reactive, and Apparent Power | Power Factor | Electronics Textbook", Allaboutcircuits.com, 2018. [Online]. Available: https://www.allaboutcircuits.com/textbook/alternating-current/chpt-11/true-reactive-and-apparent-power/. [Accessed: 31- Mar- 2018].

[3]"North American Electric Reliability Corporation", En.wikipedia.org, 2018. [Online]. Available: https://en.wikipedia.org/wiki/North_American_Electric_Reliability_Corporation#/media/File:NEC-map-en.svg. [Accessed: 01- Apr- 2018].

[4]”Hacking Power Grids: A Current Problem”, wmcyberintrusion.info, 2007. [Online]. Available: http://wmcyberintrusion.info/wp-content/uploads/2017/11/HackingPowerGrids2017.pdf. [Accessed: 01-Apr-2018]

[5]Hull, J., Khurana, H., Markham, T., & Staggs, K. (2012). Staying in control: Cybersecurity and the modern electric grid. *Power and Energy Magazine, IEEE,10*(1), 41-48.

[6]David W. Cooke, National Research Council, Division on Engineering Physical Sciences, & Board on Energy Environmental Systems. (2013). Cybersecurity of the Grid. In The Resilience of the Electric Power Delivery System in Response to Terrorism and Natural Disasters: Summary of a Workshop (pp. 10-14). National Academies Press.

[7] Hoang Nguyen and Klara Nahrstedt, ”*Detecting Anomalies by Data Aggregation in the Power Grid*”, 2006. [Ebook]. Available:https://www.semanticscholar.org/paper/Detecting-Anomalies-by-Data-Aggregation-in-the-Grid-Nguyen-Nahrstedt/c51c9a261fe0e17b82d9085ba99a996b8f43a80a [Accessed: 01 - Apr - 2018]

[8]Rui Neves-Silva，Lakhmi C. Jain，Robert J. Howlett.Intelligent Decision Technologies: Proceedings of the 7th KES International Conference on Intelligent Decision Technologies (KES-IDT 2015) (Smart Innovation, Systems and Technologies),p508 https://books.google.ca/books?id=ucvWCQAAQBAJ&pg=PA508&lpg=PA508&dq=#v=onepage&q&f=false

[9]"68–95–99.7 rule", *En.wikipedia.org*, 2018. [Online]. Available: https://en.wikipedia.org/wiki/68–95–99.7_rule. [Accessed: 03- Apr- 2018].