Global Power Consumption Report

Authors: Jingtao Cao(301311412),

Liyan Ma(301308359),

Sharad Kochar(301326043),

Weicong Wang(301326659),

Zihao Zhao(301297708)

Course: CMPT 318 Spring 2020

Simon Fraser University

Instructor: Uwe Glaesser

Abstract

This report explores the characteristics of the electricity consumption data and different types of analytic approaches have been proposed based on data's characteristics. This report starts with deciding a time window for deep analysis. The time window is chosen so that the interpretation of its patterns and features is meaningful and generalizable. The analysis uses Hidden Markov Models to train models of different characteristics. The overall process predominantly consists of feature engineering, parameter tuning and comparison against test data.

For feature engineering, the correlation matrix is used to explore the relations between each pair of features. Principal component analysis is used to further improve feature selection.

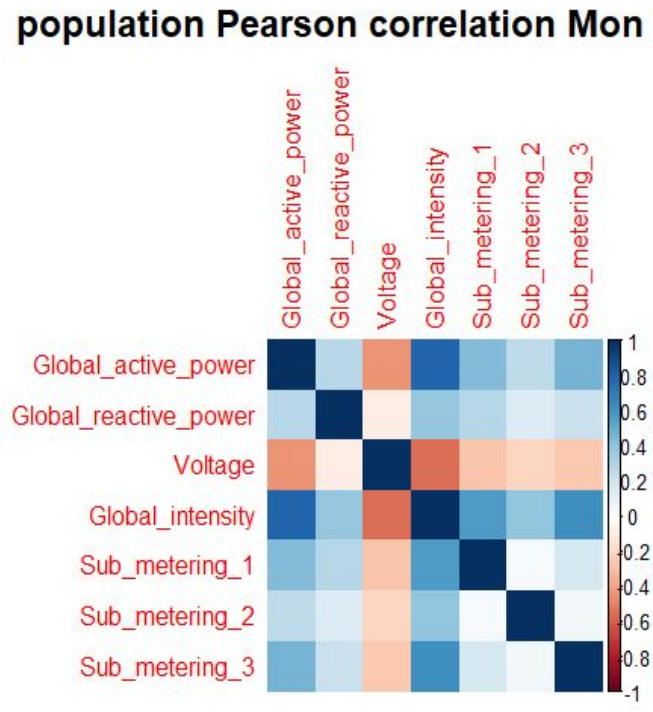
For parameter tuning, the report shows the characteristics of models trained with different numbers of states. The characteristics include log-likelihood and BIC values.

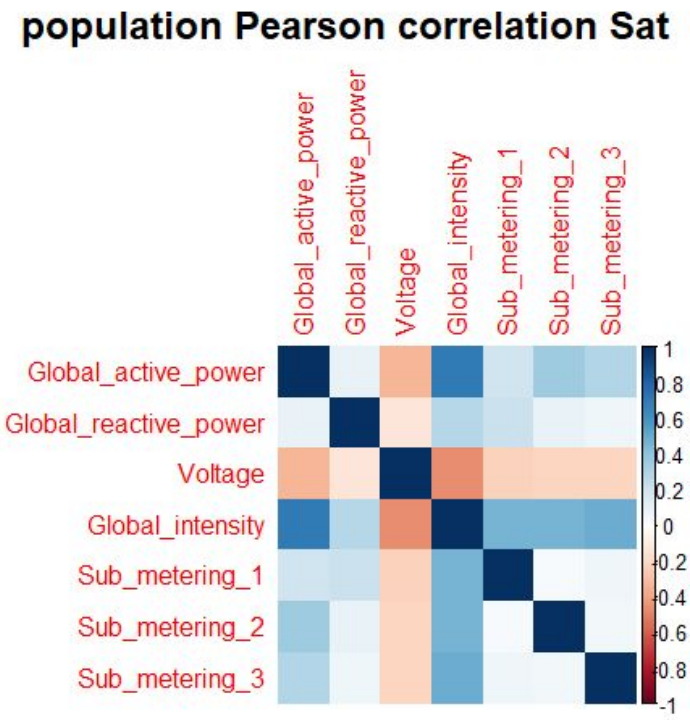
For comparison among models, the log-likelihood and BIC values are recorded both from the train data and test data. An optimized model needs to fit into a sweet spot between accuracy and complexity, while avoiding overfitting.

The report also includes Moving Average method to detect anomalous data points which are distinct from the average of preceding points.

In addition to HMMs, SVM is used to analyze the time window and its characteristics in another perspective.

Data Exploration



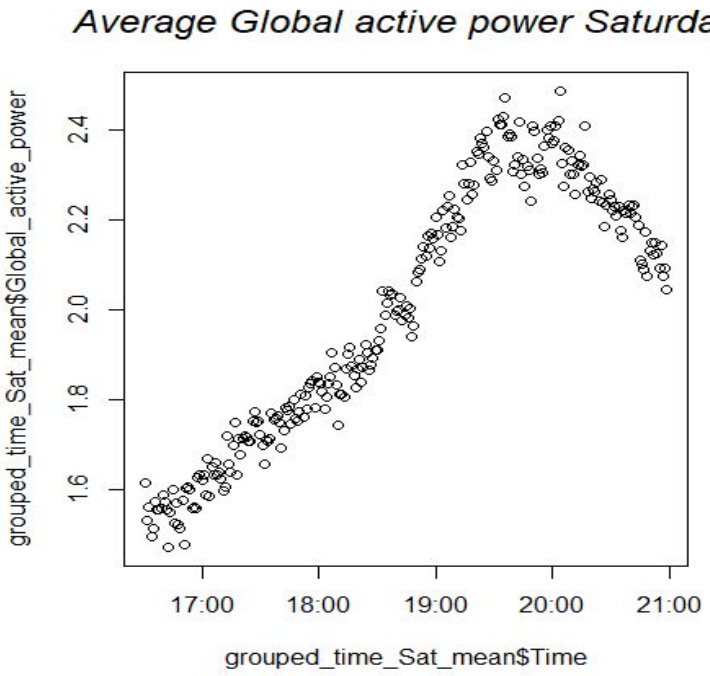


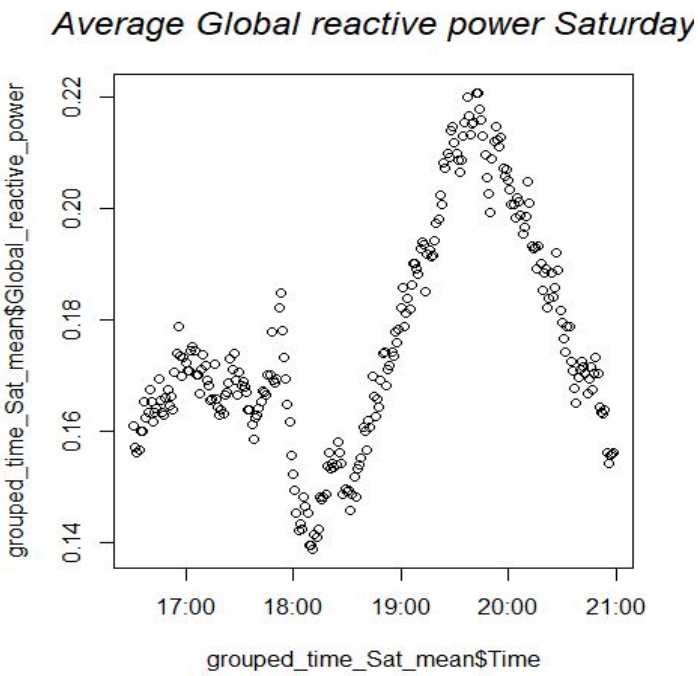
From the two correlation matrices, there is no significant difference between Saturday and Monday. By observing the correlation values of each pair of features, Global_intensity and Global_active_power share a positive relationship. Sub_metering_1 and Global_intensity also share a positive relationship. On the other hand, Voltage and Global_active_power follow a negative relationship, and Voltage and Global_intensity follow a negative relationship.

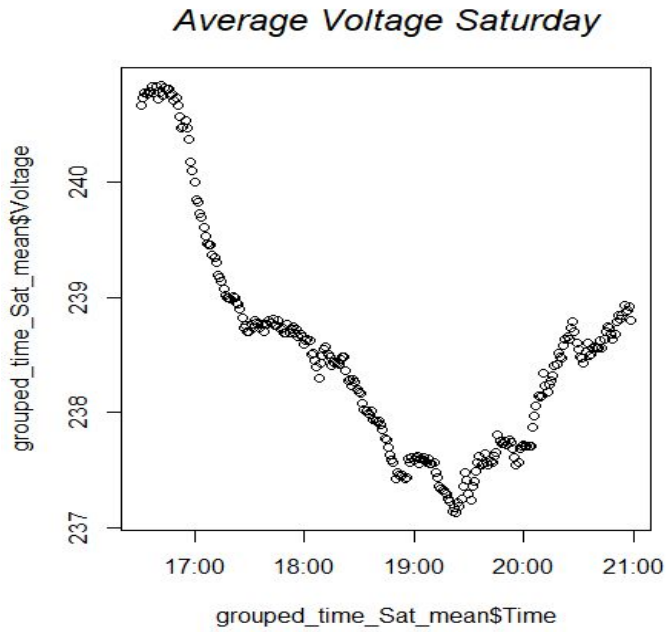
The relationships of other pairs are either positive or negative but they do not form a strong correlation between each other.

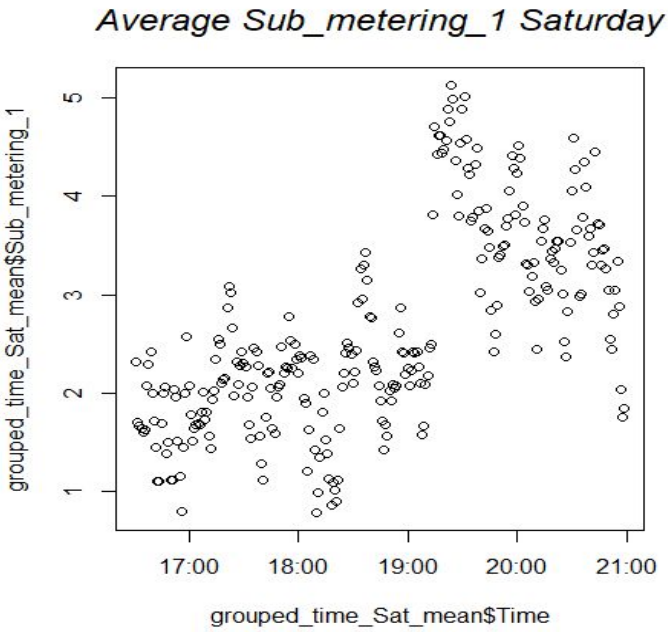
For each feature, the average of data points is calculated and graphed. The graphs are plotted to help decide a time window on Saturday and Monday that shows clear patterns. In this way, models generated from the clear patterns are generalizable and interpretable. Below are the graphs of the pattern of each feature on Monday and Saturday, from 4:30pm to 9:00pm.Furthermore, feature correlation will help in selecting specific features for feature engineering where more distinct features will help us to analyze the data well and similar features which will be reduced using PCA technique.

Graphs

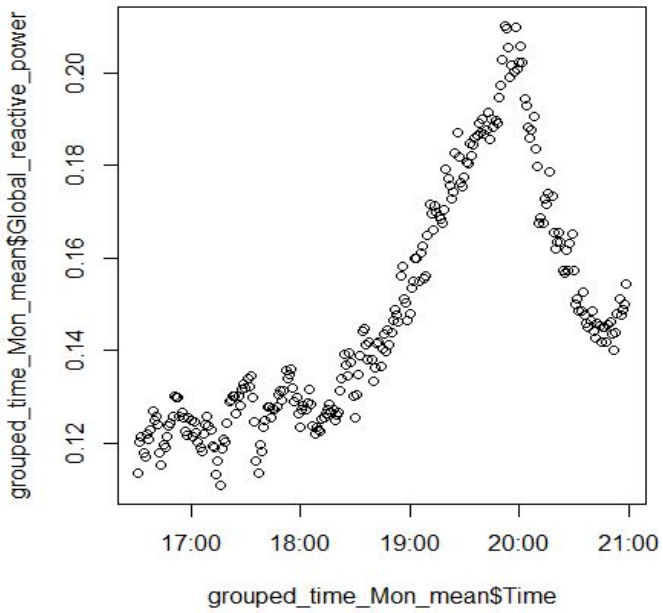

Average Global active power Saturday
grouped_time_Sat_mean$Global_active_power
1.6
1.8
2.0
2.2
2.4
17:00
18:00
19:00
20:00
21:00
grouped_time_Sat_mean$Time


Average Global reactive power Saturday
grouped_time_Sat_mean$Global_reactive_power
0.14
0.16
0.18
0.20
0.22
17:00
18:00
19:00
20:00
21:00
grouped_time_Sat_mean$Time

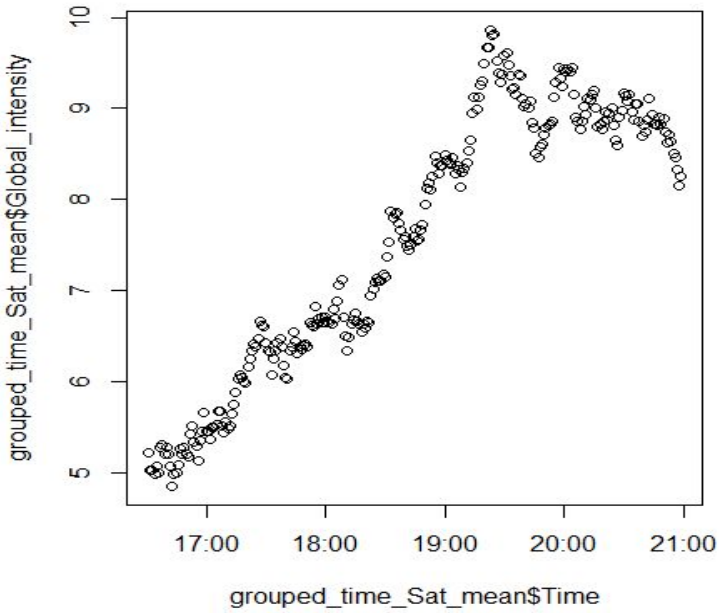

Average Voltage Saturday
grouped_time_Sat_mean$Voltage
237
238
239
240
17:00
18:00
19:00
20:00
21:00
grouped_time_Sat_mean$Time


Average Sub_metering_1 Saturday
grouped_time_Sat_mean$Sub_metering_1
1
2
3
4
5
17:00
18:00
19:00
20:00
21:00
grouped_time_Sat_mean$Time

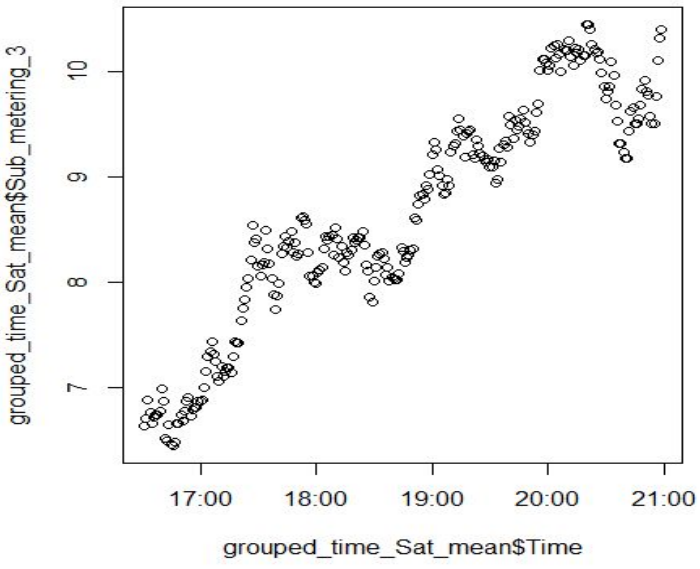
Average Global reactive power Monday



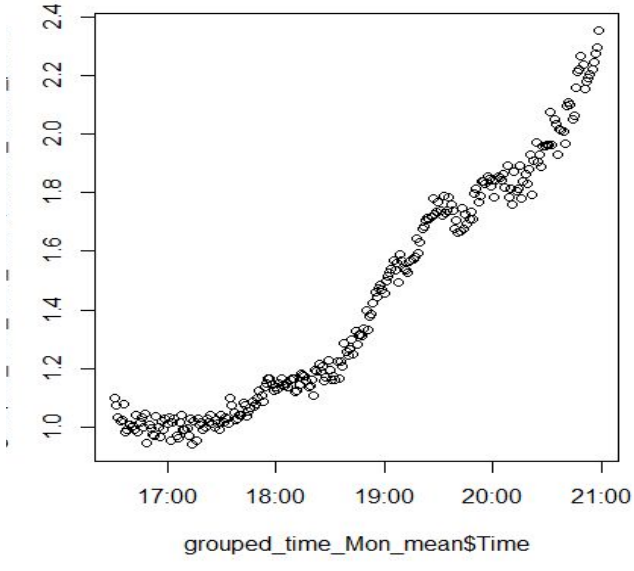
Average Global Intensity Saturday

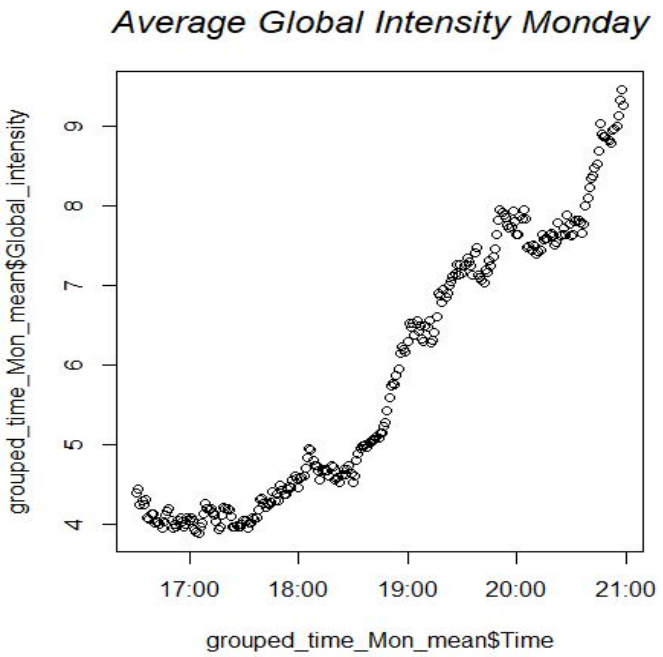


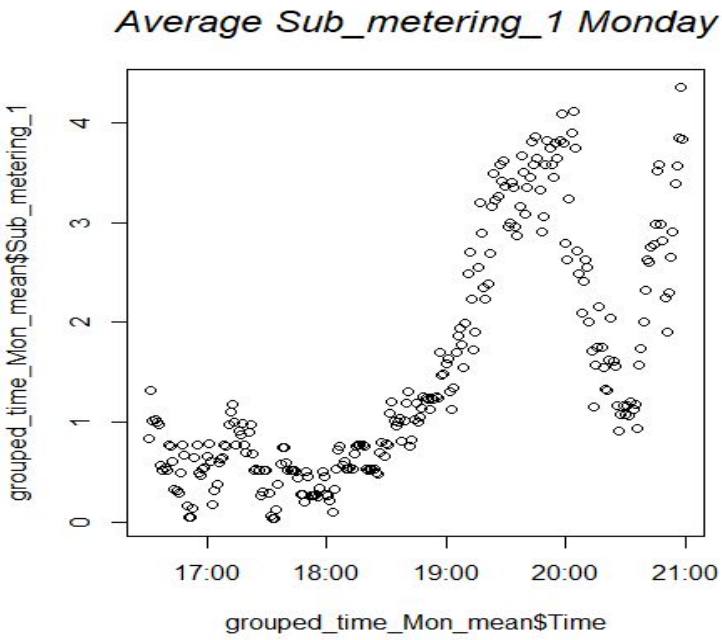
Average Sub_metering_3 Saturday

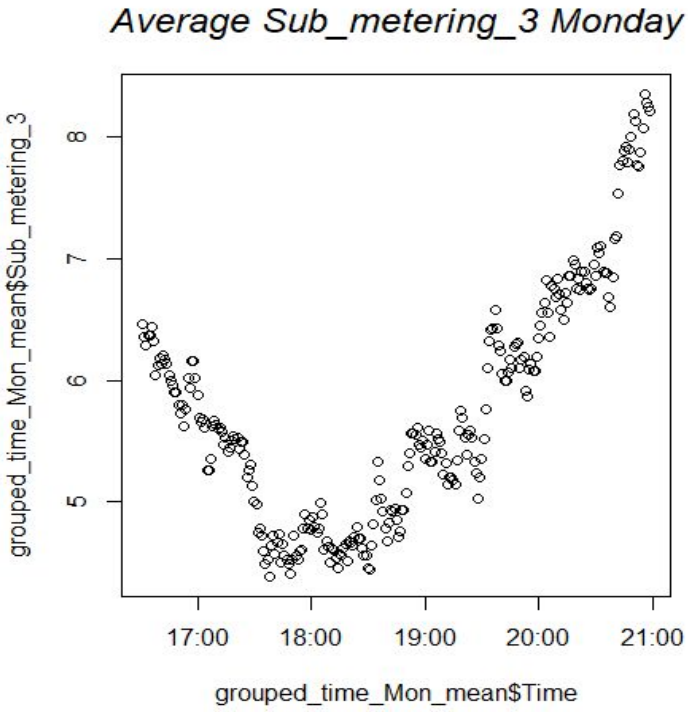


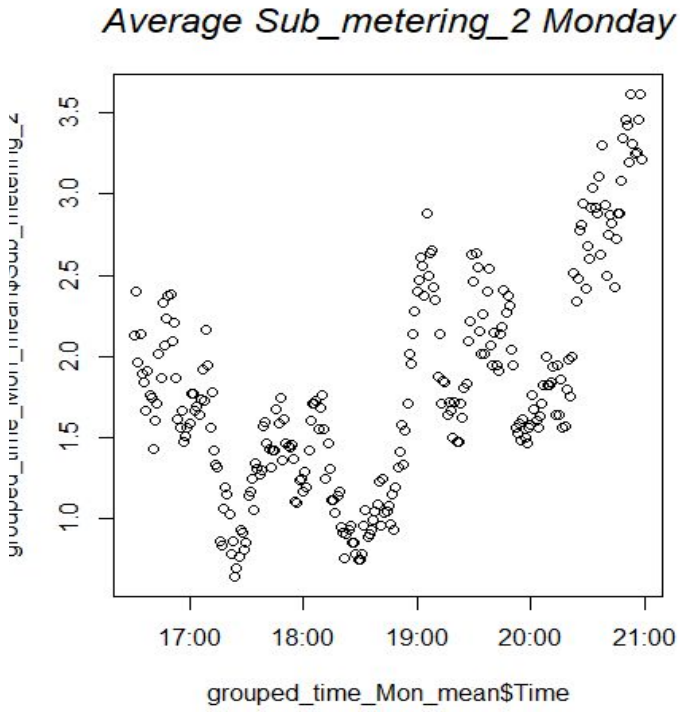
Average Global active power Monday

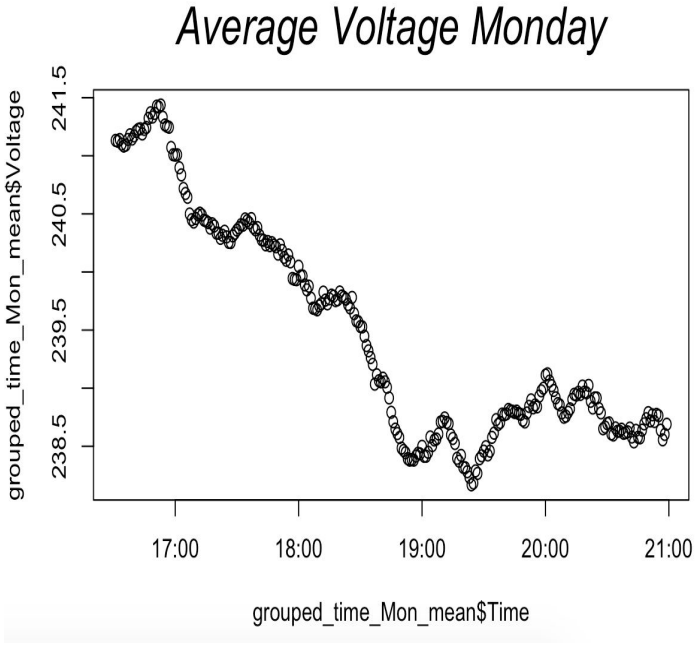



Average Global Intensity Monday
grouped_time_Mon_mean$Global_intensity
4
5
6
7
8
9
17:00
18:00
19:00
20:00
21:00
grouped_time_Mon_mean$Time


Average Sub_metering_1 Monday
grouped_time_Mon_mean$Sub_metering_1
0
1
2
3
4
17:00
18:00
19:00
20:00
21:00
grouped_time_Mon_mean$Time


Average Sub_metering_3 Monday
grouped_time_Mon_mean$Sub_metering_3
5
6
7
8
17:00
18:00
19:00
20:00
21:00
grouped_time_Mon_mean$Time


Average Sub_metering_2 Monday
1.0
1.5
2.0
2.5
3.0
3.5
17:00
18:00
19:00
20:00
21:00
grouped_time_Mon_mean$Time



From these graphs, Global_active_power, Global_reactive_power, Voltage, Global_intensity and Sub_metering_3 form clear trends during the chosen time window. Their clear patterns indicate that the models generated from them are interpretable and accurate, while not overfitting into noise and random values. The trends of some features are similar. For example, Global_Active_power and Global_Intensity both increase steadily from 4:30pm to 8:00pm. After reaching the peak, they slowly decrease. Features that share similar trends will not be fed into the training process since it will be more efficient to analyze one distinct pattern than two similar ones. However, the patterns between Monday and Saturday are occasionally different. Sub_metering_3 on Monday decreases first and then increases while Sub_metering_3 on Saturday keeps on increasing during the time window.

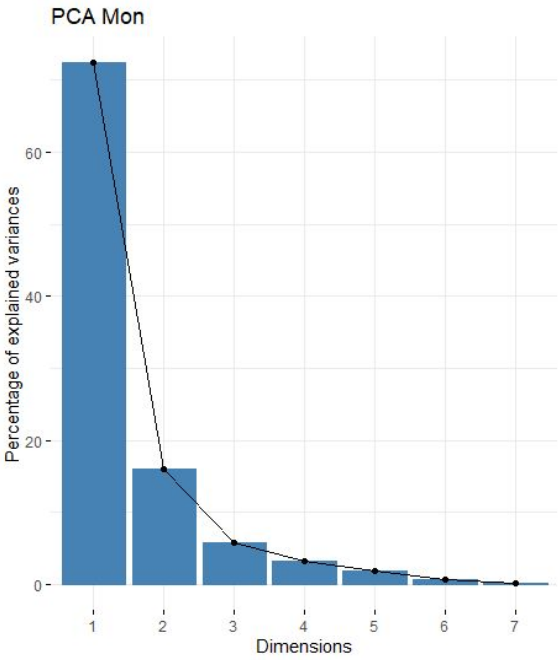
Furthermore, Average Global_Intensity has a constant upwards trend on Monday from 4:30 to 9:30 however on Saturday Global_Intensity has a similar constant upward trend from 4:30 to 7:30 and after 7:30 it starts decreasing. A similar trend on weekend and

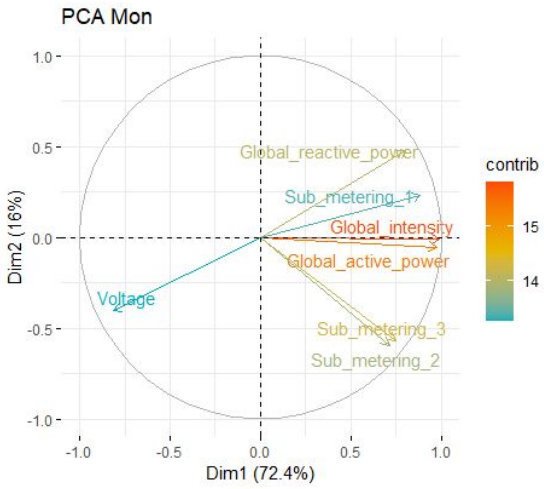
weekday Average_Global_Reactive_Power on Monday and Saturday where there is a sharp increase in both the values between 6:00 - 8:00 and after that there is a sharp decline in value on Saturday and Monday.Another, similar trend is in Average_Voltage which has a sharp decline between 4:30 to 7:30 and after that it slowly takes an upwards trend.

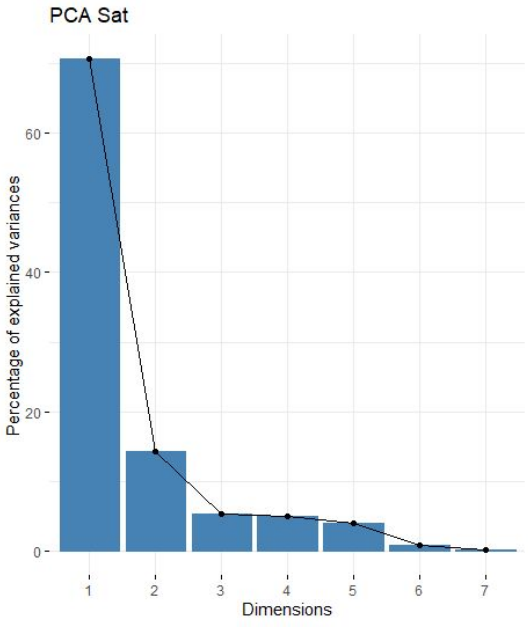
Average Sub_metering_2 on both the weekend and weekday is scattered across the time interval and doesn’t follow a particular trend however it follows the same trajectory on both the days.

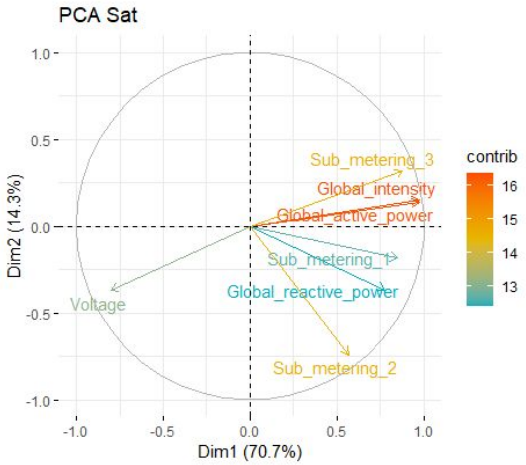
Feature Engineering

To further visualize the patterns in each feature, Principal component analysis generates the two graphs of the feature patterns on Saturday and Monday, as shown below.









Univariate Hidden Markov Model

In order to train an accurate and less complex model, we need to find the feature that shows a clear pattern. In other words, a clear pattern will have less noise, anomalies and extreme data points. Thus, models trained from such a feature will be more accurate and not prone to overfitting. From the graphs of average of each feature, Global_active_power on Saturday and Monday are clear in their patterns. From the correlation matrices, Global_active_power has a closer relationship with other features. In addition, from the interpretation of PCA values, the vector Global_active_power is close to the horizontal. Thus, Global_active_power is used to train the Univariate Hidden Markov model.

Multivariate Hidden Markov Model

To find the best combination of features, based on the correlation matrices, Voltage and Sub_metering_3 have distinct patterns from the main feature Globle_active_power. Global_reactive_power is also a good candidate as its pattern is clear. Based on the PCA vectors, the angle between Global_active_power and Global_intensity is small enough that one of them can present such patterns. The vector of Voltage is distinct from other vectors and, thus it is a good feature to be included in training the model. Sub_metering_2, Global_reactive_power and Sub_metering_3 are candidates from the

PCA graphs. However, Sub_metering_2 and Sub_metering_3 do not form an ideal pattern.

The best combination under current stage would be Globle_reactive_power, Globle_active_power and Voltage. However, this choice is subject to adjustment as the models of different combinations are actually trained. Furthermore, feature selection depends upon the dataset and since it also depends upon the model so by selecting these three features suited well on our model however it may not be the best choice in different scenarios.

Training and Testing

According to the suitable variable we get from feature engineering. We would like to train and test to find the best state of each combination. From above, we know Global_active_power, Voltage, Global_reactive_power and Global_intensity are suitable variables. Because Global_active_power has a closer relationship with other features. We set Global_active_power as a dependent variable. Combine saturday and monday data. And to make it more efficient, we select one hour of the total data and partition it into a training set and a testing set. During the training part, we would train and test HMM models from state equals 4 to state equals 20 to find the best state for each model.For univariate models, the dependent variable **(Global_reactive_power)** is used to train.

For Multivariate model:

Two combination: **(Voltage& Global_active_power)** , **(Global_reactive_power & Global_active_power)** and **(Global_intensity & Global_active_power)**

Three combinations: **(Global_intensity & Global_active_power & Voltage)** , **(Global_reactive_power & Global_active_power & Global_intensity)** and **(Global_reactive_power & Global_active_power & Voltage).**

Training part:

1. **univariate model (Global_active_power)**

State	4	5	6	7	8	9	10	11
BIC	2998.632	2873.146	2761.715	2794.459	2684.228	2460.186	2752.167	2646.481
Loglik	-1408.2268	-1301.9194	-1194.7185	-1151.6843	-1029.2421	-841.9732	-904.7955	-760.8632
12	13	14	15	16	17	18	19	20
2639.977	3085.857	2957.190	2941.125	3181.907	3359.529	3413.837	3609.783	4074.868
-658.6009	-774.6101	-595.4247	-464.6195	-454.3171	-404.5140	-285.1333	-228.6502	-298.8162

After training the univariate HMM with Global_active_power variable. BIC has a Steady decline from state equals 4 to state equals 12, then changes to increase between state equals 13 to state equals 20. However, Loglik rises continuously between 4 to 20.

BIC and Loglike are closest at state equals 19 and that is the best state in the Univariate model.

2. **Multivariate model (Voltage & Global_active_power)**

State	4	5	6	7	8	9	10	11
BIC	16459.31	16020.47	15186.60	15360.99	14452.64	14309.29	13975.52	14029.65
Loglik	-8106.883	-7835.976	-7359.635	-7379.506	-6850.080	-6695.239	-6437.266	-6365.318
12	13	14	15	16	17	18	19	20
13987.30	14130.63	13878.13	13814.32	13818.98	13930.46	14196.35	14086.73	14388.62
-6237.213	-6194.024	-5945.002	-5782.403	-5646.122	-5555.328	-5533.815	-5316.630	-5297.276

3. **(Global_reactive_power & Global_active_power)**

State	4	5	6	7	8
BIC	-1329.183	-1943.806	-2229.540	0.000	0.000
Loglik	787.3641	1146.1607	1348.4337	0.0000	0.0000

This combination would not be considered as BIC and Loglike have the weird sign.It should be caused by overfitting.

4. **(Global_intensity & Global_active_power)**

state	4	5	6	7	8	9	10	11
BIC	14346.24	13160.84	12667.77	12042.75	11640.76	11396.16	11381.48	11036.07
Loglik	-7050.346	-6406.161	-6100.222	-5720.386	-5444.141	-5238.675	-5140.245	-4868.526
12	13	14	15	16	17	18	19	20
11086.36	10870.56	10839.60	10773.75	11197.41	11138.78	11484.05	11543.38	11450.80
-4786.743	-4563.991	-4425.738	-4262.121	-4335.334	-4159.486	-4177.663	-4044.955	-3828.365

After training the two combinations of multivariate HMM, we find both BIC and Loglike from two HMMs are in the constant decreasing trend.As the slope closes to 0 to both BIC and Loglik, it cannot improve too much after state=20. So that the best state should be the largest state. And the combination **(Global_intensity & Global_active_power)** performs better than the other combination **(Voltage & Global_active_power).** Thus the best two combinations multivariate HMM is **(Global_intensity & Global_active_power)** when state = 20.

5. **(Global_intensity & Global_active_power & Voltage)**

state	4	5	6	7	8	9	10	11
BIC	28104.44	26810.26	26364.93	26107.85	25543.24	24838.29	24541.93	24626.75
Loglik	-13897.77	-13191.27	-12901.28	-12697.49	-12332.01	-11888.45	-11641.26	-11576.74
12	13	14	15	16	17	18	19	20

24000.44	23790.82	24001.75	23946.07	23936.42	24067.38	24265.44	24547.23	24359.61
-11148.73	-10921.15	-10895.92	-10729.47	-10578.11	-10489.13	-10425.78	-10396.38	-10124.35

6. **(Global_reactive_power & Global_active_power & Global_intensity)**

state	4	5	6	7	8	9	10	11
BIC	10431.743	9566.534	8817.533	7696.207	7131.043	6652.238	6404.647	6269.882
Loglik	-5061.416	-4569.405	-4127.578	-3491.667	-3125.917	-2795.425	-2572.619	-2398.306

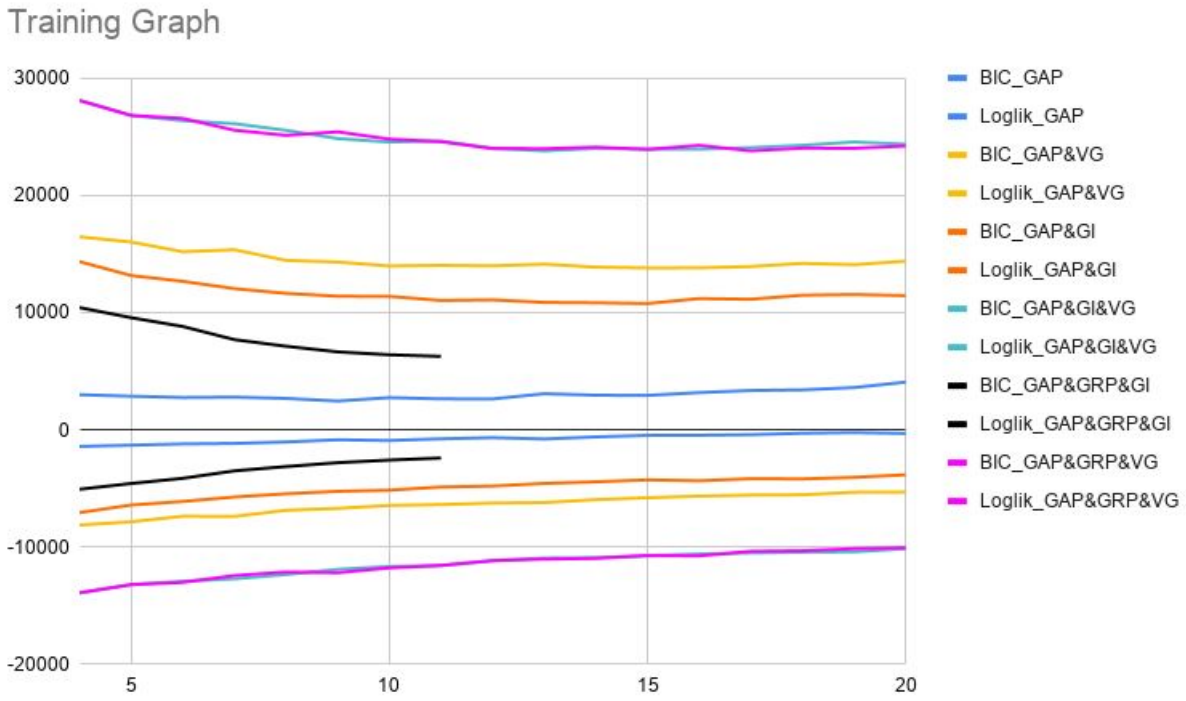
Training the three combinations multivariate HMM, the combination **(Global_reactive_power & Global_active_power & Global_intensity)** would get NAN and Inf after state = 11, so that we just train it from state = 4 to state = 11. Although it is just training until at state = 11, it gets the best performance than others. Thus the best three combinations multivariate HMM is**(Global_reactive_power & Global_active_power & Global_intensity)** and the best state is equal to 11.

7. **(Global_reactive_power & Global_active_power & Voltage)**

state	4	5	6	7	8	9	10	11
BIC	28083.85	26810.26	26556.54	25551.71	25113.78	25411.34	24792.38	24573.57
Loglik	-13887.47	-13191.27	-12997.08	-12419.42	-12117.29	-12174.98	-11766.49	-11550.15
12	13	14	15	16	17	18	19	20

23993.67	23974.86	24104.57	23914.36	24259.30	23794.66	24021.57	24006.71	24209.73
-11145.35	-11013.17	-10947.33	-10713.61	-10739.55	-10352.77	-10303.85	-10126.12	-10049.42

Training the HMM on **(Global_reactive_power & Global_active_power & Voltage)** in this model the best possible BIC and Loglik is at state 17 which is 23794.66 and -102020.85 respectively however this values are not as good as in the previous model.



Testing Part:

Now we would like to test univariate and multivariate HMM with the best combinations and state.

- For univariate model get best model when state = 19

BIC: 4490.905 & Loglike : -669.2112 (df=398)

- For multivariate model the best state is state = 20 with two combination **(Global_intensity & Global_active_power)**

  BIC: 12633.04 & Loglike : -4419.487 (df=479)

- For multivariate model the best state is state equals to 11 with three combination **(Global_reactive_power & Global_active_power & Global_intensity)**

  BIC: 6542.291 & Loglike : -2534.51 (df=186)

The best univariate model

Because we choose Global_active_power as the suitable dependent variable, so that the best univariate model should use Global_active_power as response argument and set state at 19.

The best multivariate model:

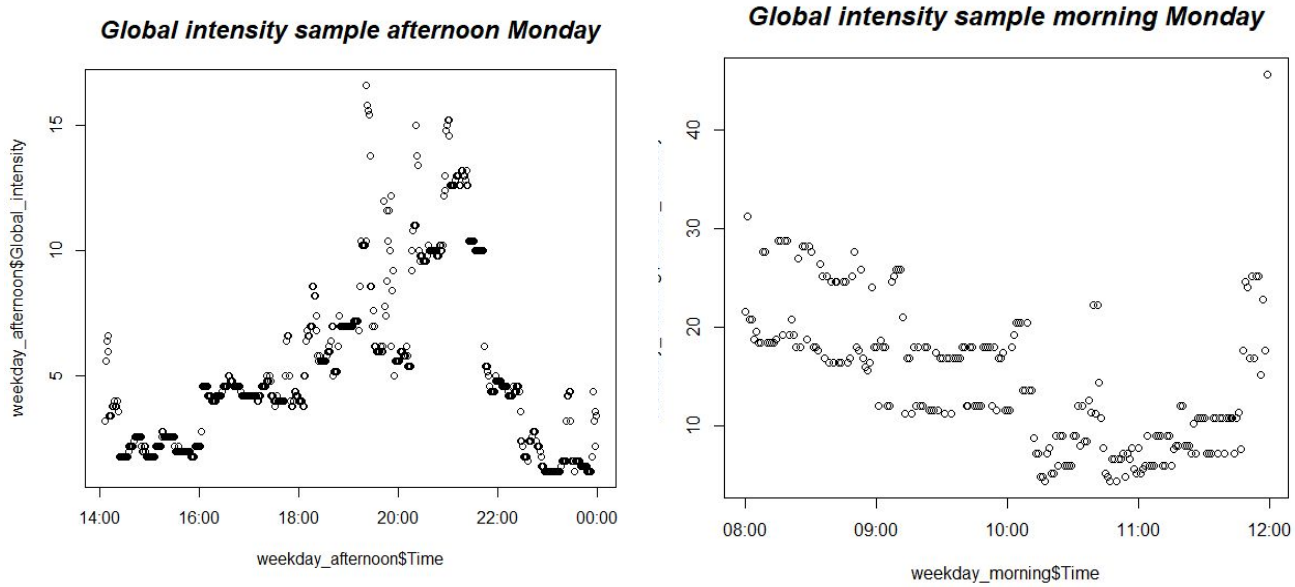
For multivariate models, there are two possible combinations (two variables and three variables). By comparing both testing results, we would suggest choosing **(Global_reactive_power & Global_active_power &Global_intensity)** as response variables and set state = 11 to build the multivariate Hidden Markov Model.

For multivariate models with two parameters the best set of combinations is **(Global_intensity & Global_active_power)** when state = 20.

Anomaly detect

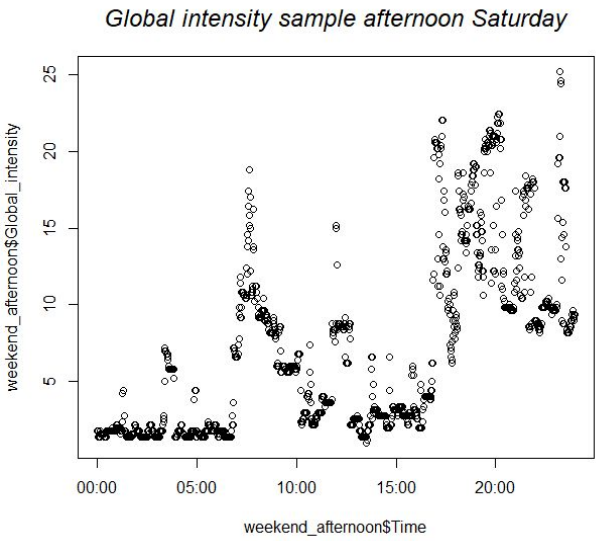
- **Choose the time window for weekdays**

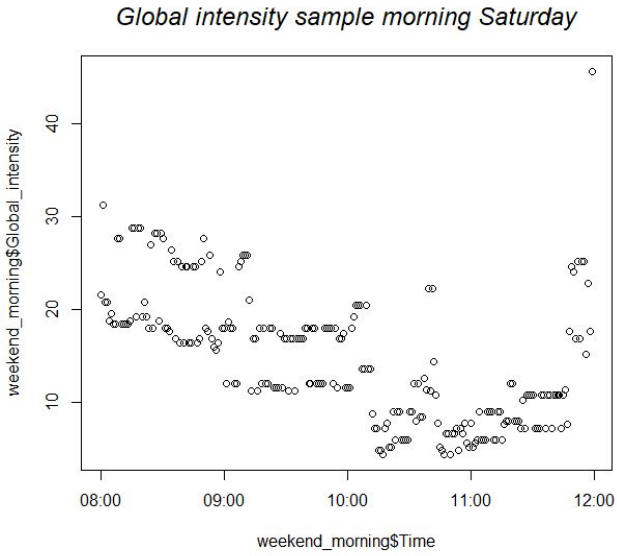
The time window of the weekday (Monday) is from “14:07:00” to “23:59:59”, according to the graph below, the graph “Global intensity sample afternoon Monday” has more variants than “Global intensity sample morning Monday”, which means the time window of “14:07:00” to “23:59:59” has more reference value.



- **Choose the time window for weekend day**

The time window of the weekend day (Saturday) is from “14:07:00” to “23:59:59”, according to the graph below, the graph “Global intensity sample afternoon Saturday” has more variants than “Global intensity sample morning Saturday”, which means the time window of “14:07:00” to “23:59:59” has more reference value.





So the time window will be "14:07:00" to "23:59:59" for both weekday and weekdays.

Monday

Anomalies: Global_reactive_power ---------- **threshold**	text1	text2	text3	text4	text5
0.1	86	86	90	267	264
0.2	7	7	9	148	135
0.3	1	1	1	75	72

Anomalies: Global_active_power ---------- **threshold**	text1	text2	text3	text4	text5
0.5	327	317	332	487	497
1	139	147	146	399	400
2	24	24	26	234	237

Anomalies: Voltage ---------- **threshold**	text1	text2	text3	text4	text5
1	146	146	202	583	586
2	36	36	98	580	586
3	4	4	65	580	586

Anomalies: Goblal_intensity ---------- **threshold**	text1	text2	text3	text4	text5
0.25	298	298	312	562	568
0.5	182	182	189	529	548
0.75	145	145	148	490	524

Anomalies: Sub_metering_1 ---------- **threshold**	text1	text2	text3	text4	text5
25	n/a	n/a	n/a	n/a	n/a
50	n/a	n/a	n/a	n/a	n/a
75	n/a	n/a	n/a	n/a	n/a

Anomalies: Sub_metering_2 --- **threshold**	text1	text2	text3	text4	text5
25	n/a	n/a	n/a	n/a	n/a
50	n/a	n/a	n/a	n/a	n/a
75	n/a	n/a	n/a	n/a	n/a

Anomalies: Sub_metering_3 --- **threshold**	text1	text2	text3	text4	text5
25	n/a	n/a	n/a	n/a	n/a
50	n/a	n/a	n/a	n/a	n/a
75	n/a	n/a	n/a	n/a	n/a

Saturday

Anomalies: Global_reactive_power --- **threshold**	text1	text2	text3	text4	text5
0.1	214	214	214	769	727
0.2	26	26	26	426	390
0.3	2	2	2	210	202

Anomalies: Global_active_power ---------- **threshold**	text1	text2	text3	text4	text5
0.5	708	688	708	1076	1108
1	349	330	349	871	897
2	91	74	91	558	549

Anomalies: Voltage ---------- **threshold**	text1	text2	text3	text4	text5
1	359	359	359	1303	1304
2	59	59	59	1300	1302
3	10	10	10	1299	1300

Anomalies: Goblal_intensity ---------- **threshold**	text1	text2	text3	text4	text5
0.25	746	746	746	1270	1263
0.5	525	525	525	1208	1199
0.75	423	423	423	1137	1109

Anomalies: Sub_metering_1 ---------- **threshold**	text1	text2	text3	text4	text5
25	n/a	n/a	n/a	n/a	n/a
50	n/a	n/a	n/a	n/a	n/a
75	n/a	n/a	n/a	n/a	n/a

Anomalies: Sub_metering_2 --- **threshold**	text1	text2	text3	text4	text5
25	n/a	n/a	n/a	n/a	n/a
50	n/a	n/a	n/a	n/a	n/a
75	n/a	n/a	n/a	n/a	n/a

Anomalies: Sub_metering_3 --- **threshold**	text1	text2	text3	text4	text5
25	n/a	n/a	n/a	n/a	n/a
50	n/a	n/a	n/a	n/a	n/a
75	n/a	n/a	n/a	n/a	n/a

Anomaly Detect different approach with z-score

In the anomaly detect part, we also use Z-Score to choose the threshold. This is a much more appropriate way to find anomaly points, since the datasets follow the normal distribution. The chosen weekday and weekend are Monday and Sunday, the time window is 20:30 to 24:00. The number of anomaly points in each variable is shown below.

Anomalies: Global_reactive_power ---------- **Z-Score**	text1	text2	text3	text4	text5
1	10125	10125	10125	9524	9517
2	3176	3176	3176	2462	2443
3	989	989	989	969	969

Anomalies: Global_active_power ---------- **Z-Score**	text1	text2	text3	text4	text5
1	5655	5660	5655	5669	5717
2	1035	1032	1035	1007	1024
3	0	0	0	0	0

Anomalies: Voltage ---------- **Z-Score**	text1	text2	text3	text4	text5
1	10021	10021	10021	5377	5373
2	1295	1295	1295	35	48
3	0	0	0	0	0

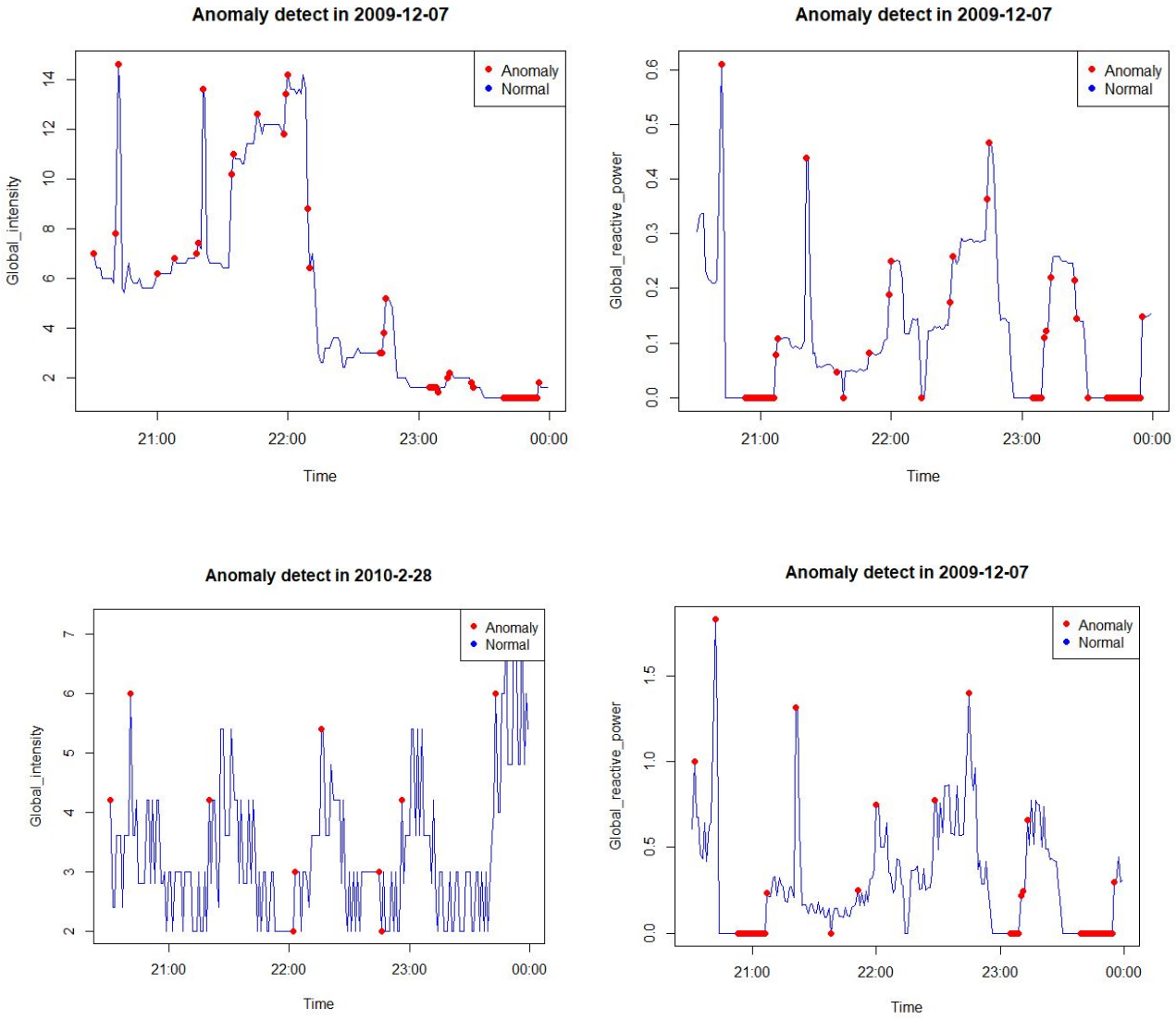
Anomalies: Goblal_intensity ---------- **Z-Score**	text1	text2	text3	text4	text5
1	10540	10540	10540	7980	8059
2	3630	3630	3630	764	773
3	1379	1379	1379	3	0

Anomalies: Sub_metering_1 ---------- **Z-Score**	text1	text2	text3	text4	text5
1	1573	1573	1573	1691	1656
2	410	410	410	432	437
3	0	0	0	0	0

Anomalies: Sub_metering_2 --- **Z-Score**	text1	text2	text3	text4	text5
1	2415	2415	2415	2338	2395
2	312	312	312	483	478
3	0	0	0	0	0

Anomalies: Sub_metering_3 --- **Z-Score**	text1	text2	text3	text4	text5
1	6772	6772	6772	8512	8452
2	694	694	694	537	521
3	0	0	0	0	0

The plot shown below is the anomaly points that are detected by the algorithm with the z-score equals to 2 in a single time window. The red points represented the anomaly points.



HMM on anomaly detect

Best multivariate (nstate = 11)

	text1	text2	text3	text4	text5
BIC	2267.4069	2189.6665	2189.6665	2189.6665	8290.3669
LogLike	-539.9	-501	-501	-501	-3551

Best univariate (nstate = 19)

	text1	text2	text3	text4	text5
BIC	1175.7519	162.9302	4226.5751	5830.7995	5862.2240
LogLike	813.7	1320	-711.7	-1514	-1530

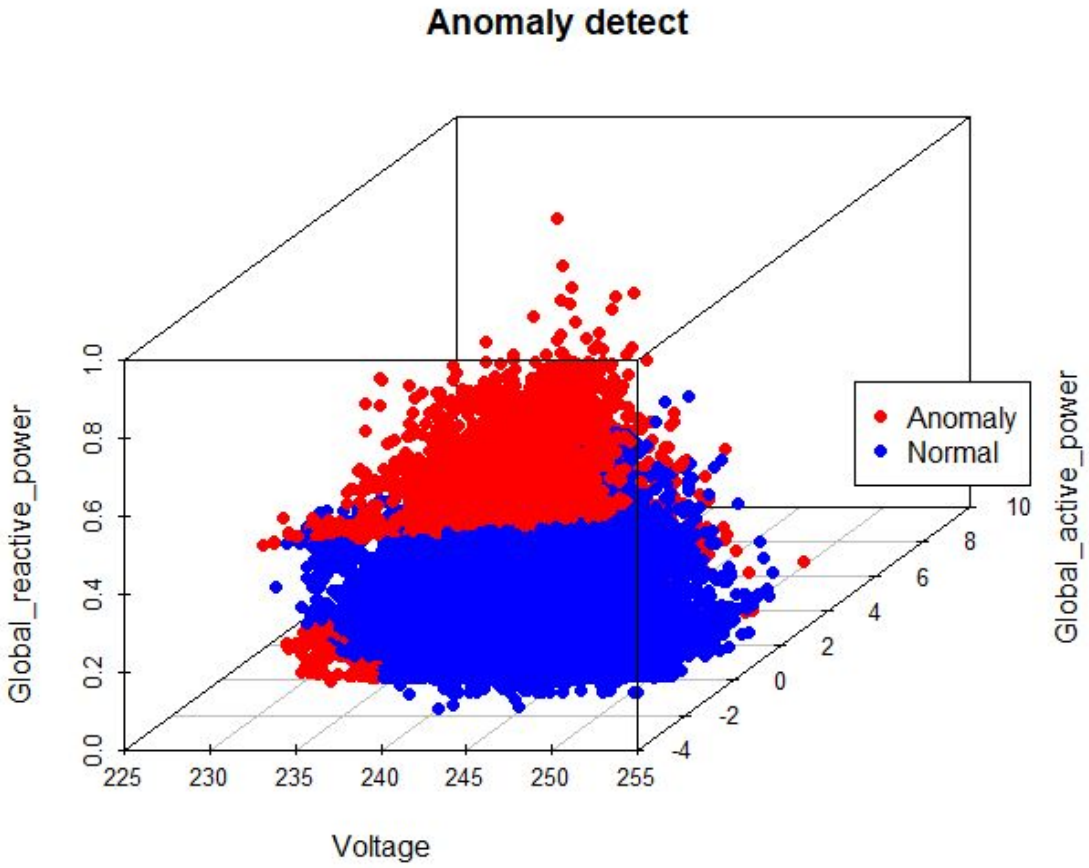
Bonus question: SVM

In this part, the one class svm classification model was generated by the function in e1071 library. The dataset train.txt and test3.txt are combined together. The time period of the dataset for the one class svm to train is 5 years, and the time window for the dataset is every monday and sunday in each week from 20:30 to 24:00.

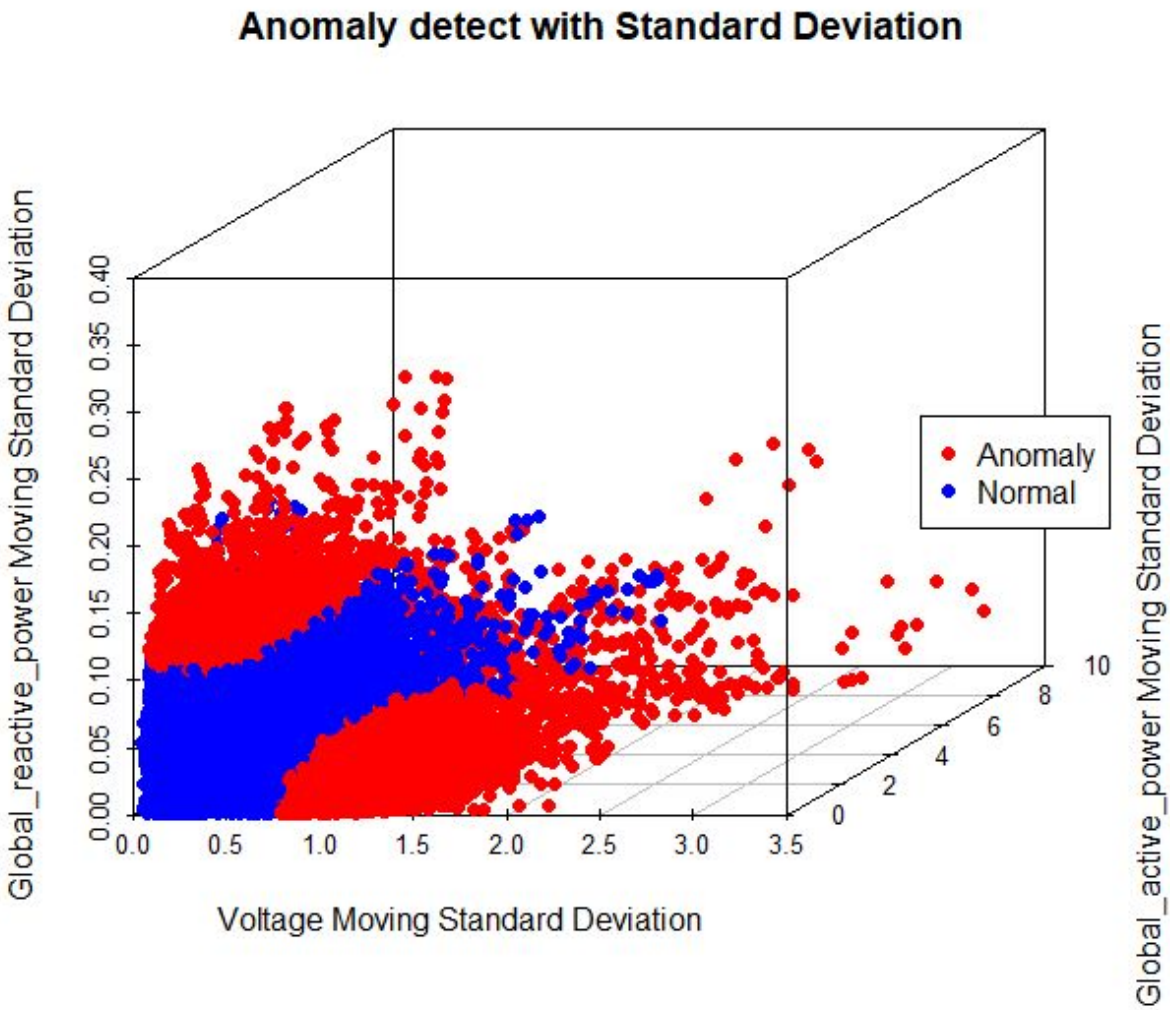
First, the chosen combination of variables is the same as the combination used in hmm **(Gobal_reactive_power, Voltage, Gobal_active_power).** The polynomial kernel was chosen for the classifier, since it is a 3 dimensional dataset for the classifier to train, and the nu parameter is 0.09.

The 3D plot result is shown below

(red dots represent anomaly points, Blue dots represent normal behavior points) from the plot polynomial kernel can classify the anomaly and normal value except some outliers.



Second, in order to detect other anomaly points that can not be detected by the previous method. The new method was needed. So, we created a new feature for the dataset, that is the moving standard deviation for each variable in the dataset. The one class SVM model was trained the same as the previous one. The result is shown below.



References

1. Tutorial for Data Analysis in R [Online]: https://www.analyticsvidhya.com/blog/2016/02/complete-tutorial-learn-data-science-scratch/
2. Hidden Markov Model by Stanford: https://web.stanford.edu/~jurafsky/slp3/A.pdf
3. Anomaly detection using R [Online]: https://towardsdatascience.com/tidy-anomaly-detection-using-r-82a0c776d523
4. Principal Component Analysis: https://towardsdatascience.com/a-one-stop-shop-for-principal-component-analysis-5582fb7e0a9c

4. Z-score [Online]: https://www.investopedia.com/terms/z/zscore.asp