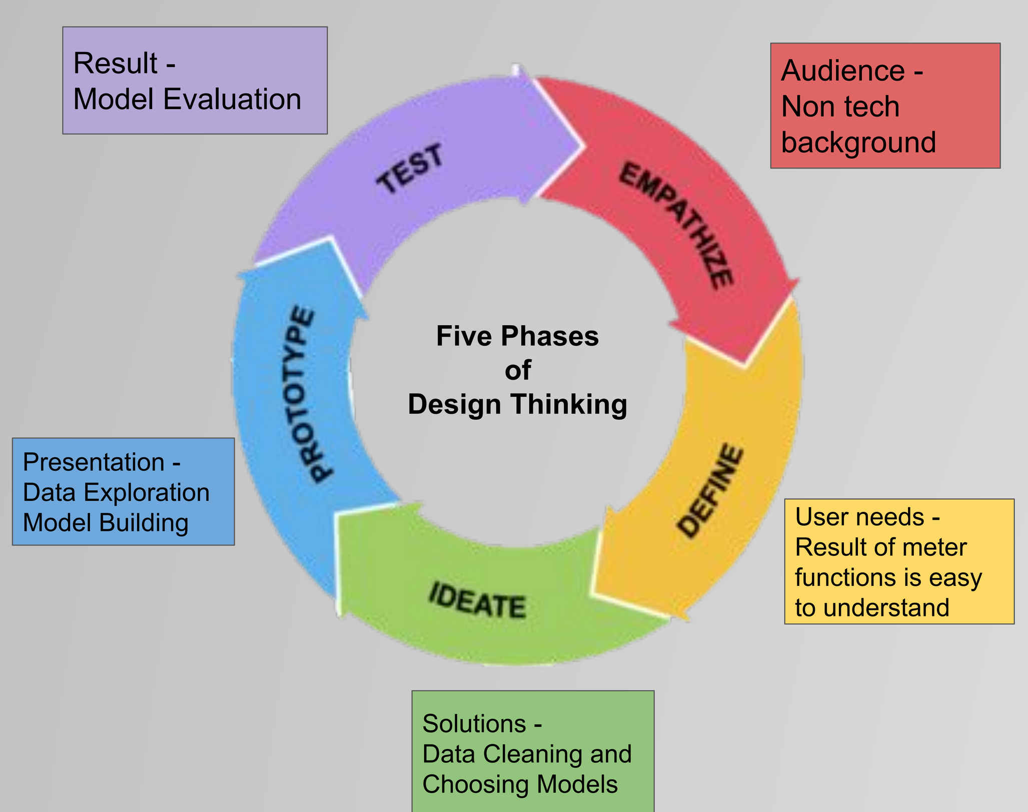
Libraries Used:

- Pandas
- Numpy
- Matplotlib
- Seaborn
- Sklearn
- Datetime
- dmba

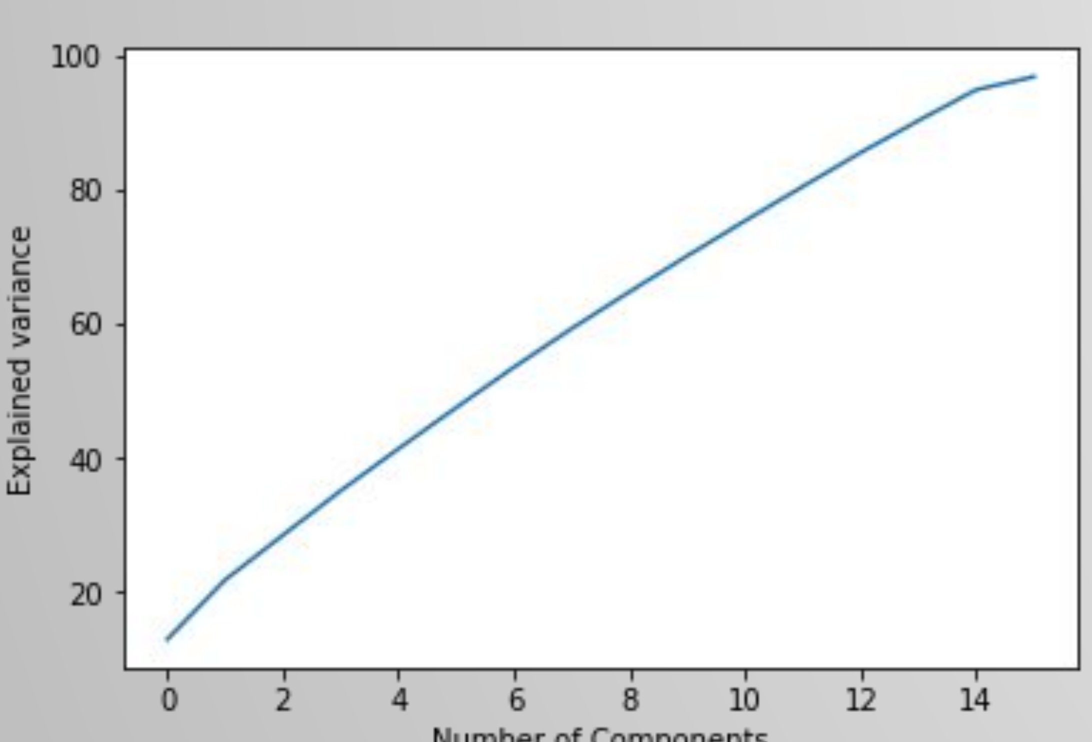
Analyze challenge with Design Thinking



Dimensionality Reduction Techniques Used:

- Principal Component Analysis
- Linear Discriminant Analysis

PCA



Analysis provided us with 16 components that explained 95% of the variance.

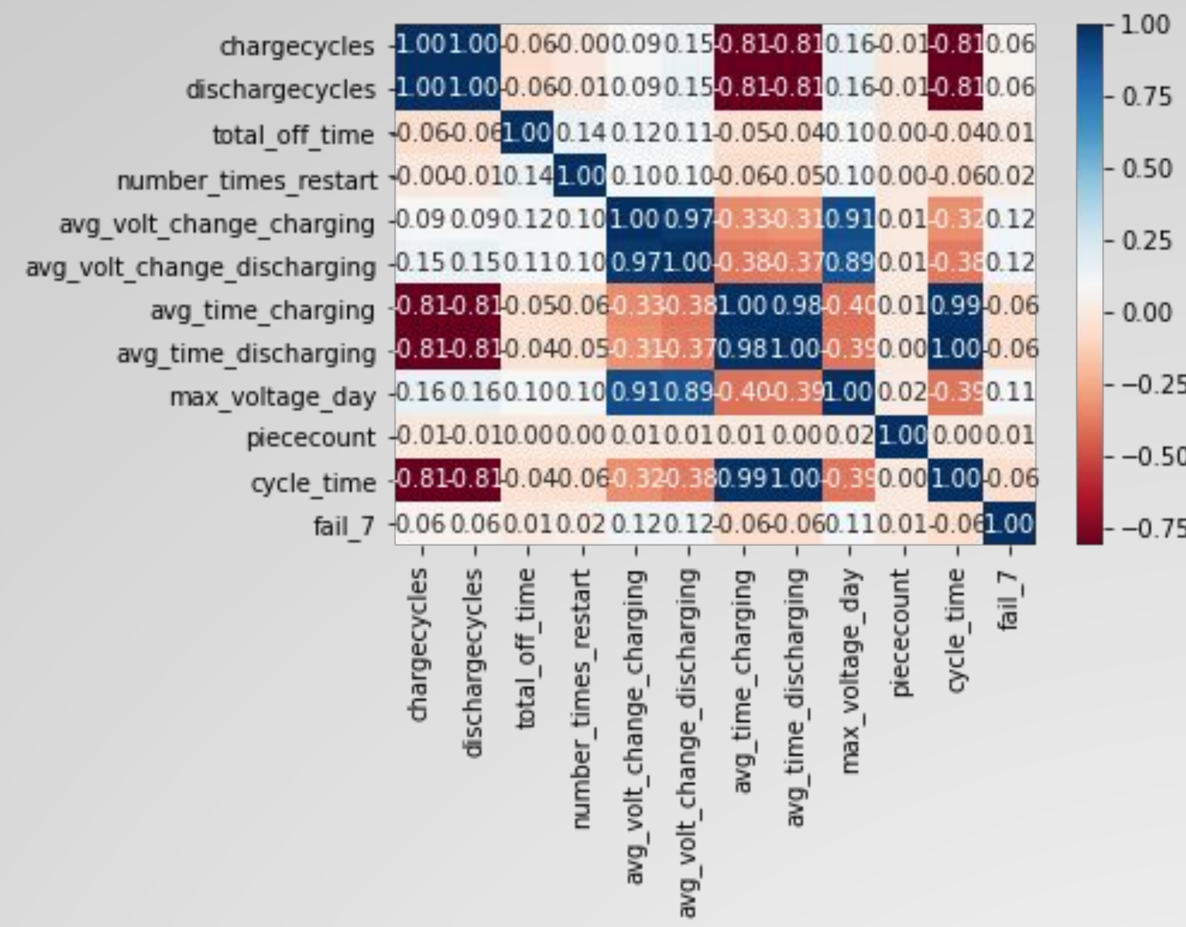
LDA

```
[[5149    0]
 [1618    0]]
Accuracy 0.7608984779074922
```

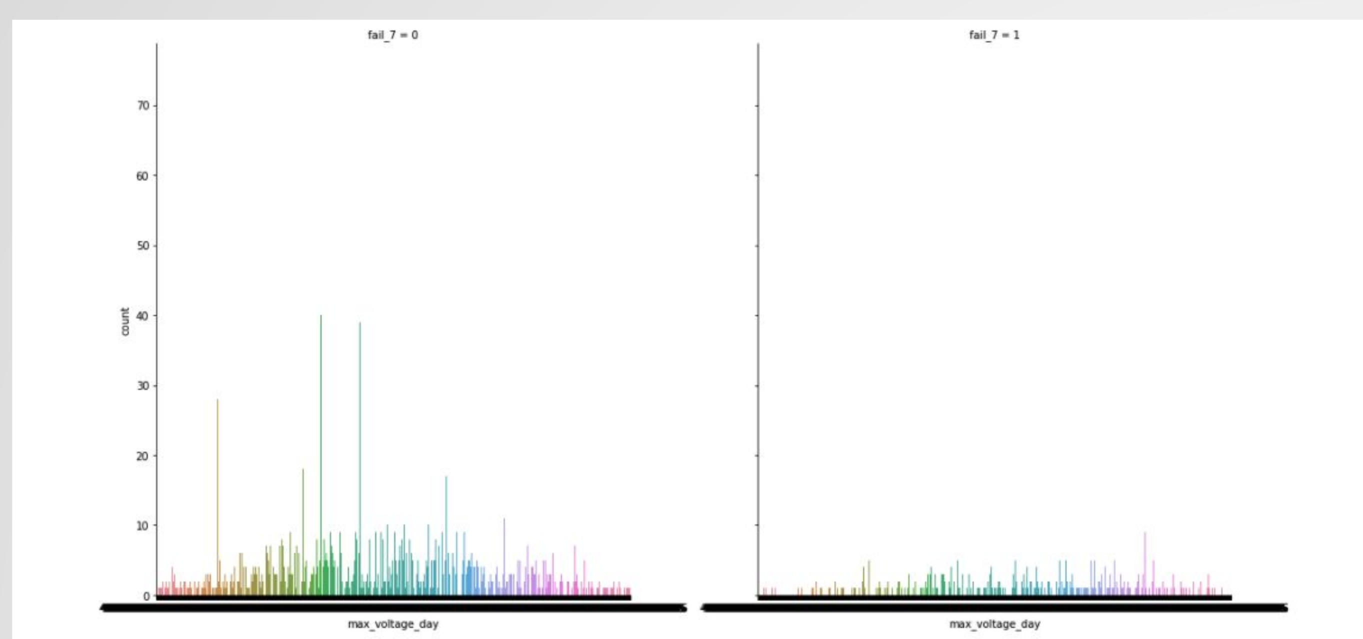
- Ran LDA using one linear discriminant
- Gives us ~76% accuracy
- Chose to go with PCA for our dimensionality reduction

Data Exploration

Correlation Matrix

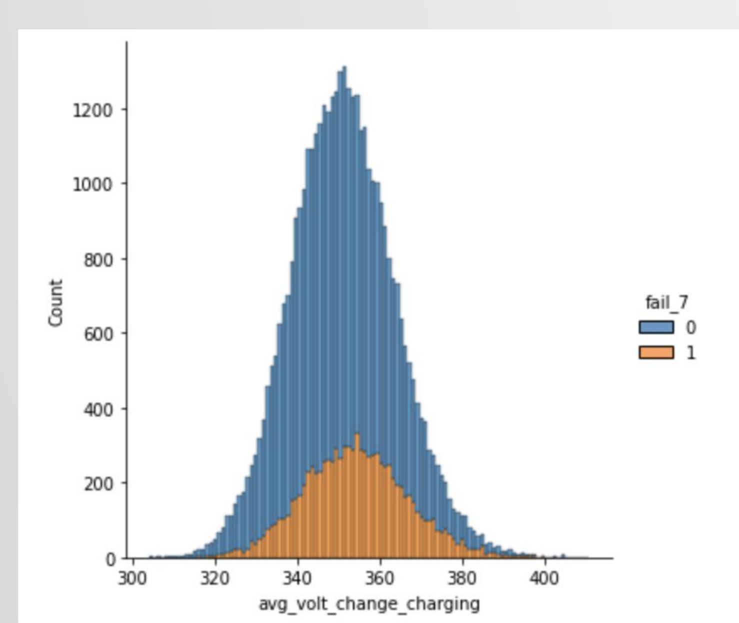


Max_Voltage_Day Distributions among Working and Failed Meters



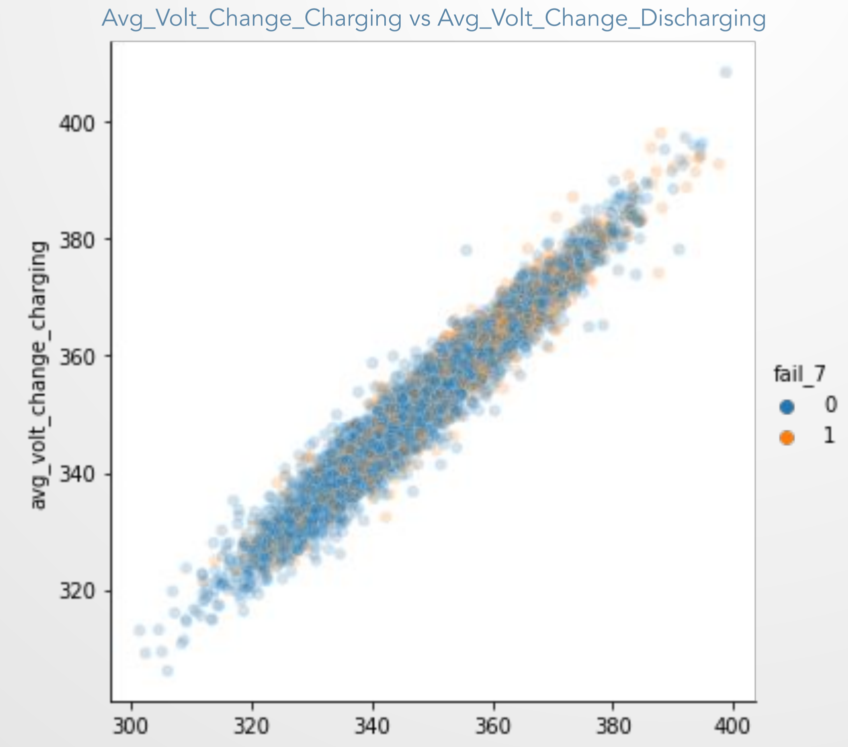
By looking at the correlation matrix we noticed 'max_voltage_day' also had a relatively high correlation with the 'fail_7' column and wanted to take a look at the distributions of 'max_voltage_day' day values across meters that did and didn't fail to see if there were and significant differences in the distributions. From the image above we noticed that the meters that failed have comparatively higher counts in the larger max_voltage_day values.

Stacked Distributions of 'avg_volt_change_charging'



Here we produced a stacked distribution of avg voltage charging broken down by working and failed meters where both distributions follow normal distribution centered around 350 volts.

Avg_Volt_Change_Charging vs Avg_Volt_Change_Discharging



For this plot we noticed that both the 'avg_volt_change_charging' and 'avg_volt_change_discharging' had relatively high correlation values to the fail_7 column in the correlation metric so we decided to create a scatter plot. Unfortunately the data was a little too dense to extract any meaningful insight so instead we plotted a sample of the data and noticed while relatively uniform it does seem that higher values in both axis correlate to a larger proportion of failed meters.

Summary

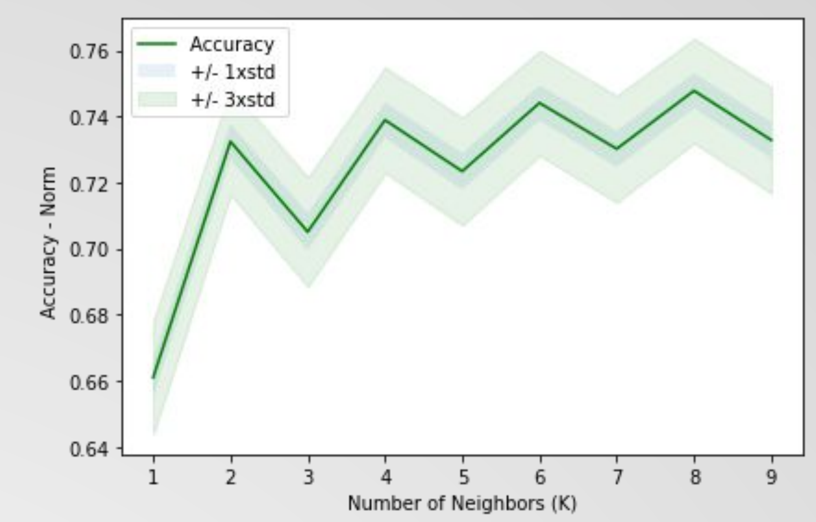
As a classification problem, we first looked at the accuracy scores where all our models returned values in the range from 0.73 to 0.77 except Logistic Regression model with normalized dataset, whose accuracy is only 0.38. The Logistic Regression fit with the PCA's has highest accuracy 0.7616.

Since none of the models were able to produce a significantly better accuracy score than the others, we then looked at the F1 scores to which we found our Logistic Regression model fit with our normalized dataset returning the largest value 0.42, but since it returned the lowest accuracy score we are not going to consider it as our choice. The Naive Bayes model fit with PCA's retuned strong precision and recall score compared to the other models. Finally, the Naive Bayes fit with PCA's returned the best F1 score of the remaining models still being considered. Therefore, the Naive Bayes fit with PCA's is our best fit model.

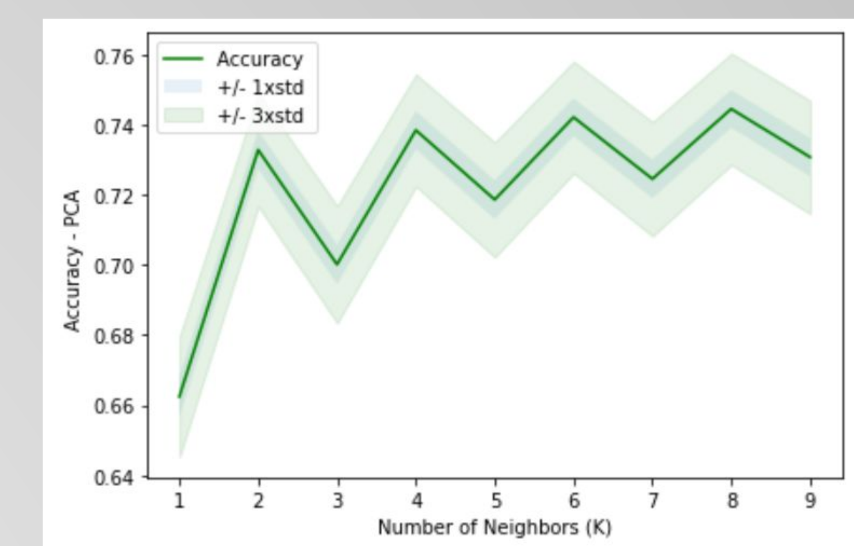
Model Building

- K-Nearest Neighbors
- Decision Tree
- Naive Bayes
- Logistic Regression

K-Nearest Neighbor Model



With normalized dataset, the best K is 8 with accuracy around 74.8%



With PCA dataset, the best K is 8 with accuracy around 74.4%

Logistic Regression

```
The intercept is -2.4001833212921158
                                 coefficient
avg_volt_change_charging            2.579951
discharging_rate_lag3               2.266364
charge_cycle_time_below_12          1.686392
Total_Days_in_Service               0.997456
avg_time_discharging                0.869621
avg_volt_change_discharging         0.761477
chargecycles                        0.720961
cycle_time                          0.675803
dischargecycles                     0.657580
avg_time_charging_lag7              0.541506
avg_time_charging_lag12             0.424249
avg_time_charging_lag13             0.356941
number_times_restart                0.329172
avg_time_charging_lag14             0.294871
avg_time_charging                   0.291521
avg_time_charging_lag3              0.243706
avg_time_charging_lag1              0.238242
avg_time_charging_lag5              0.223475
avg_time_discharging_lag4           0.147490
avg_time_charging_lag11             0.134028
discharging_rate_lag4               0.098695
avg_time_charging_lag9              0.098046
avg_time_discharging_lag10          0.079563
avg_time_discharging_lag8           0.059649
avg_time_charging_lag6              0.039104
avg_time_charging_lag2              0.031464
avg_time_discharging_lag9          -0.000864
avg_time_charging_lag8             -0.012977
charging_rate_lag3                 -0.050404
avg_time_charging_lag10            -0.074416
avg_time_discharging_lag2          -0.086722
avg_time_discharging_lag14         -0.092129
avg_time_discharging_lag11         -0.103632
avg_time_discharging_lag5          -0.122133
discharging_rate_lag6              -0.329819
discharging_rate_lag7              -0.346893
discharging_rate_lag5              -0.620780
max_voltage_day                    -0.777712
charging_rate_lag7                 -1.173917
charging_rate_lag4                 -1.205387
charging_rate_lag6                 -1.582351

Accuracy: 0.38208754248559187
precision score:  0.27274342572850035
recall score:  0.9487021013597033
f1-score:  0.4236820314656362
```

```
intercept  -1.3221137168156991
      coefficient
PC16     1.331101
PC2      0.304818
PC12     0.053244
PC15     0.039175
PC9      0.033756
PC10     0.011102
PC3      0.008691
PC8      0.007722
PC6     -0.012931
PC5     -0.015477
PC11    -0.022534
PC13    -0.040000
PC14    -0.061958
PC7     -0.080834
PC4     -0.081949
PC1     -1.670785

Accuracy: 0.7616373577656274
precision score:  0.5242718446601942
recall score:  0.03337453646477132
f1-score:  0.06275421266705404
```

Naive Bayes

Normalized Dataset

```
Confusion Matrix (Accuracy 0.6198)

       Prediction
Actual     0     1
     0 12105  8337
     1  1952  4671

Accuracy: 0.7608984779074922
Precision score: 0.3590867158671587
Recall score: 0.7052695153253813
F1 score: 0.4758799857368448
```

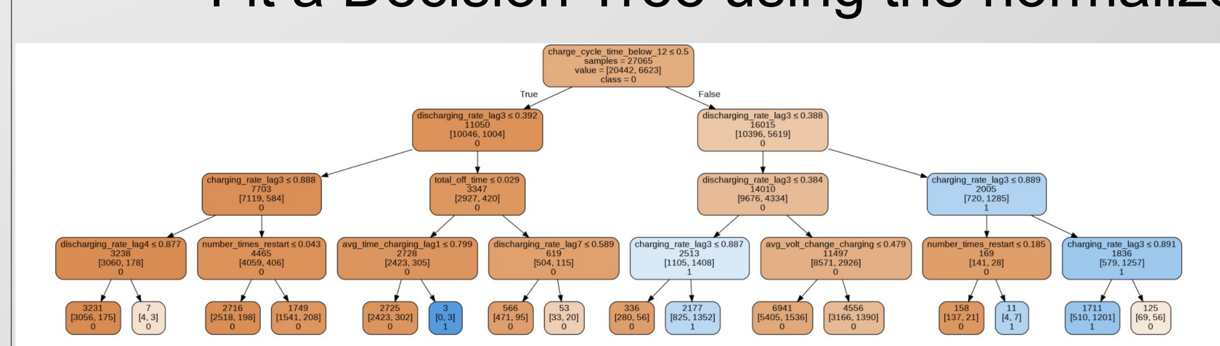
PCA

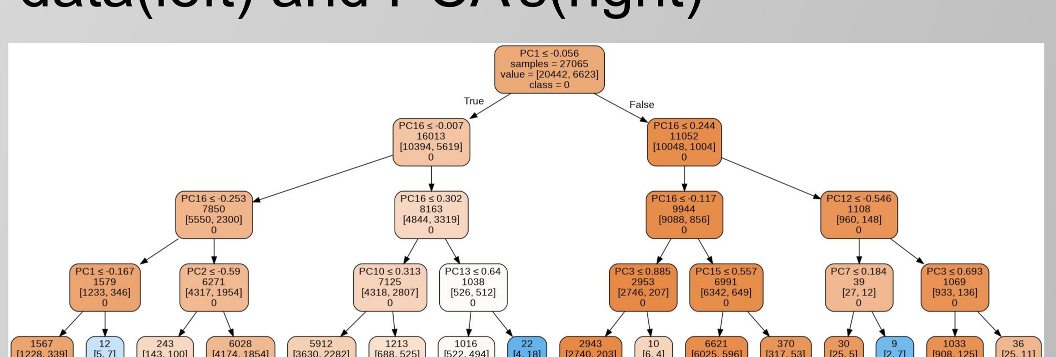
```
Confusion Matrix (Accuracy 0.7546)

       Prediction
Actual     0     1
     0 20054   388
     1  6253   370

Accuracy: 0.7591251662479681
Precision score: 0.48192771084337 35
Recall score: 0.09888751545117429
F1 score: 0.1641025641025641
```

Fit a Decision Tree using the normalized data(left) and PCA's(right)





Summary Table for Evaluation of Models

Model		KNN with K = 8		Decision Tree		Logistic Regression		Naïve Bayes	
Metric	Dataset	Normalized Dataset	PCA	Normalized Dataset	PCA	Normalized Dataset	PCA	Normalized Dataset	PCA
	Accuracy_Score	0.7477	0.7328	0.7544	0.7603	0.3829	0.7616	0.7609	0.7591
	Precision Score	0.4000	0.3304	0.3750	0.4474	0.2727	0.5243	0.3591	0.4819
	Recall Score	0.1100	0.1143	0.0408	0.0105	0.9487	0.0334	0.7053	0.0989
	F1-score	0.1726	0.1699	0.0736	0.0205	0.4237	0.0628	0.4759	0.1641