Surya Keswani
May 17, 2020

# Prediction and Analysis of Customer Churn

## I. Introduction

The Customer Churn dataset, included as an excel file in the provided code directory, contains 20,000 examples, each with 12 attributes, describing features of customers of a mobile phone provider. Customer churn signifies which customers have stopped using one company's product or service during a certain time frame. The file's fields include data on each customer such as their education, income, the value of their house, their monthly usage situation (like the number of over 15 mins calls, average call duration), and whether they churned within a month when the data was captured. The data was uniformly sampled from the full database. The goal of the data set is to predict the class variable LEAVE representing whether each customer decided to quit the company or not. The class variable, LEAVE, is the last variable on each line, and its legal values are LEAVE and STAY. Below is a list of each of the 12 attributes included in the dataset.

*College*: Is the customer college-educated?
*Income*: Annual income
*Overage*: Average overcharges per month
*Leftover*: Average % leftover minutes per month
*House*: Value of dwelling (from census tract)
*Handset price*: Cost of phone
*Over 15 min calls per month*: Average number of long (>15 mins) calls per month
*Average call duration*: Average call duration
*Reported satisfaction*: Reported level of satisfaction
*Reported usage level*: Self-reported usage level
*Considering change of plan*: Was the customer considering changing his/her plan?
*Leave*: Class variable (whether customer left or stayed)

## II. Processing the Data

Before the data could be analyzed or interpreted, the raw data had to be numerically encoded. For example, the attribute house is already numerically encoded as the price of the house of each customer. Attributes such as reported usage level were reported as very little, little, avg, high, or very high. Below are tables that show how each piece of raw data was converted to a numerical value. Any attributes that were already encoded as numeric values were left in their raw form.

College

| Raw Data | Converted Data used for Analysis |
|---|---|
| “zero” | 0 |
| “one” | 1 |

Reported Satisfaction

| Raw Data | Converted Data used for Analysis |
|---|---|
| “very_sat” | 5 |
| “sat” | 4 |
| “avg” | 3 |
| “unsat” | 2 |
| “very_unsat” | 1 |

Reported Usage Level

| Raw Data | Converted Data used for Analysis |
|---|---|
| “very_high” | 5 |
| “high” | 4 |
| “avg” | 3 |
| “little” | 2 |
| “very_little” | 1 |

Considering Change of Plan

| Raw Data | Converted Data used for Analysis |
|---|---|
| “actively_looking_into_it” | 5 |
| “considering” | 4 |
| “perhaps” | 3 |
| “no” | 2 |
| “never_thought” | 1 |

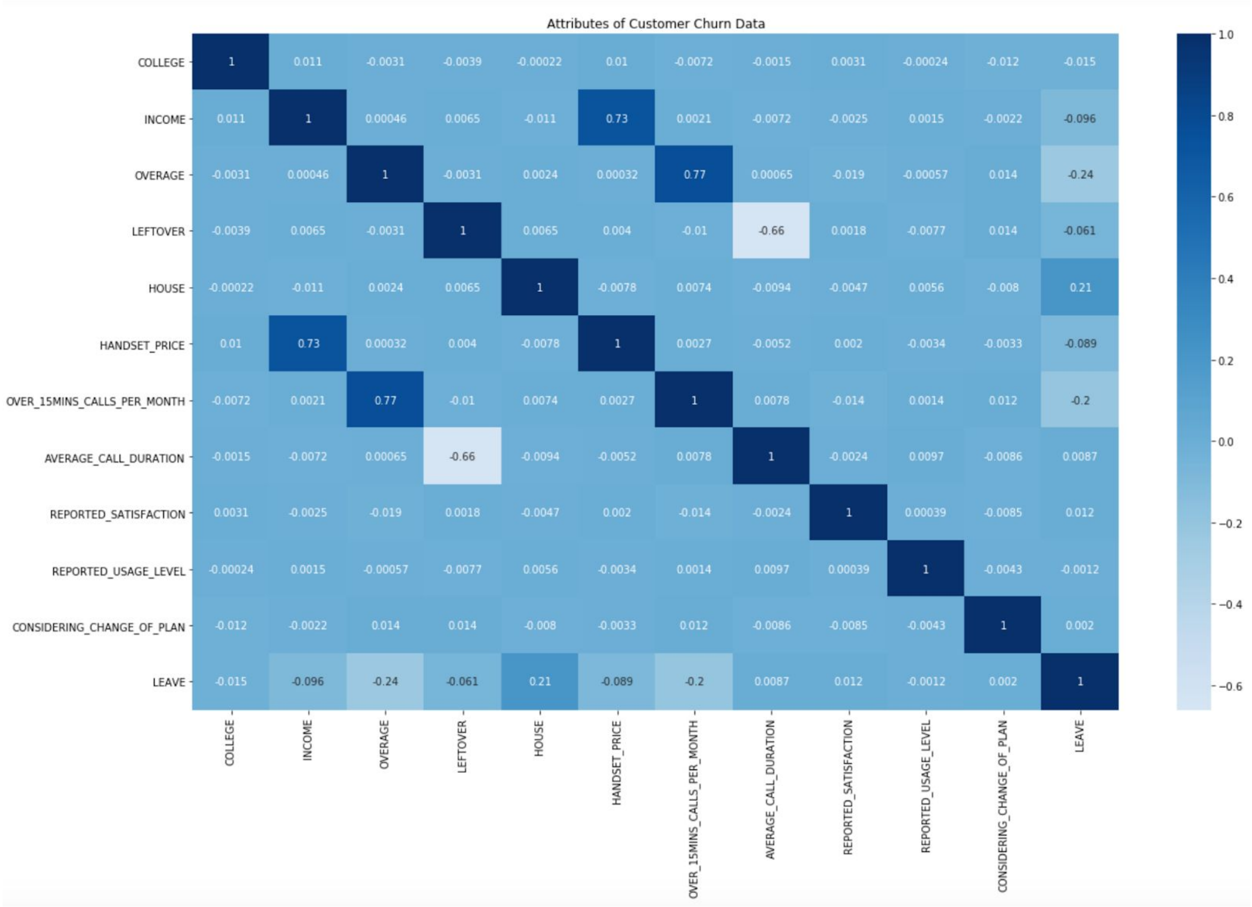
Leave

| Raw Data | Converted Data used for Analysis |
|---|---|
| “LEAVE” | 0 |
| “STAY” | 1 |

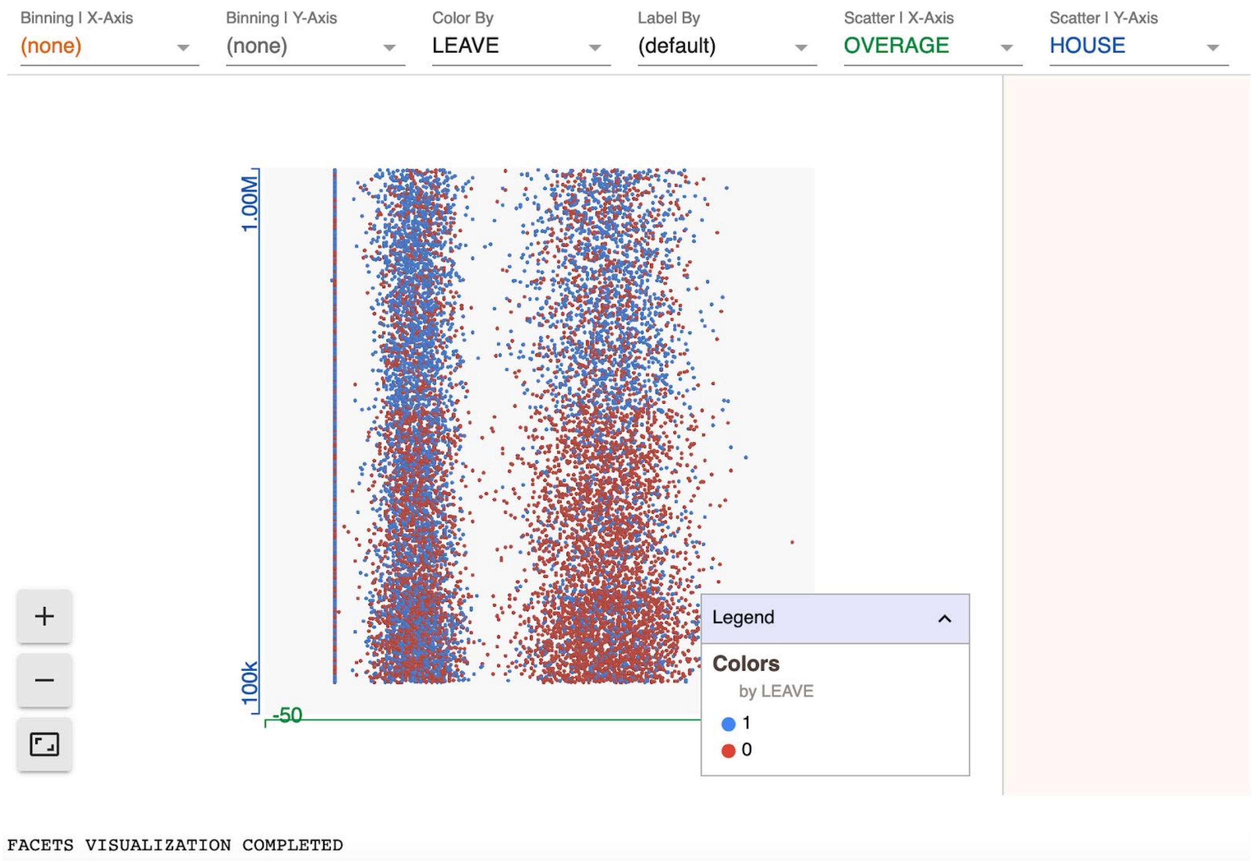
## III. Understanding the Data

A few data visualizations have been created to get a better understanding and some preliminary insights into the dataset.

The first visualization, depicted below, is a heatmap of each of the customer churn attributes mapped to one another. The closer the value to one, the more positively correlated the attributes are. The closer the value to -1, the more negatively correlated the attributes are. The heat maps reveal that almost all of the attributes are not strongly correlated in any way at all. The strongest correlation is between the attribute for over 15-minute calls and overage. Intuitively this makes sense. The more minutes used on a cell phone plan  the more the plan will cost. Income and handset prices have a strong correlation as well. Average call duration and leftover minutes are strongly inversely correlated. What I find more interesting are the correlations missing from the heatmap. It seems that reported satisfaction and reported change of plan have almost no effect on customer churn.



In addition to the heatmap, an interactive data visualization tool was implemented in the code directory. Google's Facets allows the user to visualize the data in many ways to better understand the data and look for patterns. An example of a Google Facets visualization is shown below.



Binning I X-Axis
(none)

Binning I Y-Axis
(none)

Color By
LEAVE

Label By
(default)

Scatter I X-Axis
OVERAGE

Scatter I Y-Axis
HOUSE

```
FACETS VISUALIZATION COMPLETED
```

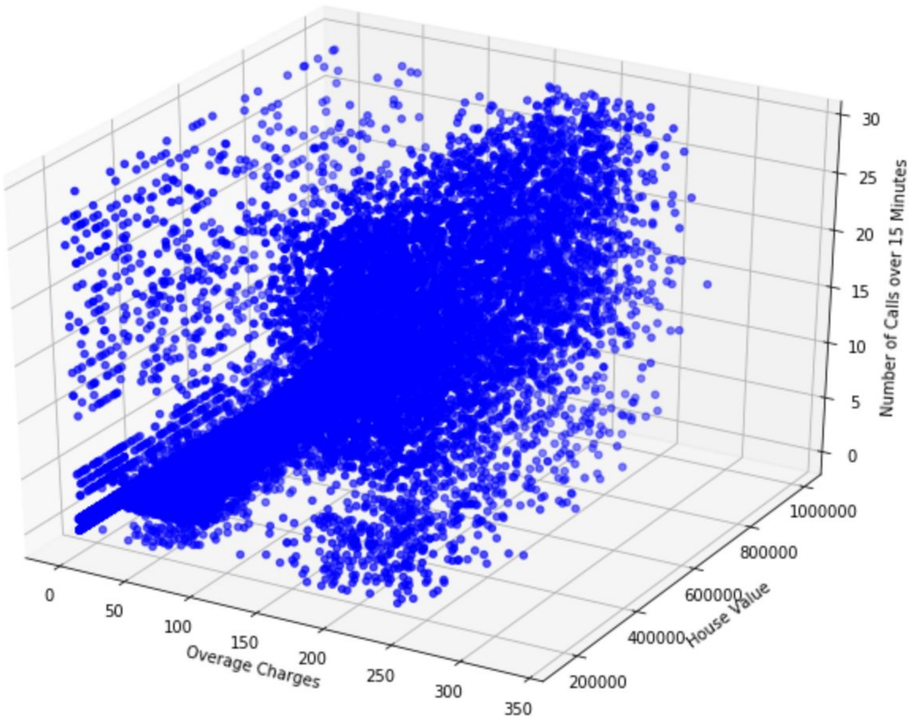
The visualization above shows the 2 most correlated factors of leave (House and Overage) mapped to one another. It seems the less valued the house and the higher the overage, the more likely a customer is going to churn and leave their current plan. Keep in mind 1 (blue dots) represent STAY and 0 (red dots) represent LEAVE. Affluent customers who own more expensive homes are more likely to stay with their current cell phone plan despite the overage and extra charges.

Another thing to note is the data is balanced. 50.74% of the 20,000 instances are labeled as STAY and the remaining are classified as customers who LEAVE.

*Note: If you are interested in experimenting with the Google Facets Visualization tool implemented in the Jupyter notebook, reload the browser after running the notebook. A glitch in the facets API does not display the visualization tool if the browser is not reloaded.*

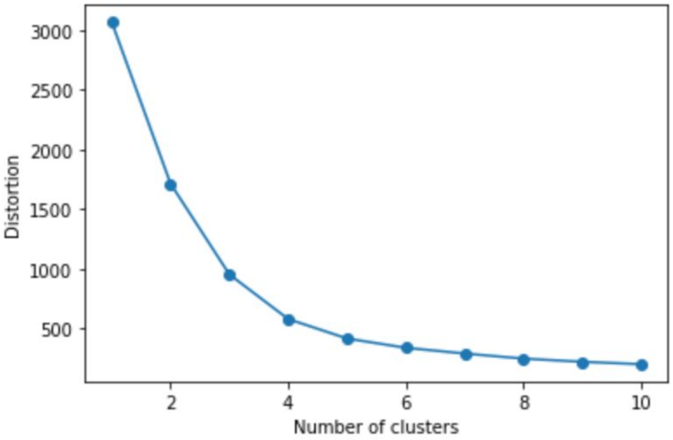
## IV. Data Exploration with K-Means Clustering

To further understand the correlations between the given attributes of the data, the K-means clustering algorithm has been implemented in the provided notebook. The goal of this project is to predict whether a customer leaves or a customer stays. The heatmap created shows the 3 more correlated variables with the label are overage, house, and over 15-minute calls per month. Below is a 3D graph of each of these attributes.

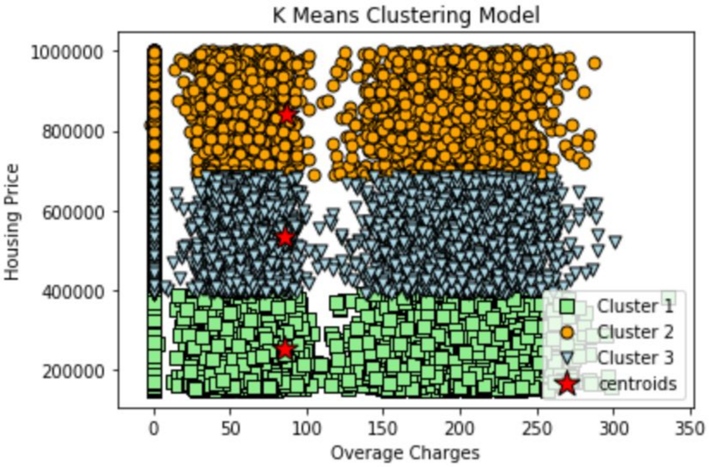


It is clear that overage and 15-minute calls will be correlated because more phone usage leads to a higher bill. For this reason, the over 15-minute call attribute was not used for the cluster method and K means was implemented using 2-dimensional data. The first dimension being overage charges and second, being housing price.

Picking the correct K value is an integral part of determining how effective K means will be. A K value too small will not effectively cluster the data and provide no insight. A K value too large will not provide generalized clusters and overfit the data. The Elbow method is used to ensure a proper K value if picked. The Elbow method is a graphical tool used to estimate the optimal K value of a dataset. The larger K becomes, the smaller the distortion (within-cluster sum squared error) will become because the data points will be closer to the assigned cluster centroids. The elbow method allows us to visually identify the "elbow" in the graph or at which point increasing K will minimally improve clustering (distortion begins to plateau).

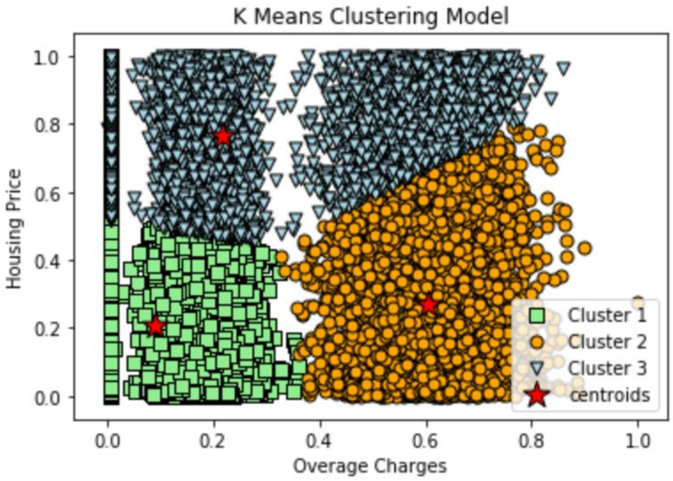


The graph above shows that as K increases after the value 3, the distortion drops minimally and any larger K value is not effective. For this reason, the K value of 3 has been chosen. Below is the result of the clustering model run in the data with a K value of 3.



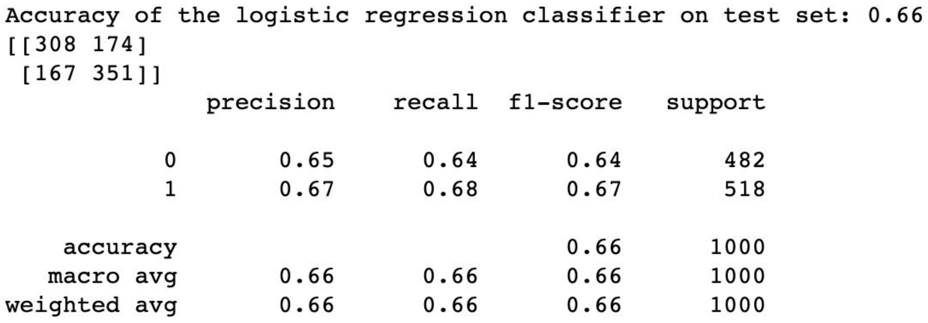
It seems that the clustering model is heavily influenced by the housing price. This is because K means is calculated via distance and the housing price dimension will overshadow the value of the overage charge dimension. For example, owning $50,000 and having $300 dollars in overage charges will result in the housing price having a much heavier weight. To fix this, the housing prices have been normalized between 0 and 1, and the overage charge normalized in the same fashion. The updated clusters can be seen below (top of page 7).

These updated clusters align more with the raw data shown in the Google facts visualization above in part III of this report. It is interesting to see how the overage charges, when normalized, so have a heavy influence on the centers of the clusters.



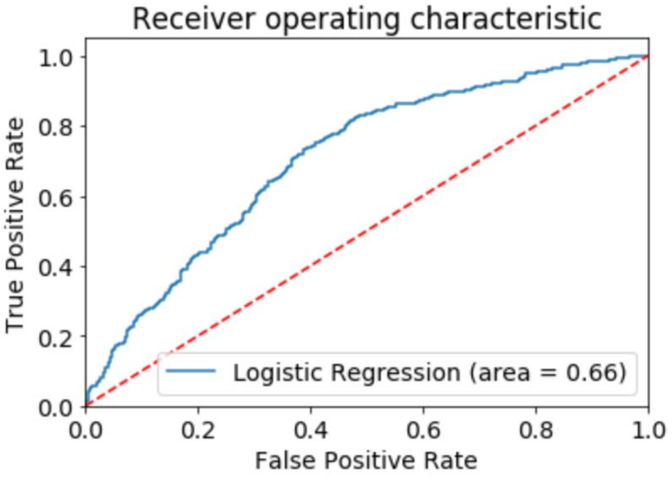
## V. Predictive Model A: Logistic Regression

The first predictive model built to analyze this dataset was Logistic Regression. The dataset was split into a training dataset and a test dataset. The training data is 95% of the original full dataset. The test data set is made of the remaining 1000 instances or 5%. Below are the results of the logistic regression classifier on the test data. The classifier is able to correctly predict 66% of the Customer Churn. The 2x2 NumPy array displayed beneath the accuracy represents the true negatives (308),  false positives (174),  false negatives (167 ), and true positives (351). The accuracy analysis reveals that both precision and recall are relatively consistent.



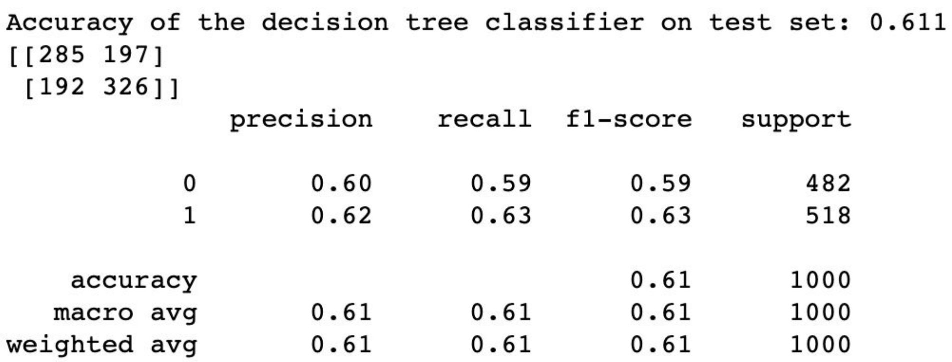
```
Accuracy of the logistic regression classifier on test set: 0.66
[[308 174]
 [167 351]]
              precision    recall  f1-score   support

           0       0.65      0.64      0.64       482
           1       0.67      0.68      0.67       518

    accuracy                           0.66      1000
   macro avg       0.66      0.66      0.66      1000
weighted avg       0.66      0.66      0.66      1000
```

In addition to the classifier summary,  the ROC curve is plotted below. The ROC curve shows the logistic regression classifier has a relatively high false-positive rate relative to the true positive rate. The heat map above shows that most of the attributes of the raw data are not heavily correlated with their labels. Furthermore, the logistic regression classifier uses every single attribute from the original data set. The hypothesis made when originally building the logistic regression classifier was that the classifier would disregard any attributes that had a low correlation with the labels for each data point. Because of this hypothesis, every single attribute was used and none were disregarded. Looking at the ROC curve, it could be said that removing some of the attributes may have improved the accuracy and validation of this classifier on this data set.

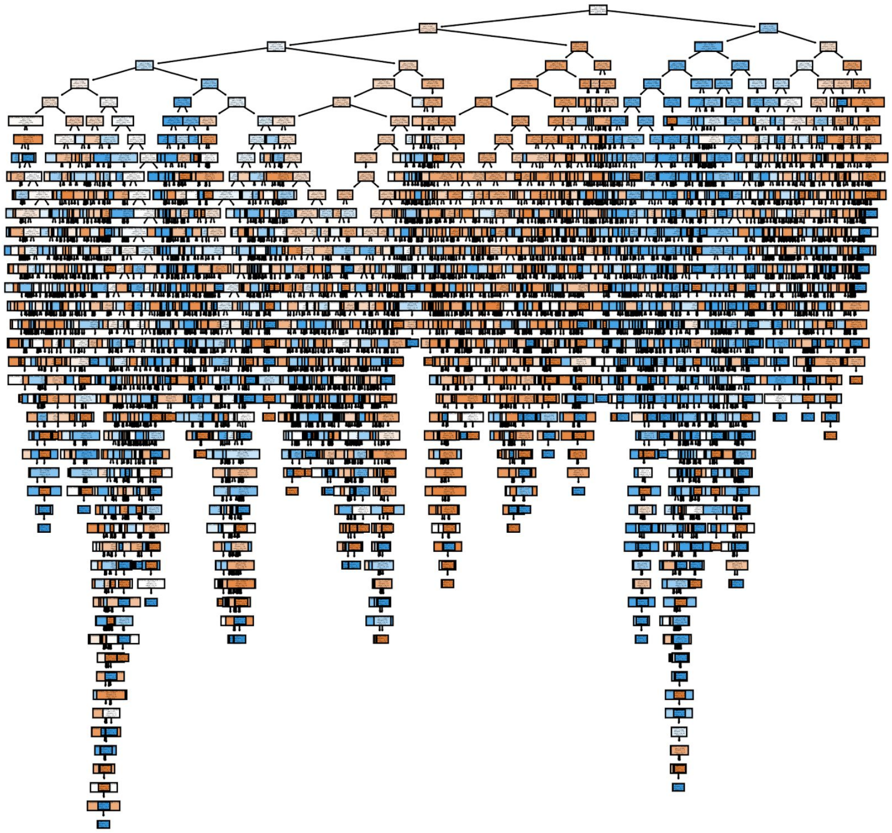


## VI. Predictive Model B: Decision Tree

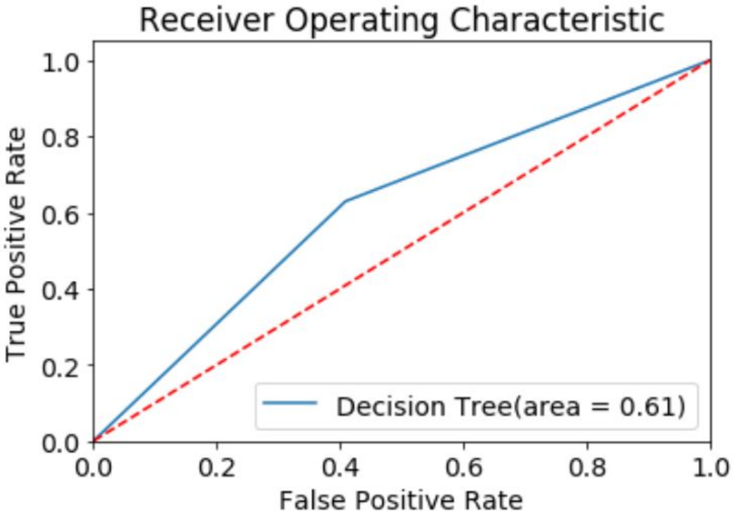
The second predictive model used to analyze this data set was a decision tree. Just like the first predictive model used, the data set was split into a train data set and test data set. The train data set consisted of 95% of the original data set and the remaining 5% of the original data set was used as test data for the decision tree. Below are the results of the decision tree as well as an image of the resulting decision tree. Each of the decision tree nodes is labeled but due to the enormous size of the tree, it is difficult to see the information provided in each node. As can be seen, by the accuracy metrics below, the decision tree underperforms the logistic regression classifier. The accuracy of the decision tree classifier is 61.1%.



```
Accuracy of the decision tree classifier on test set: 0.611
[[285 197]
 [192 326]]
              precision    recall  f1-score   support

           0       0.60      0.59      0.59       482
           1       0.62      0.63      0.63       518

    accuracy                           0.61      1000
   macro avg       0.61      0.61      0.61      1000
weighted avg       0.61      0.61      0.61      1000
```



The ROC curve for the decision tree classifier is shown below. Just like the logistic regression ROC curve,  the ROC curve for the decision tree classifier shows a relatively high false-positive rate in comparison to the true positive rate.



## VII. Conclusions

The analysis done on this data provides insight into customer churn. One interesting conclusion is that reported satisfaction or considering changing plan attributes were relatively independent of the customer churn label. This seems peculiar considering how important customer satisfaction would be in predicting customer churn. The analysis done does show that overage prices seem to be a significant factor in predicting customer churn. The overage charges specifically with lower-income customers create more customer churn. One idea to solve this would be to reduce overage charges in general or to set up different payment options for lower-income customers to pay off their overage charges to avoid those customers churning and moving to another phone provider.

In addition to these conclusions, it is important to note how almost every attribute has a normal distribution amongst all of its possible values. This seems highly peculiar and makes it difficult to analyze this data and get a higher accuracy rating using the predictive models.

## VIII. Instructions to Run the Code

To run all the models and data visualization shown in this report, open the Customer Churn Notebook in the provided Customer Churn Project file. This notebook will contain all the instructions on how to run the code. To run the notebook, keep all the associated files in the same folder.

## IX. Sources

Please note to complete this analysis many tutorials and scikit learn documentation was referenced to build these predictive models, charts, visualizations, and other data analytics tools provided in the jupyter notebook. All of these sources are listed below.

1. Building A Logistic Regression in Python, Step by Step: link
2. K-Means Clustering with scikit-learn: link
3. 3D Scatter Plot with Python and Matplotlib: link
4. Preprocessing Data: link
5. Multidimensional data analysis in Python: link
6. Python Data Visualization — Heatmaps: link