Understanding and Predicting User Retention

Joseph Reyes **Author**
jreyes1007@gmail.com email

Introduction

The goal of this project is to understand what features, if any, can serve as predictors for determining user retention and provide ways the company can leverage insights discovered. This report answers a few important questions. First, what percentage of the users sampled were retained? Second, can machine learning algorithms be used to reliably predict user retention? Last, how can the company leverage the insights discovered to improve its rider retention?

Preprocessing

In this phase the raw data set was cleaned in order to facilitate analysis in subsequent phases. The signup and last trip dates were used to derive the class labels by adding 6 months to the signup date and comparing this date with the last trip date. A 1 was used to represent users that had completed a trip 30 days prior to the pull date and 0 otherwise. About 24% of the observed users were retained. This means there was a significant class imbalance in the dataset. To correct this, random samples from the class in excess were discarded until class counts were balanced.

Figure 1: Class Imbalance

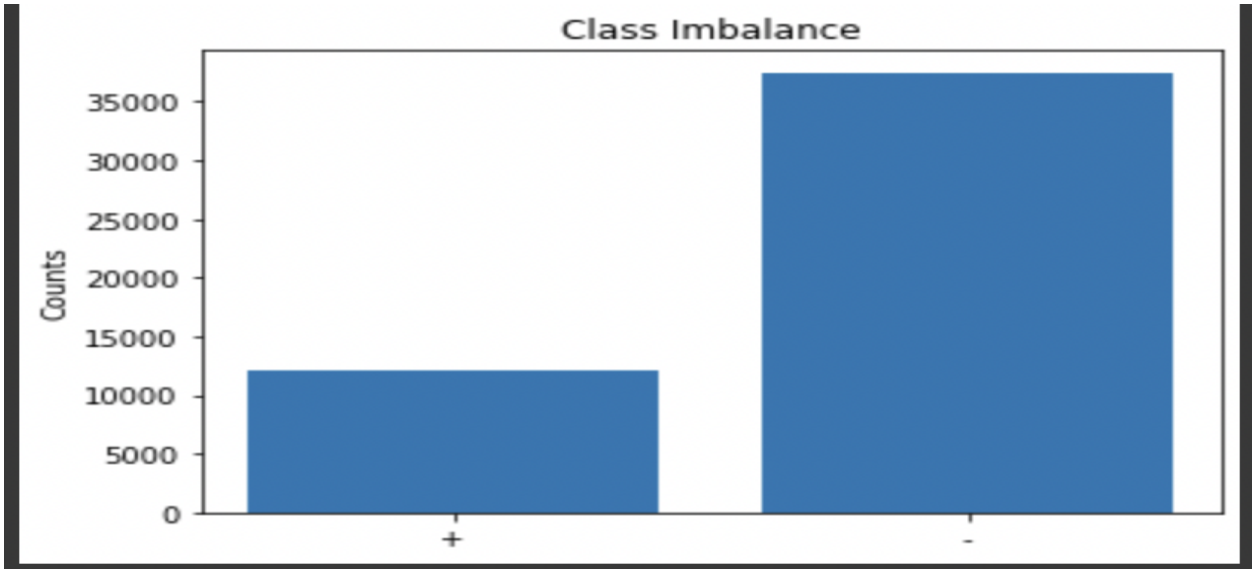


Figure 2: Balanced Classes

To put the categorical data in a form that is more useful to the machine learning algorithms, one hot encoding was used. The number of distinct elements for each categorical feature was determined. A new attribute was created for each. A 0 or 1 was used to indicate the pertinence of the feature to the user.

Mean and median values of continuous attributes were considered as replacements for their missing data. A distribution plot was used to identify the central tendency of the attribute. Because of the presence of outliers, which heavily influences mean values, the median of the attribute was used as a representative value of the feature, and thus was used to replace the missing data. A miniscule amount of users had missing phone data. These samples were discarded since a replacement for that data could not be determined.

Figure 3: Average Rating of Driver

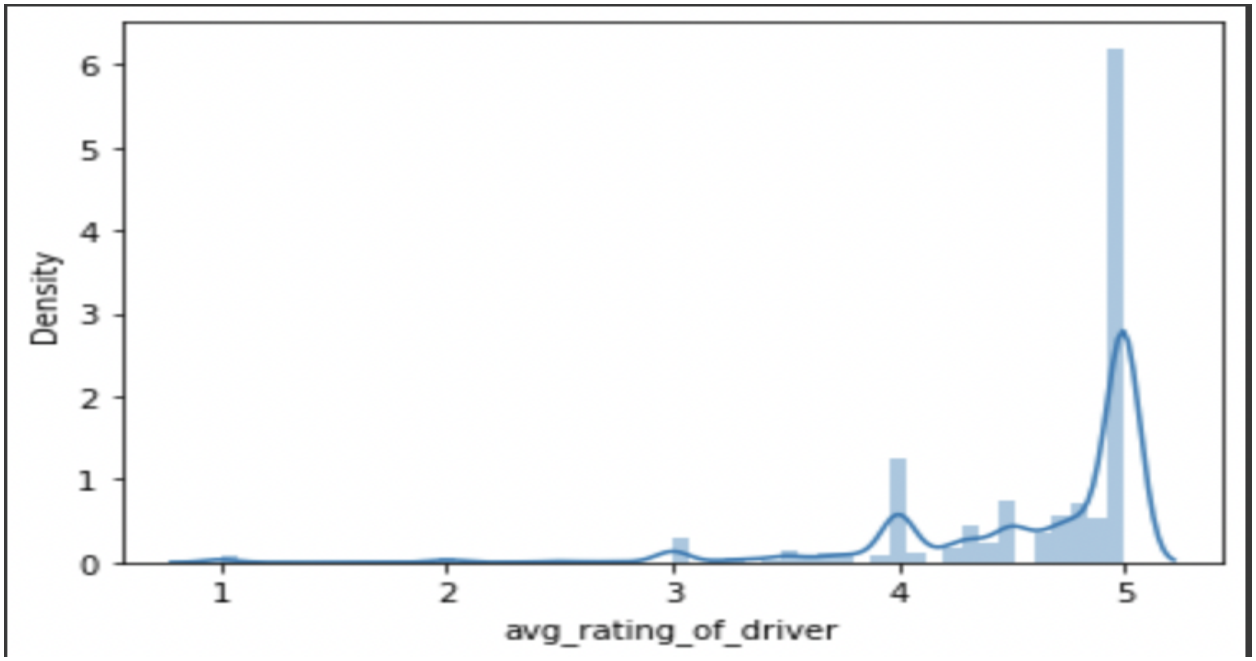


Figure 4: Average Rating by Driver

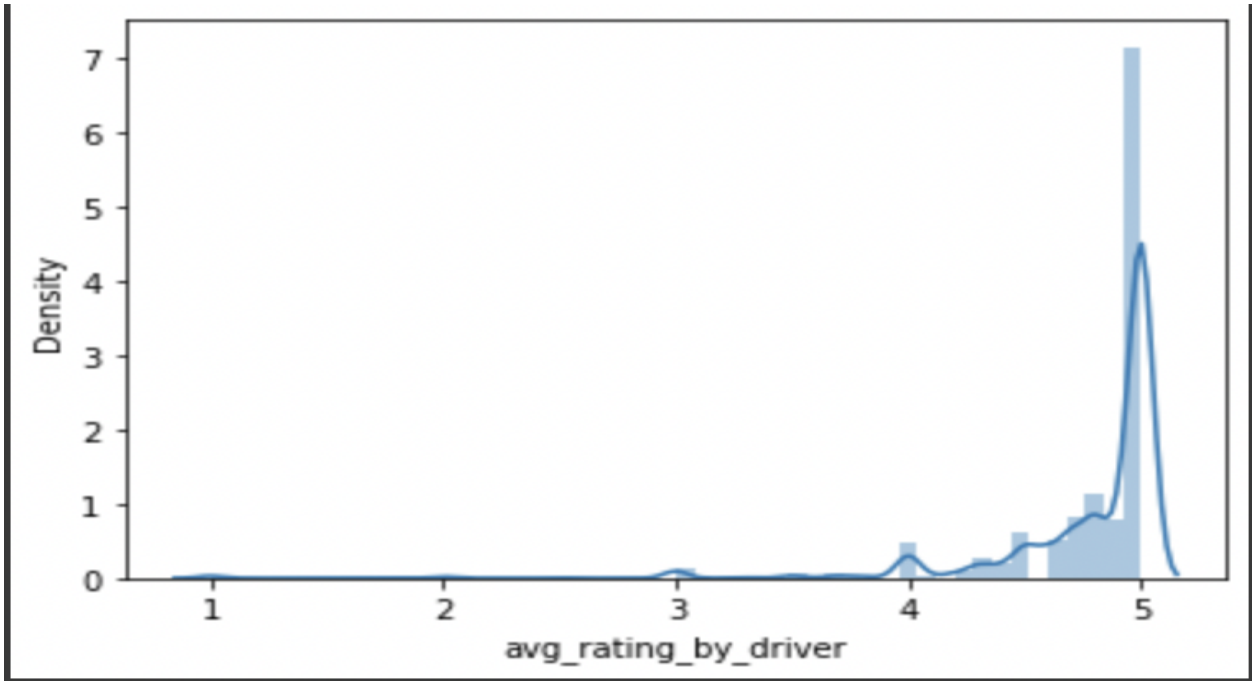
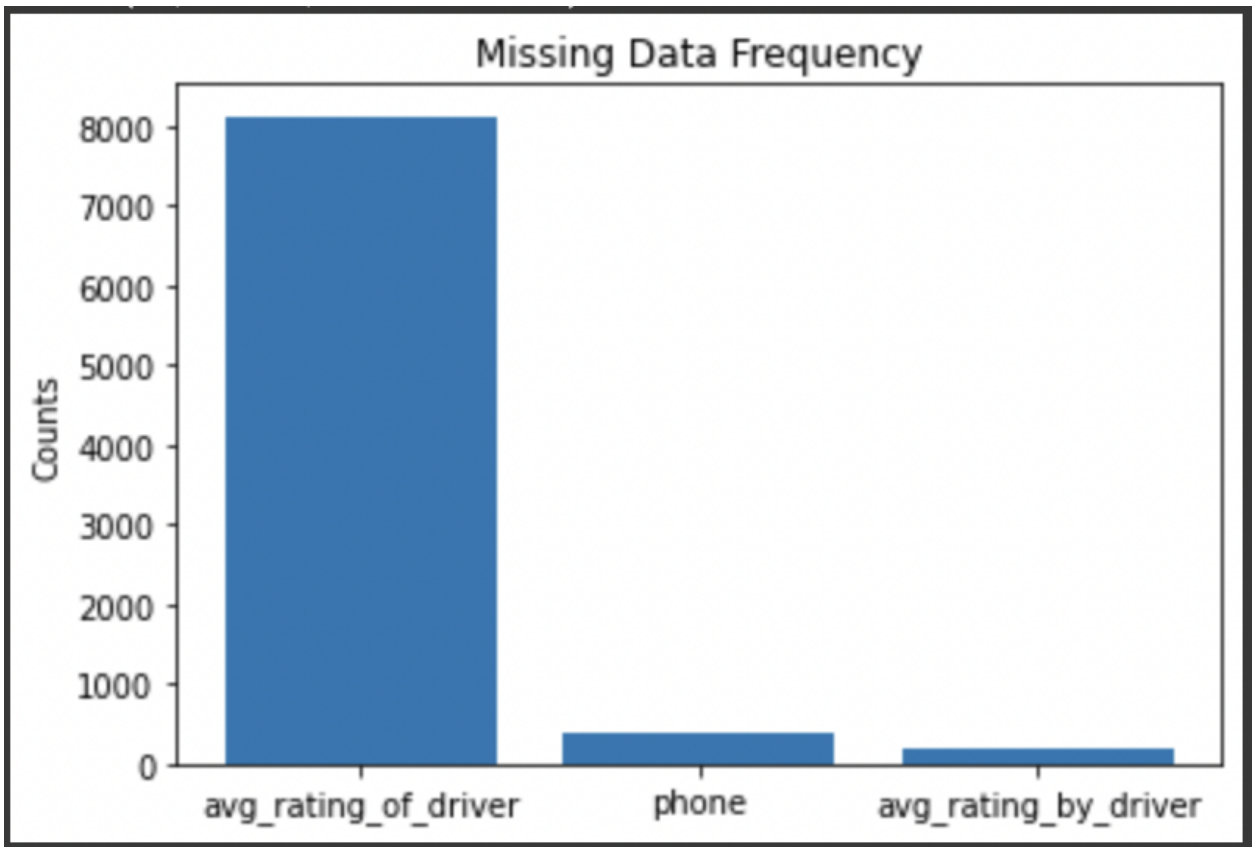


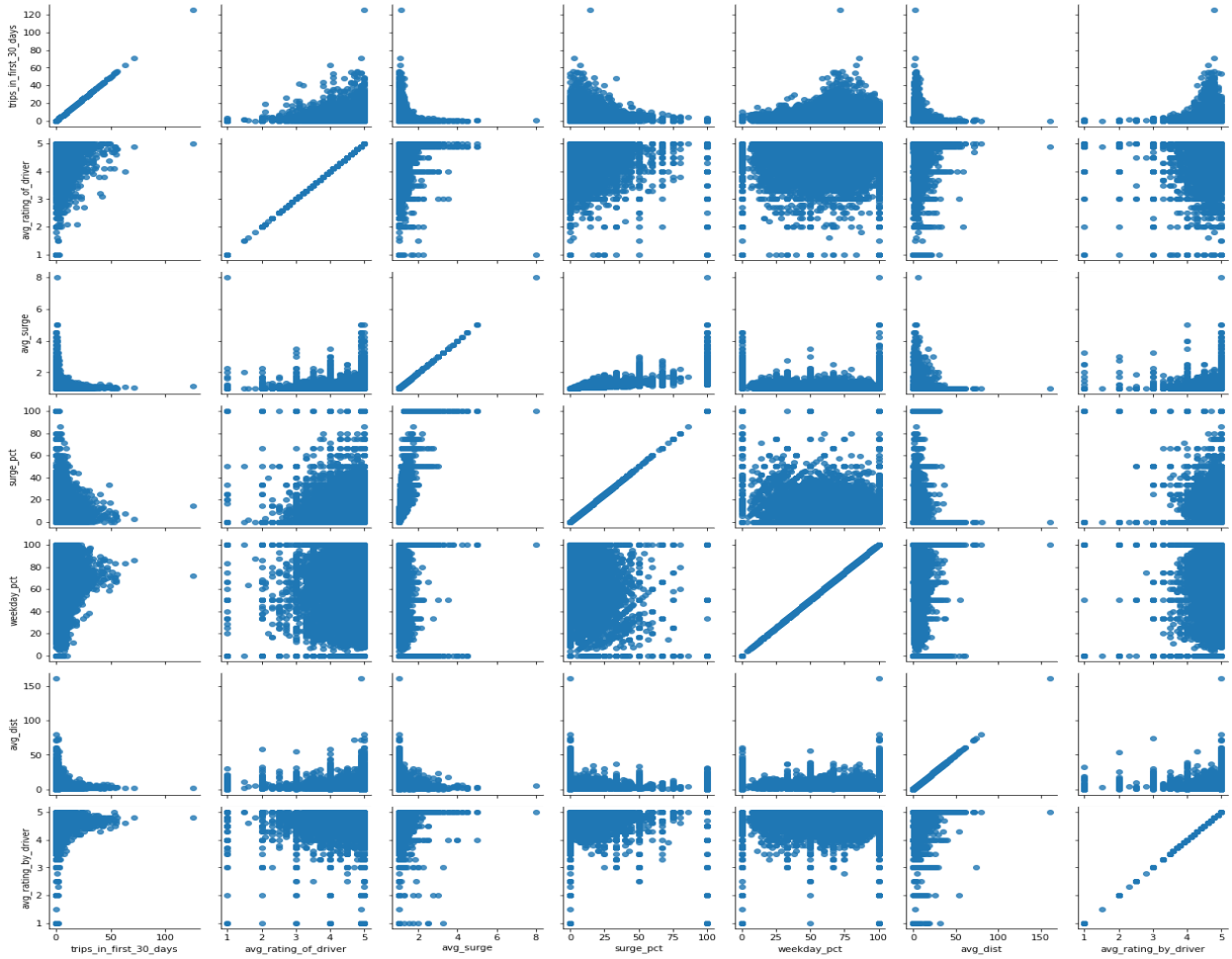
Figure 5: Counts of Missing Data



Exploratory Data Analysis

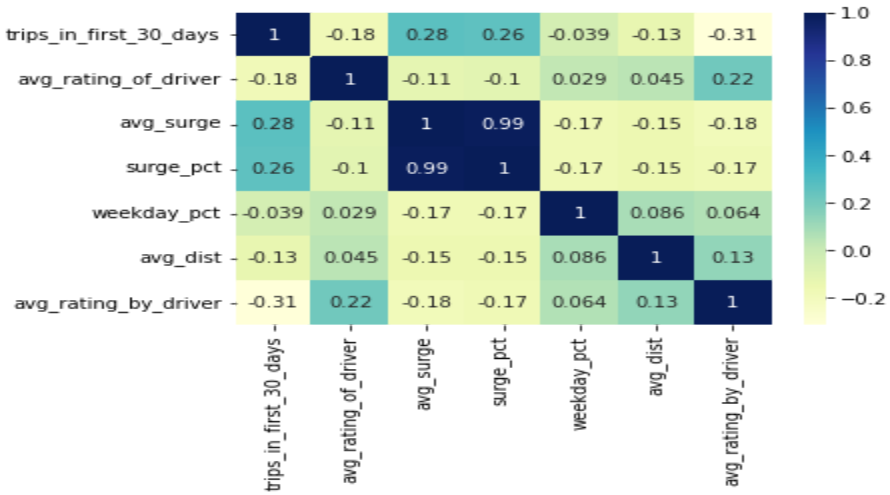
The focus in this phase was deriving meaningful insights to gain a better understanding of the data. To identify correlations between continuous features, scatter plots were produced using seaborn. This helped identify any possible dependencies between attributes. For most features, it was clear there was no correlation. However, some of the features displayed a monotonic relationship.

Figure 6: Relationships Between Continuous Features



To determine the strength and direction of the correlation between continuous features, Spearman's coefficient was determined.

Figure 7: Correlation Between Features



To identify correlations between dichotomous features and the target, a chi-squared test was used. Based on the p-value of each test, there appeared to be a connection between the categorical attributes and the target. A point biserial correlation was used to quantify the relationship between continuous features and the target. There was a noticeable positive correlation between the number of trips a user completed within the first 30 days of signing up and retention. This suggests that stakeholders may want to focus on ways to increase the number of trips a user completes within the first 30 days (perhaps by promotional offers, special perks, etc.).

Table 1: Contingency Table

retained_user	0	1
upgraded_user		
0	8062	5914
1	4052	6200

I can't overstate how useful performing PCA was for my analysis. After plotting the transformed data on a two-dimensional plot, I noticed amazing clusters and subclusters. This gave me valuable insight as to what machine learning algorithms to consider.

Figure 8: Transformed Data With Scaling

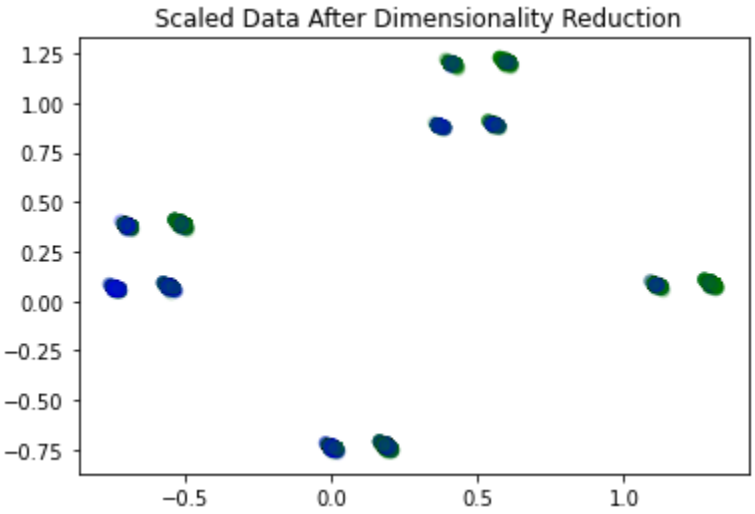
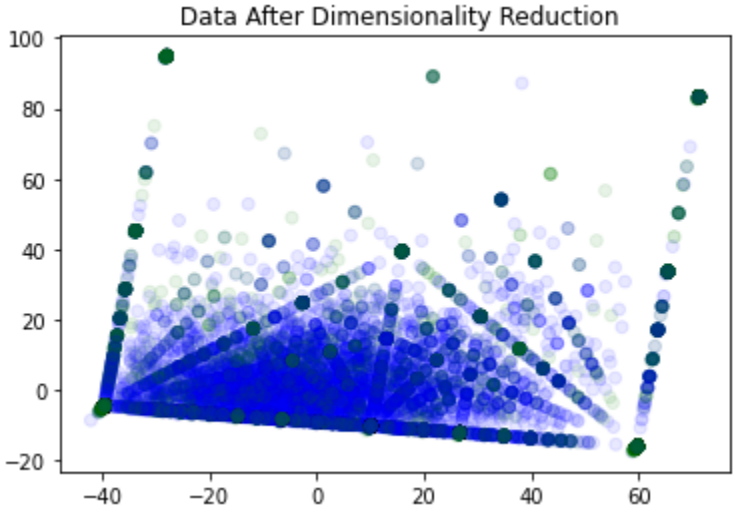


Figure 9: Transformed Data Without Scaling



To gain a better understanding of the features, the maximum, minimum, mean, and standard deviation was determined. The max and min values helped detect outliers in the data. The mean gave an indication of the feature's average value, whereas the standard deviation indicated how dispersed the values were from the mean.

Predicting User Retention With Machine Learning

Neural Networks

Since fully connected neural networks generally perform well on vector data stored in 2D tensors of shape (samples, features) dense layers were used. Hyper parameter optimization was done by varying the number of hidden layers and the capacity (the number of artificial neurons) of each layer. Networks containing 1-3 hidden layers with 4, 16, and 32 neurons were evaluated. A model consisting of two hidden layers and 16 neurons achieved the best performance. Interestingly, the models never showed signs of overfitting regardless of how many epochs were used. Sometimes models can begin to learn patterns that are specific to the training data, causing performance on validation data to worsen. This is the well-known overfitting problem that can appear when dealing with neural networks. Models with dropout, L1, and L2 regularization schemes were also evaluated.

Figure 10: Neural Network Loss

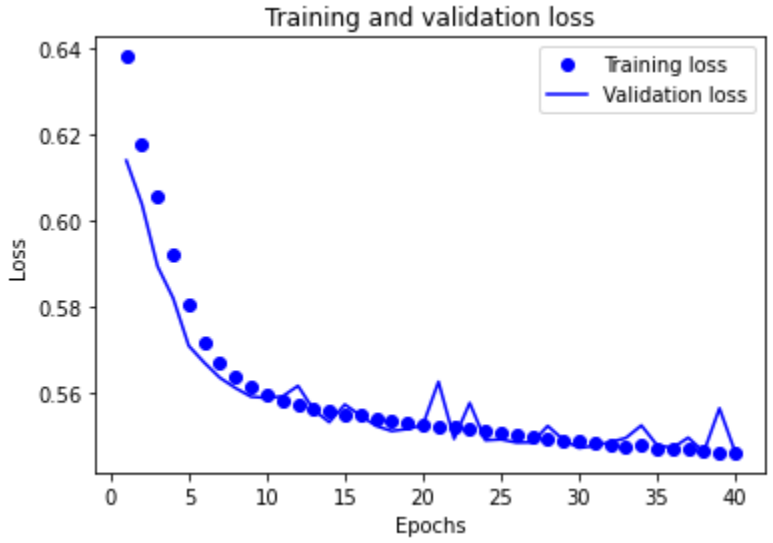


Figure 11: Neural Network Accuracy

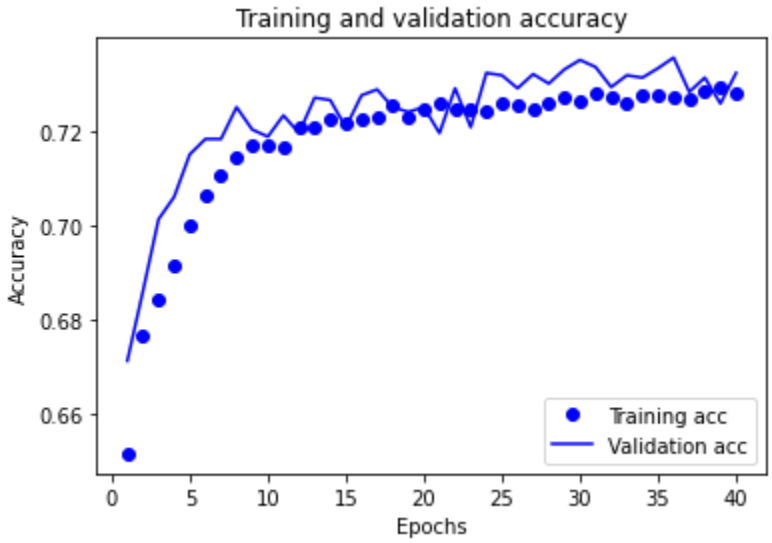


Figure 12: Neural Network True Positive Rate vs False Positive Rate

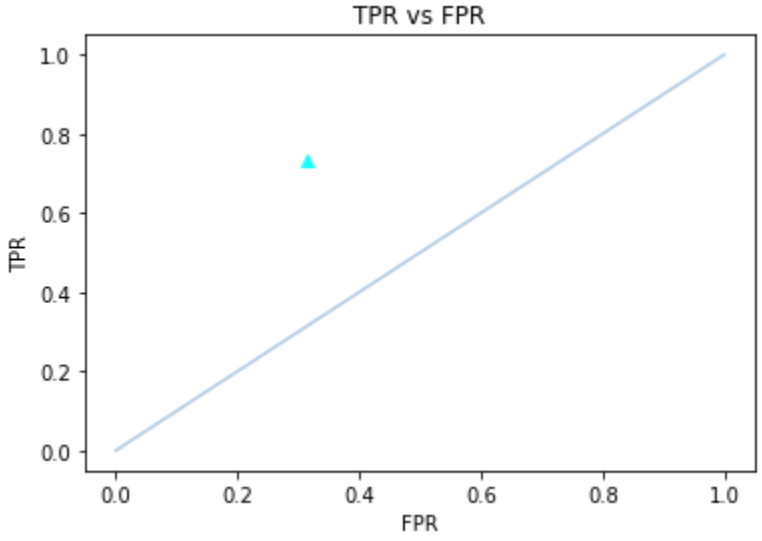


Figure 13: Neural Network Confusion Matrix

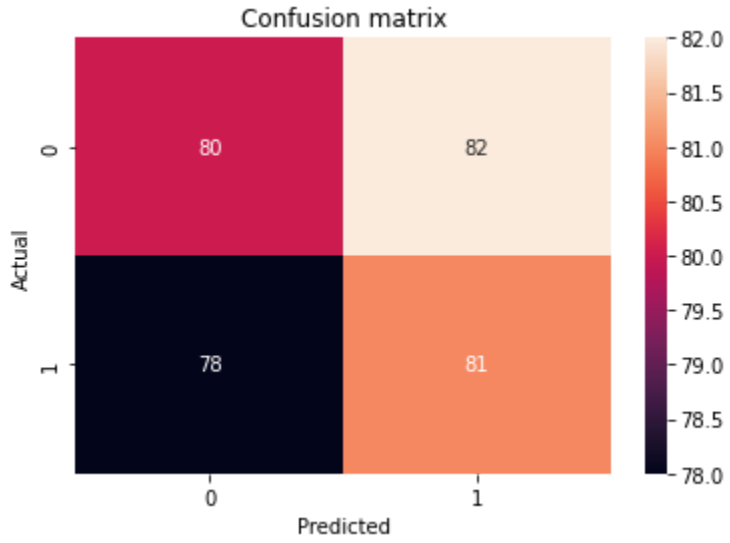


Table 2: Neural Network Metrics

Accuracy	True Positive Rate	Precision	False Positive Rate	True Negative Rate	False Negative Rate	Prevalence
0.70875	0.732894	0.693858	0.314751	0.685249	0.267106	0.49325

K-Nearest Neighbors

K-Nearest Neighbors was an obvious choice because identical class labels tended to be near one another in the 2D feature space. L2 norm was used to measure similarity between the data points. Of all the machine learning algorithms evaluated, this one performed the best.

Figure 14: K-Nearest Neighbors True Positive Rate vs False Positive Rate

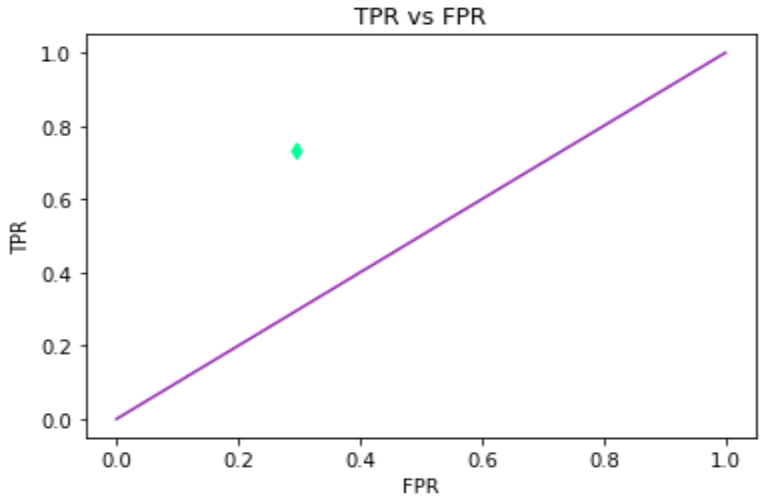


Figure 15: K-Nearest Neighbors Confusion Matrix

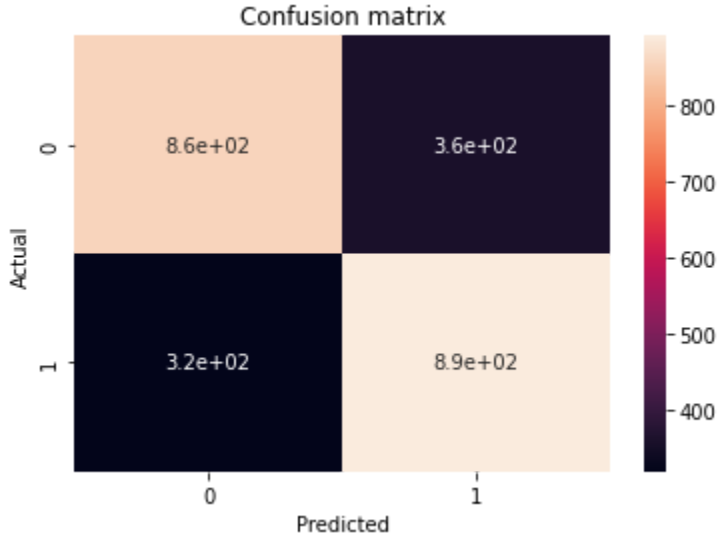


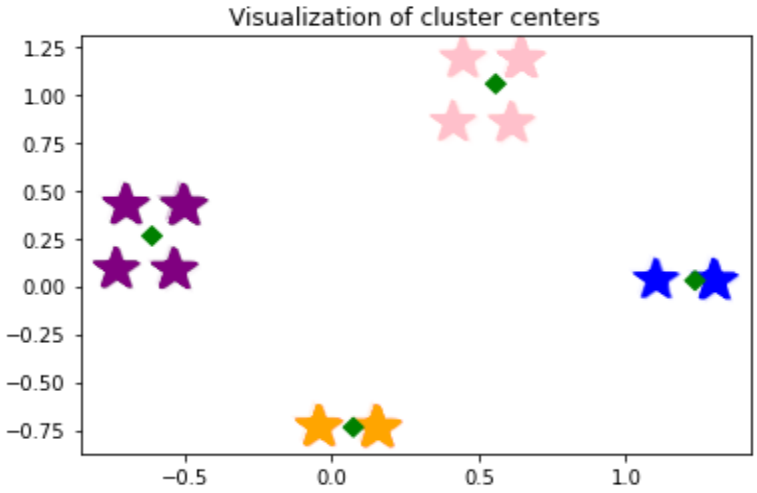
Table 3: K-Nearest Neighbors Metrics

Accuracy	True Positive Rate	Precision	False Positive Rate	True Negative Rate	False Negative Rate	Prevalence
72.8	0.735537	0.713141	0.295136	0.704864	0.264463	0.499381

K-Means

K-means was used to associate data points with one of four clusters. Four clusters were chosen because it is apparent from the plot of the transformed data that there are four main clusters. This can be useful because knowing which cluster a data point is in can be an indication of its class label.

Figure 16: K-Means



Logistic Regression

Logistic regression is a popular machine learning algorithm for binary classification problems. It is a linear method whose parameters can be determined by using maximum likelihood estimation.

Figure 17: Logistic Regression True Positive Rate vs False Positive Rate

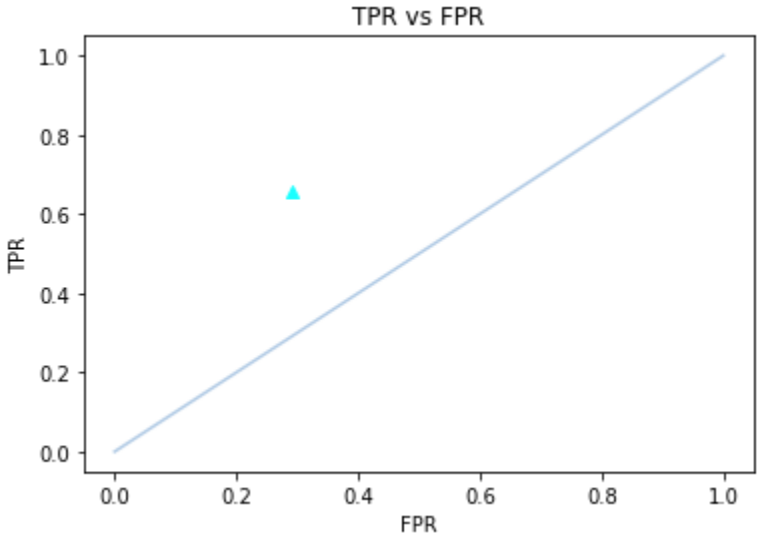


Figure 18: Logistic Regression Confusion Matrix

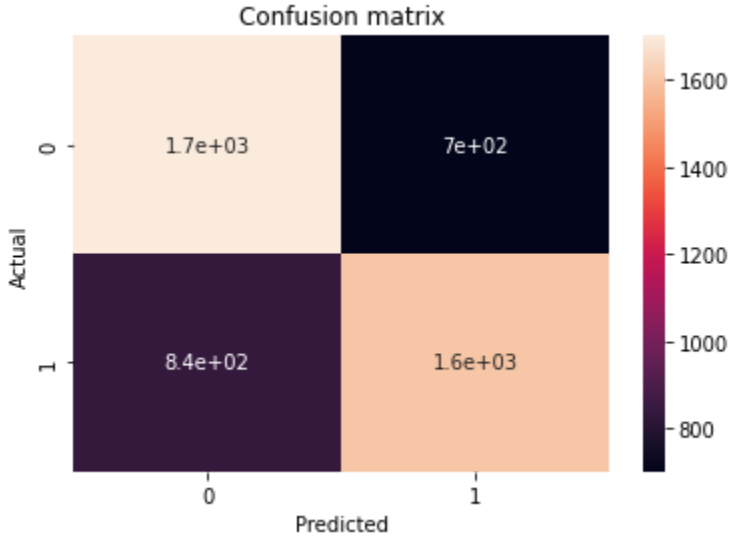


Table 4: Logistic Regression Metrics

Accuracy	True Positive Rate	Precision	False Positive Rate	True Negative Rate	False Negative Rate	Prevalence
69.5	0.655483	0.695312	0.292256	0.707744	0.344517	0.504333

Recommendations

As mentioned before, there was a noticeable positive correlation between the number of trips a user completed within the first 30 days of signing up and retention. This suggests that stakeholders may want to focus on ways to increase the number of trips a user completes within the first 30 days (perhaps by promotional offers, special perks, etc.). I also suggest research be done to figure out the root cause and meaning of the four clusters that were discovered after using PCA. This may lead to some valuable insights that can help the company improve its rider retention. Machine learning algorithms tend to perform better when they have a lot of data to learn from, so, in order to improve performance, acquiring more data may be beneficial.