Final Project

December 1, 2022

Team Member Names: Madeline Witters

Project Title: Predicting Customer Churn and Identifying Attributes of At-Risk Customers

Problem Statement

One of the most important metrics of success in any business is customer retention. A business's customer churn rate can have very significant financial impacts that affect the company in a multitude of ways. A Harvard Business Review article recently stated that merely increasing a company's customer retention rates by 5% can increase profits upwards of 25%; conversely, the cost of obtaining a new customer can be anywhere from 5 to 25 times as expensive as retaining an existing one.

It therefore follows that if a company is able to predict which customers are at risk of leaving, they can use this information to better position themselves in a variety of ways, such as: creating an intervention plan for customers at risk of leaving, calculating potential loss of revenue in the next quarter, or simply better understanding their customer demographic and various market segments.

In this project, there are two central research questions that I will aim to answer:

1. Can I create a model that will predict customer churn with a reasonable accuracy rate?
   - Furthermore, does one model type outperform another in predicting customer churn?
2. What features/customer attributes (present in the dataset) are most important in predicting whether a customer will churn?
   - Additionally, what does interpretation of these attributes reveal (for example, are younger customers more at risk of churning)?

Data Source

For this project I am using the "Bank Customer Churn" dataset, sourced from Kaggle. The dataset has 10,000 data points and 12 variables. Each row represents an individual customer. Variables are both categorical and quantitative, and include demographic variables (age, gender, country), as well as industry-specific variables (balance, credit_card, etc.). The response variable "churn" is categorical, containing a 1 if the client left the bank and a 0 if they remained a customer.

Methodology

My methodology for this project consisted of three distinct parts:

- Exploratory Data Analysis
- Variable Selection

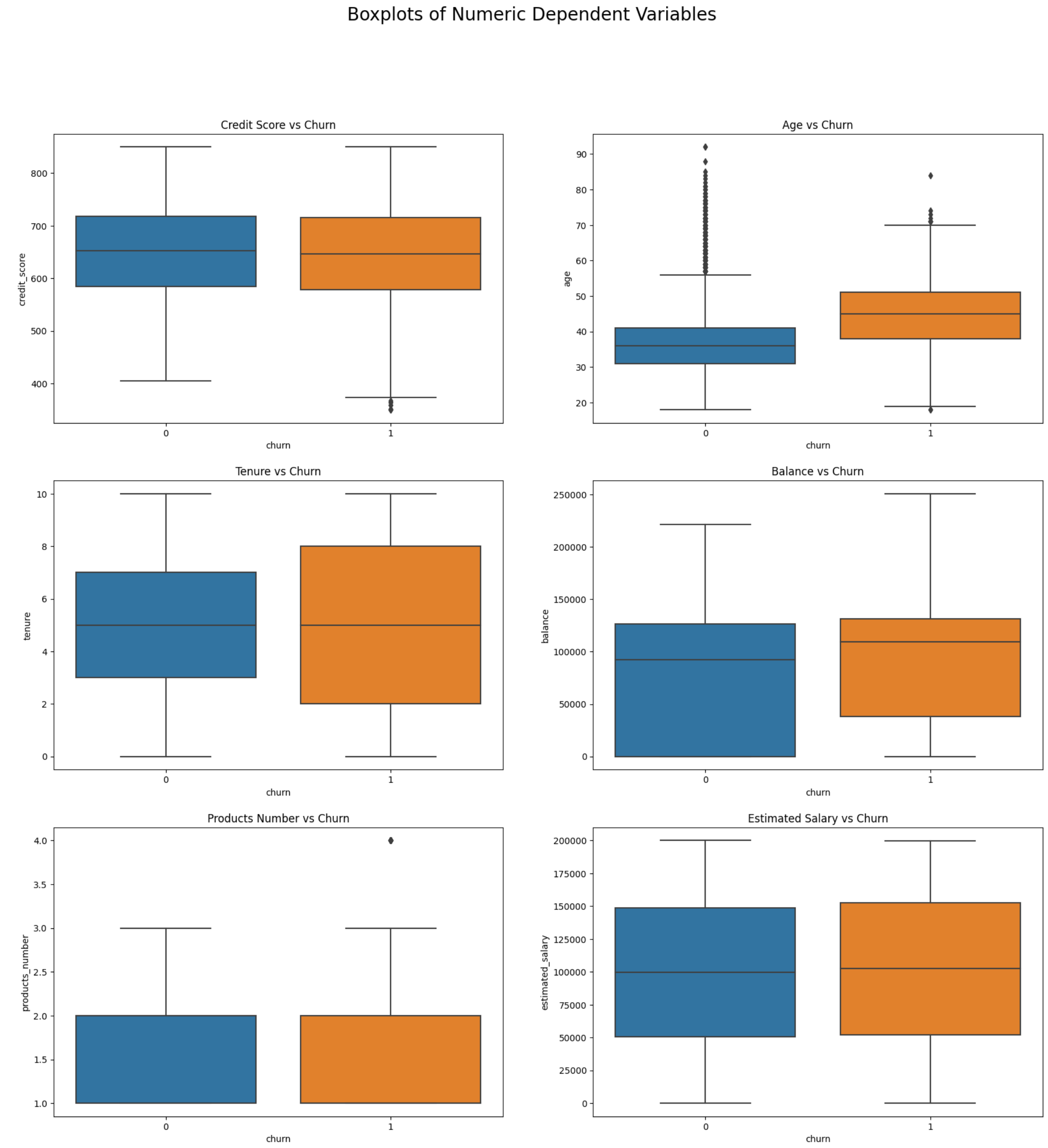
- Modeling

I will begin with the exploratory data analysis.

Exploratory Data Analysis

I began by performing some basic data cleaning. This primarily consisted of checking for any missing data, and dropping irrelevant features present in the dataset. There was no missing data in the dataset, so no imputation or row removal was necessary. There was only one irrelevant feature in the dataset, customer_id, a unique identification number for each customer. I used customer_id to confirm there were no duplicates present in the dataset, and then dropped it.

Next, I explored the remaining predictor variables via visual analysis. I began by creating boxplots for all the numeric variables. See below:

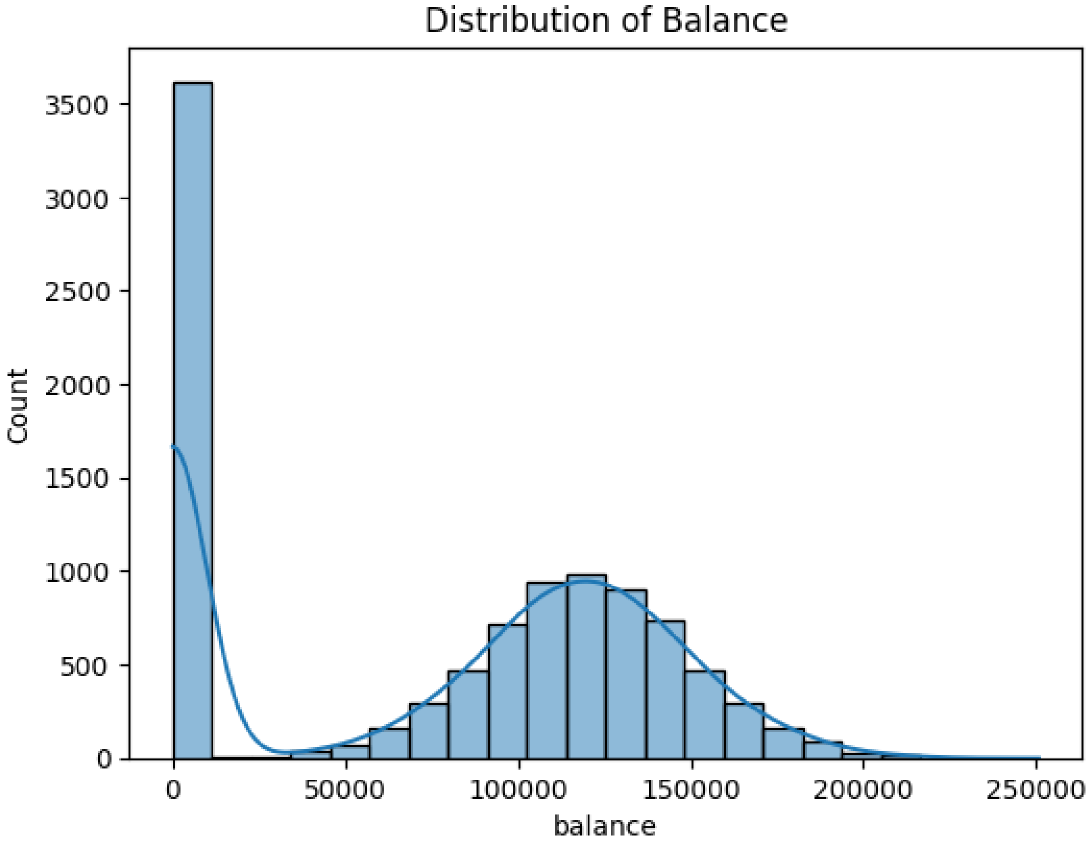


The boxplots revealed a couple interesting findings. First, it appears that the variables Credit Score, Products Number, and Estimated Salary all have very little impact on whether or not a customer churns. The boxplots for all of these variables were nearly identical when comparing the customer churn group (1) to the non-churn group (0). There was no visible distinction between the medians, and the inner-quartile range was nearly identical. Based off these boxplots alone, it would seem that these three variables will have little role in predicting churn.

Two variables that do appear to potentially have an impact, however, are Age and Balance. With the variable Age, it appears that older customers are more likely to churn, with the median for the churn group much higher than the non-churn group. With the variable Balance, it appears that those with a higher account balance are more likely to churn than those with a lower balance.

There are a couple additional items of note present in these boxplots. First, when examining the Balance boxplot for the non-churn group, it appears that there is a very large group of customers who have an account balance of zero. This may be a sign of a skewed distribution of the Balance variable. Second, several of the boxplots appear to show potential outliers. This is particularly notable in the Age boxplots. I decided to further investigate these anomalies.

I began by examining the distribution of the Balance variable via a histogram/KDE plot. See below:

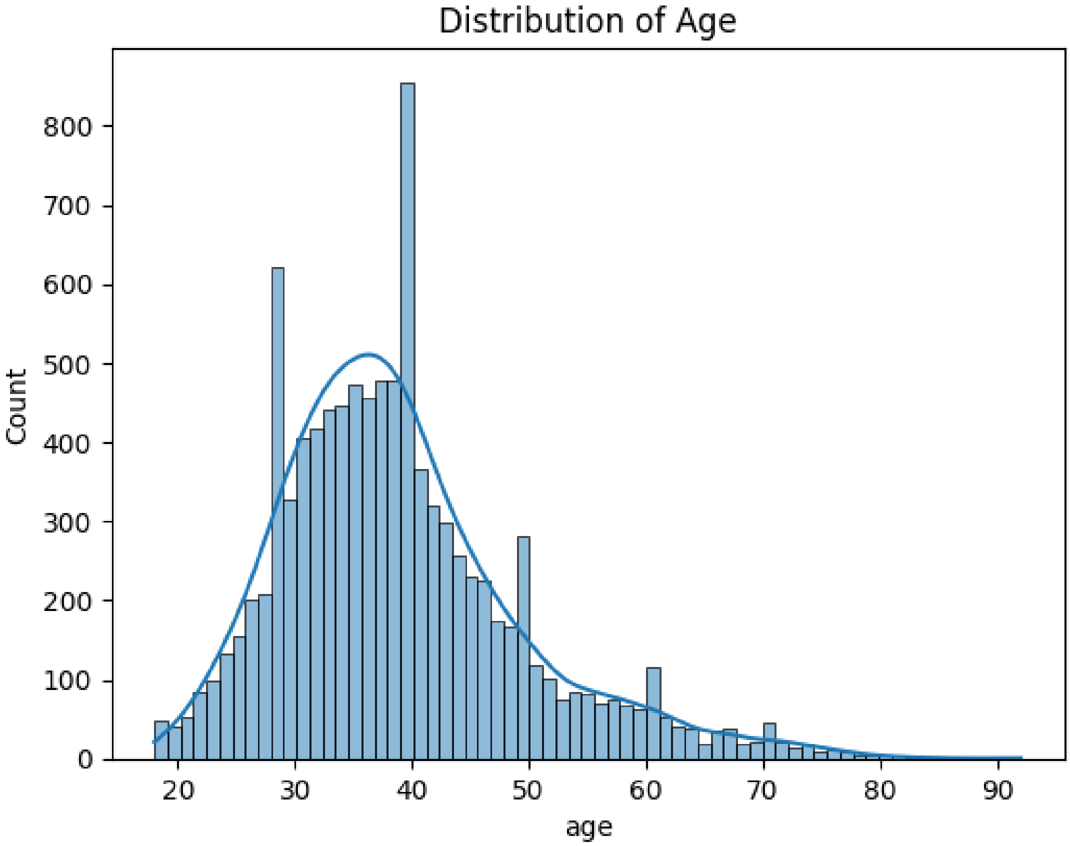


The histogram for Balance indeed confirmed a skewed distribution, with over 3500 customers having a balance of 0 in their account. While this seems odd, it could be the case that anyone who opens a new account with the bank automatically enters the system with a balance of 0 before their money is transferred over (which could also explain the shape of the non-churn balance boxplot above–those who just opened an account would be less likely to churn). Still, due to the distribution of the Balance variable, I decided to transform it to a categorical variable called “zero_balance”, with a 1 indicating the customer had a balance of $0, and a 0 indicating otherwise.

Next, I used the z-score test to check for outliers present in the numeric dependent variables. Using a threshold of 3 standard deviations away from the mean, the z-score test identified approximately 133 outliers present in the Age variable, and 60 in the Products Number variable. No outliers were identified in any of the remaining variables. Having identified the outliers, I further investigated them in order to determine whether they should be removed from the dataset.

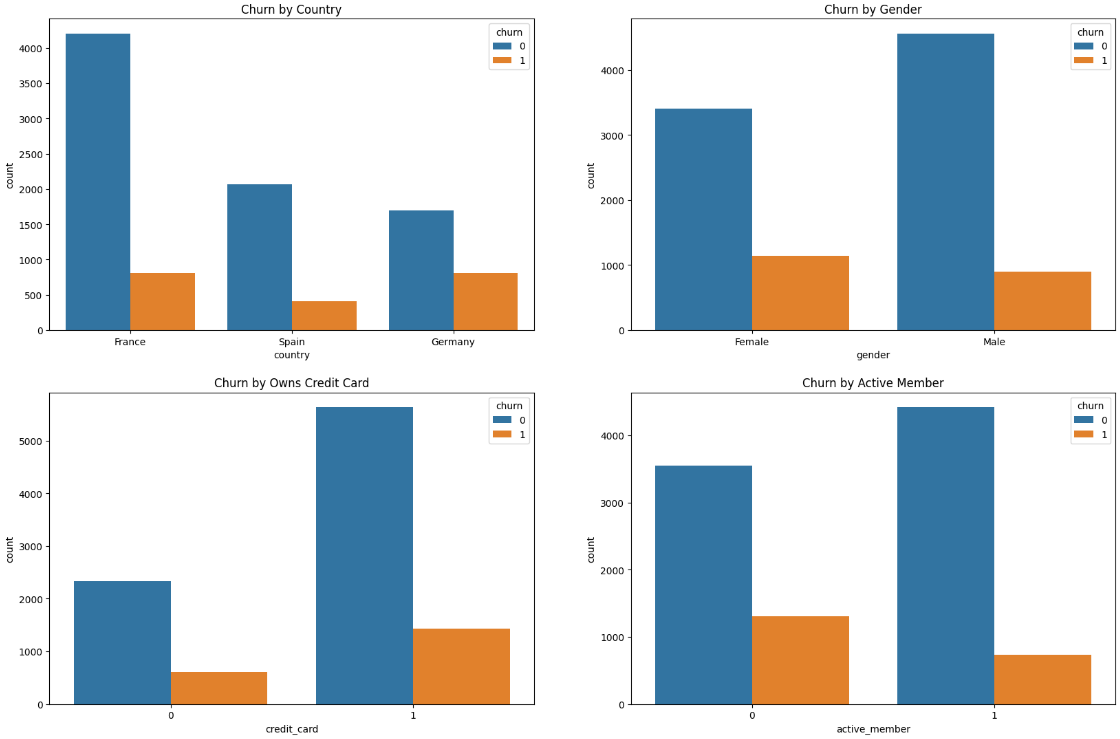
Most notably, none of the outliers appeared to represent incorrect data or data entry errors. For example, in the Age category, if one of the outliers had been an individual with the age 300, that would be a clear result of an error and could be thrown out. However, the outliers in Age primarily consisted of individuals in their 80s and 90s, who, while they may considered outliers, nonetheless represent real customers. A histogram/KDE plot of the Age variable may be found below illustrating this distribution. Likewise, the outliers in the Products Number variable also did not appear to indicate any obvious errors. Instead, it was simply the case that most customers had 1-3 products, with a small minority having 4 products.

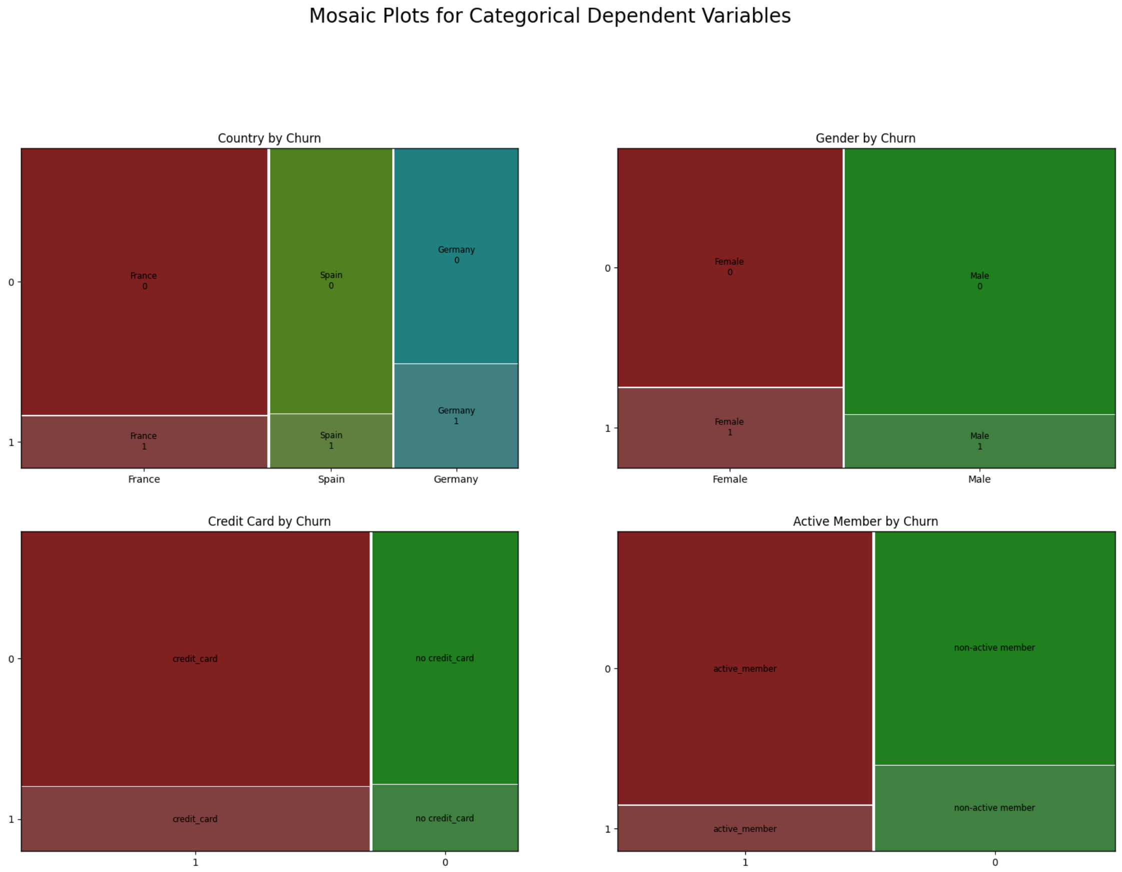
Ultimately, I decided not to remove any of the outliers from the dataset due to the fact that all of the outliers contained real customer information and did not appear to be the result of errors. Still, it should be noted that these outliers could still impact model fit. I address this issue further in the Conclusion section when discussing ways to improve and extend this project.



Having thoroughly explored the numeric dependent variables, I next turned to the categorical variables. I began by creating bar charts and mosaic plots for a basic visual analysis. See below:

Bar Charts for Categorical Dependent Variables

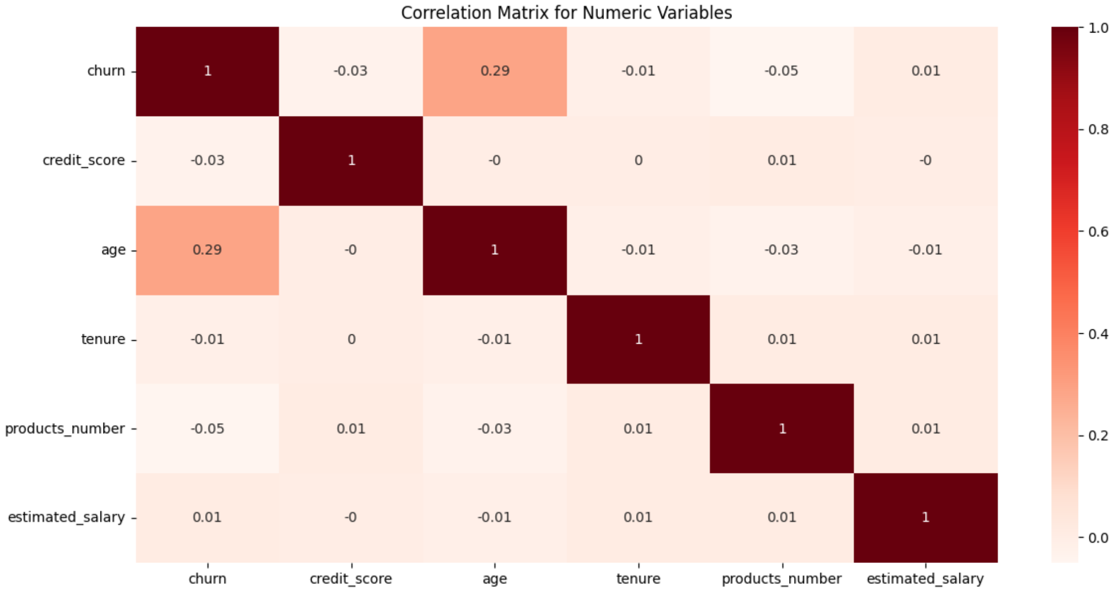




These plots contained a few interesting findings, and indicate that the variables Country, Gender, and Active Member may all play a role in predicting whether or not a customer will churn. It appears that those from Germany are more likely to churn than those from France or Spain; Females appear more likely to churn than males; and Non-Active Members are more likely to churn than Active Members. Whether or not an individual owns a Credit Card does not appear to have a large relationship with Churn.

Another interesting finding revealed by this plots is the fact that the dataset has a much higher proportion of data for customers who did not churn (0) than those who did churn (1). The official value counts for Churn are 7,963 customers who did not churn (0) and 2,037 who did churn (1). This unbalanced split has some implications for modeling which I will discuss later on.

The last part of exploratory data analysis I conducted was creating a correlation matrix for the numeric variables. High correlation between predicting variables may indicate that multicollinearity is present in the dataset, which has implications for variable selection and model building. However, the correlation matrix revealed very low correlations between numeric variables, indicating that multicollinearity will likely not be a large issue for this dataset. See below:



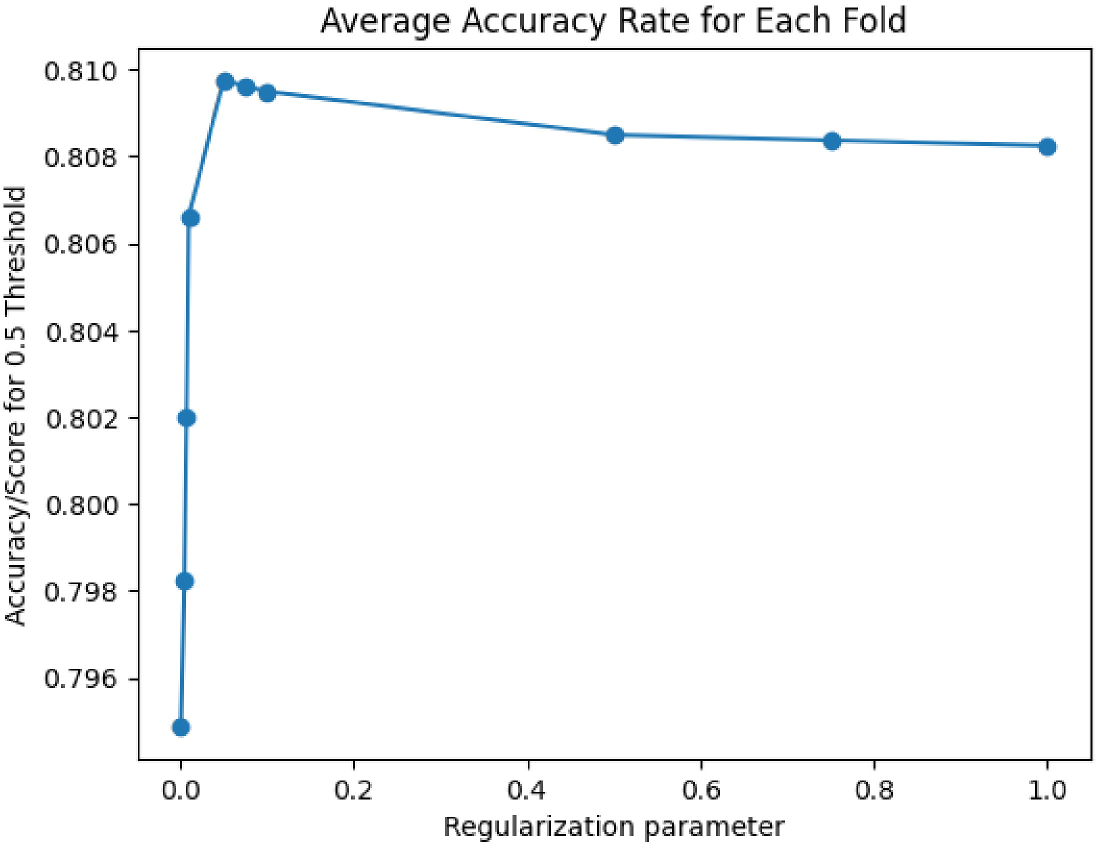
Having completed the exploratory data analysis, I conducted some one-hot encoding for the relevant categorical variables, and split the data into an 80/20 train-test split.

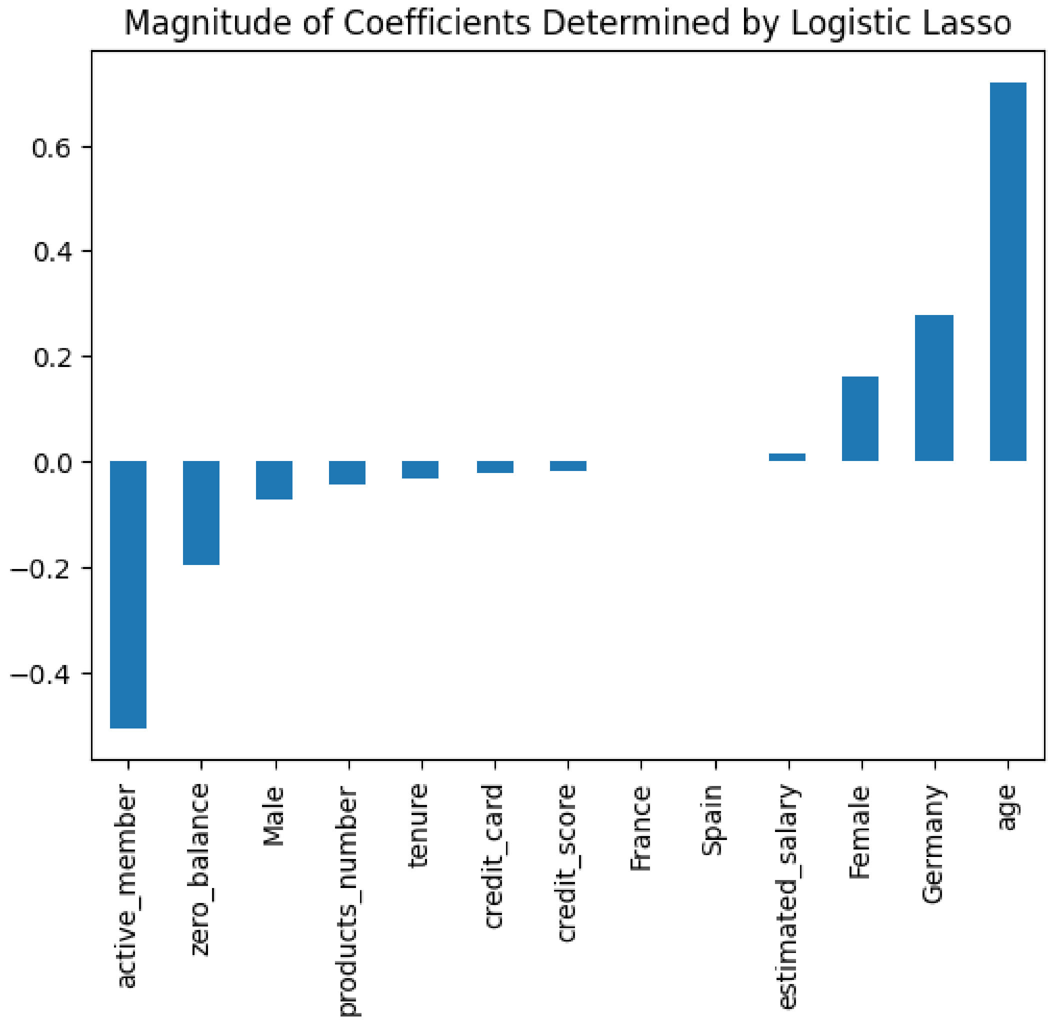
Variable Selection

I next turned to variable selection. While the correlation matrix revealed that multicollinearity may not be an issue for this dataset, variable selection can still aid by both removing irrelevant features (those with no predictive or explanatory power), and further by identifying the most important features.

I decided to use Lasso for variable selection. Traditional Lasso optimizes the least squares problem with a L1 penalty. However, my response variable is binary; thus I must perform Logistic Lasso regression to perform variable selection (essentially, this means fitting a traditional Logistic Regression model but adding in an L1 penalty). I tuned the penalization parameter for the Logistic Lasso model via 10-fold cross-validation. Ultimately, the best regularization parameter was found to be 0.05. It should be noted that in sklearn's LogisticRegressionCV, the regularization parameter describes the inverse of regularization strength; thus smaller values specificy stronger regularization.

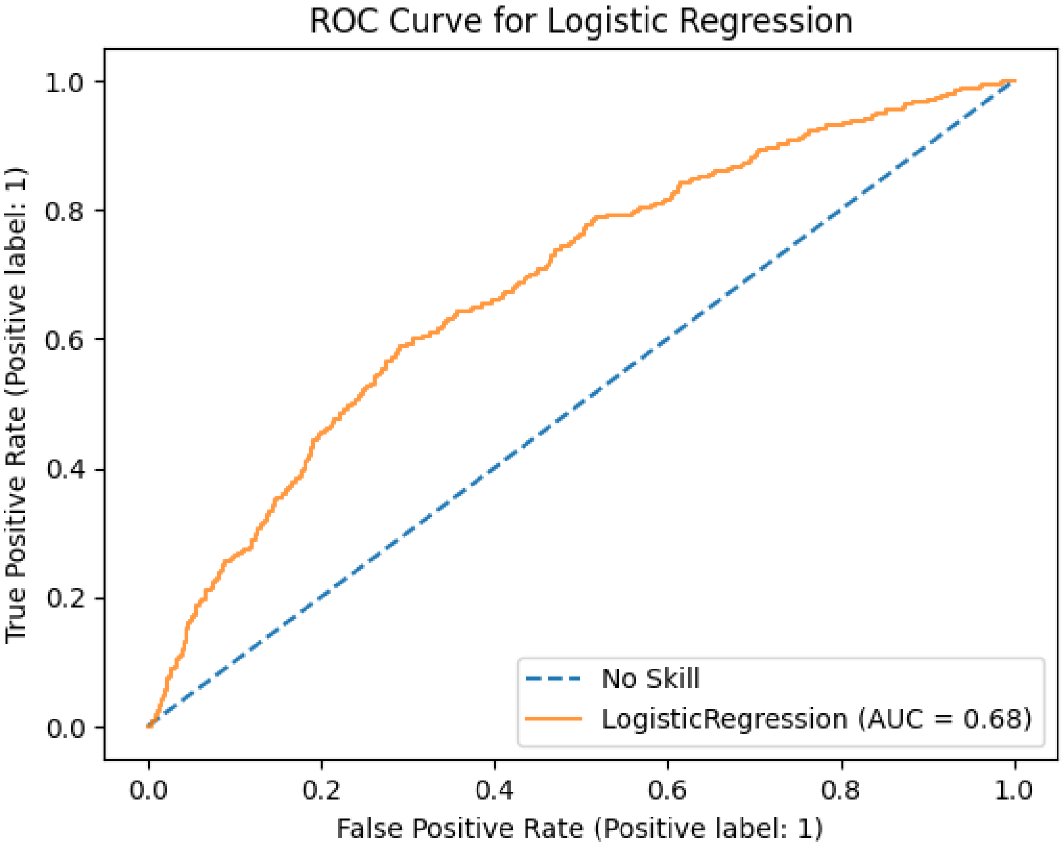
Below is a plot showing each tested regularization parameter against the average accuracy score of that particular model (using a default classification threshold of 0.5 – more on this later).

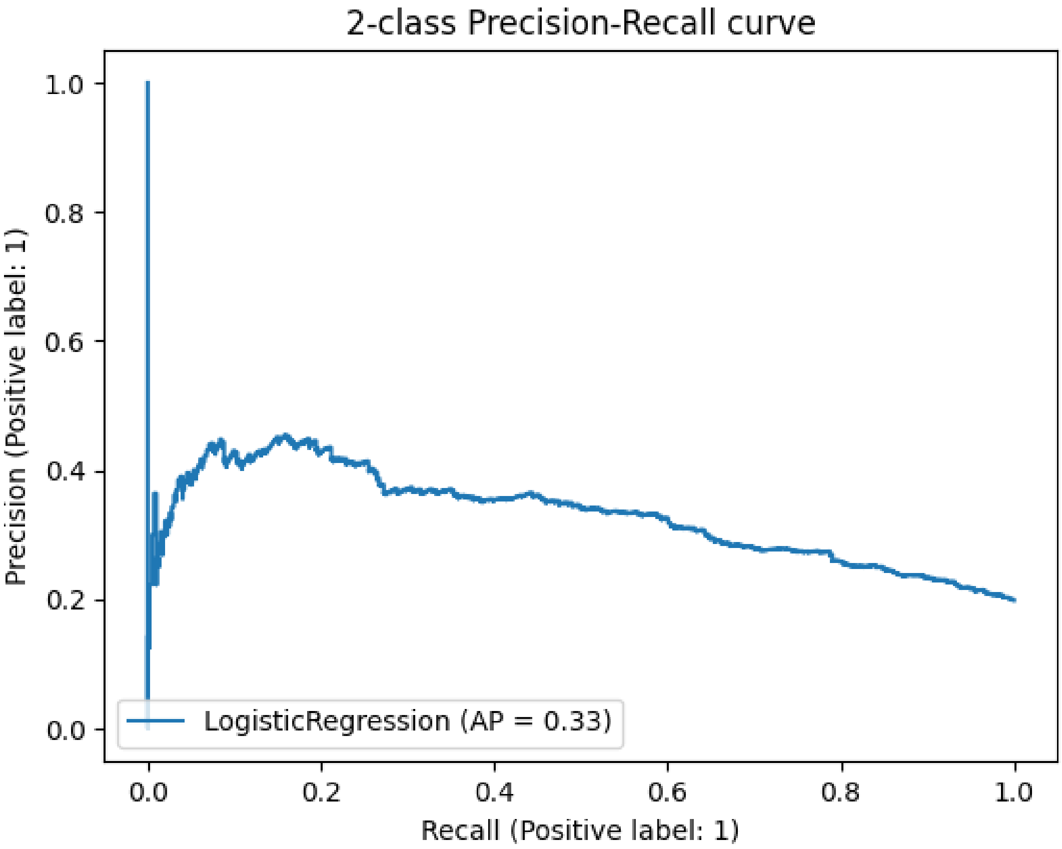

Average Accuracy Rate for Each Fold
Accuracy/Score for 0.5 Threshold
0.810
0.808
0.806
0.804
0.802
0.800
0.798
0.796
0.0
0.2
0.4
0.6
0.8
1.0
Regularization parameter



Modeling

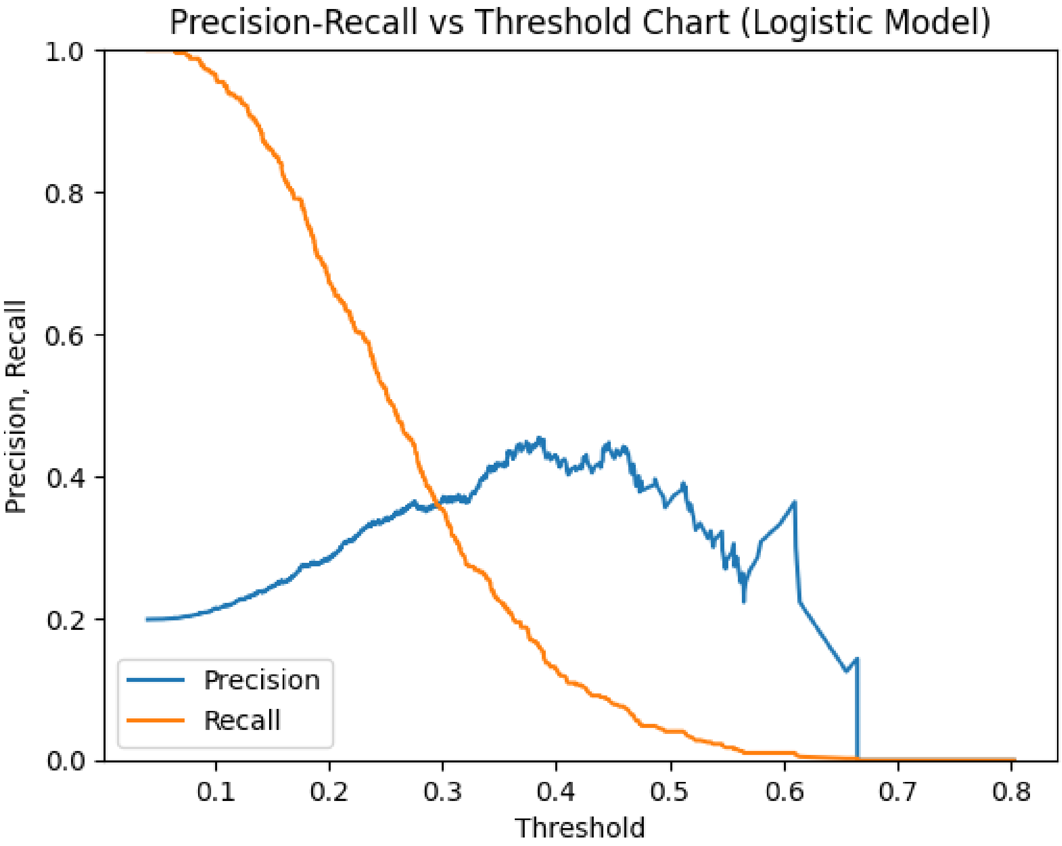
Logistic Regression


ROC Curve for Logistic Regression
True Positive Rate (Positive label: 1)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate (Positive label: 1)
No Skill
LogisticRegression (AUC = 0.68)



```
******** Logistic Regression model where threshold = 0.1 ******
Accuracy/Score is 0.291
Confusion Matrix:
[[ 200 1404]
 [  14  382]]
              precision    recall  f1-score   support

           0       0.93      0.12      0.22      1604
           1       0.21      0.96      0.35       396

    accuracy                           0.29      2000
   macro avg       0.57      0.54      0.29      2000
weighted avg       0.79      0.29      0.25      2000


******** Logistic Regression model where threshold = 0.25 ******
Accuracy/Score is 0.7045
Confusion Matrix:
[[1203  401]
```

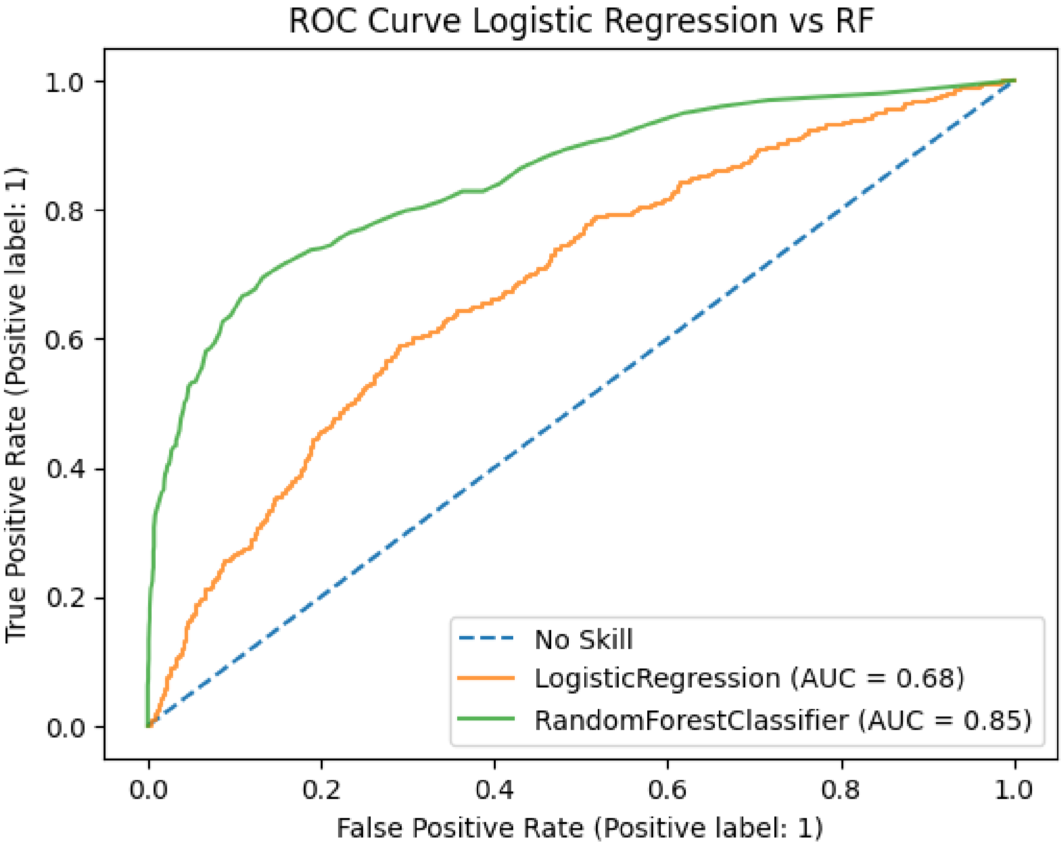
```
 [ 190  206]]
              precision    recall  f1-score   support

           0       0.86      0.75      0.80      1604
           1       0.34      0.52      0.41       396

    accuracy                           0.70      2000
   macro avg       0.60      0.64      0.61      2000
weighted avg       0.76      0.70      0.73      2000


******** Logistic Regression model where threshold = 0.3 ******
Accuracy/Score is 0.7515
Confusion Matrix:
[[1363  241]
 [ 256  140]]
              precision    recall  f1-score   support

           0       0.84      0.85      0.85      1604
           1       0.37      0.35      0.36       396

    accuracy                           0.75      2000
   macro avg       0.60      0.60      0.60      2000
weighted avg       0.75      0.75      0.75      2000


******** Logistic Regression model where threshold = 0.5 ******
Accuracy/Score is 0.7965
Confusion Matrix:
[[1577   27]
 [ 380   16]]
              precision    recall  f1-score   support

           0       0.81      0.98      0.89      1604
           1       0.37      0.04      0.07       396

    accuracy                           0.80      2000
   macro avg       0.59      0.51      0.48      2000
weighted avg       0.72      0.80      0.72      2000


******** Logistic Regression model where threshold = 0.6 ******
Accuracy/Score is 0.8005
Confusion Matrix:
[[1597    7]
 [ 392    4]]
              precision    recall  f1-score   support
           0       0.80      1.00      0.89      1604
           1       0.36      0.01      0.02       396

    accuracy                           0.80      2000
   macro avg       0.58      0.50      0.45      2000
weighted avg       0.72      0.80      0.72      2000
```

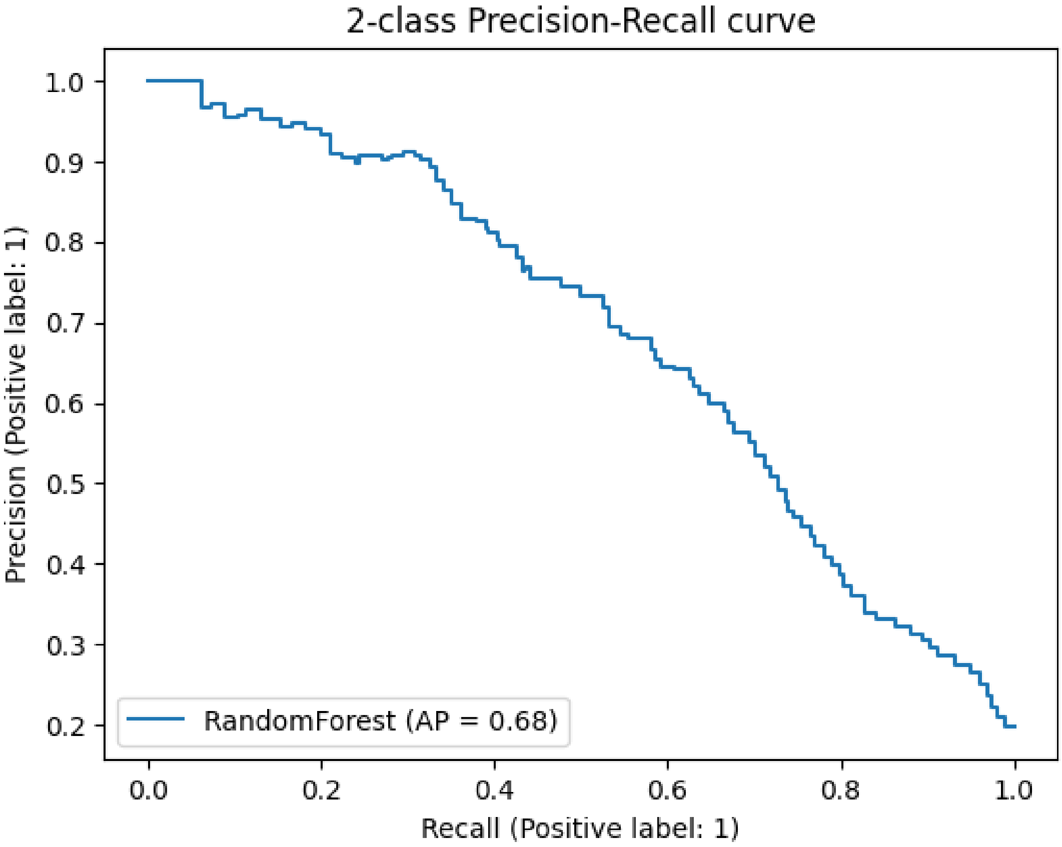


Random Forest

```
Random Forest Results
Accuracy/Score is 0.866
Confusion Matrix:
[[1537   67]
 [ 201  195]]
              precision    recall  f1-score   support

           0       0.88      0.96      0.92      1604
           1       0.74      0.49      0.59       396

    accuracy                           0.87      2000
```

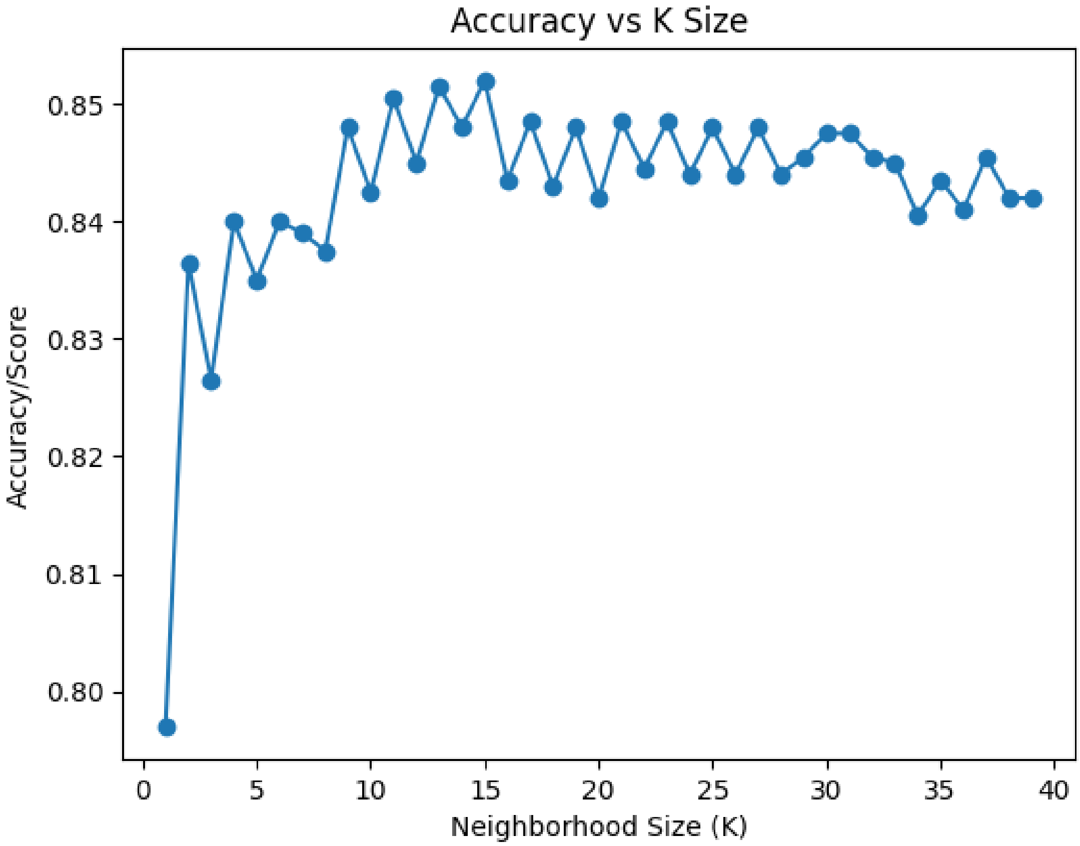
```
   macro avg       0.81      0.73      0.76      2000
weighted avg       0.86      0.87      0.86      2000
```

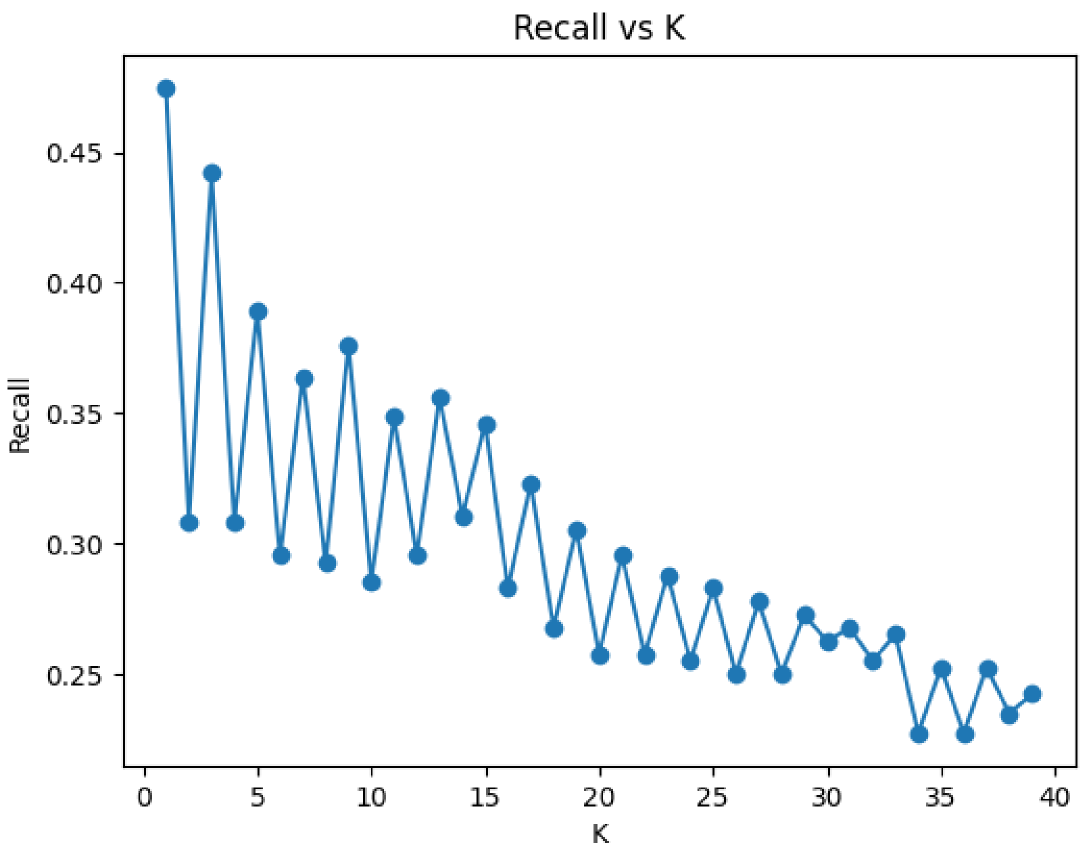




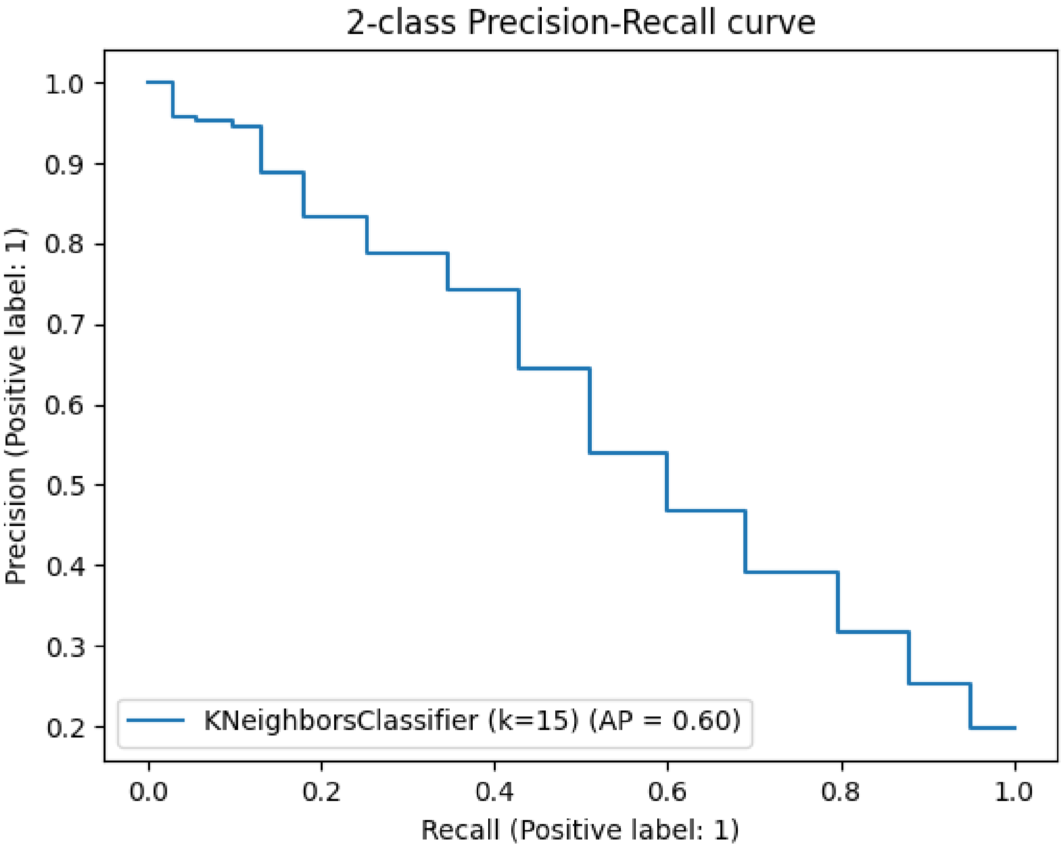
Refitting of Models

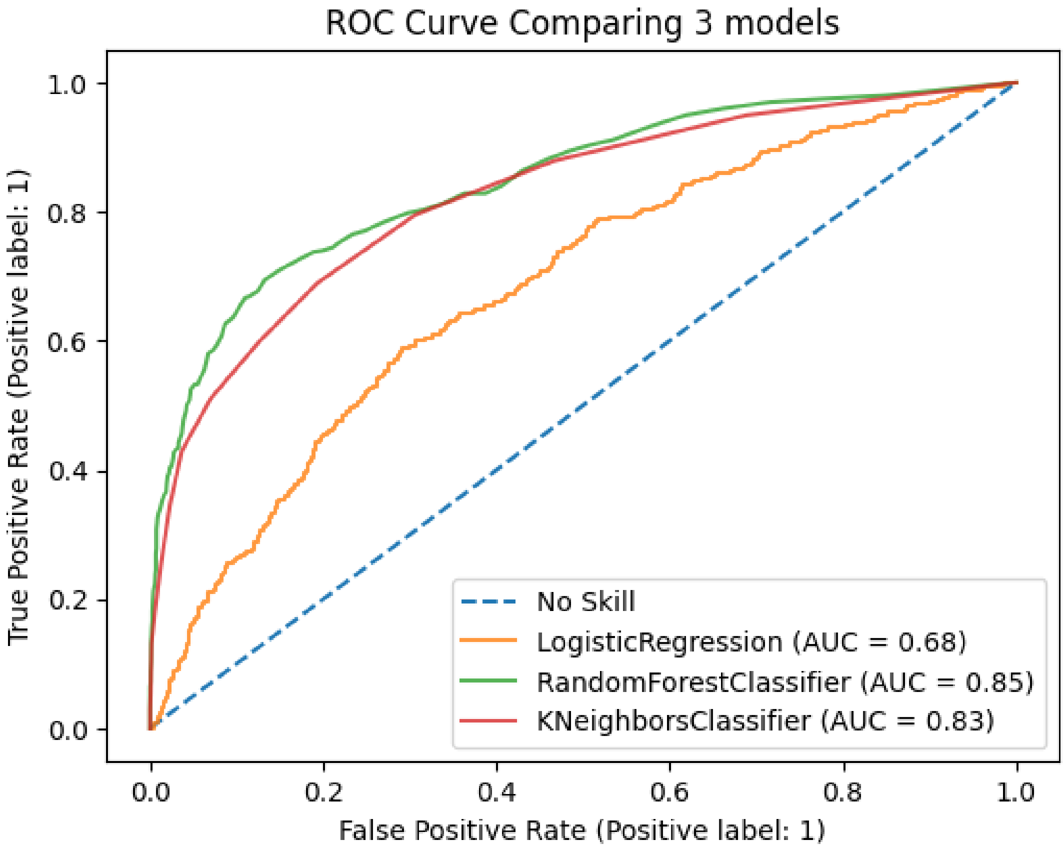
KNN Classifier


Accuracy vs K Size
Accuracy/Score
0.80
0.81
0.82
0.83
0.84
0.85
Neighborhood Size (K)
0
5
10
15
20
25
30
35
40



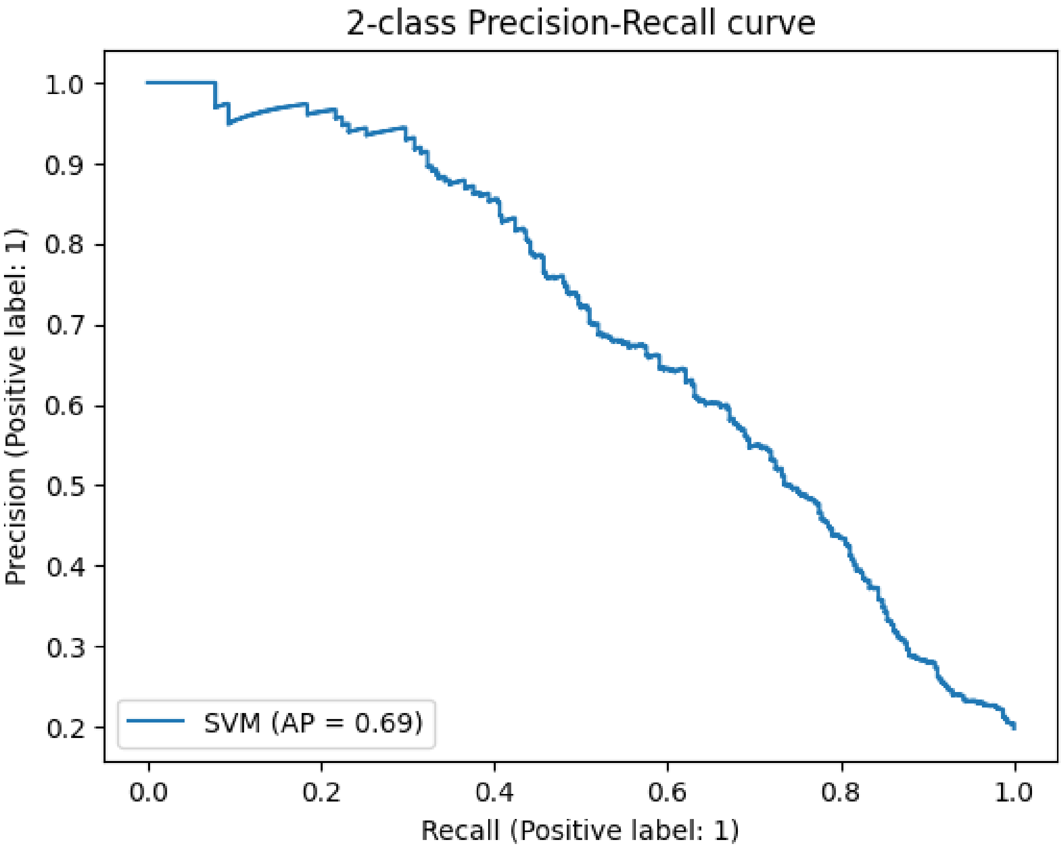
```
KNN Results when k=15
Accuracy/Score is 0.852
[[1567   37]
 [ 259  137]]
              precision    recall  f1-score   support

           0       0.86      0.98      0.91      1604
           1       0.79      0.35      0.48       396

    accuracy                           0.85      2000
   macro avg       0.82      0.66      0.70      2000
weighted avg       0.84      0.85      0.83      2000


KNN Results when k=1
Accuracy/Score is 0.797
[[1406  198]
 [ 208  188]]
              precision    recall  f1-score   support

           0       0.87      0.88      0.87      1604
           1       0.49      0.47      0.48       396

    accuracy                           0.80      2000
   macro avg       0.68      0.68      0.68      2000
weighted avg       0.80      0.80      0.80      2000
```

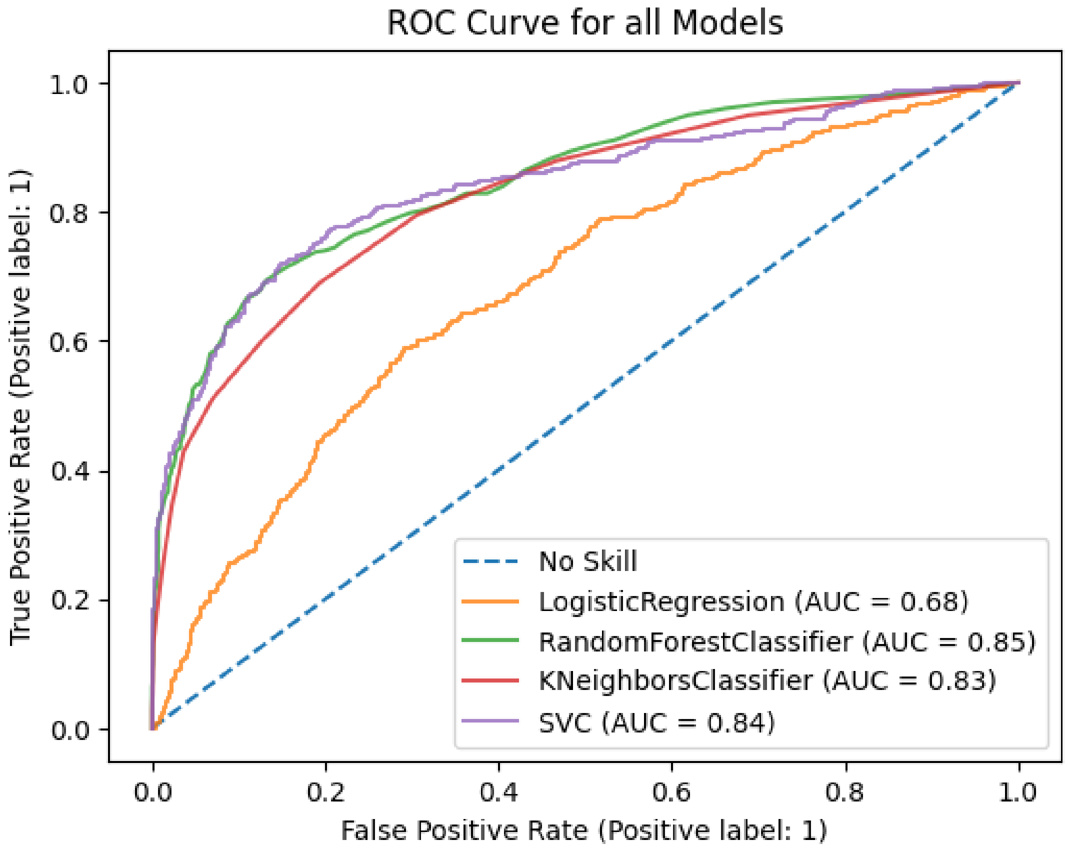




Support Vector Machine

```
SVM Results
Accuracy/Score is 0.869
[[1570   34]
 [ 228  168]]
              precision    recall  f1-score   support

           0       0.87      0.98      0.92      1604
           1       0.83      0.42      0.56       396

    accuracy                           0.87      2000
   macro avg       0.85      0.70      0.74      2000
weighted avg       0.86      0.87      0.85      2000
```


2-class Precision-Recall curve
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
Precision (Positive label: 1)
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Positive label: 1)
SVM (AP = 0.69)



Evaluation and Final Results

Conclusion

Citations

- “Bank Customer Churn”, https://www.kaggle.com/datasets/gauravtopre/bank-customer-churn-dataset
- “The Value of Keeping the Right Customers”, Amy Gallo, https://hbr.org/2014/10/the-value-of-keeping-the-right-customers