Final Report

Capstone Project: Customer Churn (E-commerce)

Shubhadeep Bhattacharya
Program: PGP DSBA
Batch: October 2019

1. Introduction

Scope and Objective of the Capstone Project

A leading e-commerce company wishes to be able to predict which of their customers is likely to churn, so as to be able to take suitable measures to prevent the customer from leaving.

We've been provided with historical data on the company's customers. The objectives of the capstone project are to mine patterns in the customer data and analyse what factors typically lead to customer churn, and build a model that will predict churn based on the important features thus identified.

Customer Churn is a problem that all businesses actively seek to address. Acquiring customers is an expensive and time consuming exercise. A customer turns profitable to a business generally over a period of time. It is hence the objective of any business to keep engaging its customers over an extended period, to be able to maximise customer life-time value. Customer churn is also an indicator of whether or not a business is successfully able to satisfy its customers. If the churn rate exceeds the industry threshold, it would indicate that the business is not performing well in delivering the value it promised to its customers.

If businesses are forewarned about which customer is likely to churn, they can take corrective measures to address the related issue, and retain their customer. If an issue is apprehended and resolved to the satisfaction of the customer, the customer is likely to stick with the business. And this study aims at helping the e-commerce company meet its customer retention objectives.

2. Exploratory Data Analysis (EDA)

- Uni-variate / Bi-variate / Multi-variate analyses to understand relationship between variables.
- Both visual and non-visual understanding of the data.

Data Overview

The dataset provided contains details of the company's customers: their attributes and behaviour on the company's e-commerce website and app. The records contain demographic details, such as gender, age, marital status, address etc. The records also reflect the customer's buying behaviour such as their preferred category, preferred login device, frequency of purchase, recency of transaction, satisfaction scores etc.

A quick survey of the data indicates that the data is recent - as many of the columns measure the customers' behaviour in the last one month. That also suggests that the business wishes to predict likelihood of churn based on fairly recent indicators of the customer's behaviour, and take corrective actions promptly to prevent customer churn. Aside from Recency, Frequency of the customers' orders is also recorded, which is an important conventional indicator of churn.

The dataset comprises records of 5,630 unique customers. And there are 20 features per record. We notice that a few categorical variables have been wrongly represented as numeric, and will have to be corrected. There's also a presence of a few missing values in the dataset, which will have to be addressed.

The features, their distribution, summary statistics, their inter-relationships and insights will be studied at length in this EDA section. We begin with studying each variable in the univariate analysis section:

Univariate Analysis

Categorical Variables

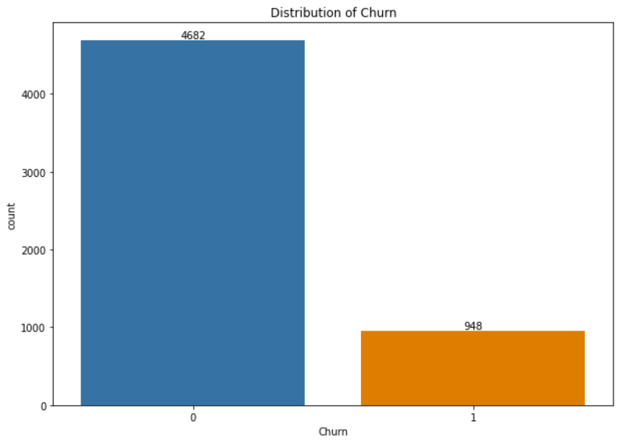
CustomerID:
The CustomerID variable is a unique label assigned to each customer record. There are 5630 unique customers in the dataset, indicating there are no duplicate records. This feature will not be used in modeling, and will be dropped presently.

Churn:

Churn is the Target Variable. We are required to build a model to predict Churn.

In the dataset, 1 indicates the customer has Churned. 0 indicated No Churn.

Around 17% of the customers in the dataset are labeled 1 for Churn. That means 83% of customers records belong to the category of No Churn (0). This suggests an imbalance in the representation of customers who are likely to Churn. This is an issue that will have to be handled at the time of building the predictive model.

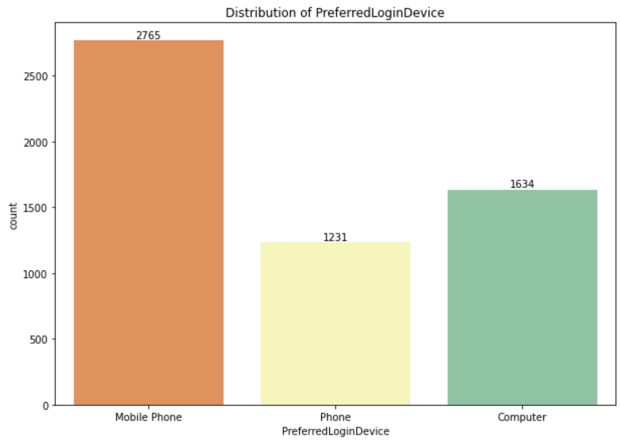


PreferredLoginDevice:
This variable indicates the device type that the customer prefers to use, to log into the company's app / website.
On initial inspection, there were 3 categories: Mobile Phone, Computer and Phone, with ~ 49%, 29% and 22% share respectively.

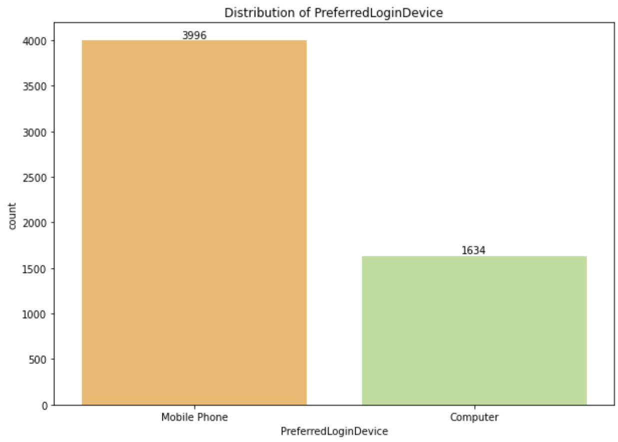
However, there's no apparent reason to have Mobile Phone and Phone as two separate categories. There is no indication that the differentiation was devised on some form of customer segmentation. So we've decided to merge both under the category 'Mobile Phone'. The consolidation resulted in 70% customers falling in the broad category that preferred to login using their phones.

Distribution before and after transformation:

Before Transformation



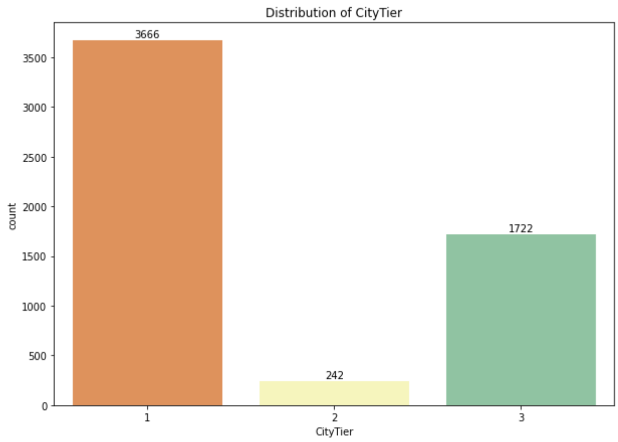
After Transformation



CityTier:

Cities have been classed as - Tier 1, Tier 2 and Tier 3. This variable indicates which Tier the customer's city falls in.

Almost two-thirds (over 65%) of customers reside in Tier 1 cities. Tier 3 accounts for 30% of the customers. Less than 5% reside in Tier 2 cities.

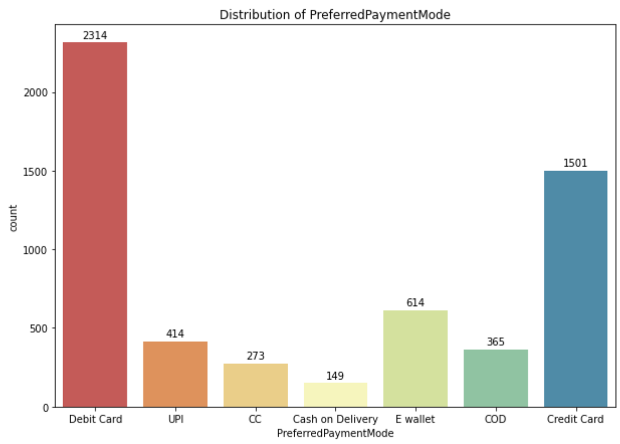


PreferredPaymentMode:
This variable indicates the preferred mode of payment for each customer. On initial inspection, there were 7 categories for this variable: Debit Card, Credit Card, E wallet, UPI, COD, CC, Cash on Delivery.

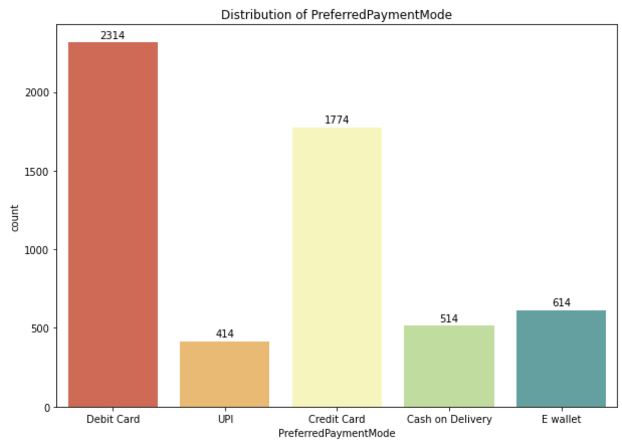
Now, it would be reasonable to assume that CC and Credit Card are the same category. And COD and Cash on Delivery also indicate the same mode of payment. So we have merged CC and Credit Card under Credit Card. And we've consolidated COD and Cash on Delivery under Cash on Delivery.

Debit Card ranks as the favoured payment mode for over 41% of the customers. The consolidated Credit Card category accounts for almost 32% of the customers.

Before Transformation



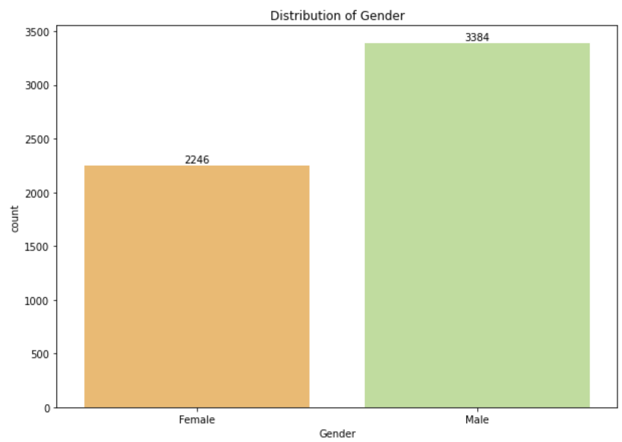
After Transformation



Gender:

This variable indicates whether the customer is male or female.

In this dataset, ~ 60% of the customers are Male, and 40% Female.

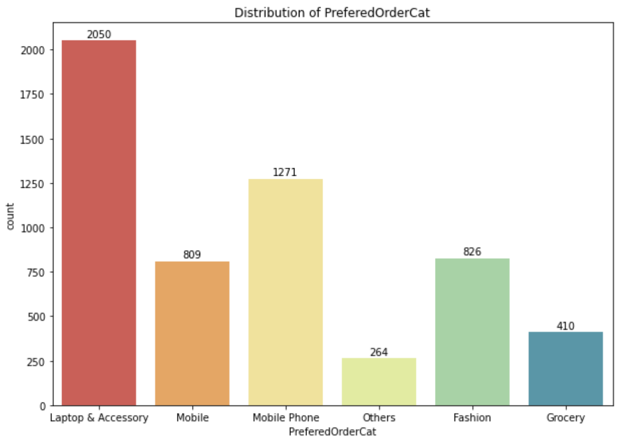


PreferedOrderCat:
This variable indicates the Category of products that the customer is most prone to buying. On first inspection, there were 6 listed categories: Laptop & Accessory, Mobile Phone, Fashion, Mobile, Grocery, Others

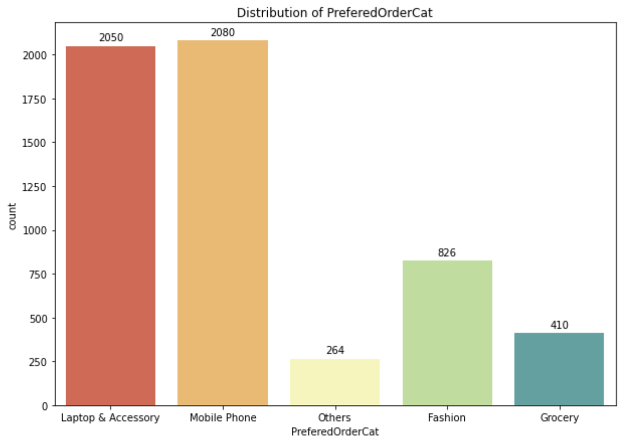
It would be reasonable to assume that Mobile Phone and Mobile indicate the same category of products. So we have merged both under Mobile Phone.

<u>With the transformation, Mobile Phone emerges as the leading category, with Laptop & Accessories a very close second with 37% and 36% share respectively</u>.

Before Transformation



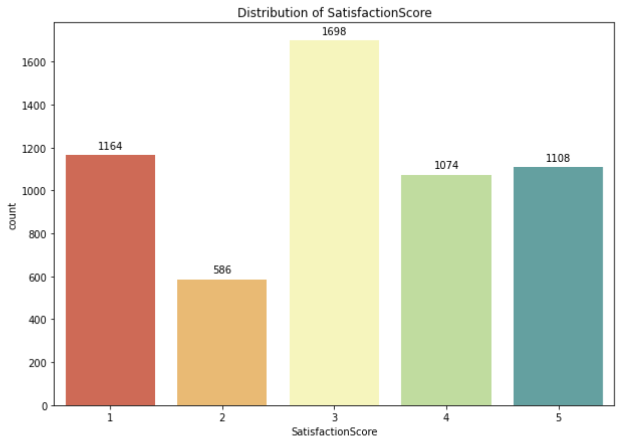
After Transformation



SatisfactionScore:

This is a score ranging from 1 to 5 that the customer has provided as a rating for his/her satisfaction level. We assume 1 implies the customer is unhappy with the service, and 5 implies that he is happy with the service.

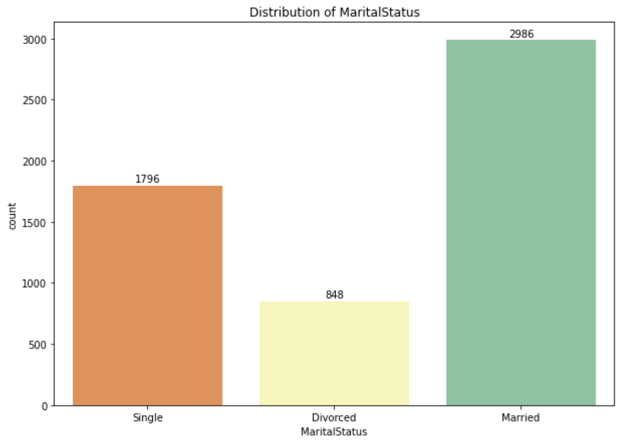
The rating of 3 or average is the largest category, with 30% customers having given this rating.
20% have given the service a score of 1, the lowest.
~ 40% of customers have given the service a score of 4 or 5.



MaritalStatus:

This variable indicates whether the customer is single, married or divorced.

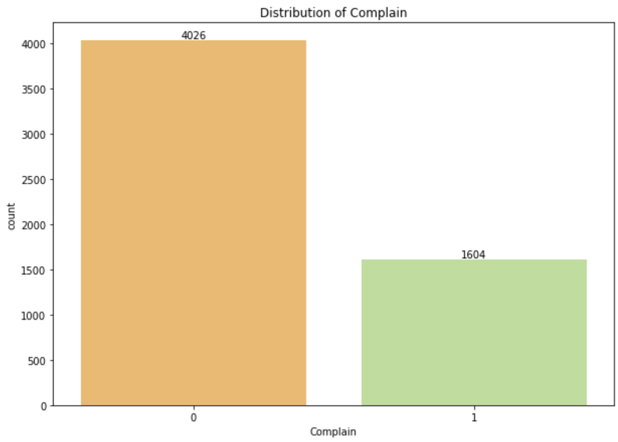
53% of the customers are married. Around 32% are single. And 15% are divorced.



Complain:

This variable indicates whether or not the customer has filed a complaint in the last one month. 1 indicates that a complaint was filed. 0 indicates no complaint was filed in the last one month.

Over 28% of the customers have raised a complaint in the last month. That is a significant proportion of customers who are dissatisfied with the service.

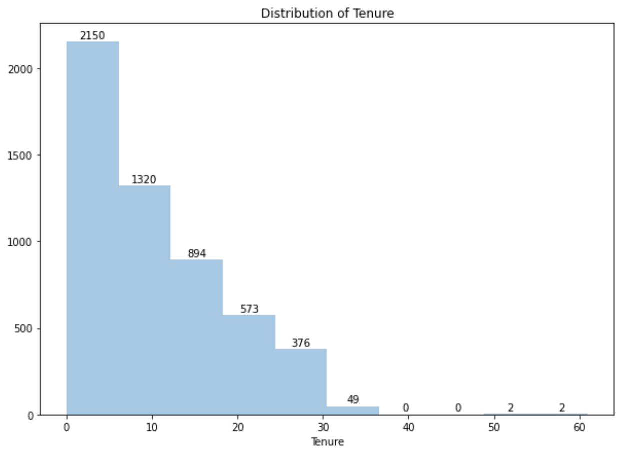


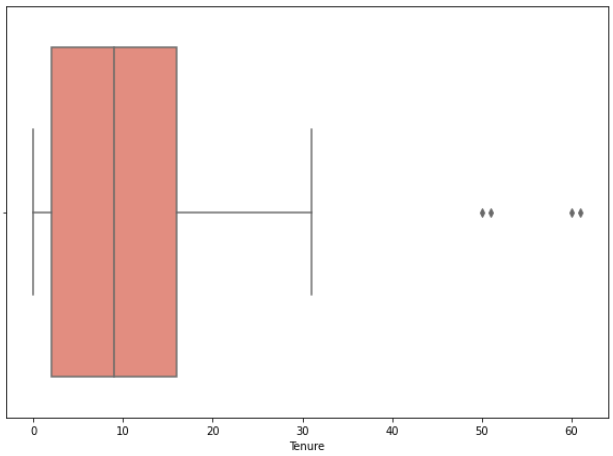
Numeric Variables

Tenure

Summary Statistics:

count	5366
mean	10.19
std	8.56
min	0
25%	2
50%	9
75%	16
max	61





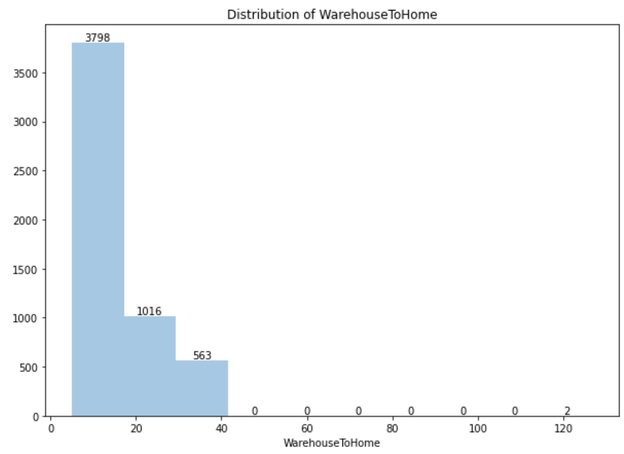
The variable indicates how long the customer has been with the company. We assume it is measured in months.
40% customers feature in the 0 to 5 month range, with over 1000 customers in the 0 to 1 month tenure. This indicates a large chunk of new customers.
50% customers have been with the company for 9 months or less.
75% of the customers have been with the company for around 16 months only.

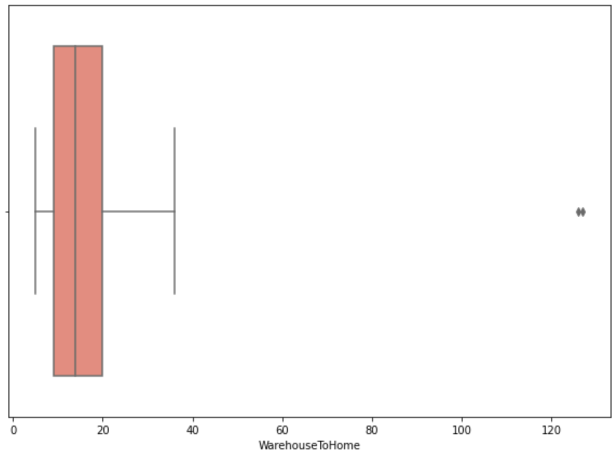
This indicates that the ecommerce service has seen significant growth in customers in the last year or so.

WarehouseToHome

Summary Statistics:

count	5379
mean	15.64
std	8.53
min	5
25%	9
50%	14
75%	20
max	127.00





This is a measure of the distance of the customer's home address to the nearest warehouse.

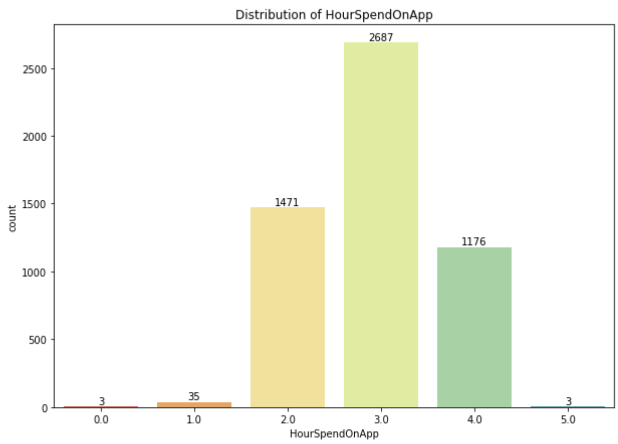
Half the customers live within 14 distance units from the nearest warehouse, and three-fourths of customers, less than or equal to 20.

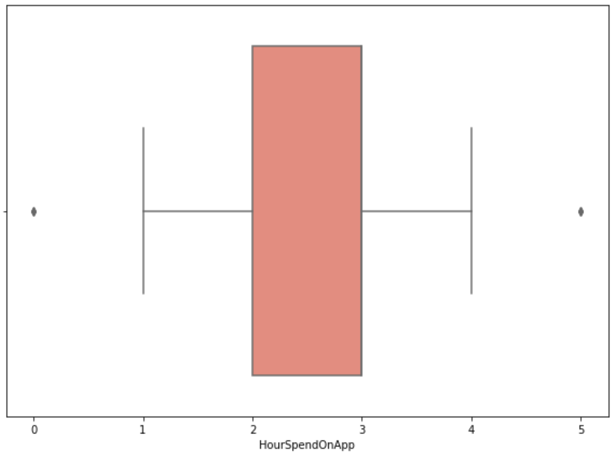
There is a couple of extreme values, for customers that live more than 120 units from the warehouse. This is largely causing the skew in the distribution.

HourSpendOnApp

Summary Statistics:

count	5375
mean	2.93
std	0.72
min	0
25%	2
50%	3
75%	3
max	5





This is a measure of the hours the customer has spent on the company's app or website.

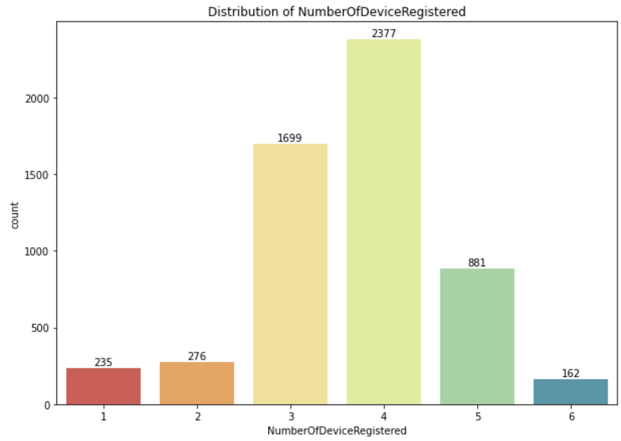
Over 99% of customers (5334) are recorded having spent between 2 to 4 hours. 3 Hours appears to be the modal observation.

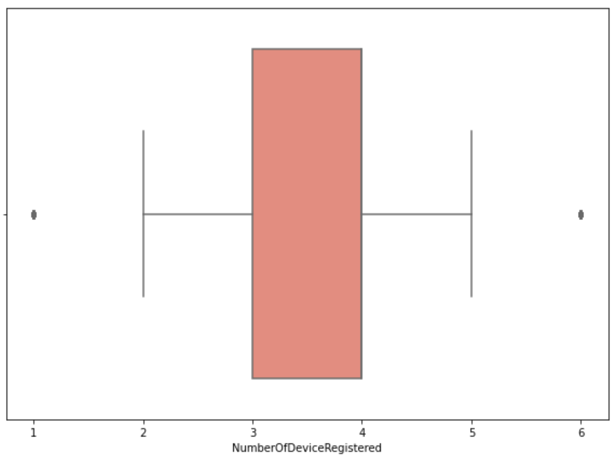
There are a few who have spent 0. We assume they qualify as less than 1 hour.

NumberOfDeviceRegistered

Summary Statistics:

count	5630
mean	3.69
std	1.02
min	1
25%	3
50%	4
75%	4
max	6





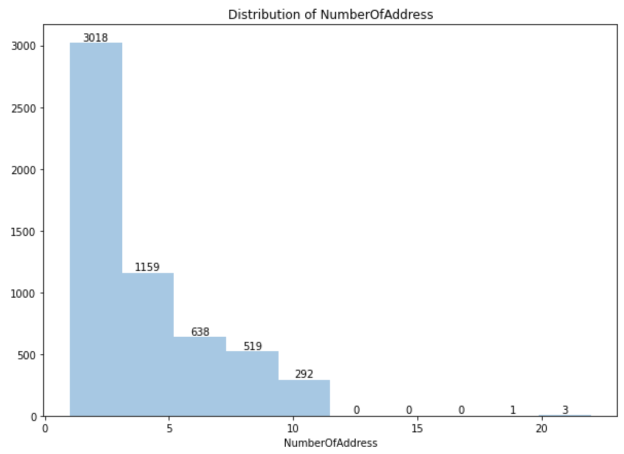
This is a count of the number of devices that has been registered by the customer to log into the company's app or website.

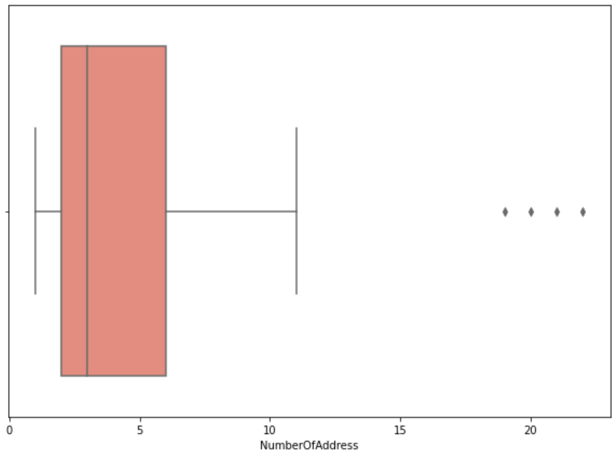
Close to 90% of the customers have 3 to 5 devices registered, 4 is the most popular number, with 42% of customers in the group.

NumberOfAddress

Summary Statistics:

count	5630
mean	4.21
std	2.58
min	1
25%	2
50%	3
75%	6
max	22





This is a count of the number of addresses a customer has registered with the service.

The median number is 3, which means 50% of customers have 3 or fewer addresses registered.

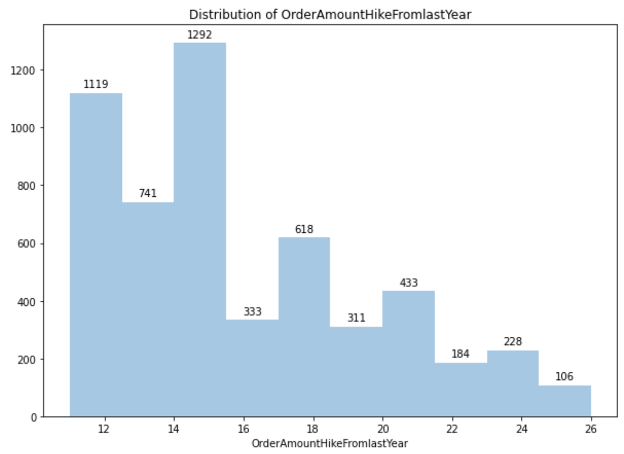
However the distribution is right skewed, given a significant number with a large number of addresses, many in the 4 to 12 addresses range.

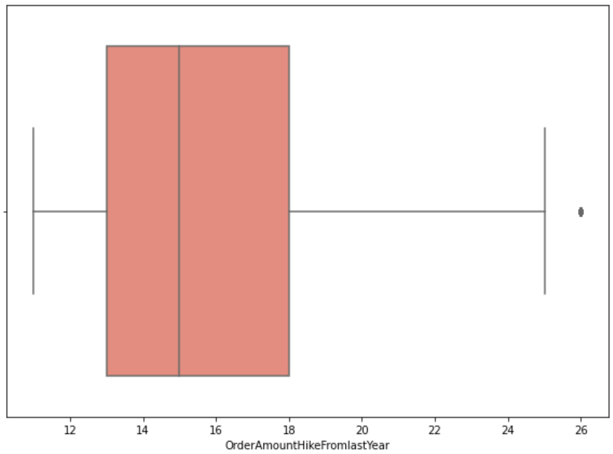
There are a few extreme values as well, around 20.

OrderAmountHikeFromlastYear

Summary Statistics:

count	5365
mean	15.71
std	3.68
min	11
0.25	13
0.50	15
0.75	18
max	26





This variable measures the increase in order value from the previous year levels.

We find that the minimum hike is 11%. And the median hike is 15%. There is a long tail, with some customers who have seen up to 26% hike in order amount.

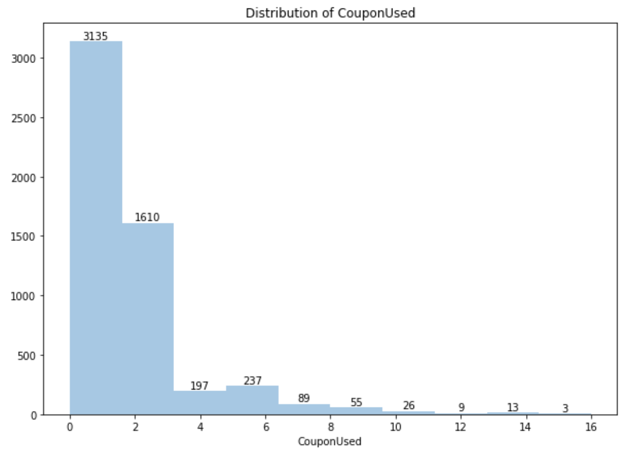
Now, there are 3288 customers with less than 12 month Tenure. And at least 500 customers with 0 tenure - hence it is difficult to reconcile how all customers have a positive order amount hike from last year.

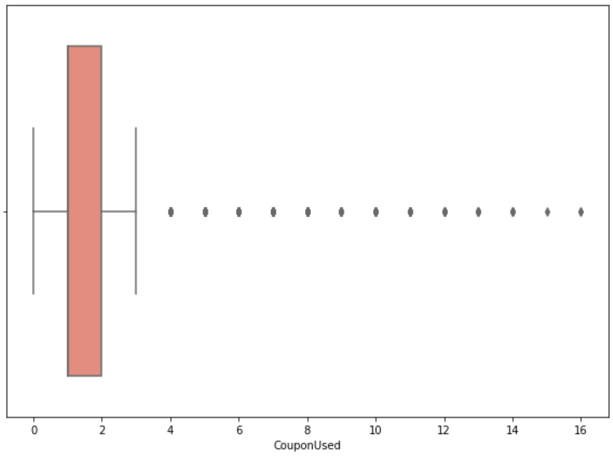
Hence we assume that this is a technical variable, computed by business to indicate the Monetary aspect of the customer's behaviour.

CouponUsed

Summary Statistics:

count	5374
mean	1.75
std	1.89
min	0
0.25	1
0.50	1
0.75	2
max	16





This is a count of coupons used by the customer in the last one month.

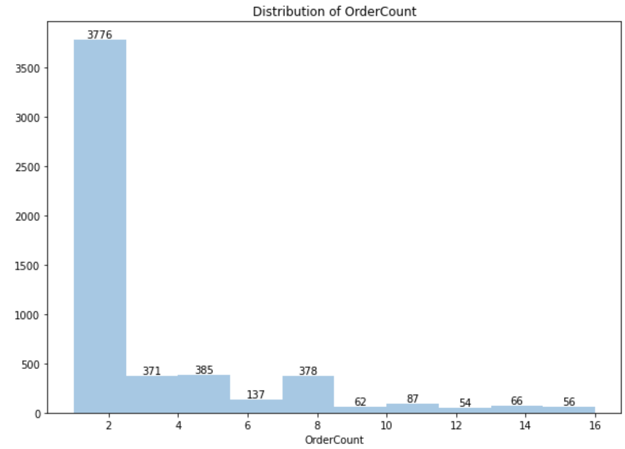
The median for CouponUsed is 1, indicating half of the customer base uses 1 or no coupons. A sizeable number, over 1200 customers, used 2 coupons in the last month.

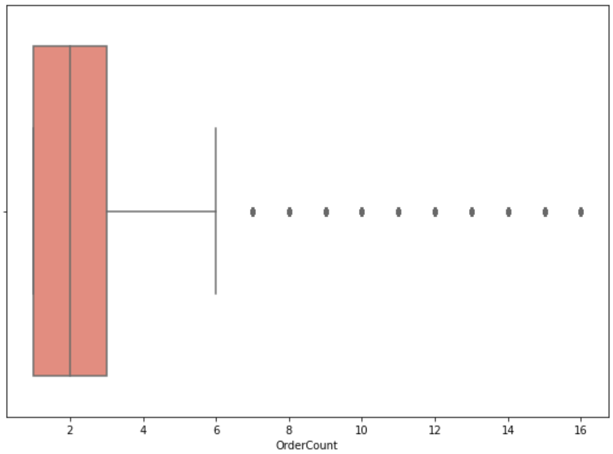
However the distribution is right skewed, given a significant number who have used multiple coupons, resulting a in a long tail, extending to some super users with up to 16 coupons.

OrderCount

Summary Statistics:

count	5372
mean	3.01
std	2.94
min	1
0.25	1
0.50	2
0.75	3
max	16





This is a count of the orders placed by the customer in the last one month.

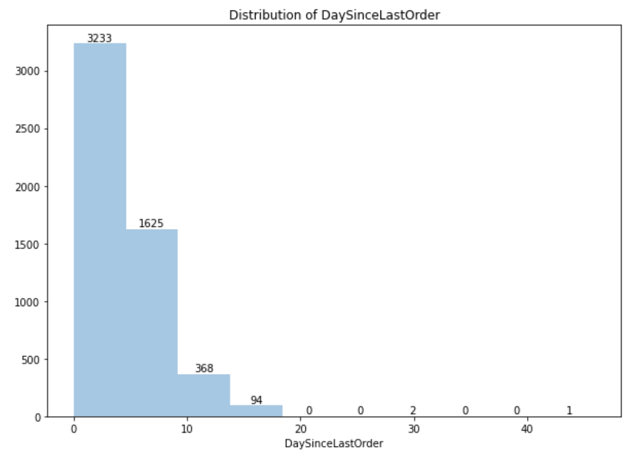
The minimum order count is 1. This indicates that all customers in the dataset have placed at least 1 order in the last month. Hence this is a set of active users.

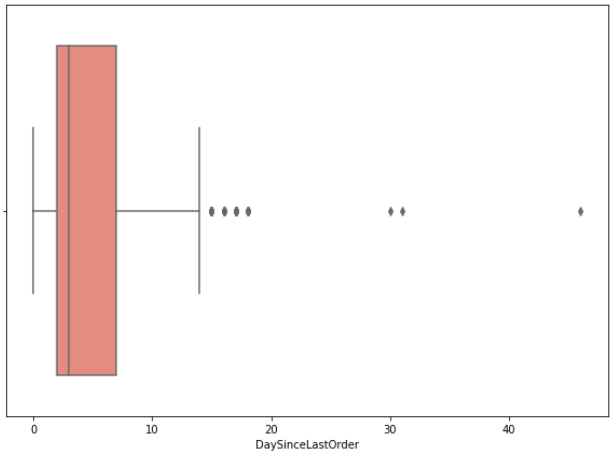
3776 customers, or around 70% of customers have placed 1 or 2 orders in the last month. Again, there is a long tail of customers who have placed larger number of orders, some even up to 16 orders in the last month. This has led to a skewed distribution.

DaySinceLastOrder

Summary Statistics:

count	5323
mean	4.54
std	3.65
min	0
0.25	2
0.50	3
0.75	7
max	46





This variable records the days since the customer last placed an order on the app / website.

The median value indicates that half of the customers have placed an order in the last 3 days. And three-fourths of the customer base have ordered in the last week.

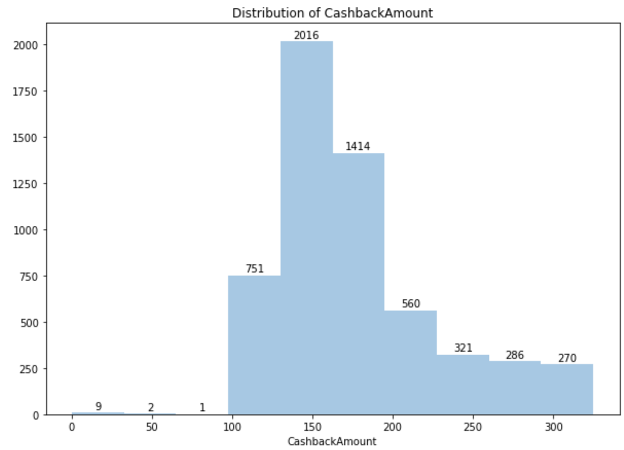
And from the histogram one sees that almost all customers have placed an order in the last 20 days/ There are only a couple of outliers beyond that threshold.

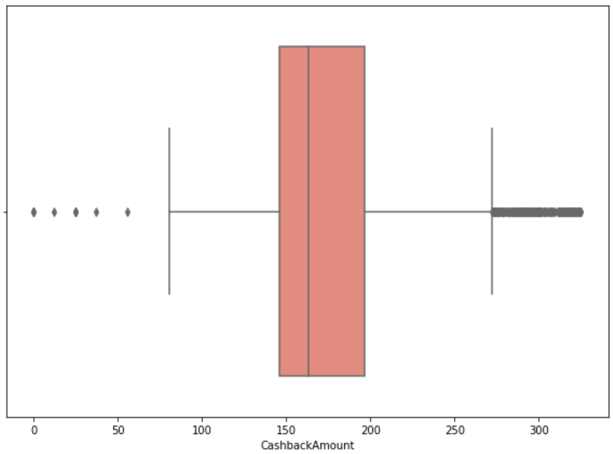
It is likely the value of 46 in erroneous, since the OrderCount variable indicates that all customers have placed an order in the last one month, so the max value should be 31 days.

CashbackAmount

Summary Statistics:

count	5630
mean	177.22
std	49.21
min	0
0.25	145
0.50	163
0.75	196
max	324





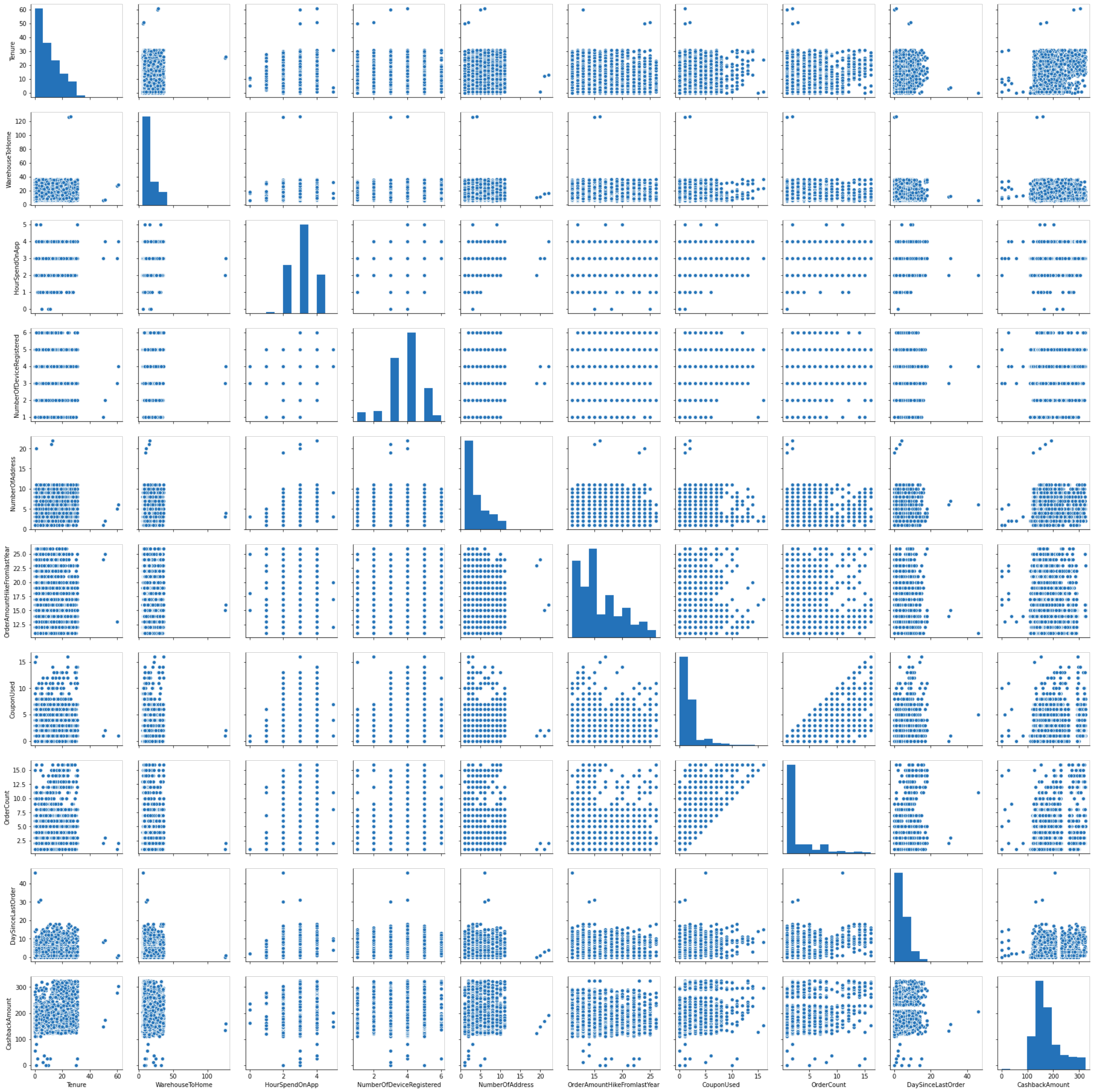
This is given as the average amount of cash back received in the last month by the customer.

Barring around 12 customers, all others appear to have received cash back ranging from around 100 to 324 units.

The median cash back amount is 163, and mean, 177. This indicates a slight right skew, with a significant number having received higher cash back amounts.

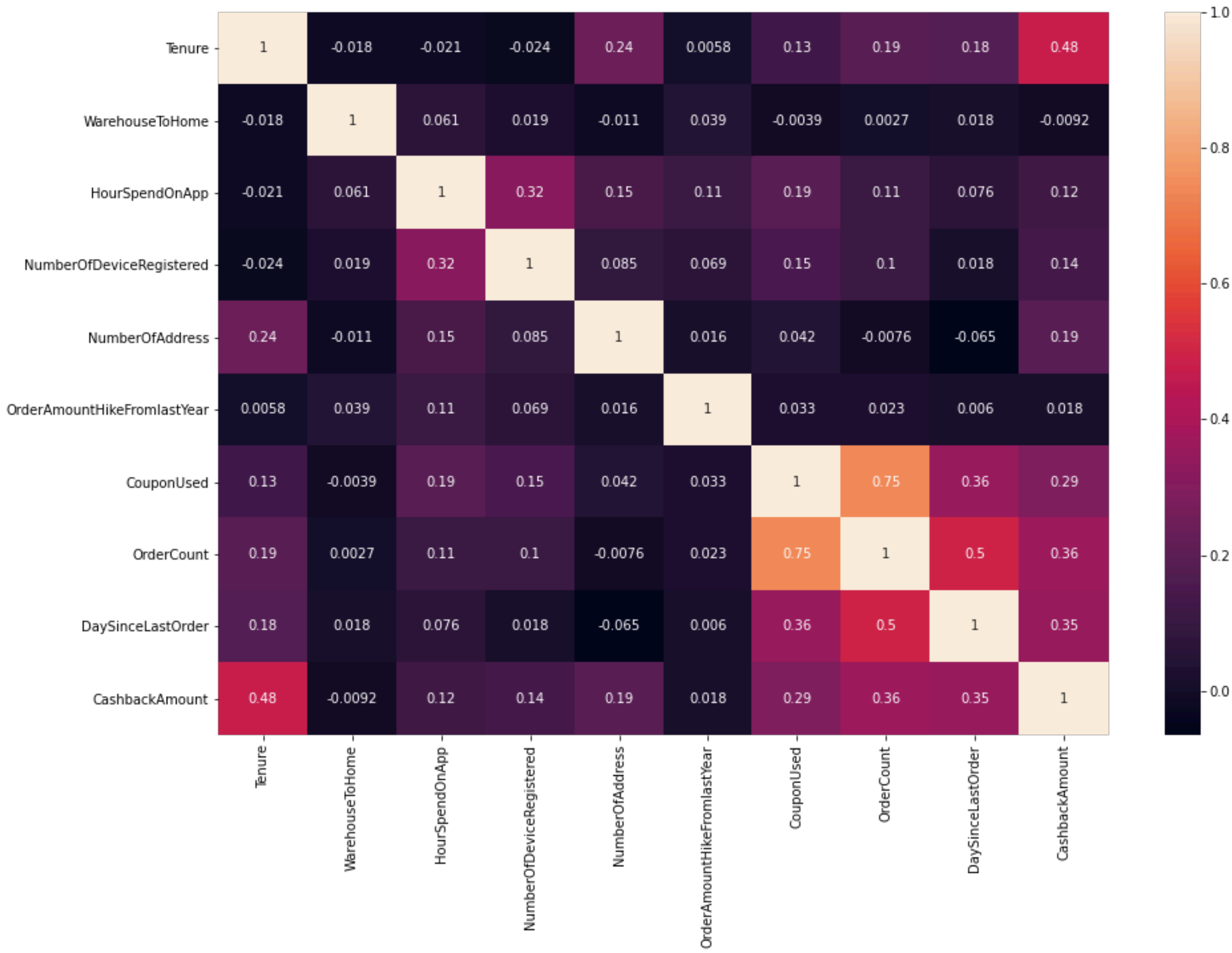
Bivariate Analysis

Assessing Correlation between Numeric Variables using a Pair Plot and Heat Map.



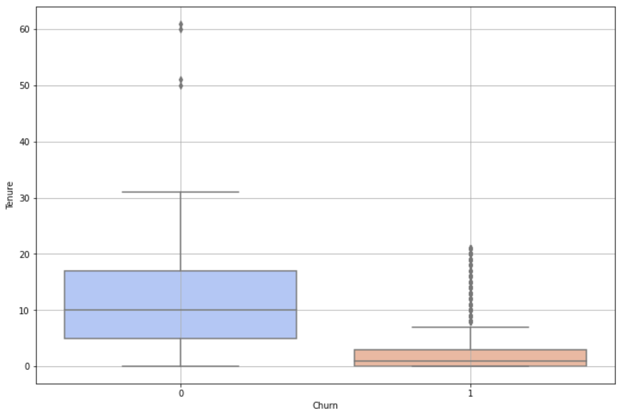
The above pair plot is a snapshot of how each numeric variable in the data set is correlated with other numeric variables. No apparent pattern is visible to suggest any strong correlation, except maybe in OrderCount and CouponsUsed. But this can be clarified using a Heat Map.

The Heat Map below indicates that among numeric variables, only CouponsUsed and OrderCount appear to have a strong correlation; rest are not significantly correlated.

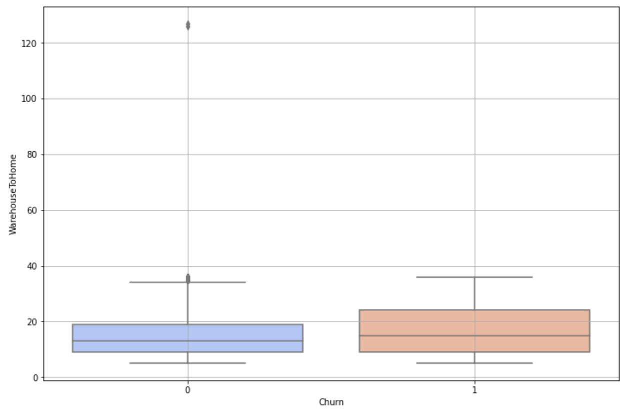


Bivariate analysis of Churn (target variable) with independent Numeric variables

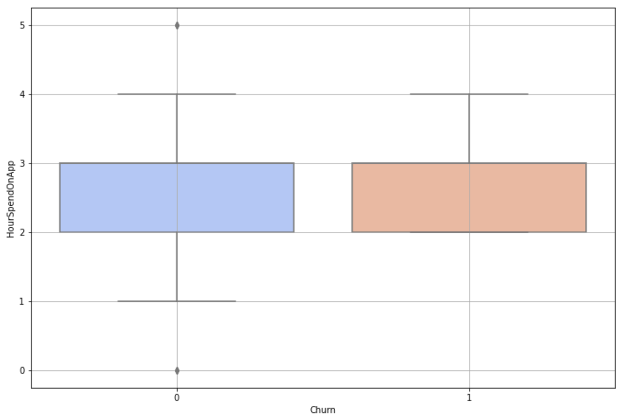
Tenure



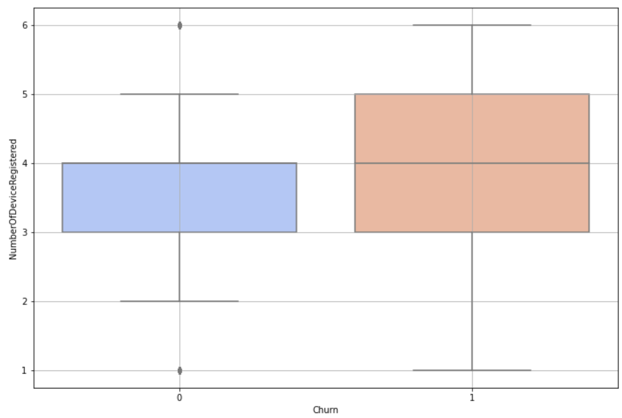
WarehouseToHome



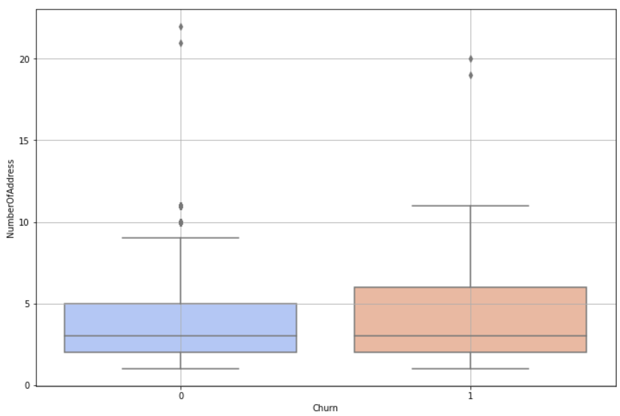
HourSpendOnApp



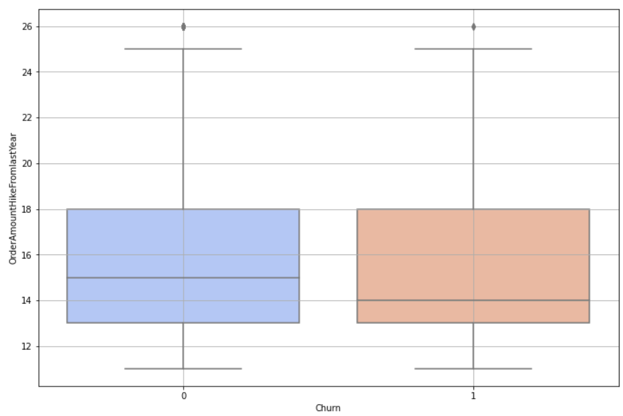
NumberOfDevicesregistered



NumberOfAddress

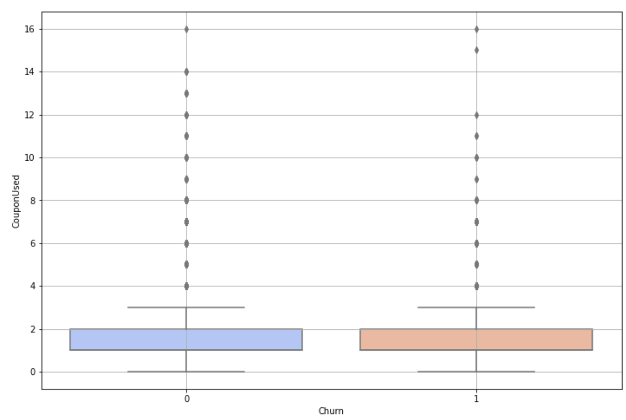


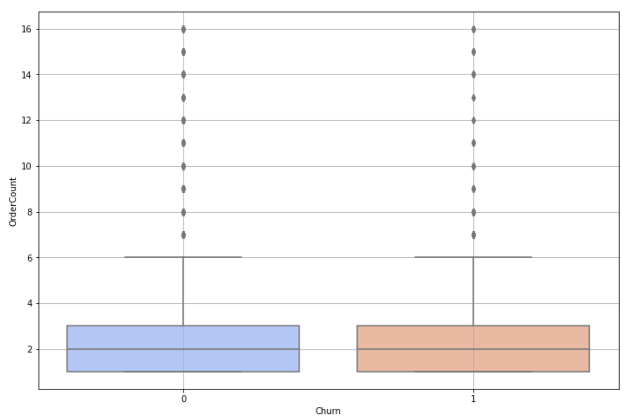
OrderAmountHikelastYear



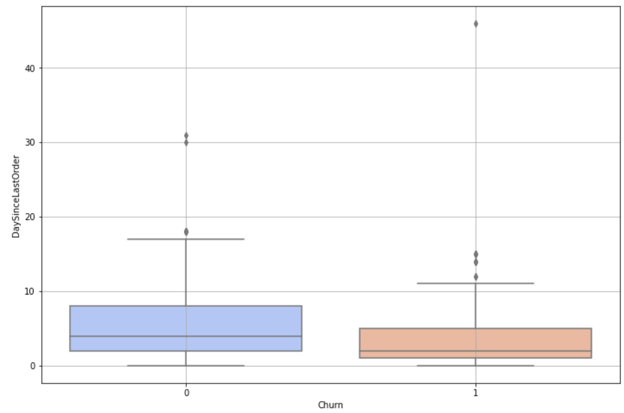
CouponUsed

OrderCount

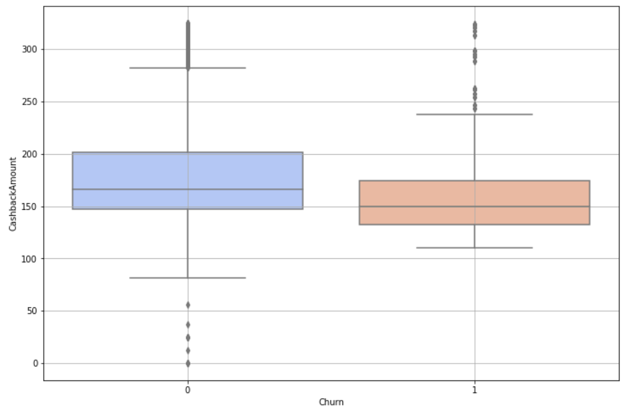




DaySinceLastOrder



CashbackAmount



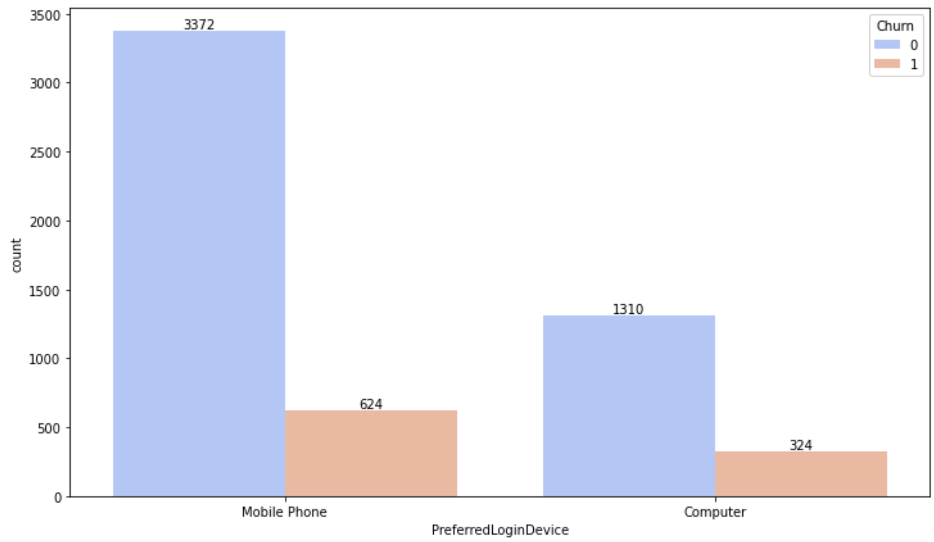
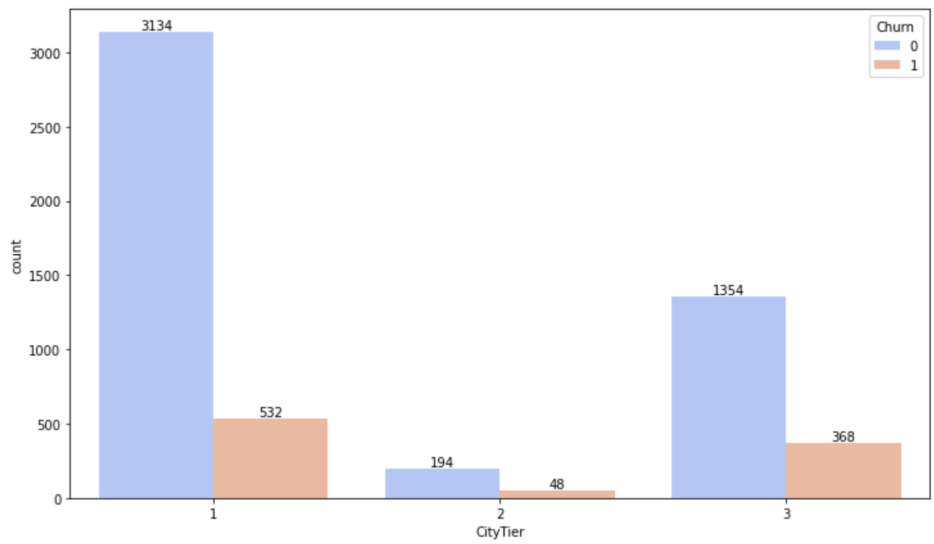
From the above box plots, two numeric variables emerge as important discriminators of whether or not a customer is likely to Churn: Tenure and DaySinceLastOrder

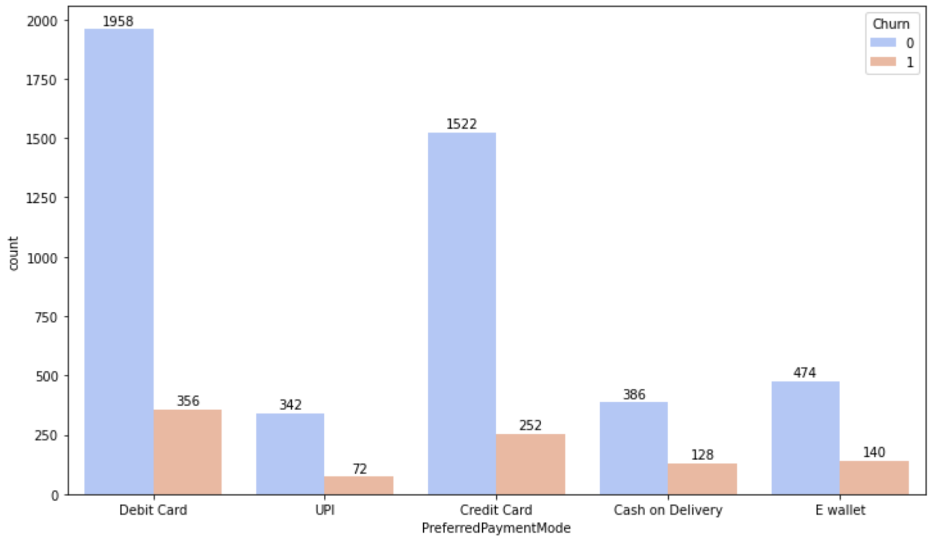
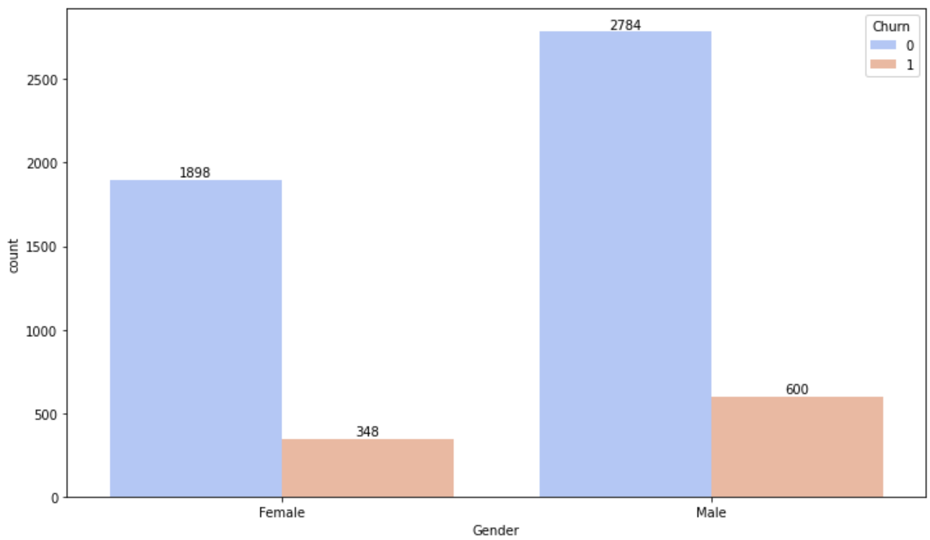
Tenure appears to be a strong discriminator for Churn. Tenure of over 8, and especially over 20 strongly increases the probability of No Churn.

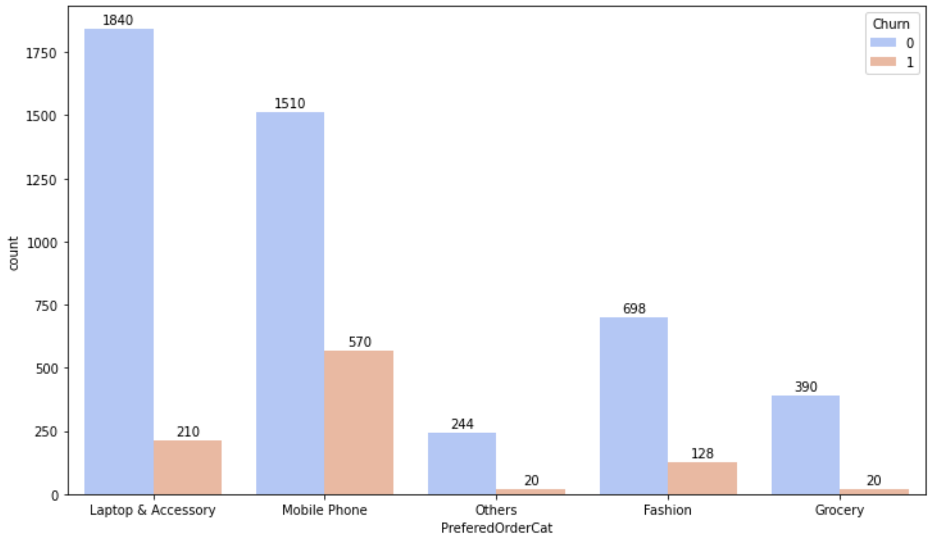
DaySinceLastOrder appears to indicate that customers that Churn have a comparatively lower median value of DaySinceLastOrder. We can see that if the DaySinceLastOrder is greater than 7, the chances of Churn are quite low.

Apart from the 2 mentioned above, CashbackAmount appears to have some discriminatory ability as well. We see that the lower levels of CashbackAmount are very likely Not Churn. And similarly the higher levels of CashbackAmount also are likely to be Not Churn. Basically, it indicates the outliers are unlikely to Churn, so not sure whether it can be as strong a predictor as the other 2 for the typical values.

Bivariate analysis of Churn (target variable) with independent Categorical variables

Variable	Observation
PreferredLoginDevice 	Customers that prefer logging in using their computers are more likely to churn than ones that prefer to log in using their phones: 20% of customers that prefer to log in using a computer, churn. 16% of customers that prefer logging in with their phone, churn.
CityTier 	Tier 1 city customers are less likely to churn, compared to Tier 2 and 3 customers. 15% of customers from Tier 1, churn. 20% of customers from Tier 2, churn. 21% customers from Tier 3, churn.
PrefferedPaymentMode	

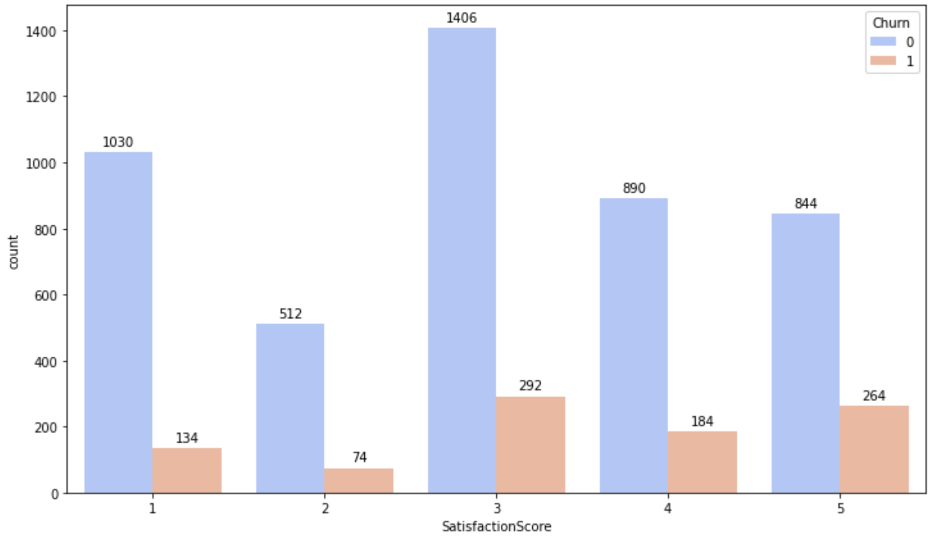
 	Customers that prefer Cash on Delivery are most likely to churn. Customers that prefer Credit Cards, the least likely. 25% of customers who prefer Cash on Delivery, churn. 14% of customers who prefer Credit Card, churn. 15% of customers who prefer Debit Card, churn. 23% of customers who prefer E-wallet, churn. 17% of customers who prefer UPI, churn.
Gender  	Men are slightly more likely to churn than Women. 15% of female customers, churn. 18% of male customers, churn
PreferedOrderCat	



Customers that prefer Mobile show significant churn rate. In contrast, customers who prefer Grocery, are least likely to churn.

15% of those who prefer Fashion, churn.
5% of this who prefer Grocery, churn.
10% of those that prefer Laptop & Accessories, churn.
27% of those that prefer Mobile Phone, churn.
7.5% of those that prefer Other categories, churn.

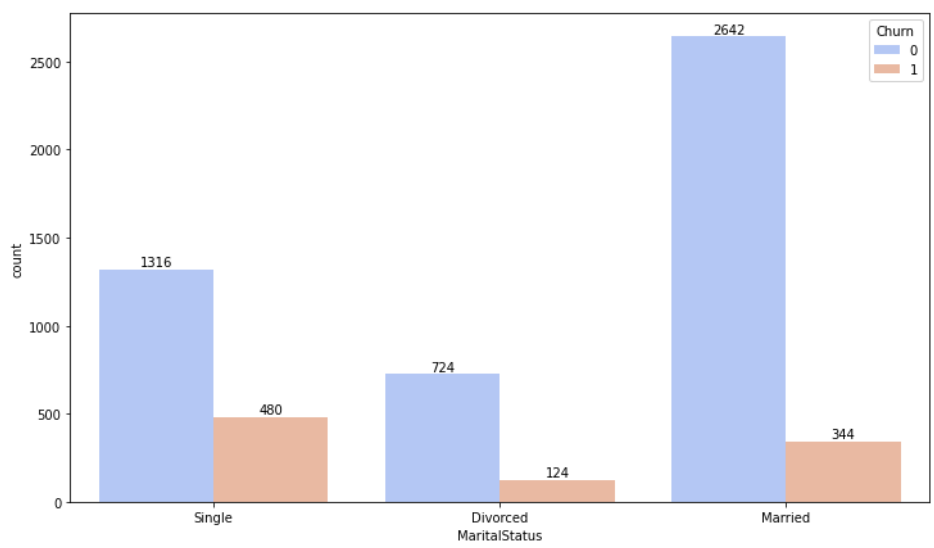
SatisfactionScore



This is a very surprising finding. The customers who give the highest rating, is most likely to churn. And the customer that gives the lowest rating is least likely to churn.

23% of Customers that rate the service at 5, churn.
17% of Customers that rate the service at 4, churn.
17% of Customers that rate the service at 3, churn.
13% of Customers that rate the service at 2, churn.
12% of Customers that rate the service at 1, churn.

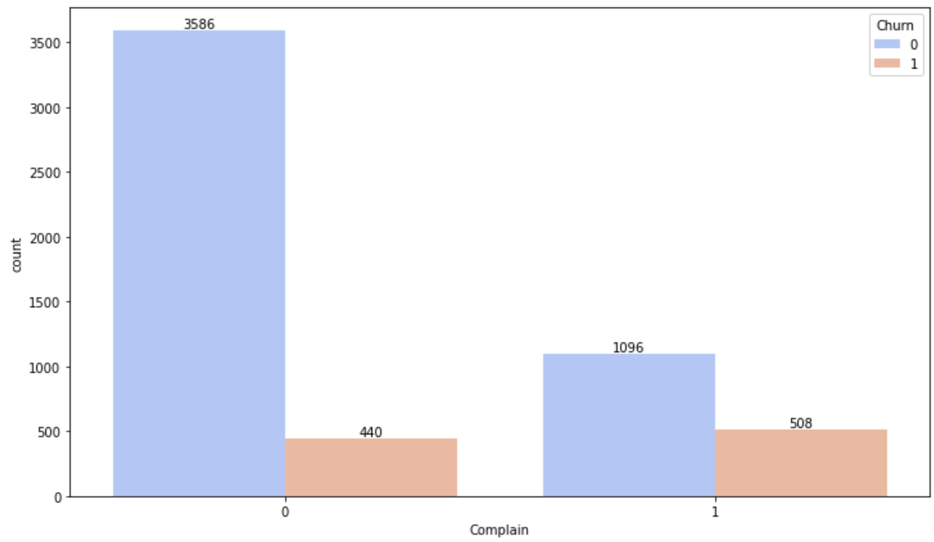
MaritalStatus



Singles are more likely to churn that Married and Divorced customers.

15% of customers that are Divorced, churn.
12% of customers that are married, churn.
27% of customers that are Single, churn.

Complain



Customers that have complained in the last one month are very likely to churn.

11% of customers who haven't complained in the last month, churn
37% of customers that complained in the last month, churn.

Summarising the key findings:

Complain is a strong indicator of Churn: 37% of customers that complained in the last month, have churned.

MaritalStatus is a strong discriminator for Churn: 27% of customers that are Single, have churned.

In the most surprising finding, customers who give the highest rating, are the most likely to churn. And the customers that give the lowest rating are least likely to churn. 23% of Customers that rate the service at 5, have churned.

PreferedOrderCat is an important indicator of likelihood of Churn: 27% of those that prefer Mobile Phone, have churned.

PreferredPaymentMode is another strong indicator of Churn: 25% of those who prefer Cash on Delivery, and 23% of those who prefer e-wallet payments, have churned.

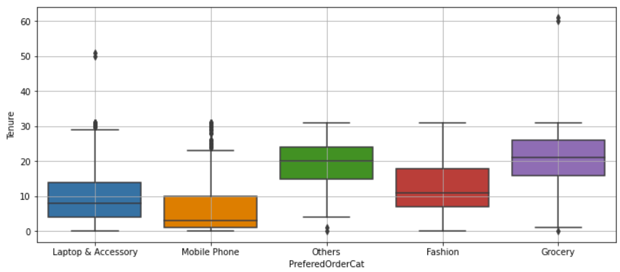
Bivariate Analysis for Missing Data imputation

There are 7 numeric columns that have missing values. We studied if there was any association between these numeric variables and the categorical variables, so that we could better impute the missing values.

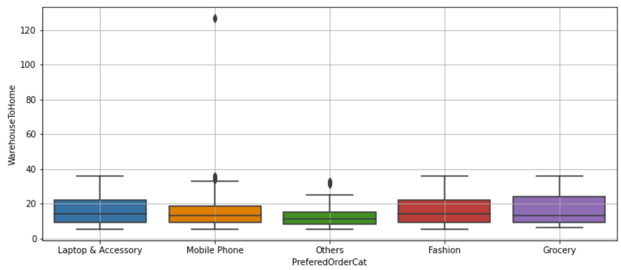
It emerged that the distribution of most of the numeric variables with missing values varied with PreferedOrderCat. Some variations were significant, while others, marginal. But among all categorical variables, PreferedOrderCat appeared most useful to segment customers.

Here are the box plots depicting the variation in the numeric variables for different classes of PreferedOrderCat.

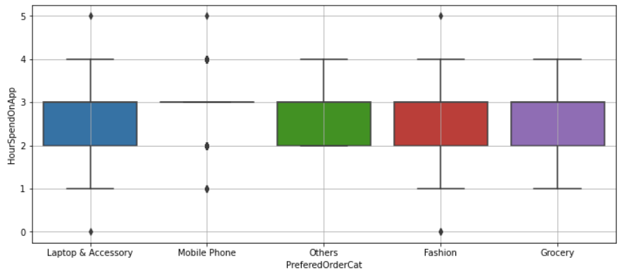
Tenure



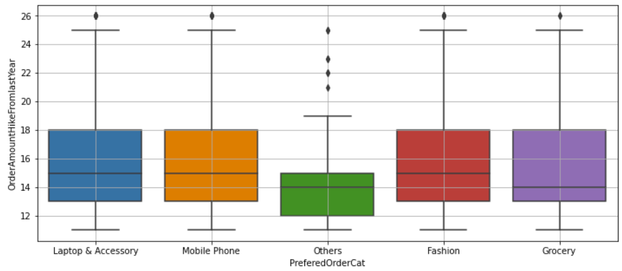
WarehouseToHome



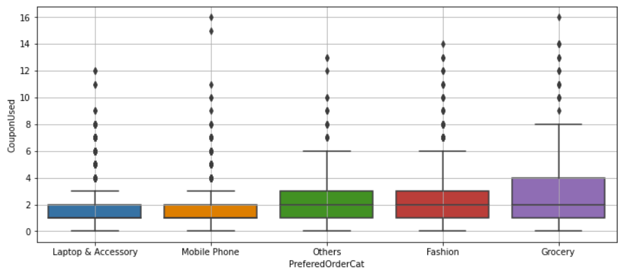
HourSpendOnApp



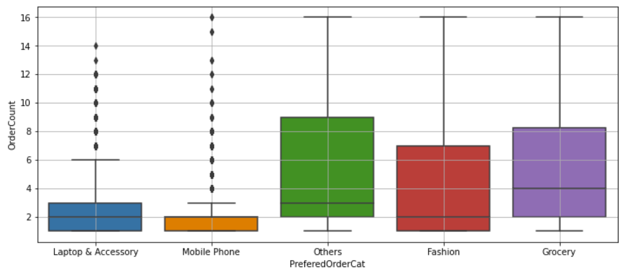
OrderAmountHikelastYear



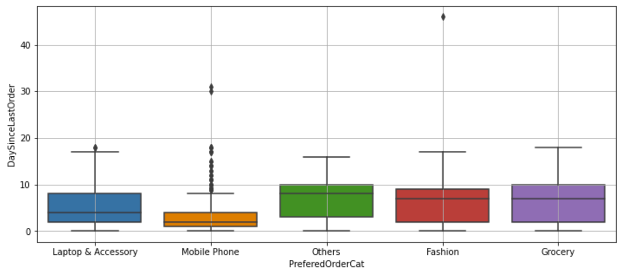
CouponUsed



OrderCount



DaySinceLastOrder



Summary of EDA Insights

From our EDA, we can summarise some patterns that indicate a comparatively higher probability of Churn. Some of the **segments that are more likely to churn than others** are:

1. People who have made a Complain in the last month: 37% of customers that complained in the last month, churn.
2. Singles: 27% of customers that are Single, churn.
3. Customers that prefer buying Mobile Phones: 27% of those that prefer Mobile Phone, churn.
4. Customers that prefer Cash on Delivery and E-wallets: ~25% of customers who prefer COD and E-wallets, churn.
5. Customers with low Tenure, typically 7 days or less.

Note: Class Imbalance in Data

Around 17% of the customers in the dataset are labeled 1: indicating Churn, and 83% of customers records are labeled 0: indicating No Churn. There is hence a substantial class imbalance between the Churn and the No Churn classes.

This imbalance is likely to affect the model's ability to predict accurately, especially for the minority class: customer who are likely to Churn (1).

To address this issue, we can consider using data augmentation techniques such as SMOTE (Synthetic Minority Oversampling Technique) to synthetically create observations for Class 1, and thus create a balanced dataset.

Another option would be to use Hyper Parameter Tuning when building our predictive model, to train the model to factor in the class imbalance.

3. Data Cleaning and Pre-processing

- Approach used for identifying and treating missing values and outlier treatment
- Need for variable transformation
- Variables removed or added

Removal of unwanted variables

We have dropped CustomerID because we won't be using it for building our predictive model.

Missing Value treatment

In the given dataset, there are 7 numeric columns that have missing values: Tenure, WarehouseToHome, HourSpendOnApp, OrderAmountHikeFromlastYear, CouponUsed, OrderCount, DaySinceLastOrder

In the EDA section, we studied if there was any association between these numeric variables and categorical variables, so that we can better impute the missing values

It emerged that most of the numeric variables with missing values vary with PreferedOrderCat. Hence PreferedOrderCat is a useful variable to segment customers: and we have used to impute missing values based on segments for each variable.

For example, if we impute missing values in DaySinceLastOrder based on the corresponding PreferedOrderCat value, we get the following range of values:

PreferedOrderCat	DaySinceLastOrder
Fashion	7
Grocery	7
Laptop & Accessory	4
Mobile Phone	2
Others	8

Only for Tenure, we have used PreferedOrderCat along with MaritalStatus to impute missing values.

PreferedOrderCat	MaritalStatus	Tenure
Fashion	Divorced	11
	Married	11
	Single	11
Grocery	Divorced	21
	Married	21
	Single	22

Laptop & Accessory	Divorced	7
	Married	9
	Single	8
Mobile Phone	Divorced	5
	Married	5
	Single	1
Others	Divorced	18.5
	Married	20
	Single	21

Outlier treatment

In the course of the EDA, we noted that the box plots of the numeric variables indicated a few values which could be classed as outliers. We identified and treated the outliers in the following manner:

- Tenure, WarehouseToHome, NumberOfAddress have a few extreme values, which are true outliers. These variables have been capped at the Q3 + (1.5 * IQR) level.

- CouponUsed and OrderCount have a long tail of large values. However, these are valid observations, and can represent a class of high users. So we have capped them at the 99 percentile levels, to maintain the integrity of the data.

- DaySinceLastOrder has a value of 46, which appears to be wrong. We have noted that all customers have placed at least one order in the last one month. Hence value for this column should not exceed 31, and this is corroborated by the data. We have hence imputed this value with the median value for the column, using segmentation based on variable PreferedOrderCat - similar to our missing value treatment.

Variable transformation

In the course of the EDA, 3 categorical variables have been modified where classes were consolidated: PreferredLoginDevice, PreferredPaymentMode, PreferedOrderCat

Missing Values were imputed, and outliers, treated.

In the section, we will transform categorical variables into numeric values to make it ready for modeling:

- We converted the data type of the following binary variables: Churn (target) and Complain, to numeric

- We converted the data type of the ordinal variable: Satisfaction Score, to numeric

- We performed One Hot Encoding for the following categorical variables, with Drop First as True: PreferredLoginDevice, CityTier, PreferredPaymentMode, Gender, PreferedOrderCat, MaritalStatus

Before we commence with modeling, we will further split the data into training and test datasets, and scale the data as a part of pre-processing. We will build and train our models using the training set, and validate it using the test set.

Data Split

The dataset is split (70:30) into training and test data sets using scikit-learn's train_test_split function. The dataset dimensions after split are as follows:

		Number of observations (rows)	Number of features (columns)
Training data	X_train	3941	26
	y_train	3941	1
Test data	X_test	1689	26
	y_test	1689	1

Scaling

- Scaling the numeric data: We used scikit-learn's **StandardScaler** library to standardise the numeric variables in the dataset.
- Note that we fit_transform the train data & only transformed the test data. Effectively we have used the means and standard deviations of the training data to transform the test dataset variables.

4. Model Building

- Model building.
- Effort to improve model performance.

The model building exercise can be categorised into 2 parts: Part 1 was building a range of models to get an overall idea of which model is better suited for the data, and the challenges involved. Part 2 of the exercise is the aspect of fine-tuning the models, using various methods, to improve model performance and build a reliable model.

Preliminary Model Building

A range of models (spanning parametric, non-parametric, kernel-based etc.) were built on the training data, to assess which model or model types are most suited for the data. The performance of these models was validated on the test data. And the performances were then tabulated and compared.

The principle metrics used to assess model performance are Recall, Precision, AUC score and Accuracy, in that particular order of importance. The choice of metrics and methods used for Validation have been discussed in the subsequent section.

Details of the models built, their performance metrics and insights have been provided in detail in the Appendix of the report. Here we summarise the performance and overall insights of Part 1 of the model building exercise:

		Recall (Churn)	**Precision (Churn)**	**Accuracy**	**AUC**
Logistic Regression	Training	0.55	0.77	0.9	0.76
	Test	0.52	0.78	0.89	0.74
Linear Discriminant Analysis	Training	0.51	0.76	0.89	0.74
	Test	0.52	0.79	0.89	0.74
K Nearest Neighbour	Training	0.77	0.95	0.96	0.88
	Test	0.54	0.79	0.9	0.76
Naive Bayes	Training	0.76	0.38	0.75	0.75
	Test	0.73	0.36	0.73	0.73
SVM	Training	0.61	0.91	0.92	0.8
	Test	0.54	0.88	0.91	0.76
Decision Tree	Training	0.69	0.84	0.93	0.83
	Test	0.57	0.7	0.88	0.76
Artificial Neural Network	Training	0.96	0.98	0.99	0.98
	Test	0.79	0.87	0.94	0.88

Inference:

- Since our objective is to Predict Churn, the Recall metric for Churn is arguably the most significant measure for us to assess the model's performance, followed by the corresponding Precision score.

- Based on the test metrics: Recall on most models (Logit, LDA, KNN, Decision Tree, SVM) range from 0.52 to 0.57. That is, around 52% to 57% of customers who actually churned were correctly identified and predicted by the models.

- The low Recall score across models is likely a consequence of the imbalance that we have noted in the dataset. Customers labeled as Churn (1) account for only 17% of the observations. This imbalance will have to be addressed when we tune our models in the next section.

- Of the above models, SVM had a comparatively high Precision (0.88), which set it apart from the rest whose precision scores were below 0.8.

- The Naive Bayes model was a striking contrast, and yielded a relatively high Recall (0.73), but had a very low Precision score (0.36).

- The Artificial Neural Network model proved to be the most promising model, and scored notably high both on Recall (0.79) and Precision (0.87).

Model Tuning, Bagging, Boosting

Model Tuning: We have noted in the last section that the imbalance in the dataset has affected the performance of many of our models. We hence tuned our models to address this imbalance, and apply appropriate hyper parameters with the aim to improve the model's performance.

We also explored Ensemble modeling (Bagging and Boosting) to improve overall model performance.

Details of the models built, their hyper-parameters, performance metrics and insights have been provided in detail in the Appendix of the report. Here we summarise the performance and overall insights of Part 2 of the model building exercise:

Model Performance Comparison:

		Recall (Churn)	Precision (Churn)	Accuracy	AUC
SVM (Tuned)	Training	1	0.9	0.98	0.99
	Test	0.91	0.8	0.94	0.93
Logit (Tuned)	Training	0.86	0.49	0.83	0.84
	Test	0.8	0.47	0.81	0.81
Artificial Neural Network (Tuned)	Training	1	1	1	1
	Test	0.85	0.93	0.96	0.92
Bagging: Random Forest	Training	0.87	0.58	0.87	0.87
	Test	0.81	0.52	0.84	0.83
Boosting: XGBoost	Training	1	1	1	1
	Test	0.89	0.88	0.96	0.93

Based on the Recall and Precision scores, the top 3 models are:

- SVM (Tuned): this model has the highest Recall score for churn at 91%. However, it scores slightly low on Precision at 80%.
- XGBoost (Tuned): this model has the second highest Recall score at 89%. This is marginally less than the score for SVM. However, the XGB model has a substantially better Precision at 88% than the SVM model's 80%. XGB also has a marginally higher AUC score than SVM.
- Artificial Neural Network (Tuned): this model ranks has the third highest Recall score at 85%, and boasts of a Precision of 93%, which is the highest among all models.

SMOTE: assorted models

To address the imbalance issue, we also tried using SMOTE (Synthetic Minority Oversampling Technique) to synthetically create observations that belong to the minority class 1.

We then built a number of models on the SMOTE training set, and the following are the Performance metrics on the Test data:

	Recall (Churn)	Precision (Churn)	Accuracy	AUC
Random Forest	0.82	0.28	0.61	0.69
XGBoost	0.67	0.70	0.89	0.81
Logit	0.79	0.47	0.81	0.80
LDA	0.81	0.45	0.79	0.80
Naive Bayes	0.54	0.28	0.72	0.64
KNN	0.52	0.20	0.60	0.57
SVM	0.57	0.24	0.66	0.62

Observations:

- We see that while SMOTE has resulted in better Recall scores for Churn across the board, it has almost in all cases affected Precision adversely. The best performing Model on Recall, RF with a score of 0.82 is also the worst performing on Precision, with a score of 0.28.
- None of the models using SMOTE offers particularly good results. The models that we have used by tuning the various hyper parameters, and using Boosting have proved to be significantly better models.
- We can conclude that SMOTE as a technique isn't particularly useful in producing better models in this instance. We will hence not include these models in the final analysis.

Modeling Summary

Of all the models built and tuned, based on the Recall and Precision scores, the following 3 models have proved to be the best predictors of customer churn in this case:

- Support Vector Machine (tuned)
- XGBoost (tuned)
- Artificial Neural Network (tuned)

5. Model Validation

- How was the model validated?

The performance of these models was validated on the test data. The principle metrics used to assess model performance are Recall, Precision, AUC score and Accuracy, in that particular order of importance.

Since our objective is to Predict Churn, the Recall metric for Churn is arguably the most significant measure for us to assess the model's performance, followed by the corresponding Precision score.

- Recall measures the proportion of actual churn that was identified correctly.
- Precision measures the proportion of churn identifications that was actually correct.

And given that our dataset is imbalanced, Accuracy will not serve as the ideal metric to evaluate the model's performance. That said, Accuracy and AUC are important indicators of the overall quality of the model.

The following are the performance metrics for the top 3 models as determined in the earlier section:

	Recall (Churn)	**Precision (Churn)**	**Accuracy**	**AUC**
SVM (Tuned)	0.91	0.8	0.94	0.93
Boosting: XGBoost	0.89	0.88	0.96	0.93
Artificial Neural Network (Tuned)	0.85	0.93	0.96	0.92

To further test the reliability of the top 3 models, we get a cross validation score for each model on the entire dataset. The mean accuracy score and standard deviation obtained on cross validation are as follows:

Model	**CV Mean Accuracy**	**CV Standard Deviation**
XGBoost	0.99	0.005
Artificial Neural Network	0.98	0.01
SVM	0.97	0.007

Of the three models, ANN shows a marginally higher variance compared to the other two.
Since accuracy is not the measure that we are using to rank models in this case, we use the cross validation score largely to corroborate the model's performance as measured by the Recall and Precision values for Churn, and also the AUC score.

We hence conclude that the following 2 models have proved to be the best predictors of customer churn in this case:

1. Support Vector Machine (tuned)
2. XGBoost (tuned)

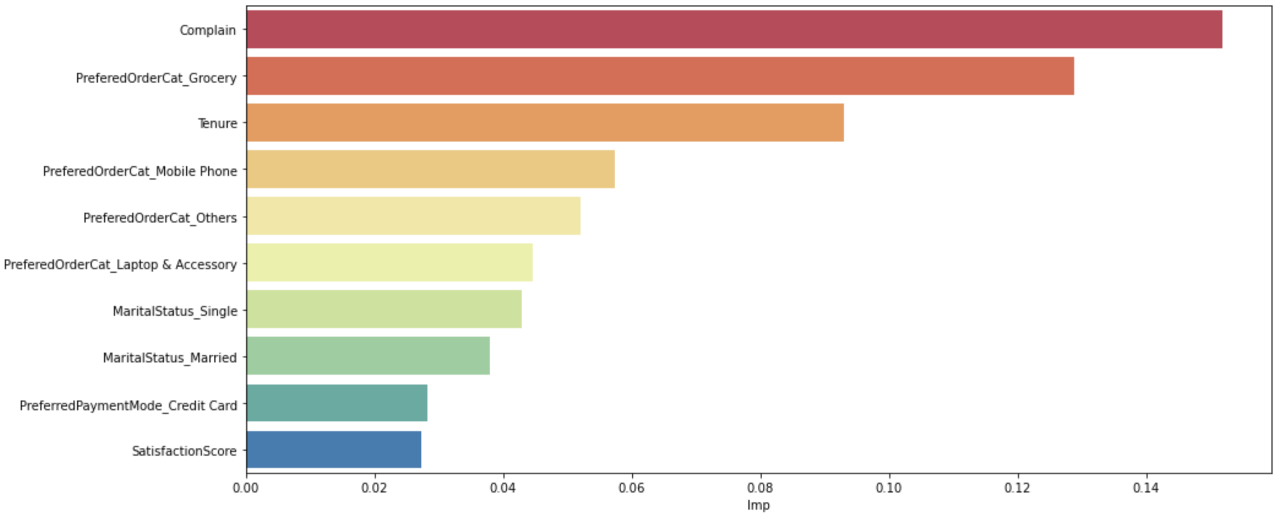
6. Final Interpretation and Recommendations

Revisiting the broad insights from EDA, where we saw that the customer segments that are more likely to churn than others are:

- People who have made a Complain in the last month
- Singles
- Customers that prefer buying Mobile Phones
- Customers that prefer Cash on Delivery and E-wallets
- Customers with low Tenure, typically 7 days or less.

<u>Insights from our models:</u>
From our modeling section, we concluded that the top 2 models are SVM (Tuned), and XGBoost (Tuned). We have used the 'rbf' kernel for the SVM model, and hence we won't know how the coefficients have been weighted. But we get an idea of the important features from our XGBoost model.

The chart below lists the <u>top 10 features and their corresponding weights, as determined by the XGBoost model</u>:



This <u>corroborates largely how our Linear models also weighted their important features</u> (See Appendix), and what we inferred in the course of our EDA.

<u>Complain, Tenure, Preferred Order Category, Marital Status, and Preferred Payment mode are by and large important discriminators of customers that churn and those that don't.</u>

Based on the above insights, the following are the recommendations to the e-commerce business, to help address the issue of churn:

Recommendations:

COMPLAINT REDRESSAL as an opportunity:

- Complain appears to be the biggest signifier of Churn. Close to 40% of customers that placed a complaint in the last month, churned.
- So the immediate opportunity would be to ensure resolving the customer's complaint to their satisfaction and preferably, delight, and hence giving them a reason to continue on the platform.
- Also, feedback from the complaints need to be actioned to improve service.

MOBILE PHONE CATEGORY as an area of focus:

- Customers that prefer buying mobile phones are the largest segment from a preferred category standpoint.
- This is also the segment that has a high probability of churn (~ 30%). Mobile phone buyers are possibly loyal to whichever platform offers them the best deals.
- Given that this segment forms a significant portion of customers, special focus to be given to the category itself, to induce them to prefer this platform over others.

SINGLES SEGMENT as an area of focus:

- Singles are twice as likely to churn than married and divorced customers.
- Perhaps the offering on the platform caters more to non-singles, and it would help to explore if offerings can be tailored keeping Singles in mind.

TENURE EXTENSION as a strategy:

- Ultimately if the above measures and the quality of service in general, succeed in keeping customers engaged on the platform for longer than 8 months to a year, the possibility of churn is significantly lowered.
- Hence a customer needs to be particularly attended to and engaged with in the initial months on the platform, to effect their continued association.

7. Appendix

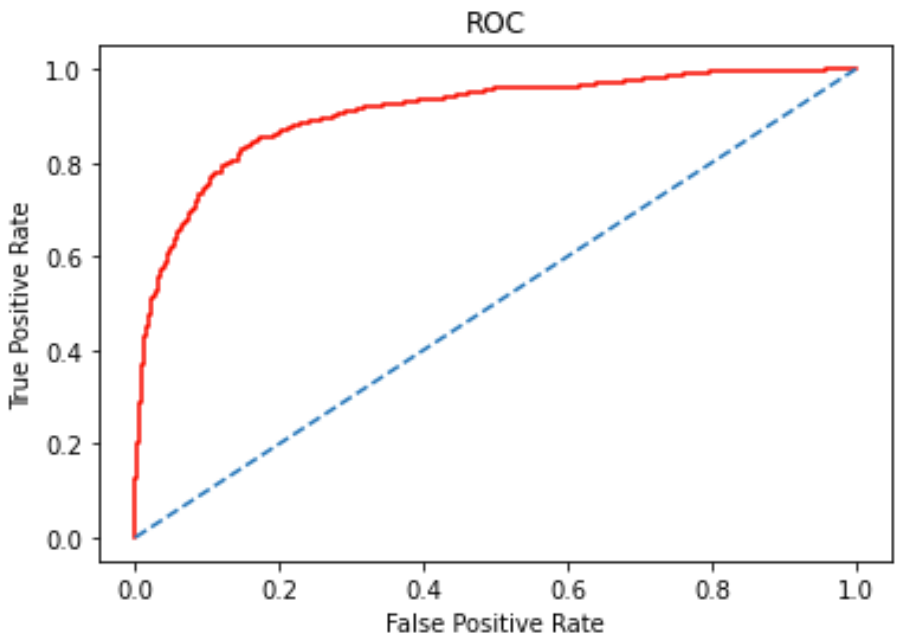
Preliminary Model building

This section details out the model performance, insights and hyper parameter tuning details of each model built for the exercise.

Logistic Regression

We used scikit-learn library's LogisticRegression algorithm to build the logistic regression model.

Logit Model's performance on the Training dataset:

Accuracy	0.90
AUC Score	0.76
ROC Curve	

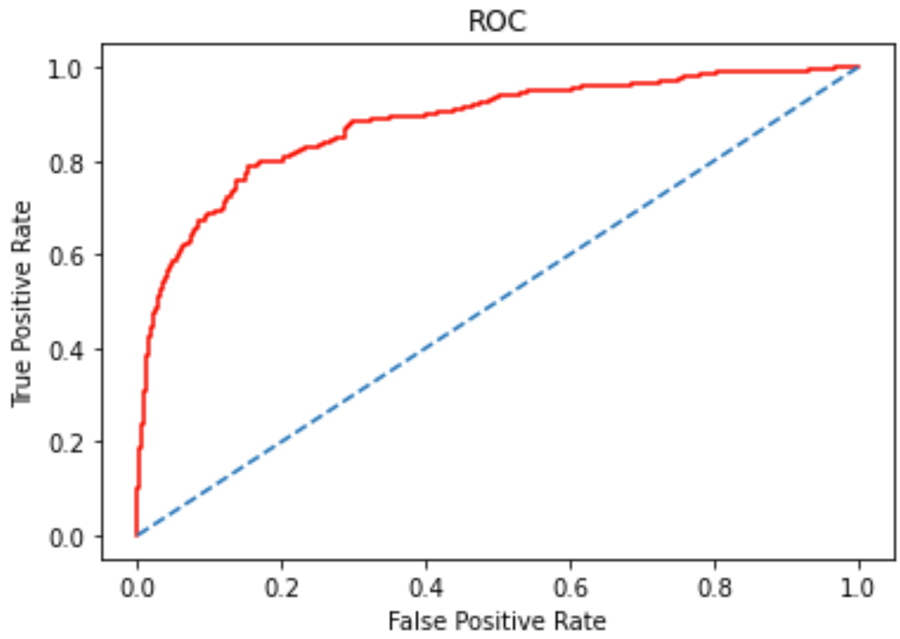
Classification Report:

	Precision	**Recall**	**F1-score**
No Churn	0.91	0.97	0.94
Churn	0.77	0.55	0.64

Confusion Matrix:

	Predicted 'No Churn'	**Predicted 'Churn'**
Actual 'No Churn'	3179	105
Actual 'Churn'	297	360

Logit Model's performance on the Test dataset:

Accuracy	0.89
AUC Score	0.74
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.91	0.97	0.94
Churn	0.78	0.52	0.62

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1355	43
Actual 'Churn'	140	151

Observations**:**

- The Logistic Regression model performs poorly on the Recall metric for Churn, which is arguably the most significant measure for us to assess whether the model is a good predictor of customers that are likely to churn.
- Based on the test metrics: Only 52% of customers who churned were correctly identified and predicted by the model.
- This performance is likely a result of the imbalance that we have noted in the dataset. Customers labeled as 1 for Churn account for only 17% of the observations. This imbalance will have to be addressed when we tune our models in the next section.

Logit Model Coefficients:

The following coefficients were assigned by the model to the various features. The values indicate the weights given to the variables by the model in deciding whether or not a customer is likely to churn.

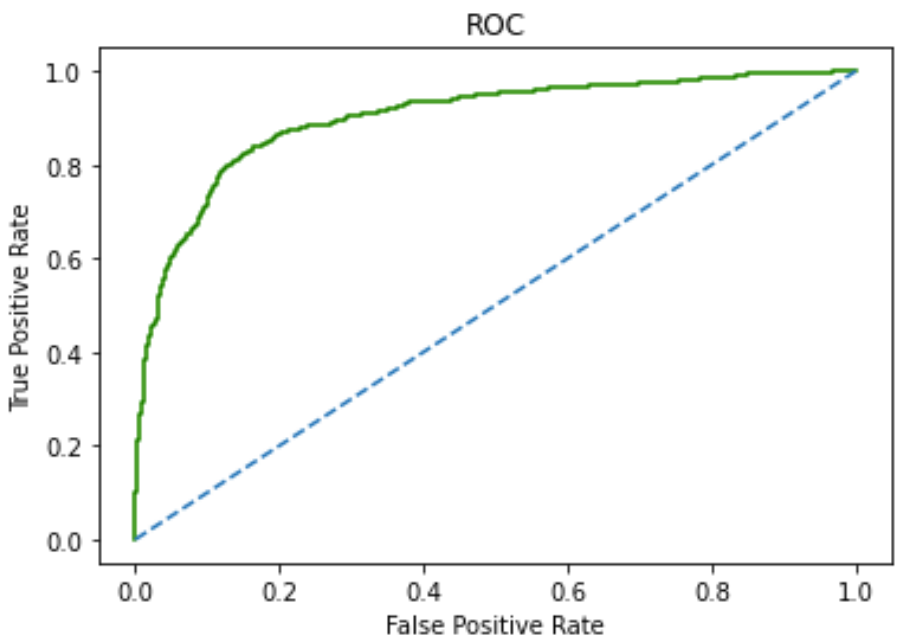
Tenure	-1.92
WarehouseToHome	0.38
HourSpendOnApp	-0.07
NumberOfDeviceRegistered	0.42
SatisfactionScore	0.30
NumberOfAddress	0.66
Complain	1.83
OrderAmountHikeFromlastYear	-0.10
CouponUsed	0.07
OrderCount	0.49
DaySinceLastOrder	-0.41
CashbackAmount	-0.56
PreferredLoginDevice_Mobile Phone	-0.52
CityTier_2	0.85
CityTier_3	0.64
PreferredPaymentMode_Credit Card	-0.69
PreferredPaymentMode_Debit Card	-0.46
PreferredPaymentMode_E wallet	0.19
PreferredPaymentMode_UPI	-0.56
Gender_Male	0.32
PreferedOrderCat_Grocery	0.17
PreferedOrderCat_Laptop & Accessory	-1.49
PreferedOrderCat_Mobile Phone	-0.39
PreferedOrderCat_Others	1.72
MaritalStatus_Married	-0.39
MaritalStatus_Single	0.63

- A negative coefficient indicates an inverse relationship. Variable Tenure with a coefficient of -1.92 indicates that a customer that has been on the platform for a longer duration is relatively less likely to churn.
- In contrast, variable Complain with a positive coefficient of 1.83 indicates that a customer who has placed a complain has a likelihood of Churn.

Linear Discriminant Analysis (LDA)

We used scikit-learn library's LinearDiscriminantAnalysis algorithm to build the LDA model.

LDA Model's performance on the Training dataset:

Accuracy	0.89
AUC Score	0.74
ROC Curve	

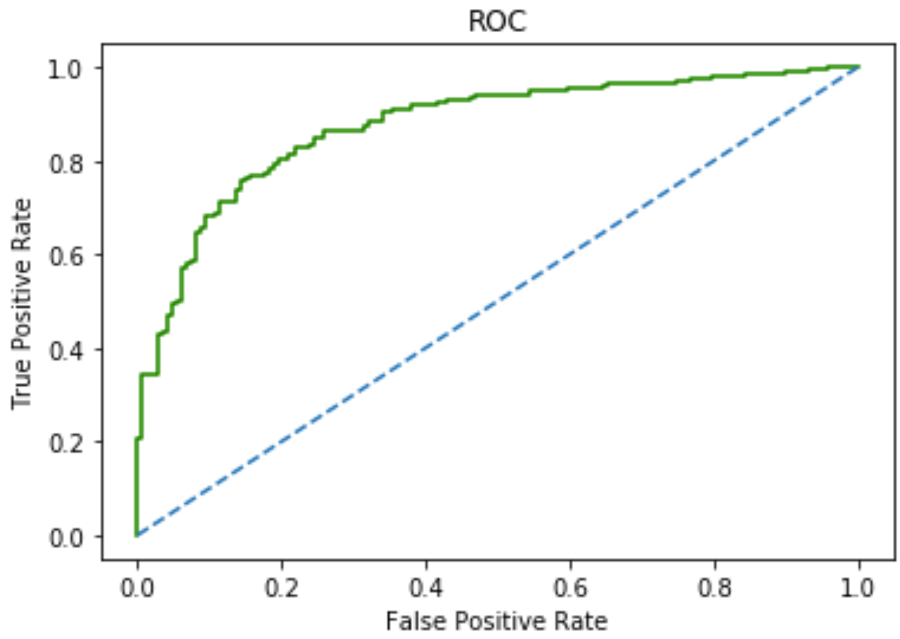
Classification Report:

	Precision	Recall	F1-score
No Churn	0.91	0.97	0.94
Churn	0.76	0.51	0.61

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	3179	105
Actual 'Churn'	324	333

LDA Model's performance on the Test dataset:

Accuracy	0.89
AUC Score	0.74
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.91	0.97	0.94
Churn	0.79	0.52	0.63

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1357	41
Actual 'Churn'	140	151

Observations:

- The LDA model is very similar in performance to the Logit model. The LDA model, too, performs poorly on the Recall metric for Churn. Based on the test metrics: Only 52% of customers who churned were correctly identified and predicted by the model.
- As noted, Recall is arguably the most significant measure for us to assess whether the model is a good predictor of customers that are likely to churn.
- This performance again is likely a result of the imbalance that we have noted in the dataset.

LDA Model Coefficients:

The following coefficients were assigned by the LDA model to the various features. The values indicate the weights given to the variables by the model in deciding whether or not a customer is likely to churn.

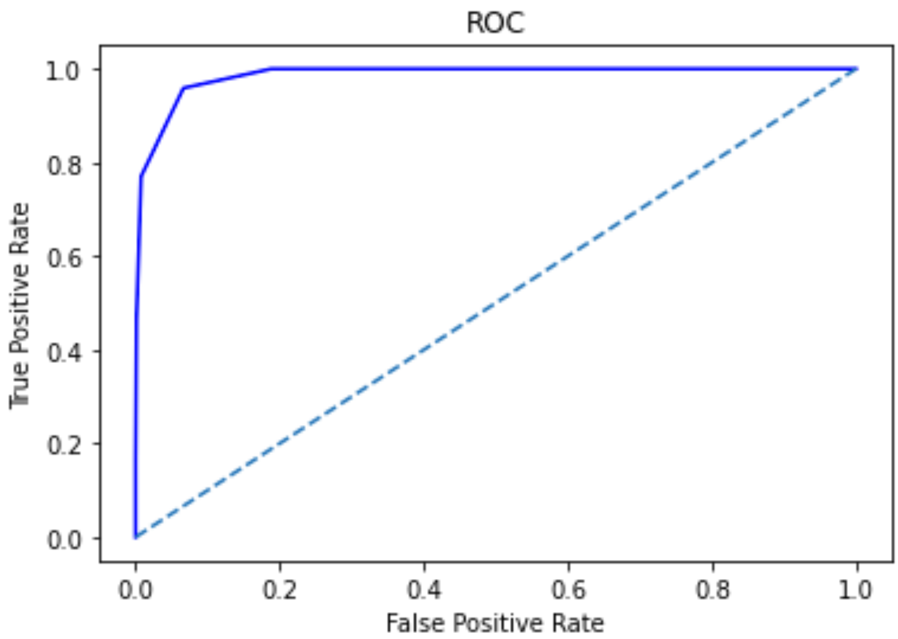
Tenure	-1.35
WarehouseToHome	0.31
HourSpendOnApp	-0.09
NumberOfDeviceRegistered	0.37
SatisfactionScore	0.29
NumberOfAddress	0.58
Complain	2.08
OrderAmountHikeFromlastYear	-0.04
CouponUsed	0.08
OrderCount	0.29
DaySinceLastOrder	-0.32
CashbackAmount	-0.55
PreferredLoginDevice_Mobile Phone	-0.39
CityTier_2	0.66
CityTier_3	0.64
PreferredPaymentMode_Credit Card	-0.96
PreferredPaymentMode_Debit Card	-0.77
PreferredPaymentMode_E wallet	-0.13
PreferredPaymentMode_UPI	-0.78
Gender_Male	0.24
PreferedOrderCat_Grocery	0.98
PreferedOrderCat_Laptop & Accessory	-1.19
PreferedOrderCat_Mobile Phone	-0.06
PreferedOrderCat_Others	1.80
MaritalStatus_Married	-0.28
MaritalStatus_Single	0.83

Even for the LDA model, Complain stands out a strong indicator of a customer’s likelihood of Churn. Also, greater the Tenure value, the less the likelihood of a customer churning.

K-Nearest Neighbor (KNN)

We used scikit-learn library’s KNeighborsClassifier algorithm to build the KNN model.

KNN Model’s performance on the Training dataset:

Accuracy	0.96
AUC Score	0.88
ROC Curve	

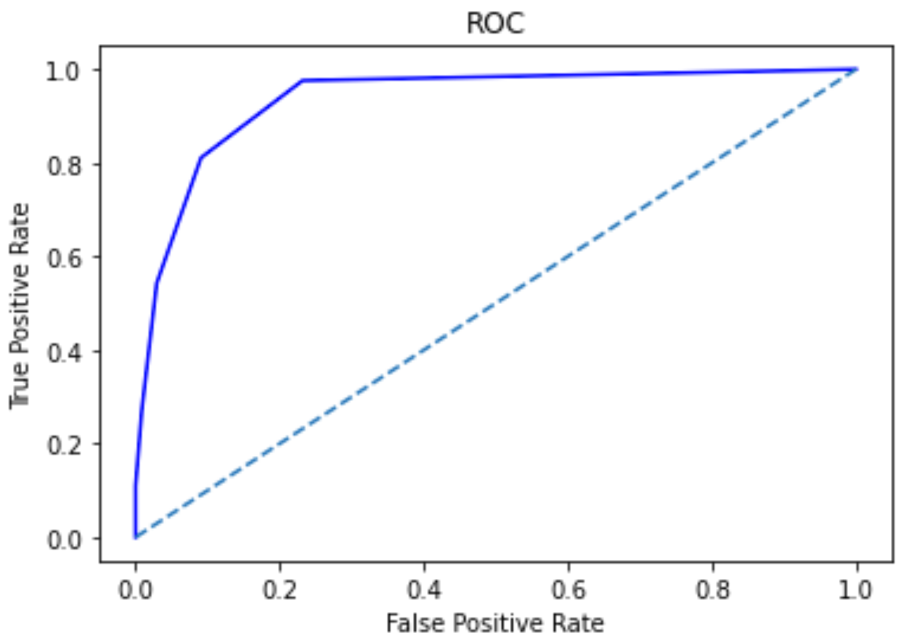
Classification Report:

	Precision	**Recall**	**F1-score**
No Churn	0.96	0.99	0.97
Churn	0.95	0.77	0.85

Confusion Matrix:

	Predicted ‘No Churn’	**Predicted ‘Churn’**
Actual ‘No Churn’	3259	25
Actual ‘Churn’	151	506

KNN Model's performance on the Test dataset:

Accuracy	0.90
AUC Score	0.76
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.91	0.97	0.94
Churn	0.79	0.54	0.64

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1357	41
Actual 'Churn'	133	158

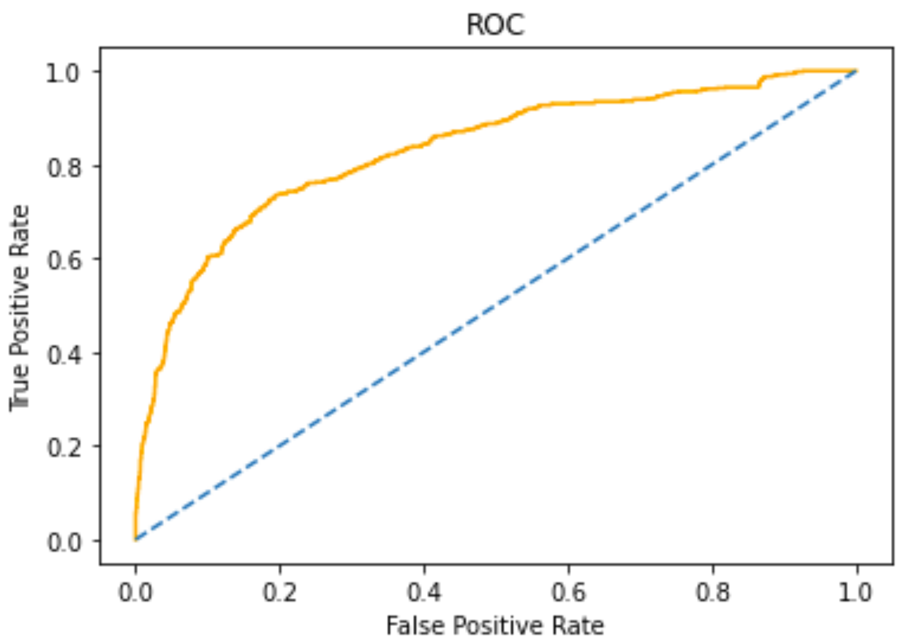
Observations:

- The model appears to be significantly underperforming on the test data. This can be taken as an indicator of overfitting. Hence the KNN model doesn't appear to be a reliable predictor of Churn.
- In any case, the Recall score on test data is low: Only 54% of customers who churned were correctly identified and predicted by the model.

Naive Bayes (NB)

We used scikit-learn library's GaussianNB algorithm to build the Naive Bayes model.

NB Model's performance on the Training dataset:

Accuracy	0.75
AUC Score	0.75
ROC Curve	

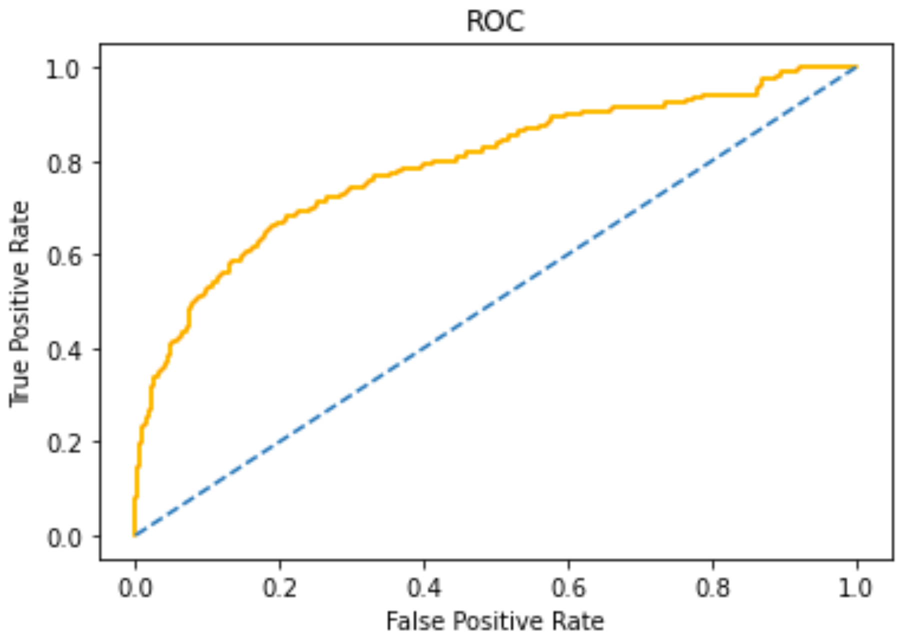
Classification Report:

	Precision	Recall	F1-score
No Churn	0.94	0.75	0.83
Churn	0.38	0.76	0.50

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	2449	835
Actual 'Churn'	156	501

NB Model's performance on the Test dataset:

Accuracy	0.73
AUC Score	0.73
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.93	0.73	0.82
Churn	0.36	0.73	0.48

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1019	379
Actual 'Churn'	80	211

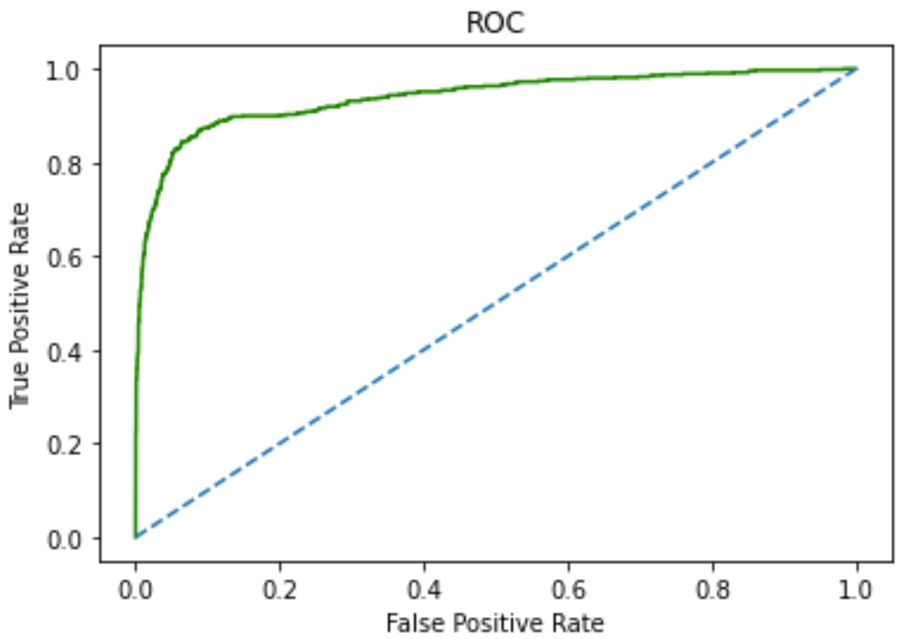
Observations:

- The Naive Bayes model performance is striking in its contrast to the other preceding models.
- The NB model has a relatively strong Recall score: 73% of customers in the test data who churned were correctly identified and predicted by the model.
- However, it scores very low on Precision. Only 36% of the customers that it predicted as Churn actually churned. Which means that 64% of customers that it predicted would churn, did not actually churn. This is a very high proportion of false positives.

Support-Vector Machine (SVM)

We use scikit-learn library's svm.SVC algorithm to build the SVM model.

SVM Model's performance on the Training dataset:

Accuracy	0.92
AUC Score	0.80
ROC Curve	

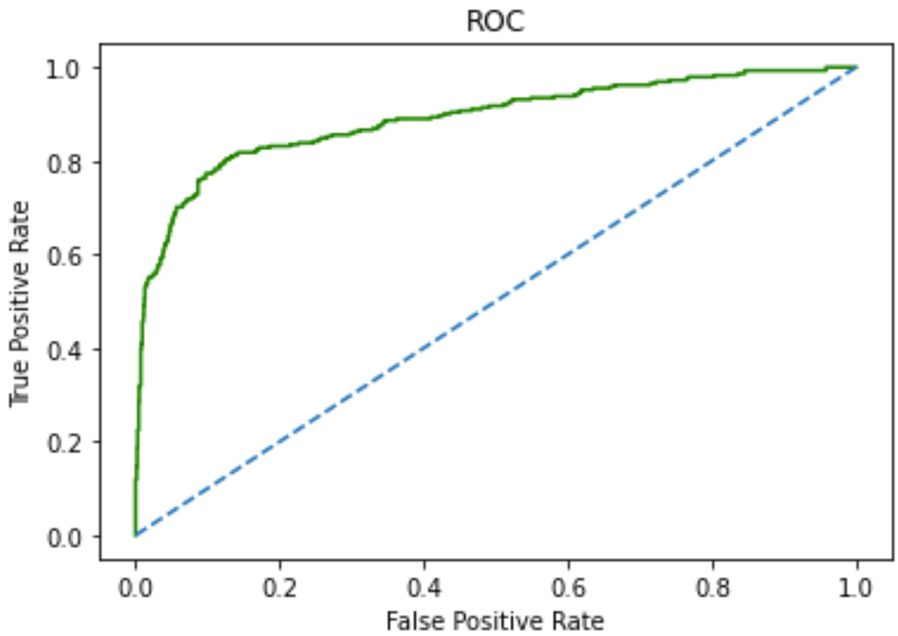
Classification Report:

	Precision	Recall	F1-score
No Churn	0.93	0.99	0.96
Churn	0.91	0.61	0.73

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	3243	41
Actual 'Churn'	255	402

SVM Model’s performance on the Test dataset:

Accuracy	0.91
AUC Score	0.76
ROC Curve	

Classification Report:

	Precision	**Recall**	**F1-score**
No Churn	0.91	0.98	0.95
Churn	0.88	0.54	0.67

Confusion Matrix:

	Predicted ‘No Churn’	**Predicted ‘Churn’**
Actual ‘No Churn’	1377	21
Actual ‘Churn’	135	156

Observations:

- Compared to the SVM model’s performance on the training data, the model appears to be underperforming on the test data, especially on Recall for Churn. However the difference in training and test metrics isn’t significant, and does not suggest overfitting.

- In any case, the Recall score on test data is low: Only 54% of customers who churned were correctly identified and predicted by the model.

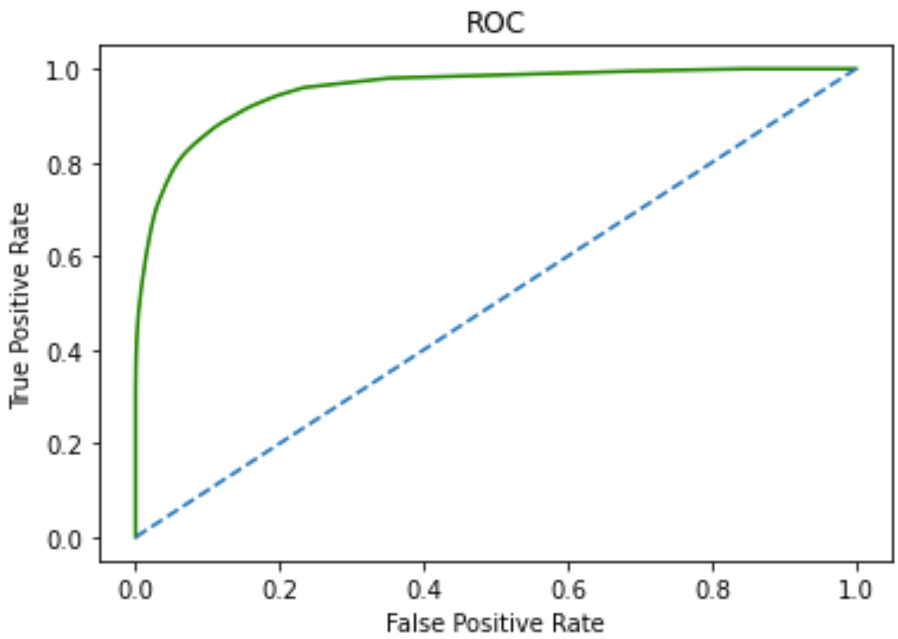
- Precision, however, is significantly higher compared to the preceding models. Around 88% of the customers that the model predicted would churn, did indeed churn.

- The imbalance in the data is likely causing the model to underperform on Recall, and we will revisit SVM in the Tuning section.

Decision Tree (Regularized)

We used scikit-learn library's DecisionTreeClassifier algorithm to build the CART model.

CART Model's performance on the Training dataset:

Accuracy	0.93
AUC Score	0.83
ROC Curve	

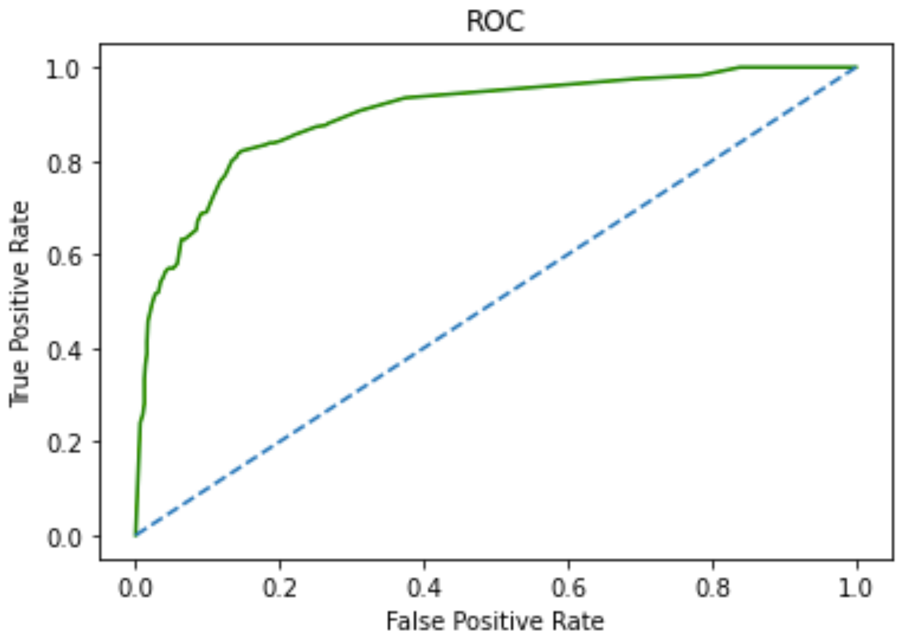
Classification Report:

	Precision	Recall	F1-score
No Churn	0.94	0.97	0.96
Churn	0.84	0.69	0.76

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	3195	89
Actual 'Churn'	201	456

CART Model's performance on the Test dataset:

Accuracy	0.88
AUC Score	0.76
ROC Curve	

Classification Report

	Precision	Recall	F1-score
No Churn	0.91	0.95	0.93
Churn	0.70	0.57	0.63

Confusion Matrix

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1328	70
Actual 'Churn'	125	166

Observations:

- Compared to the Decision Tree model's performance on the training data, the model appears to be underperforming on the test data, especially on Recall and Precision metrics for Churn. However the difference in training and test metrics isn't striking enough to necessarily suggest overfitting.

- In any case, the Recall score on test data is low: Only 57% of customers who churned were correctly identified and predicted by the model. A Precision score of 70% is also comparatively low.

The following values were assigned by the model to the various features. The values indicate the importance given to the variables by the model in deciding whether or not a customer is likely to churn.

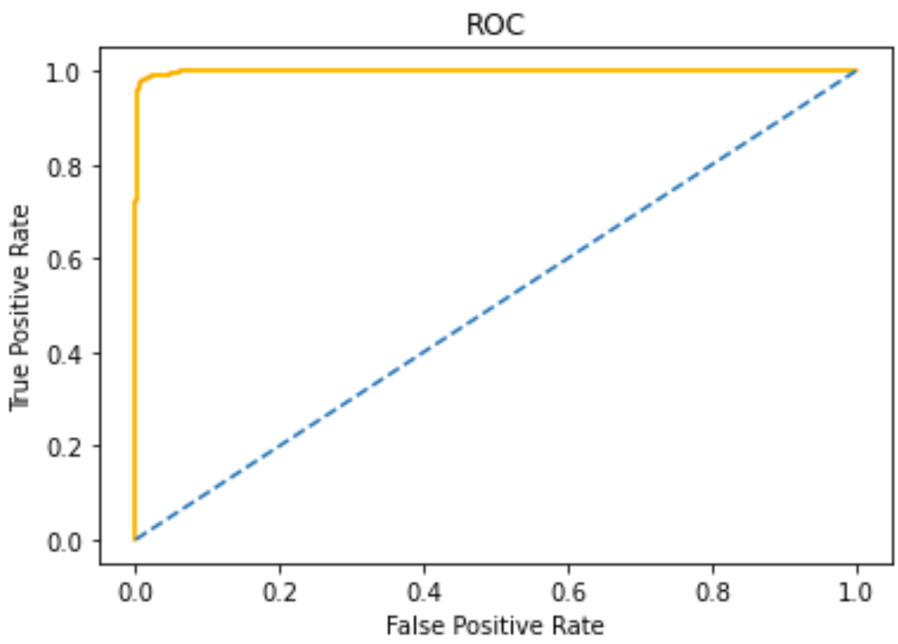
Tenure	0.484
NumberOfAddress	0.084
DaySinceLastOrder	0.082
Complain	0.079
WarehouseToHome	0.075
SatisfactionScore	0.042
NumberOfDeviceRegistered	0.029
OrderAmountHikeFromlastYear	0.021
PreferredLoginDevice_Mobile Phone	0.020
PreferredPaymentMode_E wallet	0.018
Gender_Male	0.015
CouponUsed	0.011
PreferedOrderCat_Laptop & Accessory	0.009
MaritalStatus_Married	0.007
CityTier_3	0.006
PreferredPaymentMode_Credit Card	0.006
MaritalStatus_Single	0.004
PreferedOrderCat_Mobile Phone	0.003
CashbackAmount	0.003
OrderCount	0.002
PreferredPaymentMode_Debit Card	0.000
PreferredPaymentMode_UPI	0.000
PreferedOrderCat_Grocery	0.000
PreferedOrderCat_Others	0.000
HourSpendOnApp	0.000
CityTier_2	0.000

The variable Tenure is considered most significant by far. Other factors in the top 5 are number of addresses, recency of order, whether the customer complained in the last month, and distance of customer address from the nearest warehouse.

Artificial Neural Network (ANN)

We used scikit-learn library's MLPClassifier algorithm to build the ANN model.

ANN Model's performance on the Training dataset:

Accuracy	0.99
AUC Score	0.98
ROC Curve	

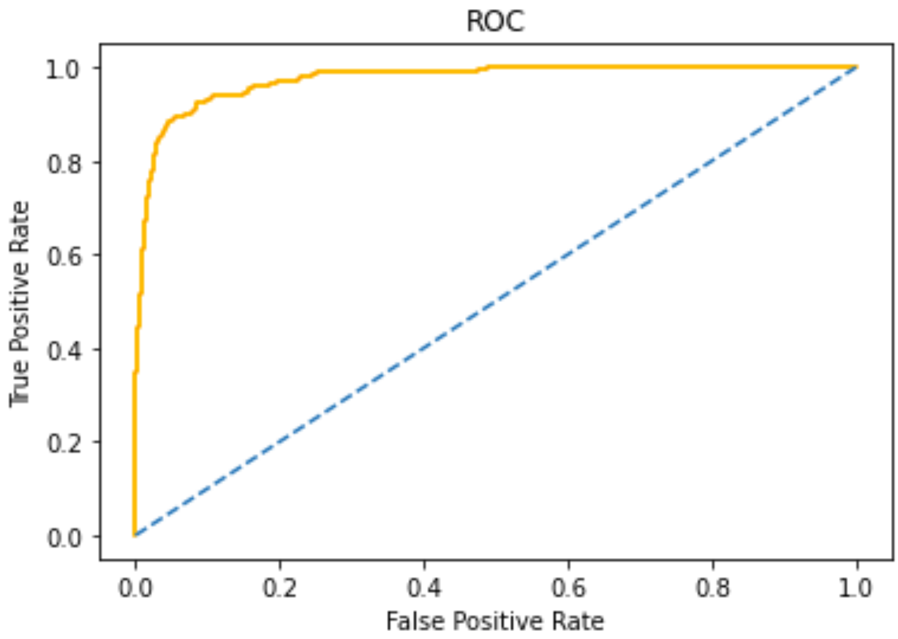
Classification Report:

	Precision	Recall	F1-score
No Churn	0.99	1.00	0.99
Churn	0.98	0.96	0.97

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	3270	14
Actual 'Churn'	25	632

ANN Model's performance on the Test dataset:

Accuracy	0.94
AUC Score	0.88
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.96	0.97	0.97
Churn	0.87	0.79	0.83

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1363	35
Actual 'Churn'	60	231

Observations:

- This is the strongest predictive model among the models we have built so far.
- A Recall value of 0.79 on the test data indicates that the model has been able to correctly predict 79% of the customers who have actually churned.
- A Precision of 0.87 on the test data is also quite strong, indicating that 87% of the customers that the model predicted would churn, did indeed churn.
- The model performance ranks the highest on almost all counts. That said, compared to the ANN model's performance on the training data, the model appears to be underperforming on the test data, on both Recall and Precision for Churn. However the difference in training and test metrics isn't striking enough to suggest overfitting.

- We will revisit ANN in the model tuning section to see if the recall can be further improved.

Model Tuning, Bagging, Boosting

This section details out the model performance, insights and hyper parameter tuning details of each model built for the exercise.

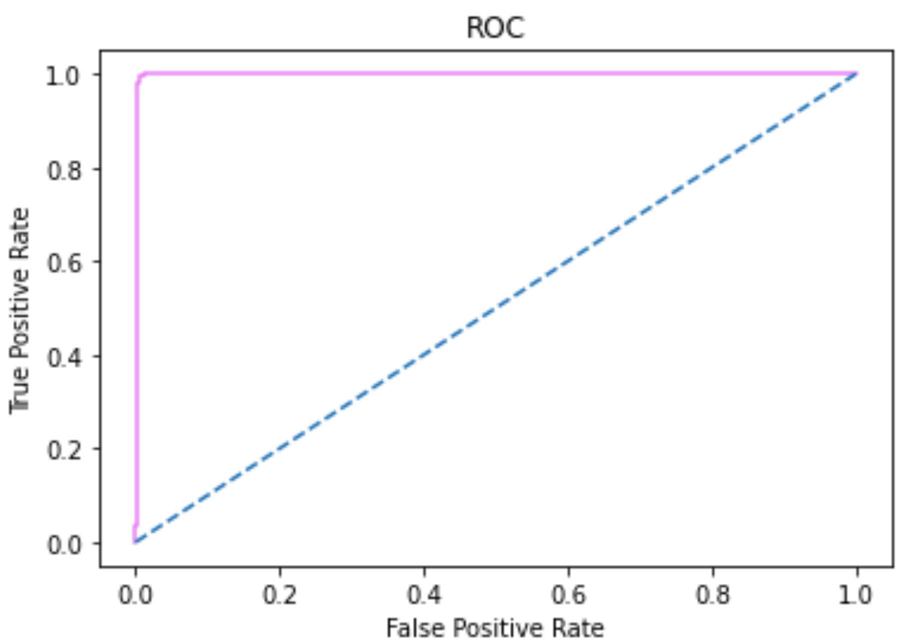
Model Tuning: We have noted in the last section that the imbalance in the dataset has affected the performance of many of our models. In this section, we will tune models to address this imbalance, and apply appropriate hyper parameters with the aim to improve the model's performance.

We will also explore Ensemble modeling (Bagging and Boosting) to improve overall model performance.

SVM (Tuned)

We used GridSearchCV to pass various options for parameters C (the penalty for error) and gamma, to obtain the optimal combination.

With the combination of the following: {'C': 2, 'gamma': 0.13, 'kernel': 'rbf', class_weight='balanced'}, the tuned SVM model gave the following results on the Training Dataset:

Accuracy	0.98
AUC Score	0.99
ROC Curve	

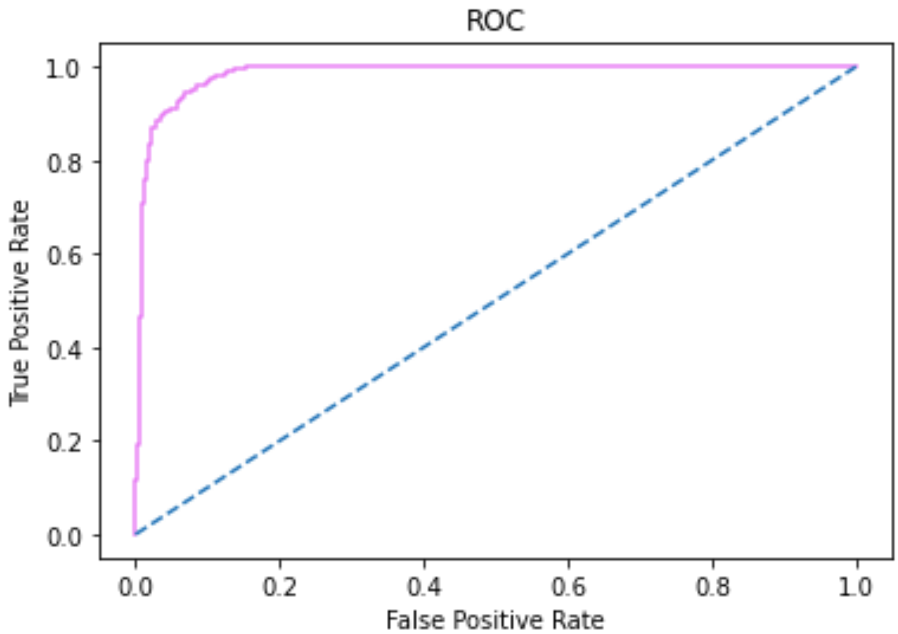
Classification Report:

	Precision	Recall	F1-score
No Churn	1.00	0.98	0.99
Churn	0.90	1.00	0.95

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	3211	73
Actual 'Churn'	0	657

SVM (tuned) Model Performance on Test dataset:

Accuracy	0.94
AUC Score	0.93
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.98	0.95	0.97
Churn	0.80	0.91	0.85

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1330	68
Actual 'Churn'	26	265

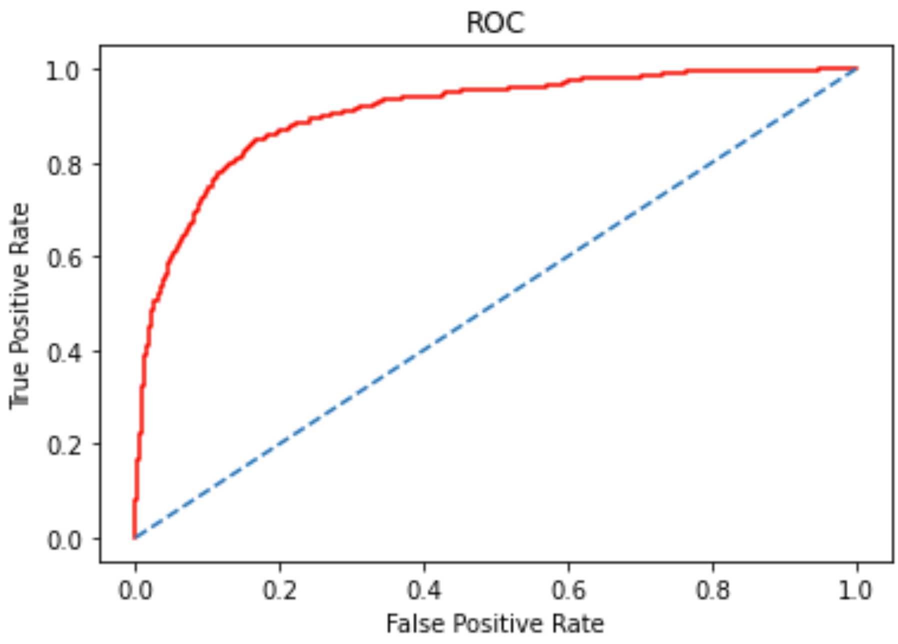
Observations:

- The Tuned SVM model is a significant improvement over the regular SVM model. A Recall score of 0.91 for Churn is the highest score among all the models we've built. This implies that the model has identified 91% of the customers that actually churned, correctly.
- Precision has been adversely affected, but a 0.8 is still a comparatively good score.
- The almost perfect performance on the Training data does suggest a possibility overfitting and that the model may not generalise well. Yet, the test performance is encouraging.

Logit (Tuned)

Logistic Regression with hyper parameter class_weight='balanced'

Logit (tuned) Model Performance on the Training Dataset:

Accuracy	0.83
AUC Score	0.84
ROC Curve	

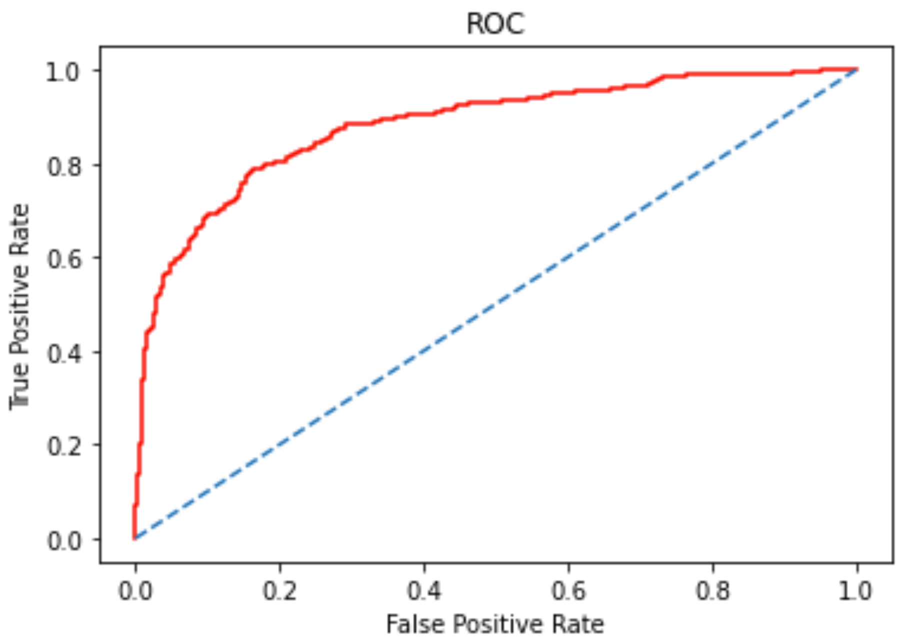
Classification Report:

	Precision	Recall	F1-score
No Churn	0.97	0.82	0.89
Churn	0.49	0.86	0.62

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	2690	594
Actual 'Churn'	95	562

Logit (tuned) Model Performance on Test dataset:

Accuracy	0.81
AUC Score	0.81
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.95	0.81	0.88
Churn	0.47	0.80	0.59

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1135	263
Actual 'Churn'	58	233

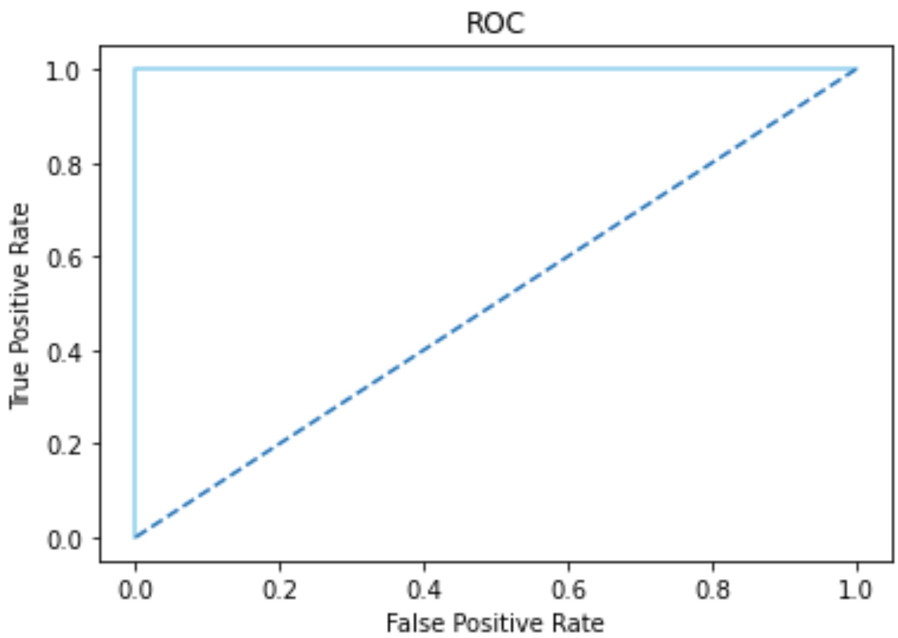
Observation:

The tuned Logistic Regression model has shown a significant improvement in Recall score, from 0.54 in the regular model to 0.80 in the tuned model. But it has also negatively affected the Precision, which is at 0.47 for the test data, a very low score.

Artificial Neural Network (Tuned)

We used GridSearchCV to pass various options for parameters hidden_layer_sizes and alpha, to obtain the optimal combination.

With the combination of the following: {'activation': 'relu', 'alpha': 0.05, 'hidden_layer_sizes': (100, 100),'solver': 'adam'}, the tuned ANN model gave the following results on the Training Dataset:

Accuracy	1.00
AUC Score	1.00
ROC Curve	

Classification Report:

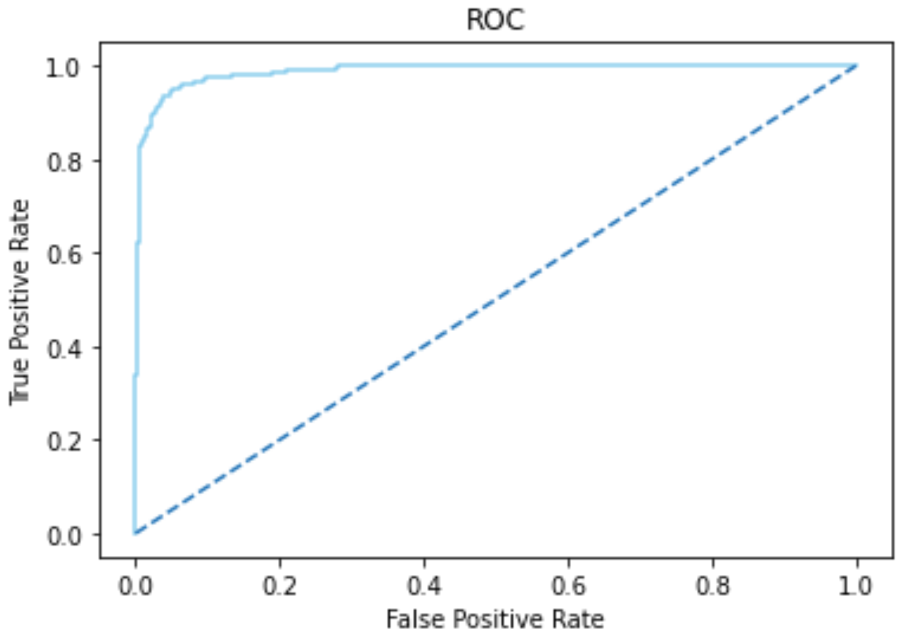
	Precision	Recall	F1-score
No Churn	1.00	1.00	1.00
Churn	1.00	1.00	1.00

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	3284	0
Actual 'Churn'	0	657

ANN (tuned) Model Performance on Test dataset:

Accuracy	0.96

AUC Score	0.92
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.97	0.99	0.98
Churn	0.93	0.85	0.89

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1380	18
Actual 'Churn'	44	247

Observations:
- The tuned ANN model is an improvement on the regular ANN model. A recall is 0.85 and a precision of 0.93 are both impressive scores.
- This implies that the model has identified 85% of the customers that actually churned, correctly.
- Precision of 0.93 indicates: 93% of those predicted to churn actually did churn.
- The perfect score performance on the Training data does suggest a possibility overfitting and that the model may not generalise well. Yet, the test performance is encouraging.

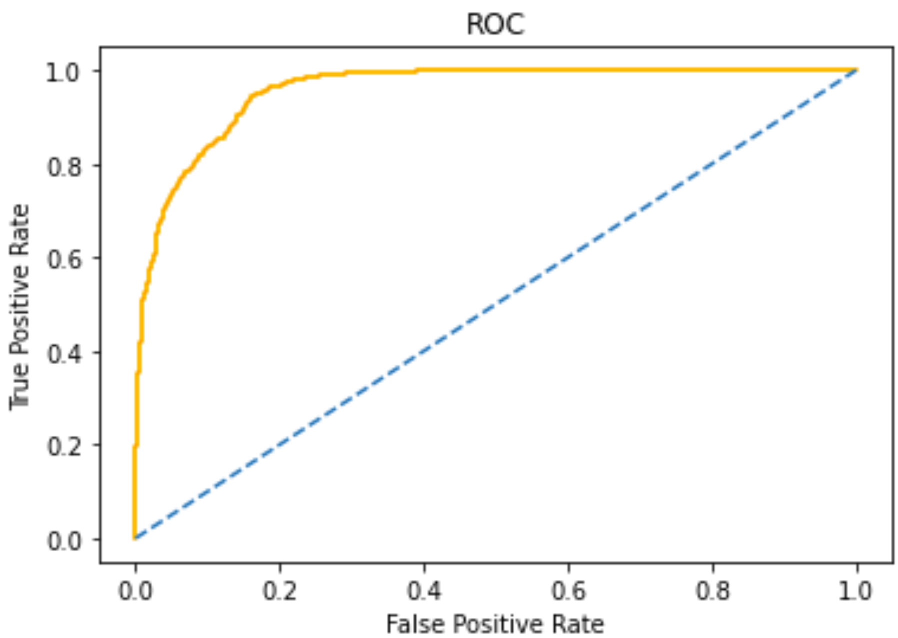
Bagging: Random Forest

For Random Forest Classification, we used scikit-learn library's RandomForestClassifier algorithm.

Tuning:
We used GridSearchCV to get the optimal parameter combination for running the Random Forest model on the data.

The best parameters obtained by this method is as follows:
{'max_depth': 10,
 'max_features': 10,
 'min_samples_leaf': 20,
 'min_samples_split': 60,
 'n_estimators': 501,
 'class_weight': 'balanced'}

Random Forest Classifier Model's performance on the Training dataset:

Accuracy	0.87
AUC Score	0.87
ROC Curve	

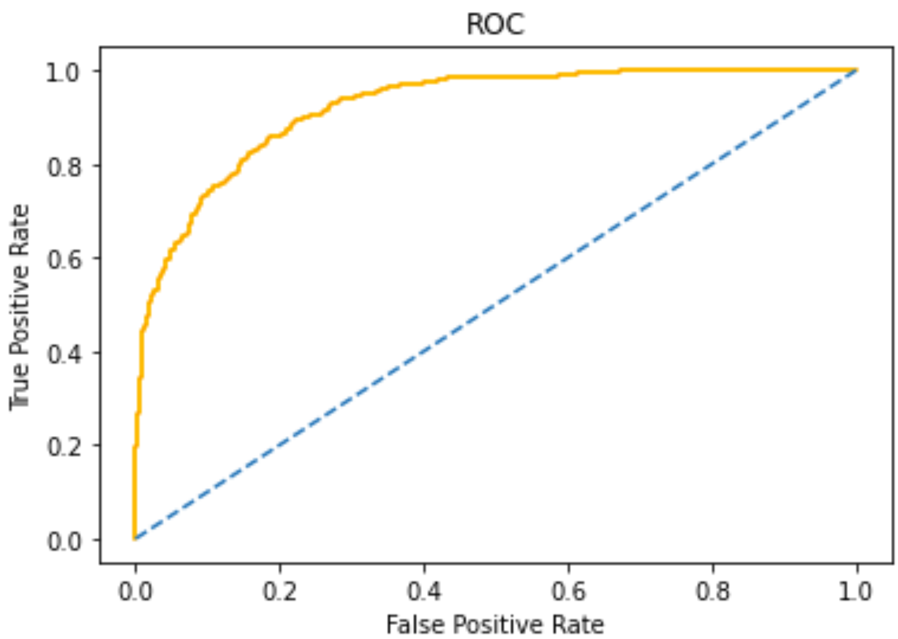
Classification Report:

	Precision	Recall	F1-score
No Churn	0.97	0.87	0.92
Churn	0.58	0.87	0.69

Confusion Matrix:

	Predicted ‘No Churn’	Predicted ‘Churn’
Actual ‘No Churn’	2864	420
Actual ‘Churn’	85	572

Random Forest Classifier Model’s performance on the Test dataset:

Accuracy	0.84
AUC Score	0.83
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.96	0.85	0.90
Churn	0.52	0.81	0.64

Confusion Matrix:

	Predicted ‘No Churn’	Predicted ‘Churn’
Actual ‘No Churn’	1183	215
Actual ‘Churn’	55	236

Observations:
The Random Forest model has a strong Recall of 0.81, but a poor Precision score of 0.52 on the test data.

Features ranked according to their importance to the Random Forest model:

Rank	Feature	Weight
1	Tenure	0.527
2	Complain	0.102
3	CashbackAmount	0.053
4	PreferedOrderCat_Mobile Phone	0.038
5	NumberOfAddress	0.037
6	DaySinceLastOrder	0.032
7	SatisfactionScore	0.032
8	WarehouseToHome	0.031
9	MaritalStatus_Single	0.024
10	CouponUsed	0.017
11	CityTier_3	0.017
12	OrderAmountHikeFromlastYear	0.016
13	PreferedOrderCat_Laptop & Accessory	0.013
14	MaritalStatus_Married	0.012
15	OrderCount	0.011
16	NumberOfDeviceRegistered	0.010
17	PreferredPaymentMode_Credit Card	0.008
18	Gender_Male	0.005
19	PreferredPaymentMode_Debit Card	0.005
20	PreferredLoginDevice_Mobile Phone	0.004
21	PreferredPaymentMode_E wallet	0.003
22	HourSpendOnApp	0.002
23	PreferedOrderCat_Grocery	0.001
24	CityTier_2	0.000
25	PreferedOrderCat_Others	0.000
26	PreferredPaymentMode_UPI	0.000

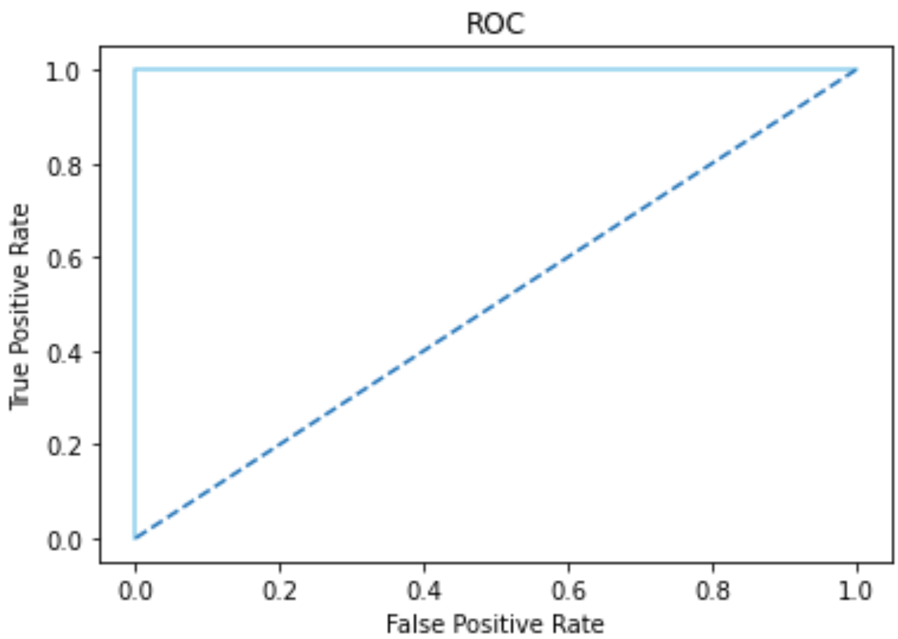
Boosting: XGBoost

For Extreme Gradient Boost Classification, we used scikit-learn library's XGBClassifier algorithm.

Tuning:
We use GridSearchCV to get the optimal parameter combination for running the XGBoost model on the data.

The parameters obtained by this method is as follows
{'colsample_bytree': 0.5,
 'gamma': 0.2,
 'learning_rate': 0.2,
 'max_depth': 7,
 'min_child_weight': 1}

XGBoost Classifier Model's performance on the Training dataset:

Accuracy	1.00
AUC Score	1.00
ROC Curve	

Classification Report:

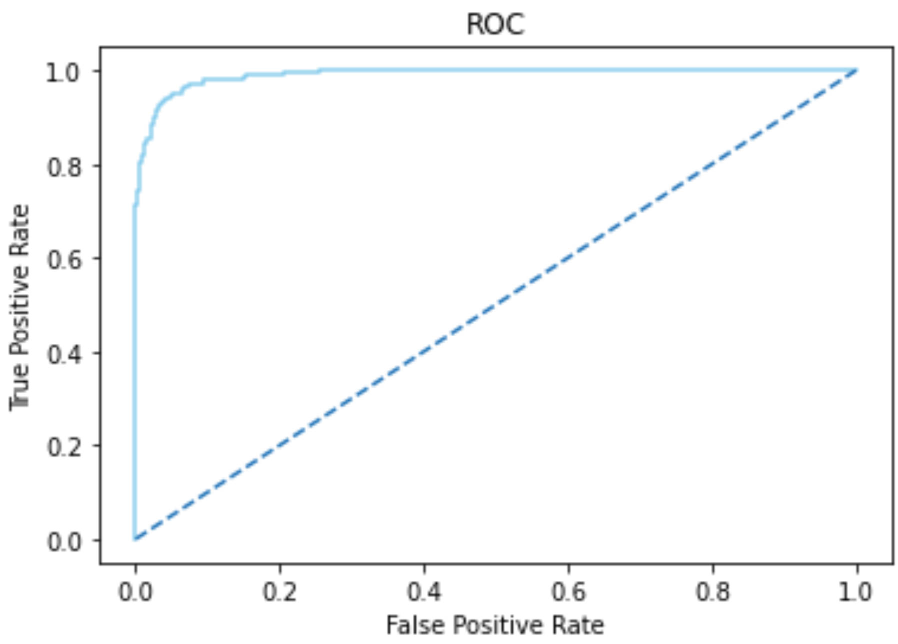
	Precision	Recall	F1-score
No Churn	1.00	1.00	1.00
Churn	1.00	1.00	1.00

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	3284	0

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'Churn'	0	657

XGBoost Classifier Model's performance on the Test dataset:

Accuracy	0.96
AUC Score	0.93
ROC Curve	

Classification Report:

	Precision	Recall	F1-score
No Churn	0.98	0.98	0.98
Churn	0.88	0.89	0.89

Confusion Matrix:

	Predicted 'No Churn'	Predicted 'Churn'
Actual 'No Churn'	1364	34
Actual 'Churn'	31	260

Observations:

- The XGB model is a strong model: a Recall is 0.89 and a Precision of 0.88 are both impressive scores.
- This implies that the model has been able to predict 89% of the customers that actually churned, correctly. Precision of 0.88 indicates: 88% of those predicted to churn, did indeed churn.
- Recall is a near second only to the SVM tuned model, and Precision is a near second only to the ANN tuned model.

- The perfect score performance on the Training data does suggest a possibility overfitting and that the model may not generalise well. Yet, the test performance is encouraging.

Features ranked according to their importance to the XGBoost model:

Rank	Feature	Weight
1	Complain	0.152
2	PreferedOrderCat_Grocery	0.129
3	Tenure	0.093
4	PreferedOrderCat_Mobile Phone	0.057
5	PreferedOrderCat_Others	0.052
6	PreferedOrderCat_Laptop & Accessory	0.045
7	MaritalStatus_Single	0.043
8	MaritalStatus_Married	0.038
9	PreferredPaymentMode_Credit Card	0.028
10	SatisfactionScore	0.027
11	NumberOfDeviceRegistered	0.026
12	PreferredPaymentMode_Debit Card	0.024
13	DaySinceLastOrder	0.024
14	PreferredPaymentMode_E wallet	0.023
15	CityTier_3	0.023
16	WarehouseToHome	0.022
17	NumberOfAddress	0.022
18	PreferredLoginDevice_Mobile Phone	0.022
19	CashbackAmount	0.021
20	PreferredPaymentMode_UPI	0.021
21	OrderCount	0.021
22	OrderAmountHikeFromlastYear	0.021
23	CouponUsed	0.019
24	Gender_Male	0.018
25	CityTier_2	0.016
26	HourSpendOnApp	0.015

This list indicates that Complain, Tenure, Preferred Order Category, Marital Status, and Preferred Payment mode are by and large important discriminators.

Supporting Documents

The related workings and code are appended in the following Jupyter notebooks:

EDA and Pre-processing	capstone_pn1_sb.ipynb
Predictive Modeling	capstone_pn2_sb.ipynb