CAPSTONE PROJECT REPORT

DSBA

greatlearning
Learning for Life

SUBMITTED BY:

Varun Kabra

Contents

1) Introduction to Business Problem

1.1) Problem Statement:

Customer churn is when the customers either switch from their incumbent service provider to its competitor or stop using the service altogether. As per the existing research, it costs five to six times more to acquire customers than to retain existing customers, and this emphasizes the importance of managing churn by organization.

An E Commerce company is facing a lot of competition in the current market. It has become a challenge to retain the existing customers in the current situation. Hence, the company wants to develop a model through which they can do churn prediction of the accounts and provide segmented offers to the potential churners. In this company, account churn is a major thing because 1 account can have multiple customers. Hence by losing one account the company might be losing more than one customer.

In this project, I will build a Churn prediction model for one of India's largest E Commerce Company, for its customer base. For this, I will use "Customer Churn Data" dataset provided to me. There are 18 features and 1 target (dependent) variable for 11260 customers. Target variable indicates if a customer left (i.e. churn=yes). Since the target variable has two states (yes/no or 1/0), this is a binary classification problem.

The variables are:

'AccountID', 'Churn', 'Tenure', 'City_Tier', 'CC_Contacted_LY','Payment', 'Gender', 'Service_Score', 'Account_user_count', 'account_segment', 'CC_Agent_Score', 'Marital_Status', 'rev_per_month','Complain_ly', 'rev_growth_yoy', 'coupon_used_for_payment', 'Day_Since_CC_connect', 'cashback', 'Login_device'

At first glance, only 'AccountID' seems irrelevant to customer churn. Other variables may or may not have an effect on customer churn. We will figure out.

1.2) Need of the Project:

The primary objective of this customer churn project is to retain customers at the highest risk of churn by proactively engaging with them. Customer churn is an important metric to track because lost customers equal lost revenue. If a company loses enough customers, it can have a serious impact on its bottom line. Another reason it's critical to improve customer retention and reduce churn is that it's generally more expensive to find new customers than it is to keep existing ones. So, companies that lose customers aren't just losing the revenue from those customers—they're also stuck with the high cost of finding new customers. No matter how good a company's product or service may be, it's essential that they monitor their customer churn rate.

1.3) Business/social opportunity

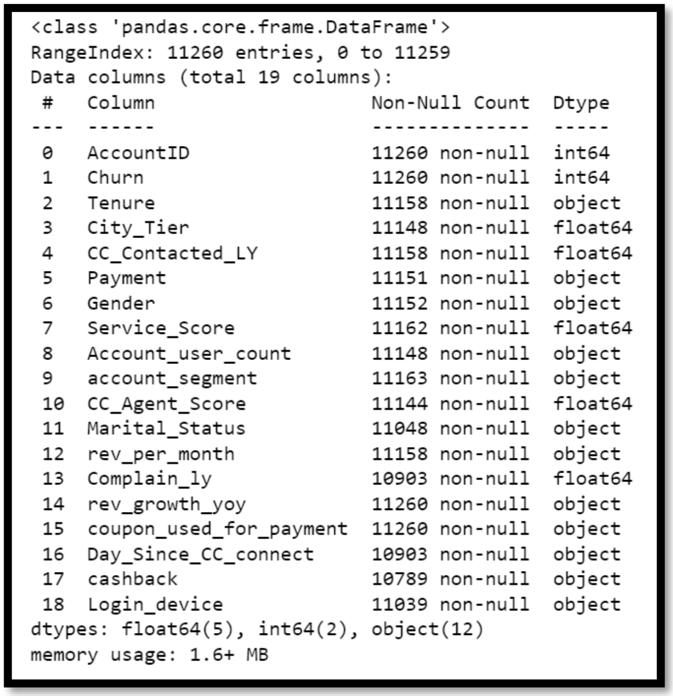
- Product Development:
  A lifecycle exists for all products and services. Customer churn can indicate that your product is nearing the end of its life cycle and may require additional development to remain relevant or meet the expectations of your customers. Speaking with churning clients will provide you with a solid indication of your product's viability in its current state, as well as assist you prioritise any new investments you need to make.

- Competition:
  Sometimes a competitor will simply charm your customer into leaving your service. It's critical to maintain strong relations with these clients because if their new service provider fails to meet their expectations, you have a decent opportunity of regaining their business. These clients become fantastic case studies once they comprehend your genuine value based on their experience on the opposite side of the fence.

- Staff Training:
  Churn might reveal a deficiency in the resources available to your personnel to meet the needs of your consumers. Not only can training help you keep clients, but it will also help you retain staff and save money on recruitment fees if you treat them properly.

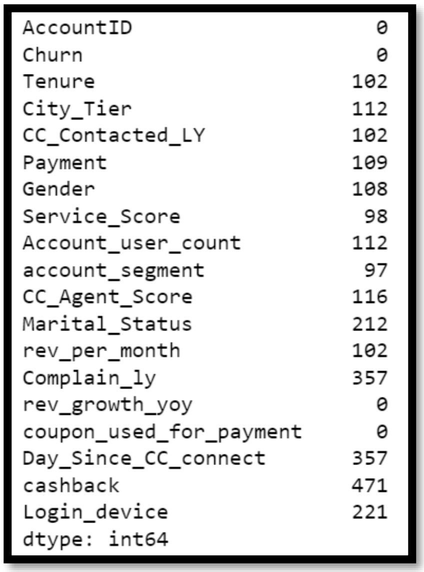
2) EDA, Data Transformation & Business Implication

- The shape of the data is: (11260,19), this signifies that there are a total of 11260 records and 19 variables of which one is a target variable (Churn).
- Below is the info of the data given to me:



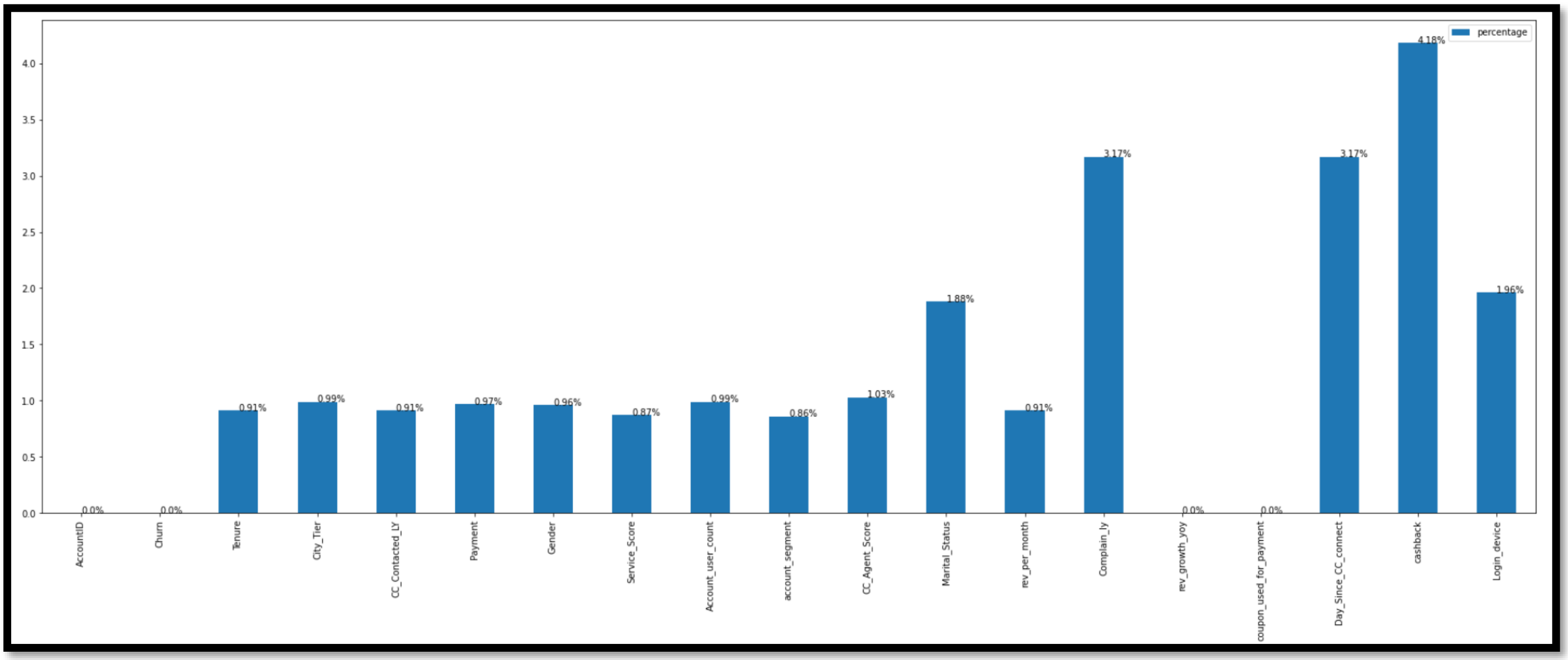
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11260 entries, 0 to 11259
Data columns (total 19 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   AccountID                11260 non-null  int64
 1   Churn                    11260 non-null  int64
 2   Tenure                   11158 non-null  object
 3   City_Tier                11148 non-null  float64
 4   CC_Contacted_LY          11158 non-null  float64
 5   Payment                  11151 non-null  object
 6   Gender                   11152 non-null  object
 7   Service_Score            11162 non-null  float64
 8   Account_user_count       11148 non-null  object
 9   account_segment          11163 non-null  object
 10  CC_Agent_Score           11144 non-null  float64
 11  Marital_Status           11048 non-null  object
 12  rev_per_month            11158 non-null  object
 13  Complain_ly              10903 non-null  float64
 14  rev_growth_yoy           11260 non-null  object
 15  coupon_used_for_payment  11260 non-null  object
 16  Day_Since_CC_connect     10903 non-null  object
 17  cashback                 10789 non-null  object
 18  Login_device             11039 non-null  object
dtypes: float64(5), int64(2), object(12)
memory usage: 1.6+ MB
```

- From the above, we can observe that the different datatypes are int64, object and float64. There are also some missing values which I will depict below.
- Below are the missing values per variable:

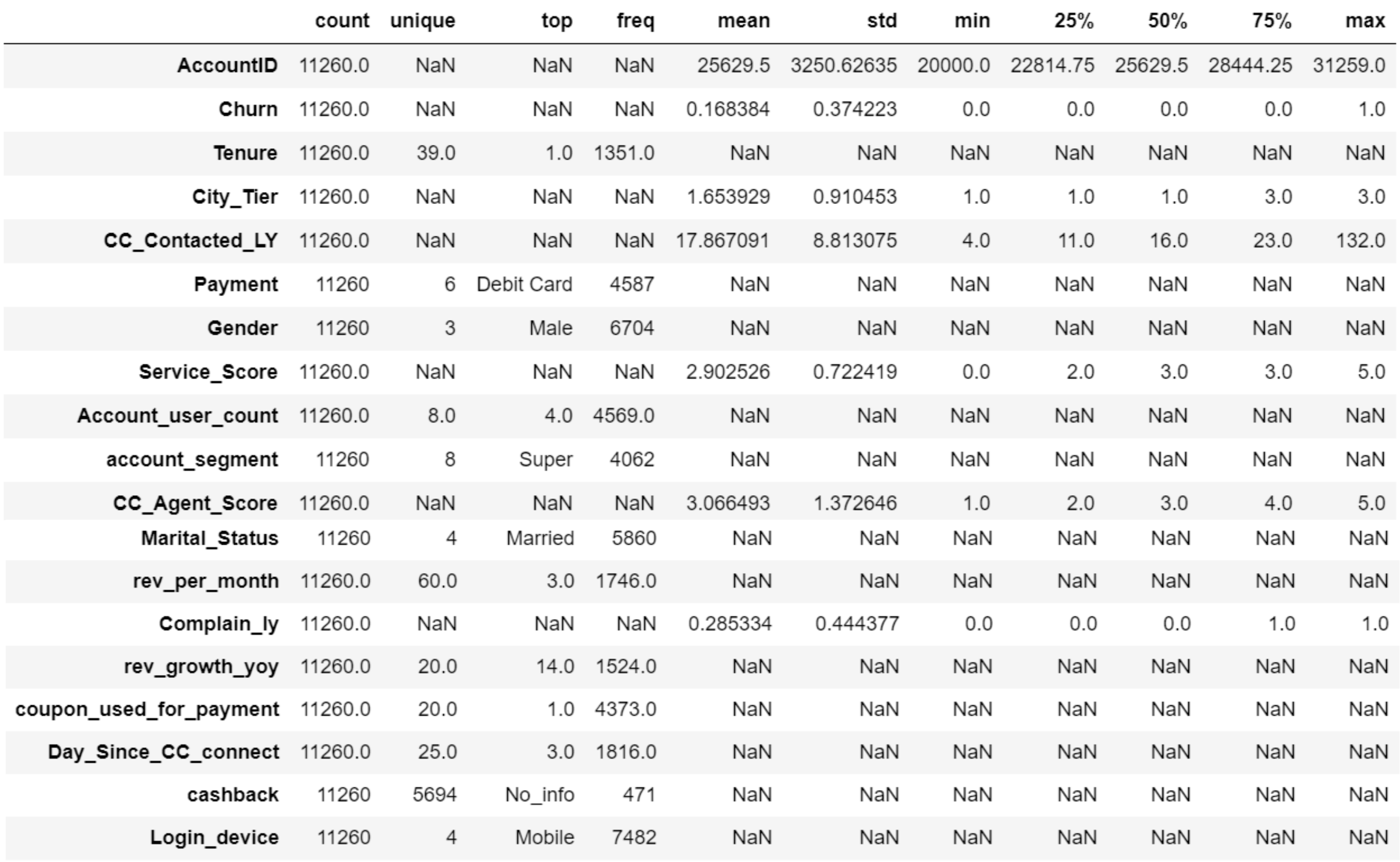


```
AccountID                    0
Churn                        0
Tenure                     102
City_Tier                  112
CC_Contacted_LY            102
Payment                    109
Gender                     108
Service_Score               98
Account_user_count         112
account_segment             97
CC_Agent_Score             116
Marital_Status             212
rev_per_month              102
Complain_ly                357
rev_growth_yoy               0
coupon_used_for_payment      0
Day_Since_CC_connect       357
cashback                   471
Login_device               221
dtype: int64
```

- Below is the percent wise missing values for each variable:



- The number of duplicate rows is 0, which means ever record in the dataset is unique.

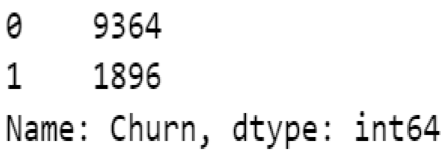


	count	unique	top	freq	mean	std	min	25%	50%	75%	max
AccountID	11260.0	NaN	NaN	NaN	25629.5	3250.62635	20000.0	22814.75	25629.5	28444.25	31259.0
Churn	11260.0	NaN	NaN	NaN	0.168384	0.374223	0.0	0.0	0.0	0.0	1.0
Tenure	11260.0	39.0	1.0	1351.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
City_Tier	11260.0	NaN	NaN	NaN	1.653929	0.910453	1.0	1.0	1.0	3.0	3.0
CC_Contacted_LY	11260.0	NaN	NaN	NaN	17.867091	8.813075	4.0	11.0	16.0	23.0	132.0
Payment	11260	6	Debit Card	4587	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Gender	11260	3	Male	6704	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Service_Score	11260.0	NaN	NaN	NaN	2.902526	0.722419	0.0	2.0	3.0	3.0	5.0
Account_user_count	11260.0	8.0	4.0	4569.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
account_segment	11260	8	Super	4062	NaN	NaN	NaN	NaN	NaN	NaN	NaN
CC_Agent_Score	11260.0	NaN	NaN	NaN	3.066493	1.372646	1.0	2.0	3.0	4.0	5.0
Marital_Status	11260	4	Married	5860	NaN	NaN	NaN	NaN	NaN	NaN	NaN
rev_per_month	11260.0	60.0	3.0	1746.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Complain_ly	11260.0	NaN	NaN	NaN	0.285334	0.444377	0.0	0.0	0.0	1.0	1.0
rev_growth_yoy	11260.0	20.0	14.0	1524.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
coupon_used_for_payment	11260.0	20.0	1.0	4373.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Day_Since_CC_connect	11260.0	25.0	3.0	1816.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
cashback	11260	5694	No_info	471	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Login_device	11260	4	Mobile	7482	NaN	NaN	NaN	NaN	NaN	NaN	NaN

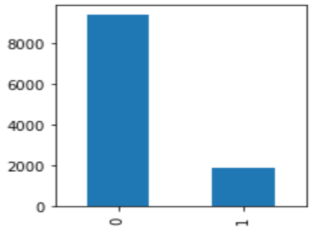
- From above, key observations are that:
    1) The mean of ‘CC_Agent_Score’ that is the satisfaction score given by customers of the account on customer care service provided by company is 3.066493.
    2) The mean of ‘Service_Score’ which is the satisfaction score given by customers of the account on service provided by company is 2.902526.

- I always look for missing values and try to handle them. The dataset we are using does contain some missing values. Since outliers exist in the dataset, I will be imputing such values using median.

- Target Variable:

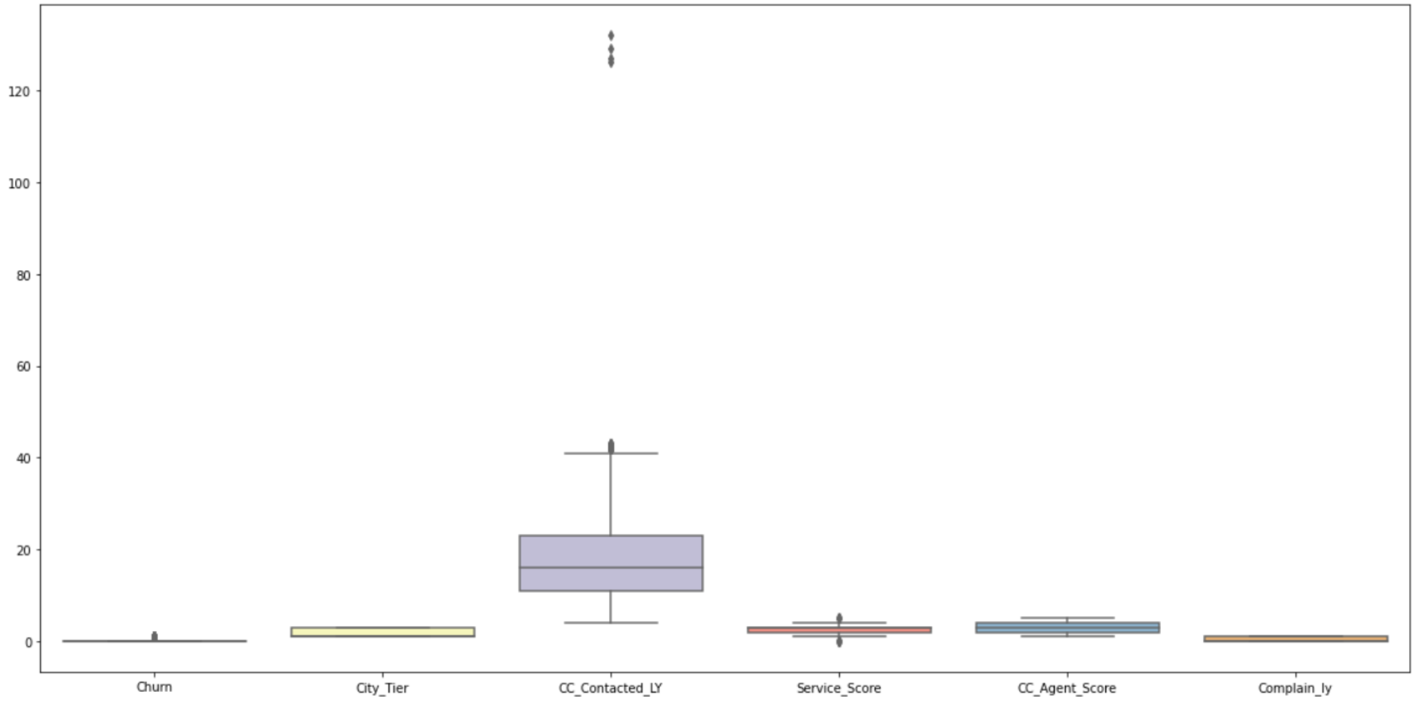


```
0    9364
1    1896
Name: Churn, dtype: int64
```

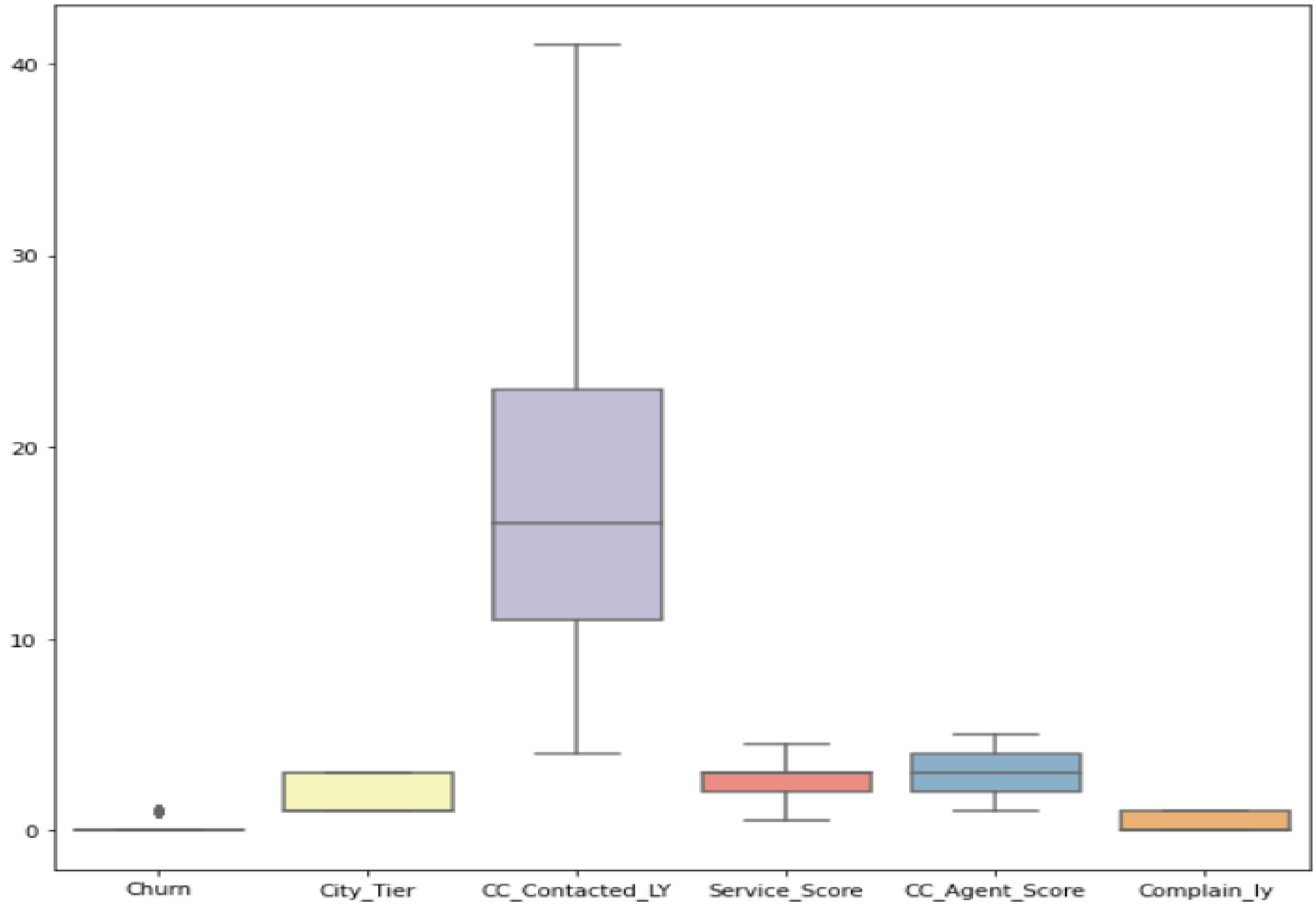


Target variable has imbalanced class distribution. Positive class (Churn=1) is much less than negative class. (Churn=0). Imbalanced class distributions influence the performance of a machine learning model negatively. I will use upsampling or downsampling to overcome this issue. It is always beneficial to explore the features (independent variables) before trying to build a model.

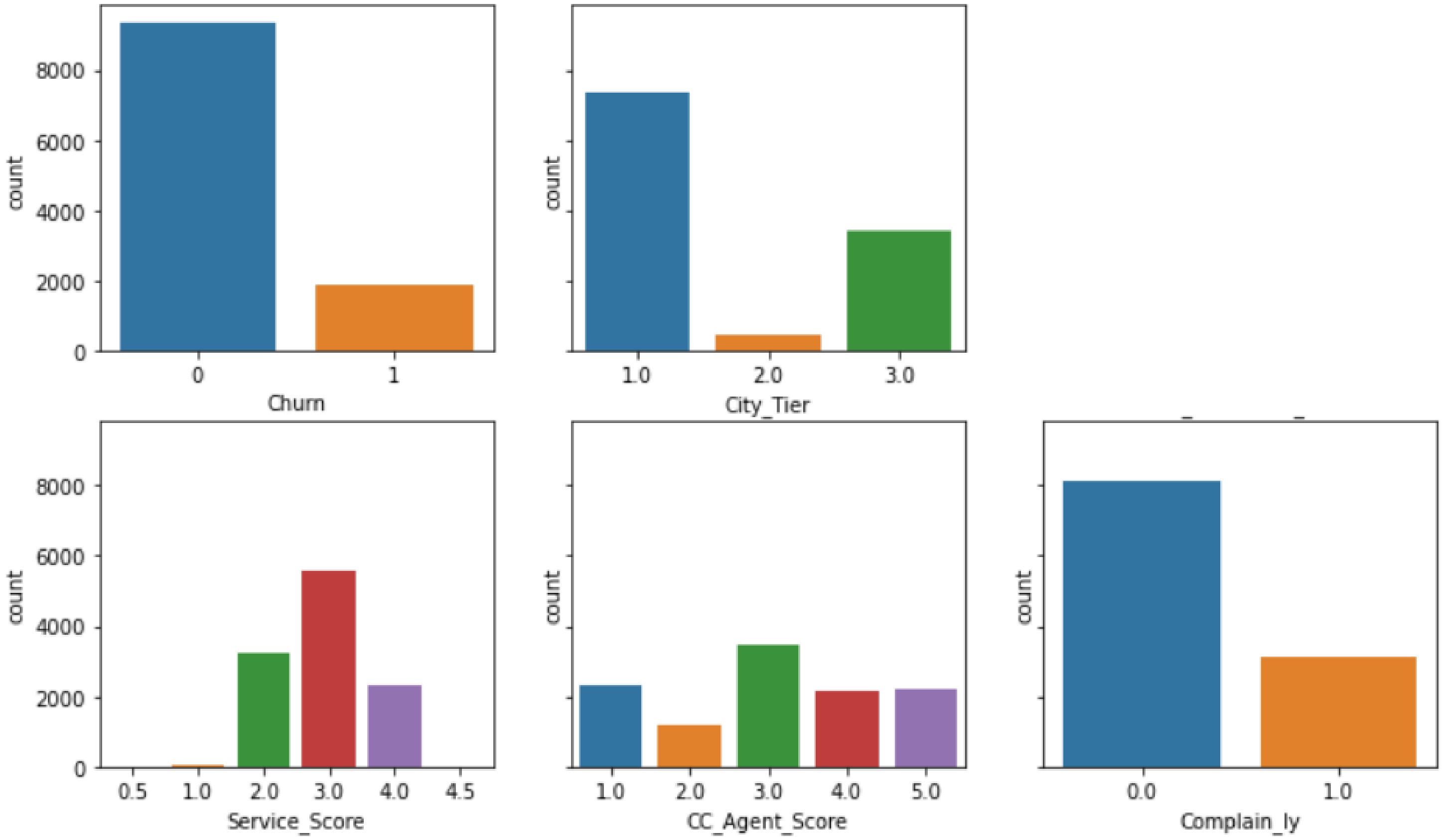
- **Univariate Analysis:**



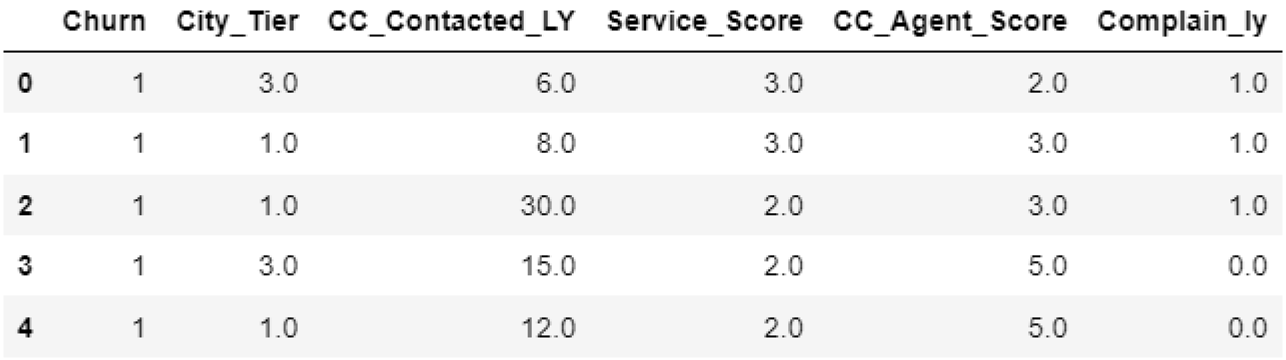
- From the above plots, we observe that outliers exist in ‘CC_Contacted_LY’ and ‘Service_Score’. So, I treat it and obtain the below plot after the outlier treatment:
-



- Bar-plots for various numeric variables:

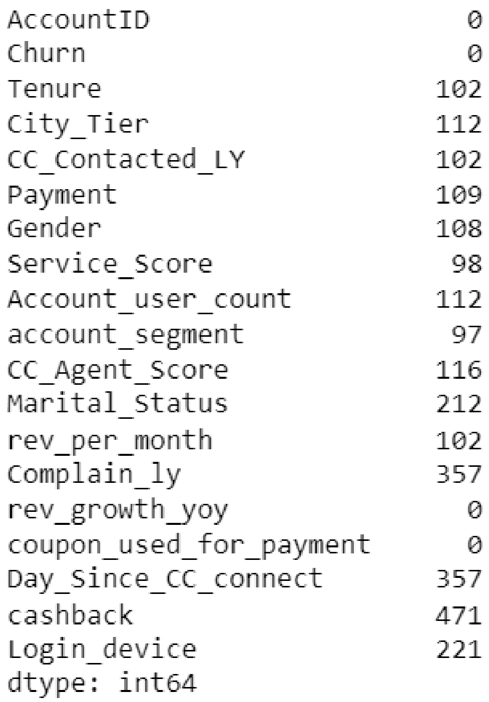


- Below is the output for head of the numerical dataset:

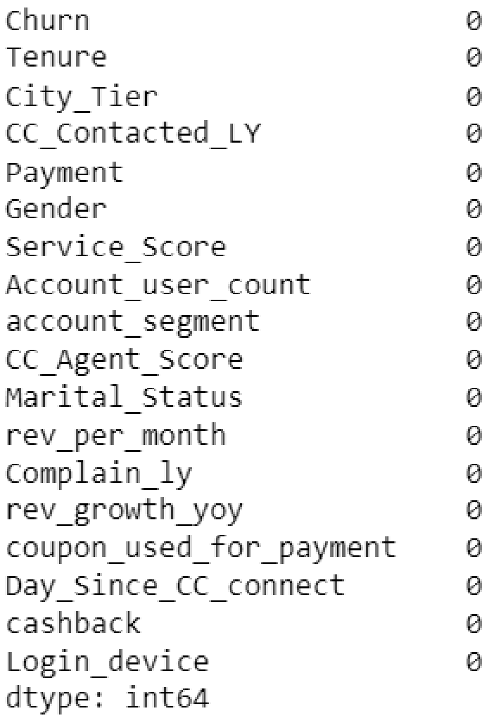


	Churn	City_Tier	CC_Contacted_LY	Service_Score	CC_Agent_Score	Complain_ly
0	1	3.0	6.0	3.0	2.0	1.0
1	1	1.0	8.0	3.0	3.0	1.0
2	1	1.0	30.0	2.0	3.0	1.0
3	1	3.0	15.0	2.0	5.0	0.0
4	1	1.0	12.0	2.0	5.0	0.0

- Missing Values before and after Imputing with median:

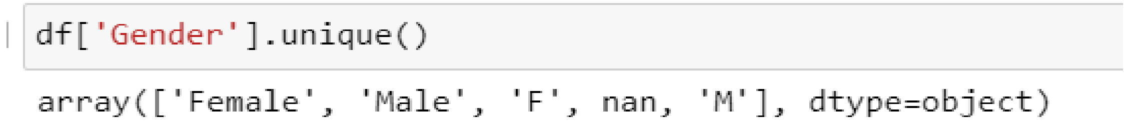


```
AccountID                   0
Churn                       0
Tenure                    102
City_Tier                 112
CC_Contacted_LY           102
Payment                   109
Gender                    108
Service_Score              98
Account_user_count        112
account_segment            97
CC_Agent_Score            116
Marital_Status            212
rev_per_month             102
Complain_ly               357
rev_growth_yoy              0
coupon_used_for_payment     0
Day_Since_CC_connect      357
cashback                  471
Login_device              221
dtype: int64
```



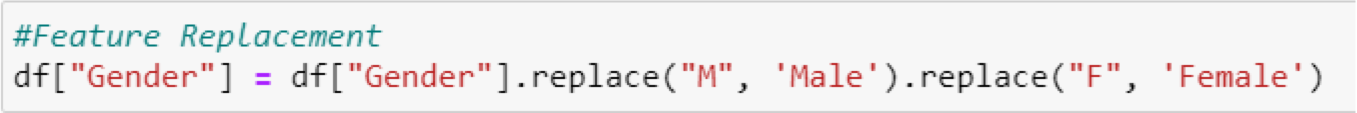
```
Churn                      0
Tenure                     0
City_Tier                  0
CC_Contacted_LY            0
Payment                    0
Gender                     0
Service_Score              0
Account_user_count         0
account_segment            0
CC_Agent_Score             0
Marital_Status             0
rev_per_month              0
Complain_ly                0
rev_growth_yoy             0
coupon_used_for_payment    0
Day_Since_CC_connect       0
cashback                   0
Login_device               0
dtype: int64
```

- I do not require ‘AccountID' for my analysis and prediction, so I have removed it.
- Since ‘Male’,’Female’,’M’,’F’ exist in the gender column, I convert ‘F’ and ‘M’ to ‘Female’ and ‘Male’ respectively, since each of these mean Female and Male, and are just unwanted additions to the category.



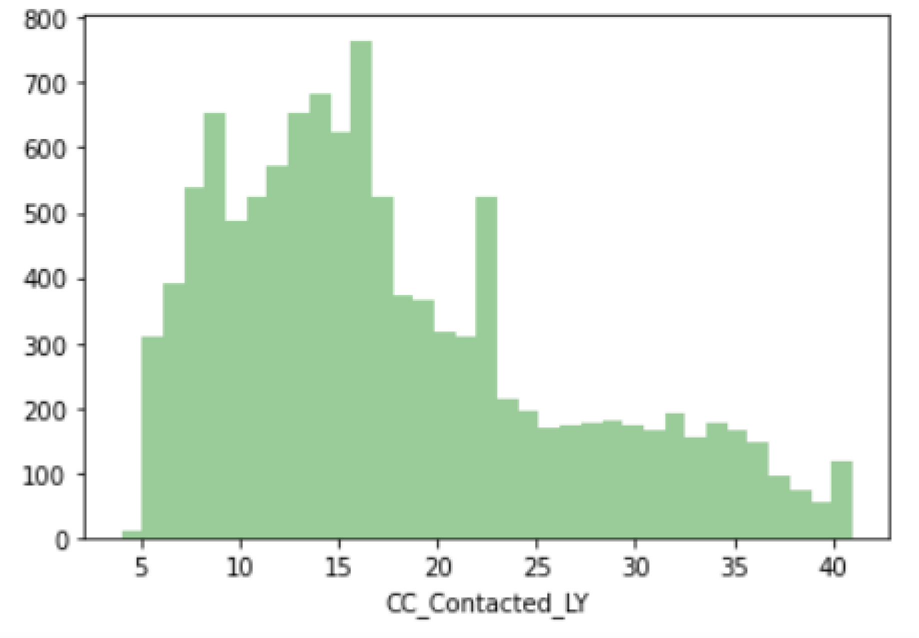
```
df['Gender'].unique()

array(['Female', 'Male', 'F', nan, 'M'], dtype=object)
```

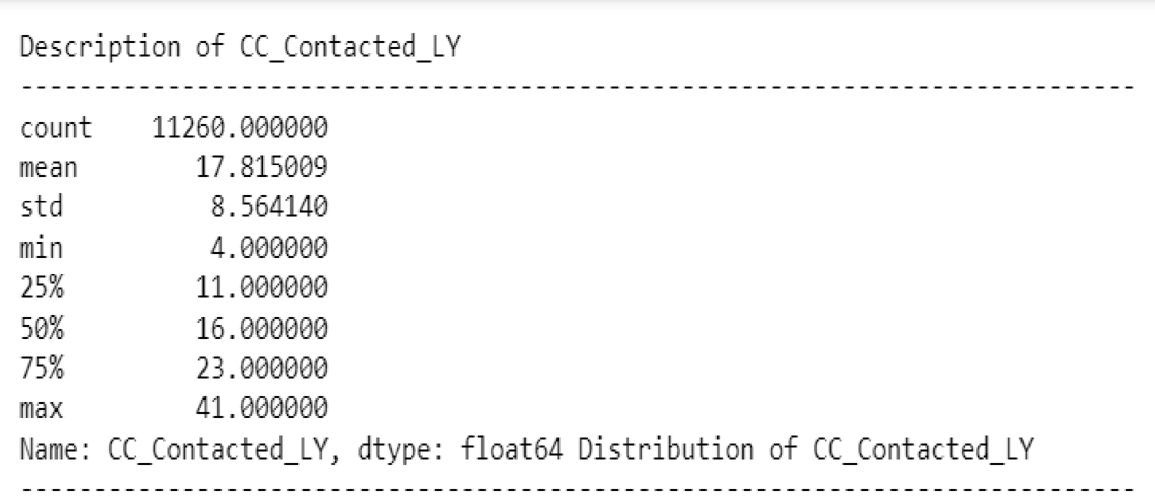


```
#Feature Replacement
df["Gender"] = df["Gender"].replace("M", 'Male').replace("F", 'Female')
```

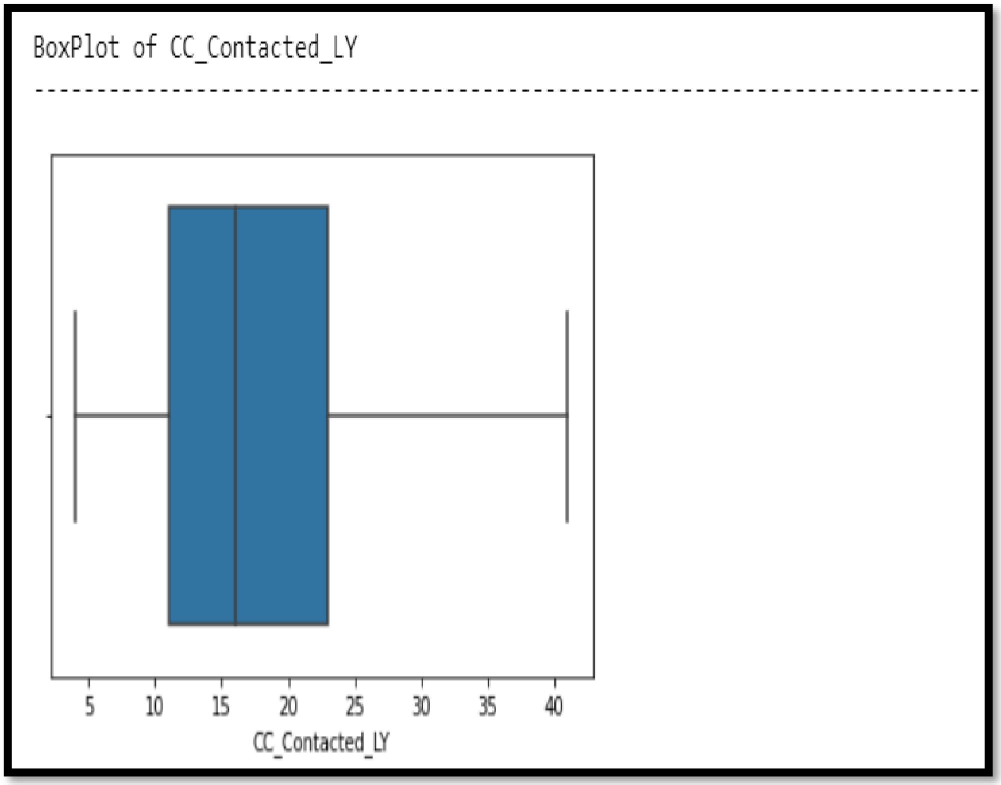
- CC_Contacted_LY:
  It is the number of times all the customers of the account have contacted customer care in last 12 months.



From the above plot we can observe that this does not follow a normal distribution.



```
Description of CC_Contacted_LY
----------------------------------------------------------------------------
count    11260.000000
mean        17.815009
std          8.564140
min          4.000000
25%         11.000000
50%         16.000000
75%         23.000000
max         41.000000
Name: CC_Contacted_LY, dtype: float64 Distribution of CC_Contacted_LY
----------------------------------------------------------------------------
```

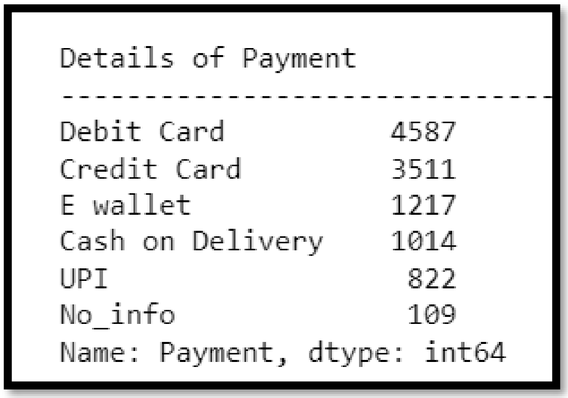


```
BoxPlot of CC_Contacted_LY
-----------------------------------------------------------------------
```

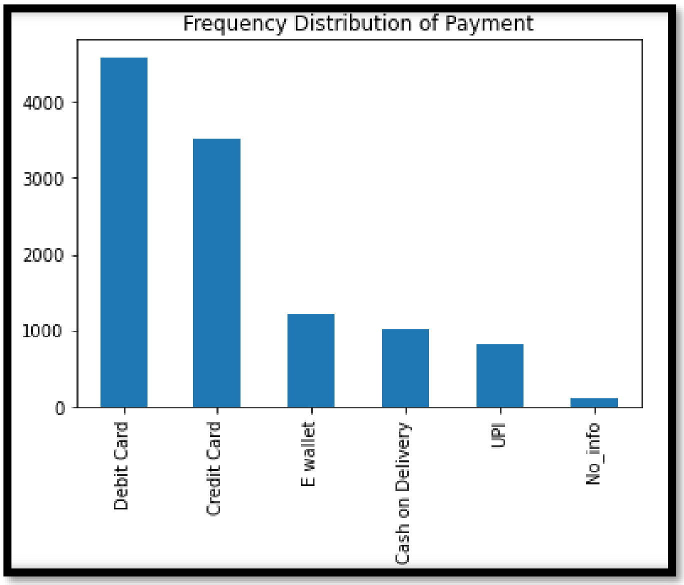
The outliers which were previously existing in this data, have been treated.

- Categorical Variables:

1) Details of Payment:

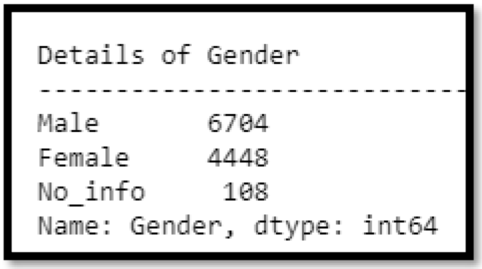


```
Details of Payment
--------------------------------
Debit Card          4587
Credit Card         3511
E wallet            1217
Cash on Delivery    1014
UPI                  822
No_info              109
Name: Payment, dtype: int64
```

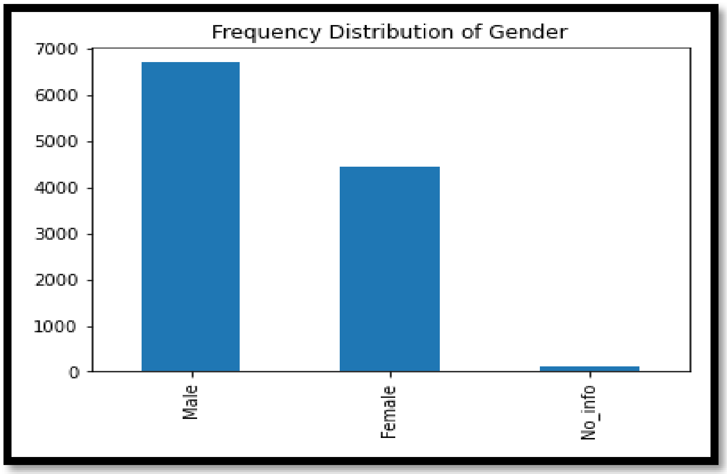


Frequency Distribution of Payment

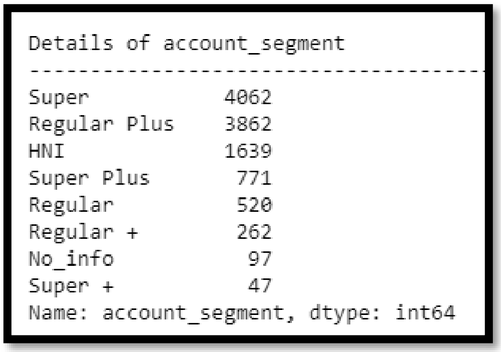
2) Details of Gender:



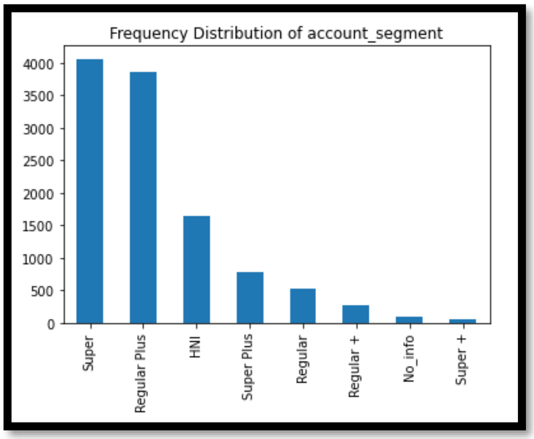
```
Details of Gender
----------------------------
Male        6704
Female      4448
No_info      108
Name: Gender, dtype: int64
```



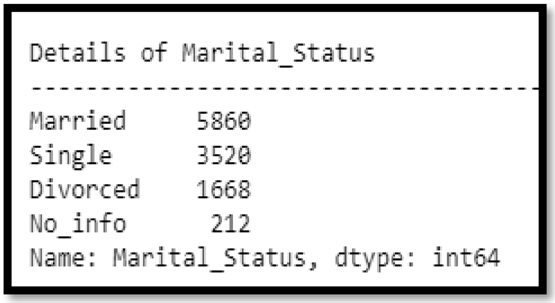
3) Details of Account_Segment:



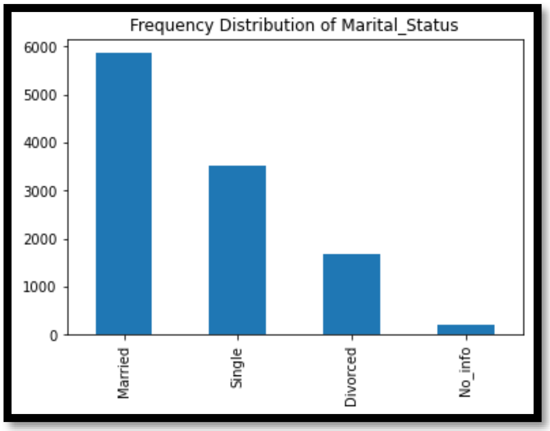
```
Details of account_segment
--------------------------------------
Super           4062
Regular Plus    3862
HNI             1639
Super Plus       771
Regular          520
Regular +        262
No_info           97
Super +           47
Name: account_segment, dtype: int64
```



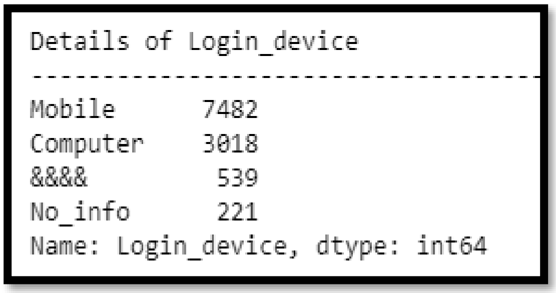
4) Details of Marital_Status:



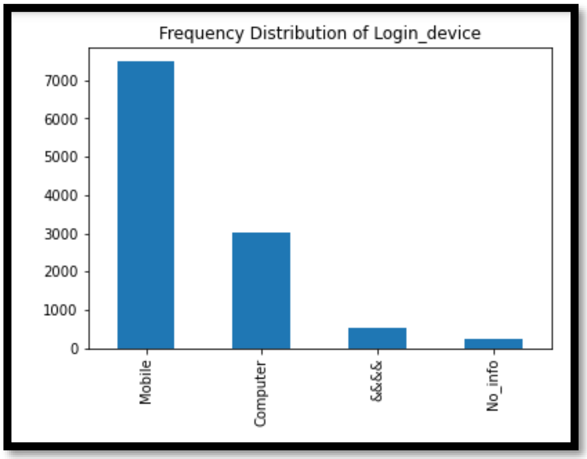
```
Details of Marital_Status
------------------------------------
Married     5860
Single      3520
Divorced    1668
No_info      212
Name: Marital_Status, dtype: int64
```



5) Details of Login_Device:



```
Details of Login_device
-----------------------------------
Mobile      7482
Computer    3018
&&&&         539
No_info      221
Name: Login_device, dtype: int64
```



- <u>Conclusions:</u>

1) In the initial dataset, only about 17% of people have churned.

2) Target Variable has imbalanced distribution.

3) Gender is the least important feature to predict 'Churn'.

4) Majority of the payments are done by the Debit Card (Count: 4587) while minimum payment is done using UPI (822). There is no information about 109 payments from the dataset provided.

5) A higher number of males exist in the given dataset (Count: 6704) while there is no information about the gender of 108 customers.

6) Super, Regular Plus and HNI take up the major account segment, while no information about 97 account segments is mentioned.

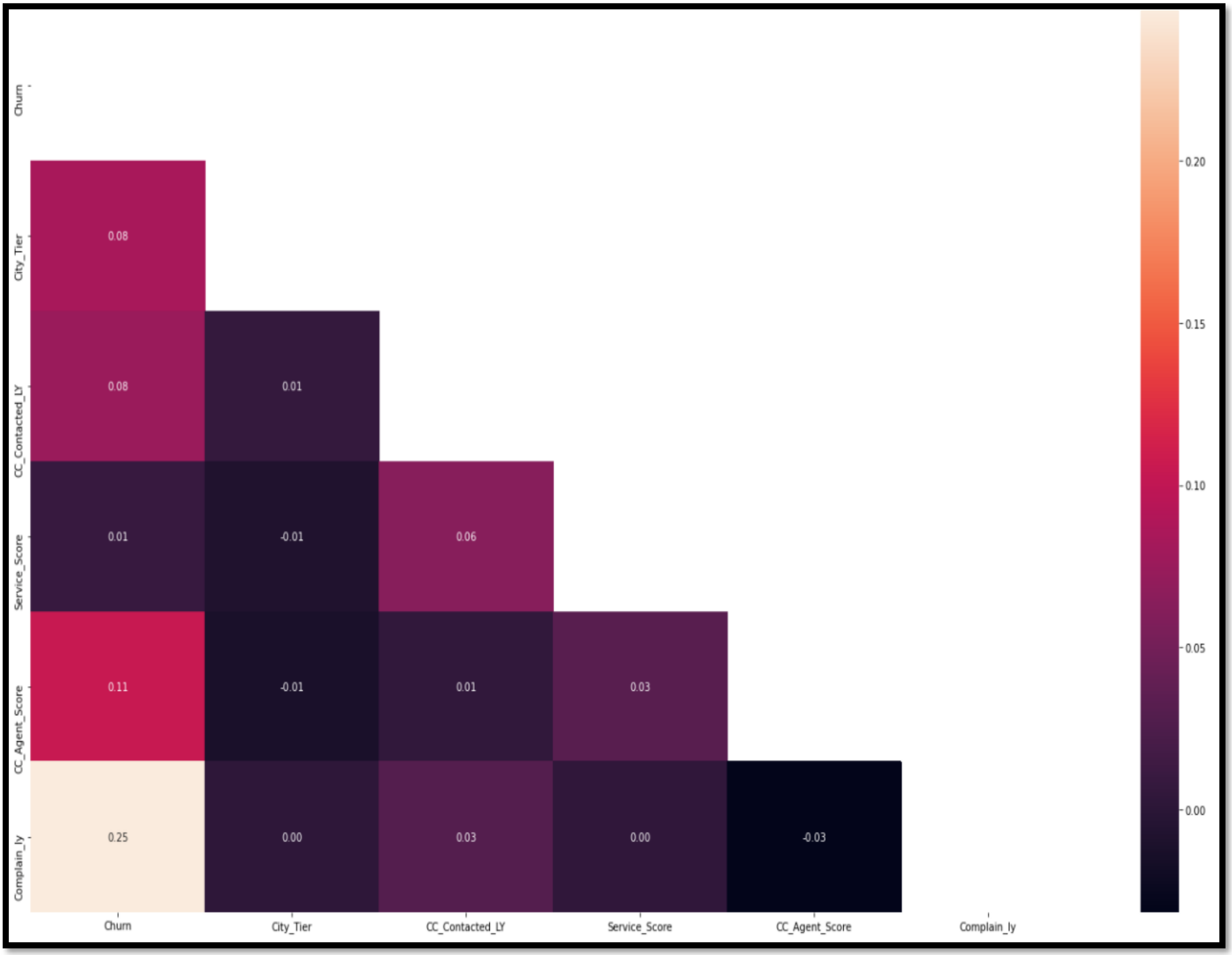
7) About 5860 customers are married, covering a major segment, while there are 3520 single customers and 1668 divorced customers. There is no mention of the marital status of 212 customers.

8) Majorly customers use mobile phone as the login device (Count: 7482).

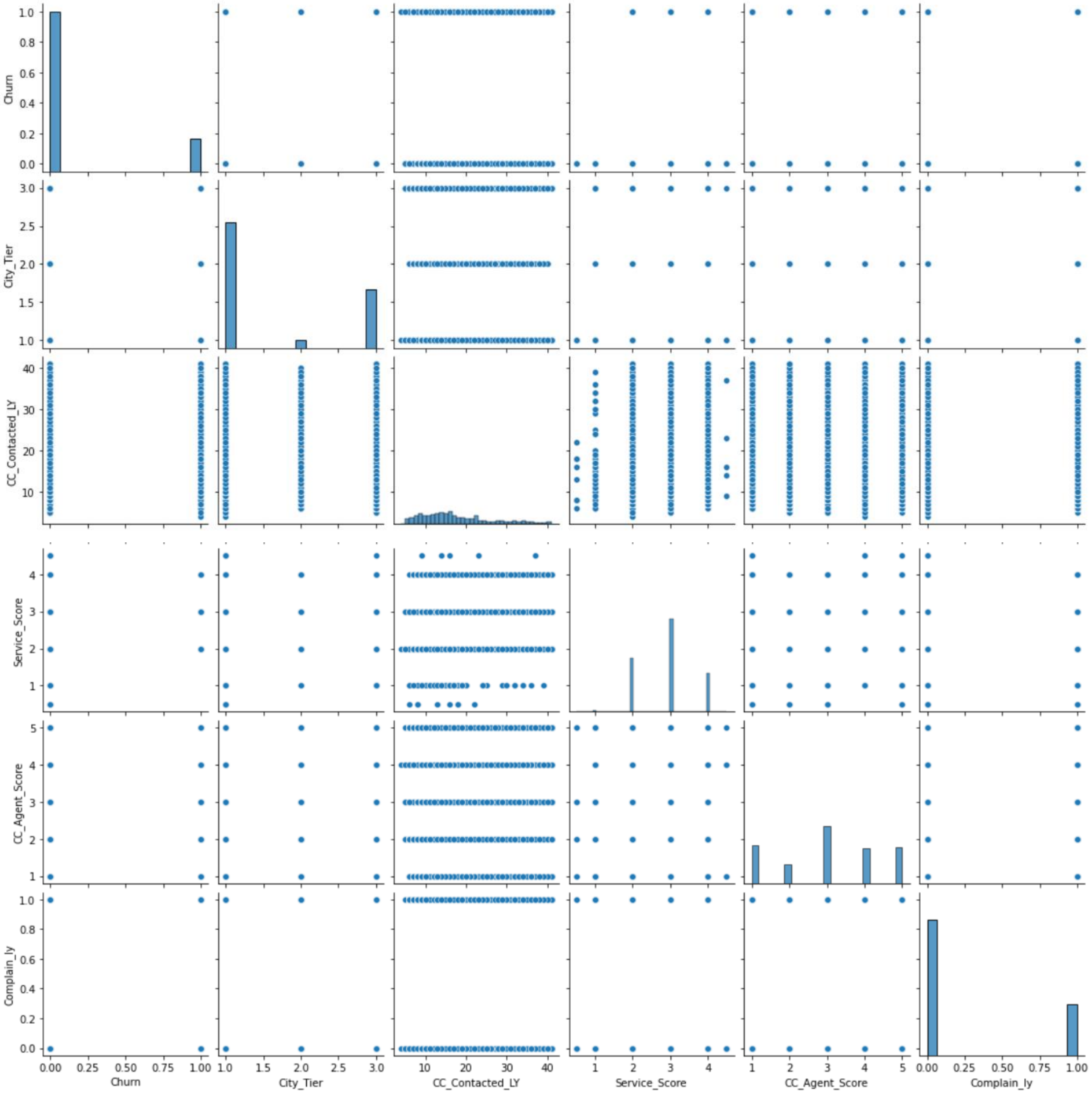
9) Mobile is a more frequently used login device for the E-commerce website.

- **Bivariate Analysis:**

1) Heatmap:



2) Pairplot:


Churn
1.0
0.8
0.6
0.4
0.2
0.0
City_Tier
3.0
2.5
2.0
1.5
1.0
CC_Contacted_LY
40
30
20
10
Service_Score
4
3
2
1
CC_Agent_Score
5
4
3
2
1
Complain_ly
1.0
0.8
0.6
0.4
0.2
0.0
0.00 0.25 0.50 0.75 1.00
Churn
1.0 1.5 2.0 2.5 3.0
City_Tier
10 20 30 40
CC_Contacted_LY
1 2 3 4
Service_Score
1 2 3 4 5
CC_Agent_Score
0.00 0.25 0.50 0.75 1.00
Complain_ly

- Conclusions:

  1) We observe that churn (target variable) is not very highly correlated with any of the independent variables.
  2) Significant variables impacting 'Churn': Tenure and Complain_1y, which means the business should retain their customers by giving them long term contracts and parallelly work on the ways they handle the customer requests.
  3) Churn is least correlated to the 'Service_Score' variable. This means that although satisfaction score is important, but it is not a dominating factor in the customer churn. This could mean that the customers are actually giving a better satisfaction rating than actually what they intend to give and it could be because of various reasons.

3) Business Insights from EDA

- From the heatmap, it is evident that 'Complain_ly' has some impact on the customer churn. This means that the way in which the Customer Care handles the customer requests is critical. The company should focus on providing a better Customer Care and minimize the need for customers to contact the same. The Satisfaction score of customers can be improved via staff training, so company should invest their efforts in giving the staff a proper training so that the way customer requests are handled is enhanced.

- Target variable has imbalanced class distribution. Positive class (Churn=1, customer leaving) is much less than negative class. (Churn=0). Imbalanced class distributions influence the performance of a machine learning model negatively. So, More data should be collected for cases when customers have stopped using the service to make the performance of the machine learning model better thereby obtaining better suggestions on areas of improvement. An analysis should be carried out for the people who have been least interactive with the website and surveys should be sent out to them so that a better and balanced data can be prepared.

- Segment-wise analysis can be performed (Clustering can be applied to group the customer into segments by understanding their behaviours from the dataset) to determine the factors that affect churn differently in different segments.

- Since most of the payment is done by debit cards and credit cards certain schemes and offers can be released for payments by Debit and Credit cards by the E-commerce website for attractive deals, this would increase the customer participation.

- Super, Regular Plus and HNI are the three major segments, so certain strategies can be built segment wise to retain customers from these segments as they take a major portion of the subscriptions.

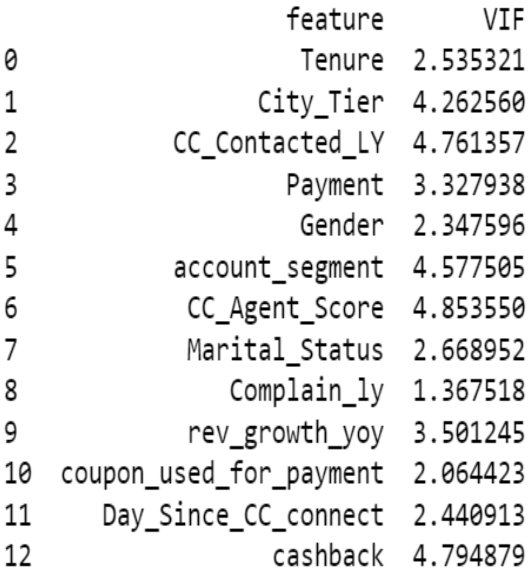
- Since customers majorly use the E-Commerce website on mobile, certain features like notifications can be prompted on the mobile phones of the users whenever an exciting deal is about to be released by the company. This would make the application more interactive.

- If a customer shows signs of churn risk, engagement should be engaged with the Customer Support Management team to increase the communication that might be missing otherwise.

4) Data Pre-processing

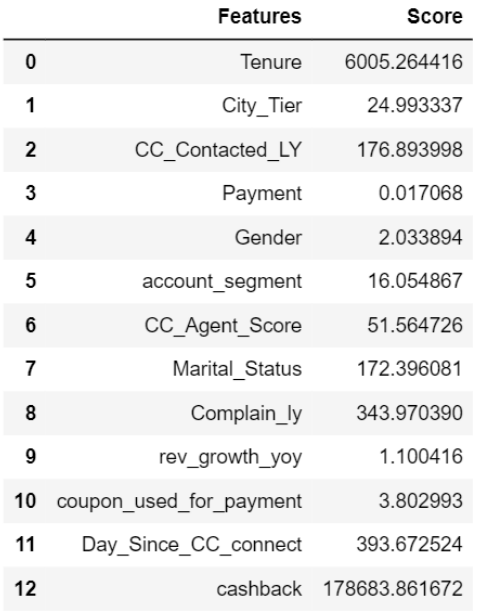
Here, I perform feature selection which shows the most important and significant features between all the features and can be used to get most required business insights. I have used two methods here: VIF and Recursive elimination to obtain below features for model building:

Using VIF:



```
                  feature       VIF
0                  Tenure  2.535321
1               City_Tier  4.262560
2          CC_Contacted_LY  4.761357
3                 Payment  3.327938
4                  Gender  2.347596
5         account_segment  4.577505
6           CC_Agent_Score  4.853550
7           Marital_Status  2.668952
8             Complain_ly  1.367518
9           rev_growth_yoy  3.501245
10 coupon_used_for_payment  2.064423
11     Day_Since_CC_connect  2.440913
12                cashback  4.794879
```

Using Recursive Feature Elimination:



	Features	Score
0	Tenure	6005.264416
1	City_Tier	24.993337
2	CC_Contacted_LY	176.893998
3	Payment	0.017068
4	Gender	2.033894
5	account_segment	16.054867
6	CC_Agent_Score	51.564726
7	Marital_Status	172.396081
8	Complain_ly	343.970390
9	rev_growth_yoy	1.100416
10	coupon_used_for_payment	3.802993
11	Day_Since_CC_connect	393.672524
12	cashback	178683.861672

From the above, I obtain a total of 13 features which are the most significant for predicting ‘churn’, so I have dropped other features and used only these 13 for my analysis.

I have already treated the missing values using median and the same has been demonstrated earlier. Median has been used, since the data given to me had outliers and mean is affected by outliers, so imputing with mean would not give us proper results.

There were no duplicates in the dataset, so the data given was clean.

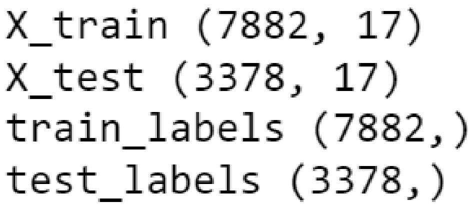
There is one modification done on the ‘Gender variable’ which has been shown earlier in the report as a part of EDA, the data from all the remaining variables have been used as given to me.

Outliers present have been treated accordingly, so our data is prepared for model training and evaluation.

No variables were added in the dataset separately.

5) Model Building

I first extract the target column into separate vectors for training set and test set. The target column here is churn.
After this I perform a split on the data splitting it into 30% for the test data and 70% as the train data as below:



```
X_train (7882, 17)
X_test (3378, 17)
train_labels (7882,)
test_labels (3378,)
```

There are a total of 7882 records for the train data and 3378 records for the test data.

I have prepared the following model procedures to analyse and review the dataset and get t-he performance a nd importance of the features available on the dataset which can gathers more information about the subjects.

Following is the list of model building procedure which will be used in this project:

(1) Logistic Regression: 13 significant features are selected by Recursive Feature Elimination and VIF methods.

(2) Decision Tree: Optimal tree had 18 splits, 'gini' criterion used. Best parameters were obtained by gridsearch algorithm

(3) Random Forest: Optimal model had 11 features

(4) Linear Discriminant Analysis

(5) K Nearest Neighbours: K values from 1 to 18; the optimal model had K as 15s

(6) Naive Bayes

(7) Gradient Boosting

(8) Extreme Gradient Boosting

For every model building precure, I have gone through the following steps:

(a) Model Prediction

(b) Model Performance

(c) ROC-AUC Graph

(d) Model Performance Metrics

- Inferences from the Logistic Regression Model:

1) Recall is the percentage of users that end up churning that the algorithm successfully finds. In our case, there is a low recall of 0.34 on train data and 0.36 on the test data which is again on the lower side. This means that our algorithm is not giving a decent percentage of customers being churned actually.
2) Precision tells us of all the users that the algorithm predicts will churn, how many of them do churn. From the above model, there is a decent precision obtained for customers who would churn for train and test data respectively (0.76 and 0.74).
3) The AUC for train and test are 0.836 and 0.84 respectively, which means that we have obtained a nice model for our prediction of customer churn (since the train and test AUC values are nearly equal)
4) Both the test and train curves hug the upper left corner and have very strong AUC values. With such strong models, we can now turn our eyes to tuning some model parameters/hyperparameters to slowly elevate our scores.
5) From the confusion matrix, we observe that a total of 6861 records were predicted accurately out of the total 7882 records for the train data. While, a total of 2943 records were accurately predicted out of the total 3378 records for the test data. From this it is evident that the model did perform better on the test data.
6) The false positive rate is essentially a measure of how often a "false alarm" will occur — or, how often an actual negative instance will be classified as positive. In our model the number of false positives are 141 and 71 on train and test data respectively.
7) The False negative values are 880 and 364 respectively for train and test data.

- Inferences from Decision Tree:

1) From the feature importance values, we observe that Tenure is the most important feature and Gender is the least important feature.
2) While using the Gini criterion to build our CART tree, we observe a good model score for train and test data, being 0.916 and 0.905 respectively. It is a more effective model than logistic regression and the train and test model scores are nearly equal and higher than the logistic regression.
3) The values of AUC for train and test data are 0.895 and 0.891 respectively. A higher AUC means better is the model's performance, and in this case we have obtained a good accuracy both in case of train and test data.
4) Precision for the predications are 0.71 and 0.7 for train and test data respectively. While these values are close to each other and seem to be decent, they are lower than the values obtained in the logistic regression model. So, it appears to be a little less precise as compared to the logistic regression model, however it has a better accuracy.
5) Recall is another key metric for our analysis and in this case, we have a much higher recall of 0.61 and 0.6 for the train and test data as compared to logistic regression model.
6) Overall, the metrics for this model seem to be better than the logistic regression model.

- <u>Inferences from Random Forest Model:</u>

1) From the model built, tenure is a highly important feature for predictions while gender is the least important feature. Which means that the customer is greatly impacted by the tenure while gender has a minimal effect on it.
2) The AUC for train and test data is 0.96 and 0.935 respectively. This so far has been the highest AUC between all the models so far. The accuracy is really high for both train and test data and the model seems to be a good predictor.
3) The precisions for train and test data are 0.86 and 0.82 respectively, which is again the highest between all the models so far, making it a better model.
4) The recall obtained for train and test are 0.65 and 0.63 respectively. While this is a much better recall from the previous models, the recall can still be enhanced for better predictions.
5) Overall, the model has correctly made 9148 predictions from a total dataset of 11260 records.
6) The False positives for train and test data are 139 and 77 respectively, which are relatively low.
7) So far, this has been the best model from the 3 models built.
8) With the Random Forest model we can actually generate probabilities for each class prediction. So we can basically have the model give us a probability for each customer of how likely the model thinks that customer is to Churn
9) In doing so, we are able to generate a list of customers who are of high value to the company and are at high risk of Churning. These are the customers with which the company would want to intervene in some way to get them to stay.
   s

- <u>Inferences from Linear Discriminant Analysis:</u>

1) The model score for the train and test data are 0.8671 and 0.8676 respectively.
2) A total of 9113 records were predicted accurately from 112s60 records.
3) The AUC for the train and test data are 0.895 and 0.891 respectively. While these AUCs are close and high, as compared to random forest these are a little low.
4) The False Positive values for train and test data are 164 and 87 respectively.
5) The recall value for the train data is 0.33 which is low, however, in case of test data the recall is 0.97. This means that the model is performing well in case of test data, however it performs poorly in train data (in terms of recall).
6) The precisions for train and test data are 0.73 and 0.88 respectively. While the precision values are good, the variation is a lot in train and test data.

- <u>Inferences for K – Nearest Neighbours:</u>

1) The train and test model accuracy obtained is 0.8435 and 0.82889 respectively.
2) The AUC for the train data is 0.8355 and for the test data it is 0.737. While a good AUC is obtained for train data, the test AUC is relatively low. Which means that the model might perform well on the train data but might perform a little poorly on test data.
3) The precisions for train and test data are 0.7 and 0.46 respectively. Here again, the precision is very low for the test data, which means that while the mssodel is giving reliable predictions for the train data, it might not give similar results for the test data.
4) Th recall for the train data is 0.12 and for the test data it is 0.07. The recall values which are significant to us in this case are really low, making the model not an ideal one.
5) Interestingly, this model made 9244 accurate predictions which has been the most as compared to above models.

- Inferences for Gaussian Naive Bayes Model:

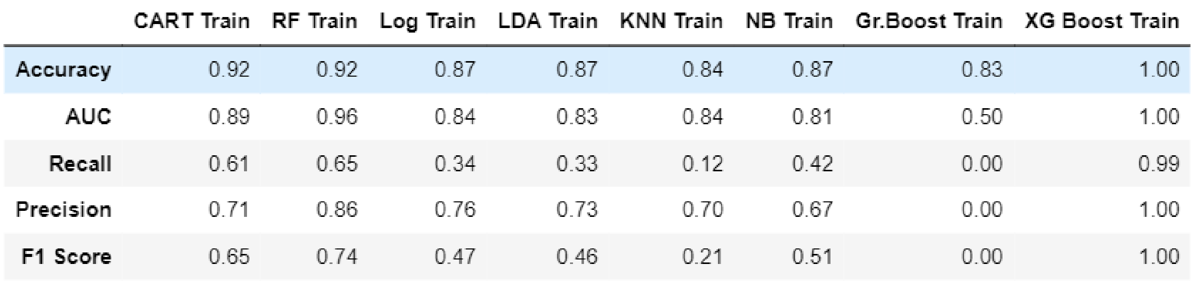
  1) From the confusion matrix, it is clear that a total of 8962 accurate predictions were made by this model, this is the least number of accurate predications in comparison to the other models.
  2) The AUCs for train and test data are 0.808 and 0.804 respectively. While these values are almost the same, making the model perform equally well on both train and test data, the values are lower as compared to other models. Making the other models a better pick.
  3) The precisions for both train and test data is 0.67.
  4) The recalls for train and test data are 0.42 and 0.44 respectively, this still needs to be increased for a better prediction.

- Inferences for Xtreme Gradient Boosting (XGB):

  1) We observe that XGB has increased the accuracy for the model tremendously.
  2) All the parameters are high, making the predictions extremely accurate.
  3) Both, train and test data have a really high precision on 1.0 and 0.94 respectively.
  4) The recalls for train and test are high as we wanted and are equal to 0.99 and 0.86 respectively.
  5) The AUC curves for both train and test data make a perfect curve, having values as 0.9999 and 0.9913 for train and test data respectively.
  6) From the confusion matrix, we observe that a total of 9335 predications are correctly made, with low false positives (0 in train and 29 in test).

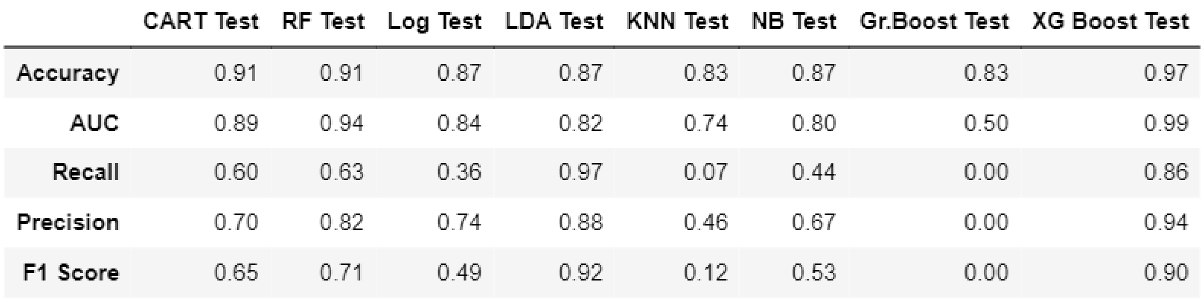
6) Model Performance Comparison

1. Performance metrics on train data:



	CART Train	RF Train	Log Train	LDA Train	KNN Train	NB Train	Gr.Boost Train	XG Boost Train
Accuracy	0.92	0.92	0.87	0.87	0.84	0.87	0.83	1.00
AUC	0.89	0.96	0.84	0.83	0.84	0.81	0.50	1.00
Recall	0.61	0.65	0.34	0.33	0.12	0.42	0.00	0.99
Precision	0.71	0.86	0.76	0.73	0.70	0.67	0.00	1.00
F1 Score	0.65	0.74	0.47	0.46	0.21	0.51	0.00	1.00

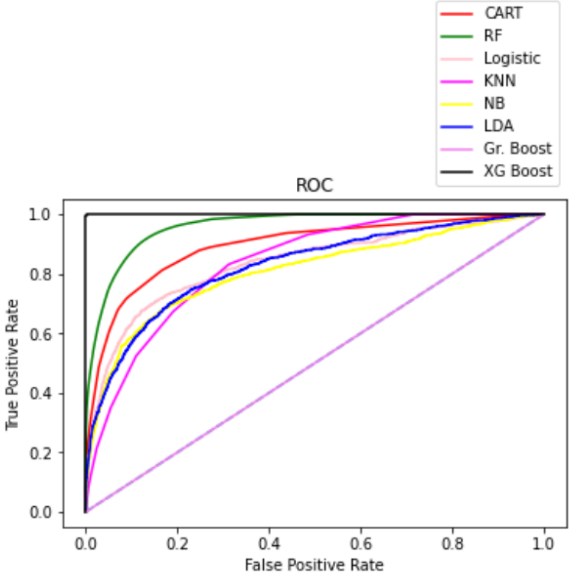
2. Performance metrics on test data:



	CART Test	RF Test	Log Test	LDA Test	KNN Test	NB Test	Gr.Boost Test	XG Boost Test
Accuracy	0.91	0.91	0.87	0.87	0.83	0.87	0.83	0.97
AUC	0.89	0.94	0.84	0.82	0.74	0.80	0.50	0.99
Recall	0.60	0.63	0.36	0.97	0.07	0.44	0.00	0.86
Precision	0.70	0.82	0.74	0.88	0.46	0.67	0.00	0.94
F1 Score	0.65	0.71	0.49	0.92	0.12	0.53	0.00	0.90

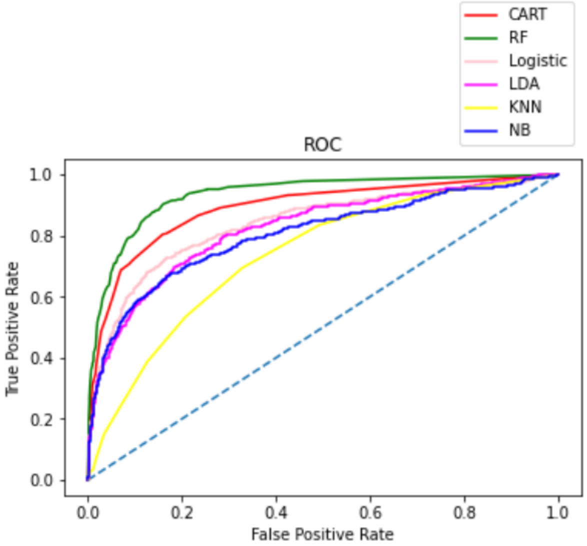
ROC-AUC on train data:

```
<matplotlib.legend.Legend at 0x20a3d695820>
```



ROC-AUC on test data:

```
<matplotlib.legend.Legend at 0x20a3d9c9340>
```



7) Business Recommendations

The purpose of this project was to develop a model that would be able to determine churning clients from a telecom company, as efficiently as possible.

Being able to identify potential churners in advance allows the company to develop strategies to prevent customers from leaving the client base. With this data in hand, companies can offer incentives, like discounts or loyalty programs, or provide additional services in an attempt to reduce the churn rate.

Below are some of the business recommendations, I would like to put forward: ss

1) From the above models, we can clearly observe that XGBoost is the most efficient model out of all the models. The performance metrics: Accuracy, AUC, Recall, Precision and F1 Score are the highest as compared to all the models. The best cut-off has the highest true positive rate together with lowest false positive rate, this scenario is observed in case of XGBoost.

2) While the recall values seem to be overfitted for the XGBoost model I have prepared (this is because of the imbalanced dataset, i.e. more number of data for people who do not churn), SMOTE can be applied to reduce overfitting in the model.

3) From my analysis, we can clearly observe tenure and Complain_1y as two important features for customer churn. This means that the company should take a look at any complaints that have been raised by account in last 12 months and identify how the customer satisfaction and experience can be made better. Once the customer satisfaction is increased, there is a lesser chance that he or she would churn.

4) In addition to the above, Gender is the least important feature to predict the customer churn. This means that Gender is not significant in the customer churn prediction.

5) When we use XGBoost to combine weaker models, we observe an extremely accurate and precise model being generated. The accuracy in case of XGBoost for train data is 1 and for test data is 0.97, making it a highly optimized model. With high precision and recall values we can observe how XGBoost has greatly made the model more reliable and efficient. The best results were obtained by applying XGBoost algorithm, and the business should go ahead with this model for churn prediction.

6) XGBoost model suggests that 493 customers would churn (on the basis of test data) sand since it has a high accuracy, the business can rely on the result and can reach out to these clients with a marketing campaign or even offering them some kind of benefit based on their profiles.

7) If a customer is showing signs of churn risk, it probably is not a great time for sales to reach out with information about additional services the customer might be interested in. Rather, that engagement should be with the CSM so they can help the customer become re-engaged and see value in the products they currently have.

8) Long term contracts should be promoted since tenure of the customers have a huge impact on the churn. So new schemes which can acquire customer and keep them engaged for a long time should be put forward by the company.

9) The company can also look forward to market more products as Combo (multi) service offerings, these new offerings would keep the customers more engaged and they would look to invest more.

10) The company can track its Net Promoter Score (NPS) by gathering information on what their customers like about the product, as well as what they don't. Numeric scores can be assigned and certain products that aren't liked much by the customers can be enhanced.

11) Since most of the users login by mobile phone, an in-app message could be a useful way of showing the customers how to find what they're after. Alternatively, a reminder email can be sent, pointing out all the helpful features they haven't used yet or if they've been inactive for a certain amount of time.