Domain Orientated Case Study – Telecom Churn

Group members:

Hemavathi A.B

NarendraKumar P

Business Problem Overview

- In the telecom industry, customers are able to choose from multiple service providers and actively switch from one operator to another. In this highly competitive market, the telecommunications industry experiences an average of 15-25% annual churn rate. Given the fact that it costs 5-10 times more to acquire a new customer than to retain an existing one, **customer retention** has now become even more important than customer acquisition.
- For many incumbent operators, *retaining high profitable customers is the number one business goal*.
- To reduce customer churn, telecom companies need to **predict which customers are at high risk of churn.**
- In this project, you will analyse customer-level data of a leading telecom firm, build predictive models to identify customers at high risk of churn and identify the main indicators of churn.

Understanding and defining churn

- There are two main models of payment in the telecom industry - **postpaid** (customers pay a monthly/annual bill after using the services) and **prepaid** (customers pay/recharge with a certain amount in advance and then use the services).
- In the postpaid model, when customers want to switch to another operator, they usually inform the existing operator to terminate the services, and you directly know that this is an instance of churn.
- However, in the prepaid model, customers who want to switch to another network can simply stop using the services without any notice, and it is hard to know whether someone has actually churned or is simply not using the services temporarily (e.g. someone may be on a trip abroad for a month or two and then intend to resume using the services again).
- Thus, churn prediction is usually more critical (and non-trivial) for prepaid customers, and the term ‘churn’ should be defined carefully. Also, prepaid is the most common model in India and Southeast Asia, while postpaid is more common in Europe in North America.
- This project is based on the Indian and Southeast Asian market.

Definitions of churn

There are various ways to define churn, such as:

- **Revenue-based churn**: Customers who have not utilised any revenue-generating facilities such as mobile internet, outgoing calls, SMS etc. over a given period of time. One could also use aggregate metrics such as ‘customers who have generated less than INR 4 per month in total/average/median revenue.
- The main shortcoming of this definition is that there are customers who only receive calls/SMSes from their wage-earning counterparts, i.e. they don’t generate revenue but use the services. For example, many users in rural areas only receive calls from their wage-earning siblings in urban areas.
- **Usage-based churn**: Customers who have not done any usage, either incoming or outgoing - in terms of calls, internet etc. over a period of time.
- A potential shortcoming of this definition is that when the customer has stopped using the services for a while, it may be too late to take any corrective actions to retain them. For e.g., if you define churn based on a ‘two-months zero usage’ period, predicting churn could be useless since by that time the customer would have already switched to another operator.
- In this project, you will use the **usage-based definition** to define churn.

High-value churn

- In the Indian and Southeast Asian markets, approximately 80% of revenue comes from the top 20% of customers (called high-value customers). Thus, if we can reduce the churn of high-value customers, we will be able to reduce significant revenue leakage.

- In this project, you will define high-value customers based on a certain metric (mentioned later below) and predict churn only on high-value customers.

Missing Data

- The “telecom_churn_data” had 99,999 rows and 226 columns

- Number of columns/variables that had significant missing values were 166. Only 60 variables had all rows with data.

High Value Customers

- Created new columns using average and total
- telecom['total_rech_data_amt_6'] = telecom['av_rech_amt_data_6'] * telecom['total_rech_data_6']
- telecom['total_rech_data_amt_7'] = telecom['av_rech_amt_data_7'] * telecom['total_rech_data_7']
- telecom['total_rech_data_amt_8'] = telecom['av_rech_amt_data_8'] * telecom['total_rech_data_8']
- telecom['total_rech_data_amt_9'] = telecom['av_rech_amt_data_9'] * telecom['total_rech_data_9’]
- Checked the average recharge done in the first two months(June & July) - the good phase
- Total amount spend would be the sum of total data recharge done & total call/sms recharges
- Took the 70 percentile of the calculated average amount
- 70 percentile is : 478.0
- Now the shape of the filtered dataset is: (30001, 222)
- Hence there are 30,001 rows where the average recharge amount is > 70th percentile for the 6th and 7th month

Checking for Churn customers

- Tagged the churned customers (churn=1, else 0) based on the fourth month: Those who have not made any calls (either incoming or outgoing) AND have not used mobile internet even once in the churn phase.
- The columns we need to use to tag churners are: total_ic_mou_9 ,total_og_mou_9, vol_2g_mb_9, vol_3g_mb_9
- After tagging churners, removed all the attributes corresponding to the churn phase (all attributes having ‘ _9’, etc. in their names).

Churn Percentage

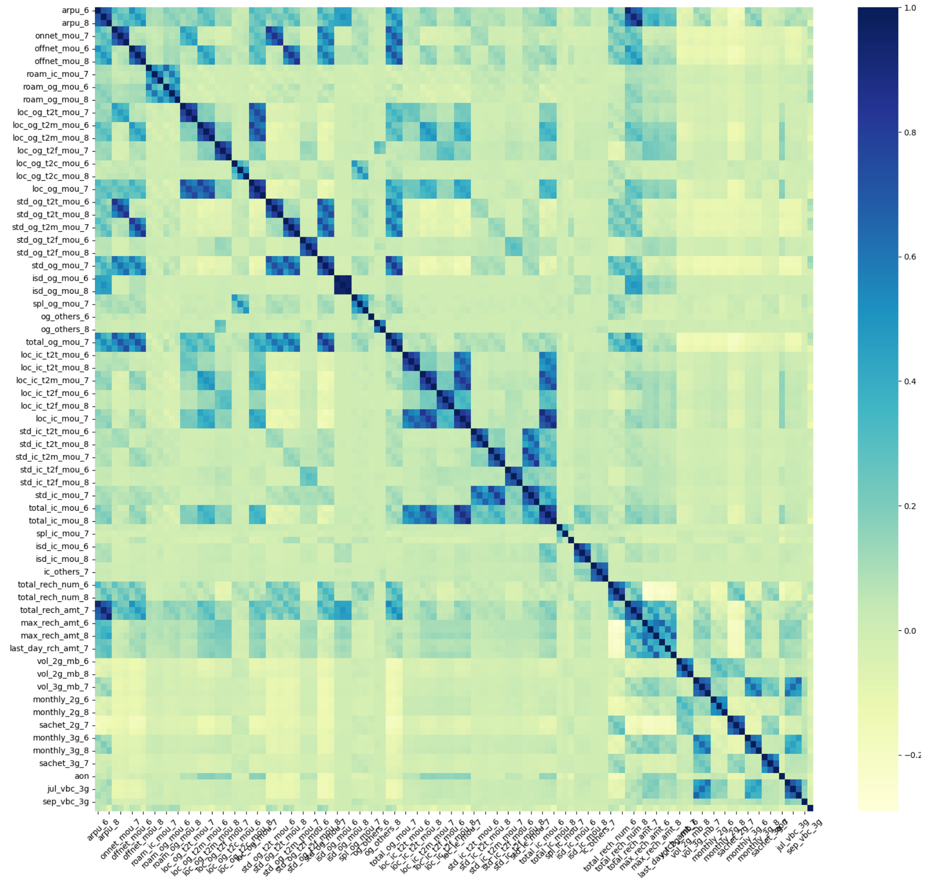
- There are 92% customers who have not churned and 8% customers who have churned.
- This is a clear case of class imbalance as most of the data sits in "Not Churn" and only 8% of the population have moved to different telecom services. Also we are not using the PCA here because,
- PCA transforms your features into principal components, which are linear combinations of the original features. This makes it difficult to interpret the model results, as the coefficients will not directly correspond to the original variables.
- While PCA reduces dimensionality, it introduces new features (the principal components), which can complicate your model development and interpretation.
- In a classification context, PCA can sometimes lead to overfitting, especially if we keep too many components that capture noise rather than signal. Hence we are moving with the classifical approach of identifying a good feature set through RFE, p-value and VIF check.

Churn	Proportion or Percentage
Not Churned	91.86
Churned	8.14

EDA or Feature Engineering

- Checked the columns with no variance in their values and drop such columns.
- Drop Columns with > 30% of missing values.
- Dropped columns that have date values.
- Deleted the 9th month columns because we would predict churn/non-churn later based on data from June, July and August months.
- Dropped rows that had null values.
- Final we have 28,504 rows and 126 columns which are still a very good amount of data to do analysis.

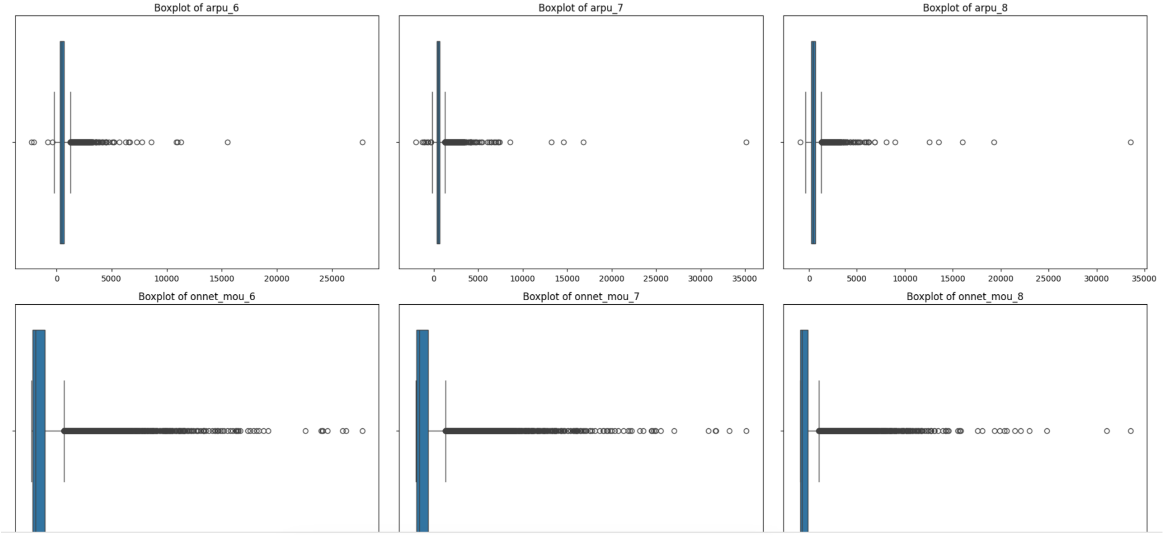
Data Visualization

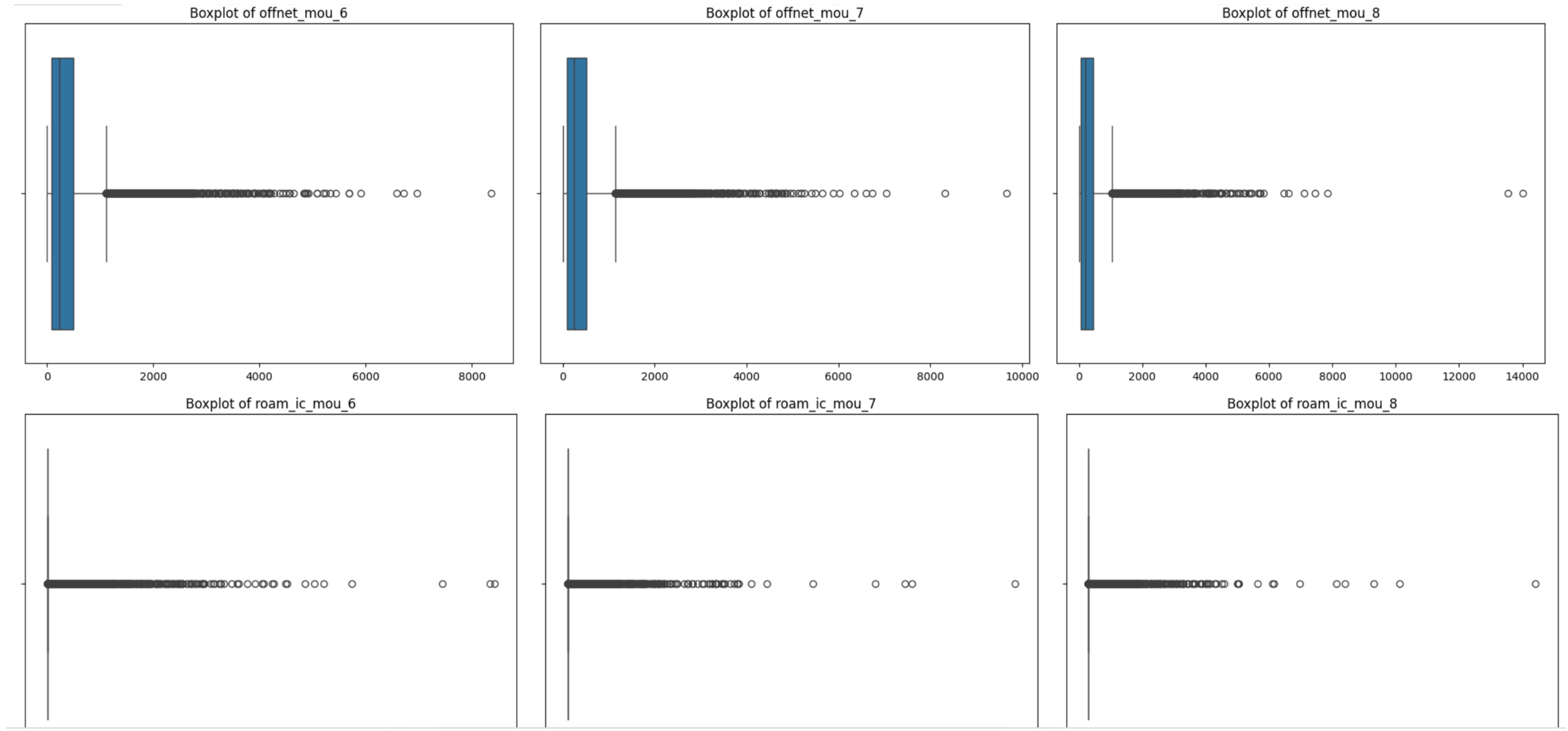


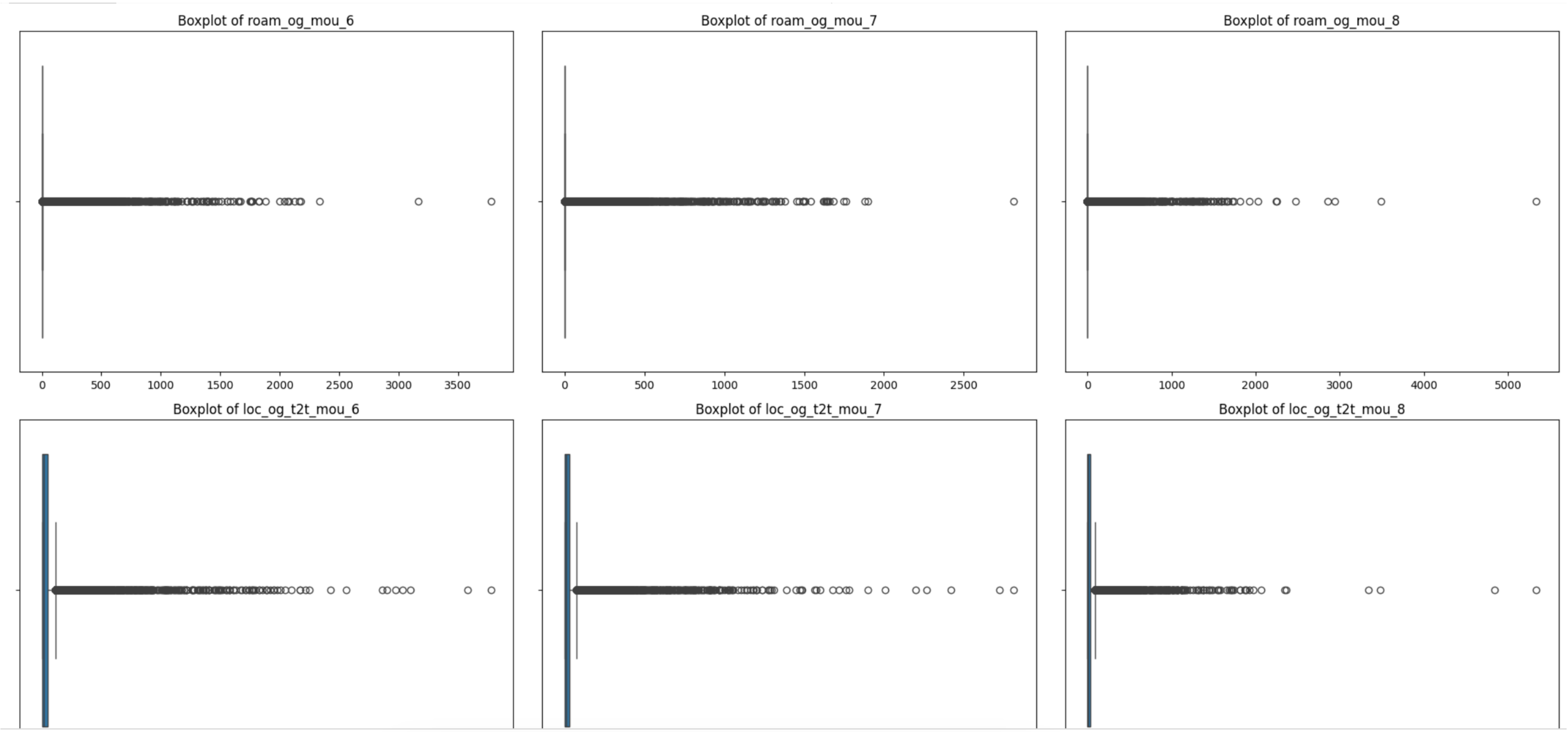
- As we can see from the graph, correlation is present between features.
- We will take care of correlated features later, or any other suitable technique for this problem.

Box plots

- We see significant outliers in all the variables/columns which are mostly coming from 8th month data.




Boxplot of offnet_mou_6
0
2000
4000
6000
8000
Boxplot of offnet_mou_7
0
2000
4000
6000
8000
10000
Boxplot of offnet_mou_8
0
2000
4000
6000
8000
10000
12000
14000
Boxplot of roam_ic_mou_6
Boxplot of roam_ic_mou_7
Boxplot of roam_ic_mou_8



Boxplot of roam_og_mou_6

Boxplot of roam_og_mou_7

Boxplot of roam_og_mou_8

Boxplot of loc_og_t2t_mou_6

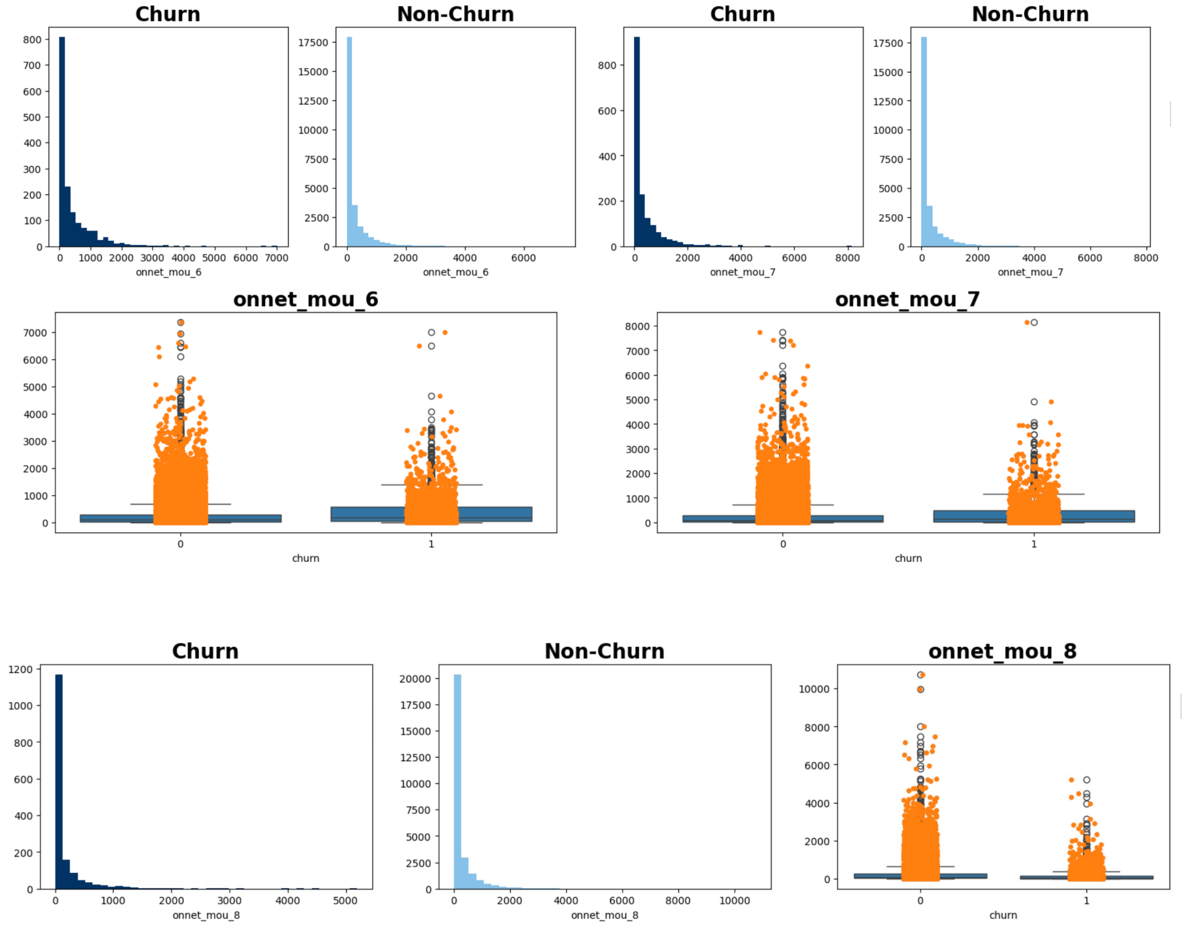
Boxplot of loc_og_t2t_mou_7

Boxplot of loc_og_t2t_mou_8

Distribution plots :

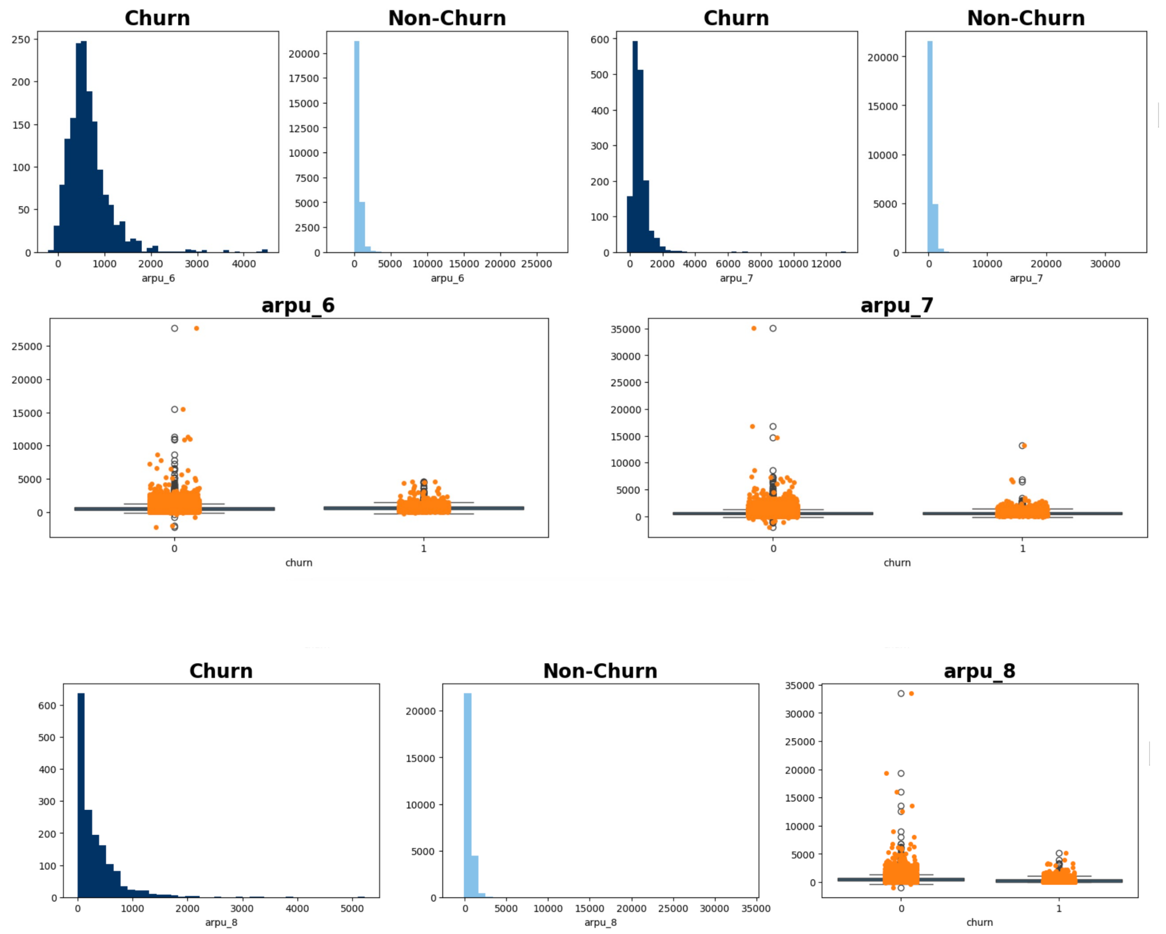
Var_Name: Minutes of usage for all kind of calls within the same operator network:

We can clearly see that Minutes of usage for all kind of calls within the same operator network is decreasing for churn customers. Also it looks like some of the customers are having high minutes of usage(outlier present)



Var_Name: Average Revenue per user

As we can see average revenue per user is decreasing for churn customers in 8th month. Also there are lots of outlier exists in revenue as some customers might using higher data and recharging frequently.

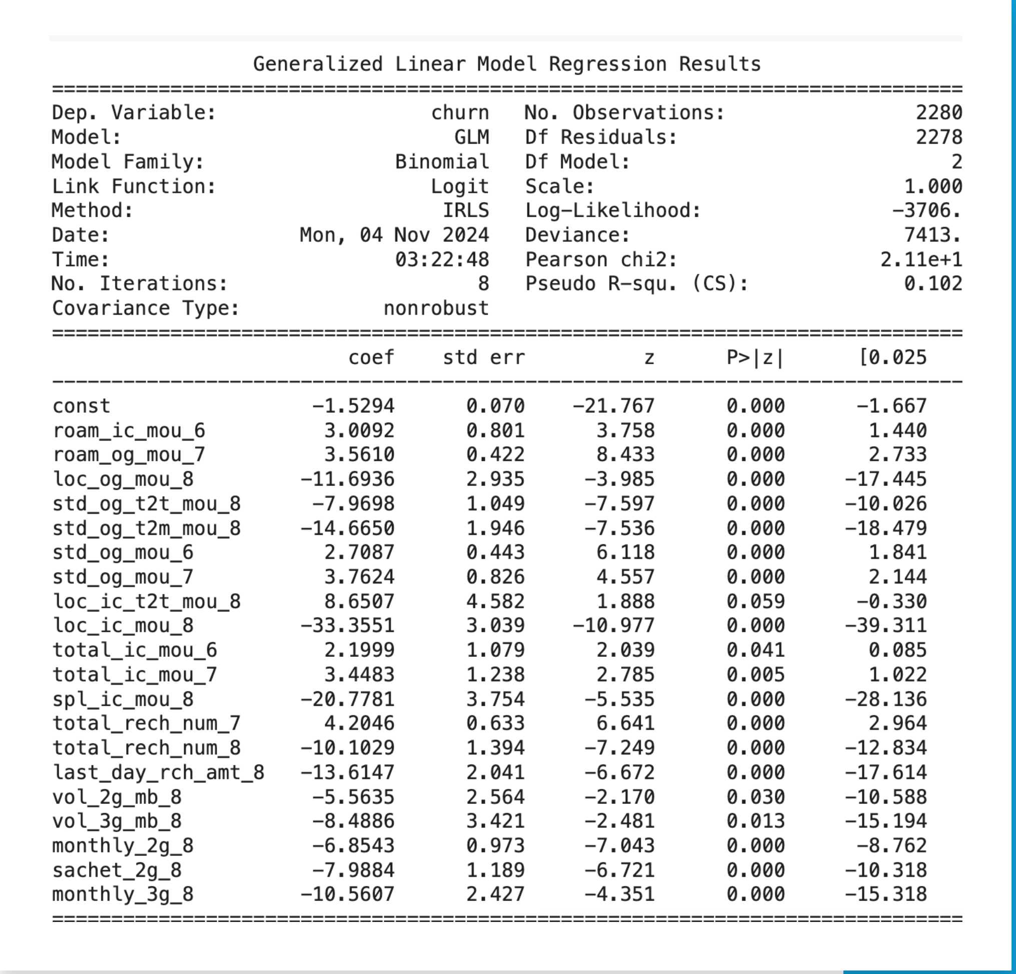


Inferences from the visualizations.

- Total recharge amount distribution is getting increased from 6 to 7th month and then getting decrease in 8th month for Churn customers. Also Max Recharge Amount is decreased in 8th month for Churn customers. Last day Recharge decreased in in 8th month for churn customers.
- We see for most of the variables the customer is churning in the 8th month.
- Offnet minutes of usage is also decreasing for churn customers in 8th month. As compared to 6th and 7th month , in 8th month there is no high minutes of usage.
- As compared to other parameters it looks like customers use less services during roaming, during talking to their operator, STD, ISD and call facilities.
- Minutes of usage for local out going calls is also decreasing for churn customers in 8th month. As compared to 6th and 7th month , in 8th month there is no high minutes of usage.
- Monthly recharge of 2g and 3g data volume in Mb decreased in 8th month for churn customers.
- For most of the customers Age on Network is around 800-900 days. There are outliers too as the maximum value is 4321.
- Significant drop in total incoming calls and total outgoing calls for churn customers , however for non churn customer it still increases.

Logistic Regression Model

- Logistic Regression model was built using RFE for feature selection.
- After dropping high VIF and p-value > 0.05 , final model is show on the right hand side.



```
                 Generalized Linear Model Regression Results
==============================================================================
Dep. Variable:                  churn   No. Observations:                 2280
Model:                            GLM   Df Residuals:                     2278
Model Family:                Binomial   Df Model:                            2
Link Function:                  Logit   Scale:                           1.000
Method:                          IRLS   Log-Likelihood:                 -3706.
Date:                Mon, 04 Nov 2024   Deviance:                        7413.
Time:                        03:22:48   Pearson chi2:                  2.11e+1
No. Iterations:                     8   Pseudo R-squ. (CS):              0.102
Covariance Type:            nonrobust
==============================================================================
                      coef    std err          z      P>|z|     [0.025
------------------------------------------------------------------------------
const              -1.5294      0.070    -21.767      0.000     -1.667
roam_ic_mou_6       3.0092      0.801      3.758      0.000      1.440
roam_og_mou_7       3.5610      0.422      8.433      0.000      2.733
loc_og_mou_8      -11.6936      2.935     -3.985      0.000    -17.445
std_og_t2t_mou_8   -7.9698      1.049     -7.597      0.000    -10.026
std_og_t2m_mou_8  -14.6650      1.946     -7.536      0.000    -18.479
std_og_mou_6        2.7087      0.443      6.118      0.000      1.841
std_og_mou_7        3.7624      0.826      4.557      0.000      2.144
loc_ic_t2t_mou_8    8.6507      4.582      1.888      0.059     -0.330
loc_ic_mou_8      -33.3551      3.039    -10.977      0.000    -39.311
total_ic_mou_6      2.1999      1.079      2.039      0.041      0.085
total_ic_mou_7      3.4483      1.238      2.785      0.005      1.022
spl_ic_mou_8      -20.7781      3.754     -5.535      0.000    -28.136
total_rech_num_7    4.2046      0.633      6.641      0.000      2.964
total_rech_num_8  -10.1029      1.394     -7.249      0.000    -12.834
last_day_rch_amt_8 -13.6147     2.041     -6.672      0.000    -17.614
vol_2g_mb_8        -5.5635      2.564     -2.170      0.030    -10.588
vol_3g_mb_8        -8.4886      3.421     -2.481      0.013    -15.194
monthly_2g_8       -6.8543      0.973     -7.043      0.000     -8.762
sachet_2g_8        -7.9884      1.189     -6.721      0.000    -10.318
monthly_3g_8      -10.5607      2.427     -4.351      0.000    -15.318
==============================================================================
```

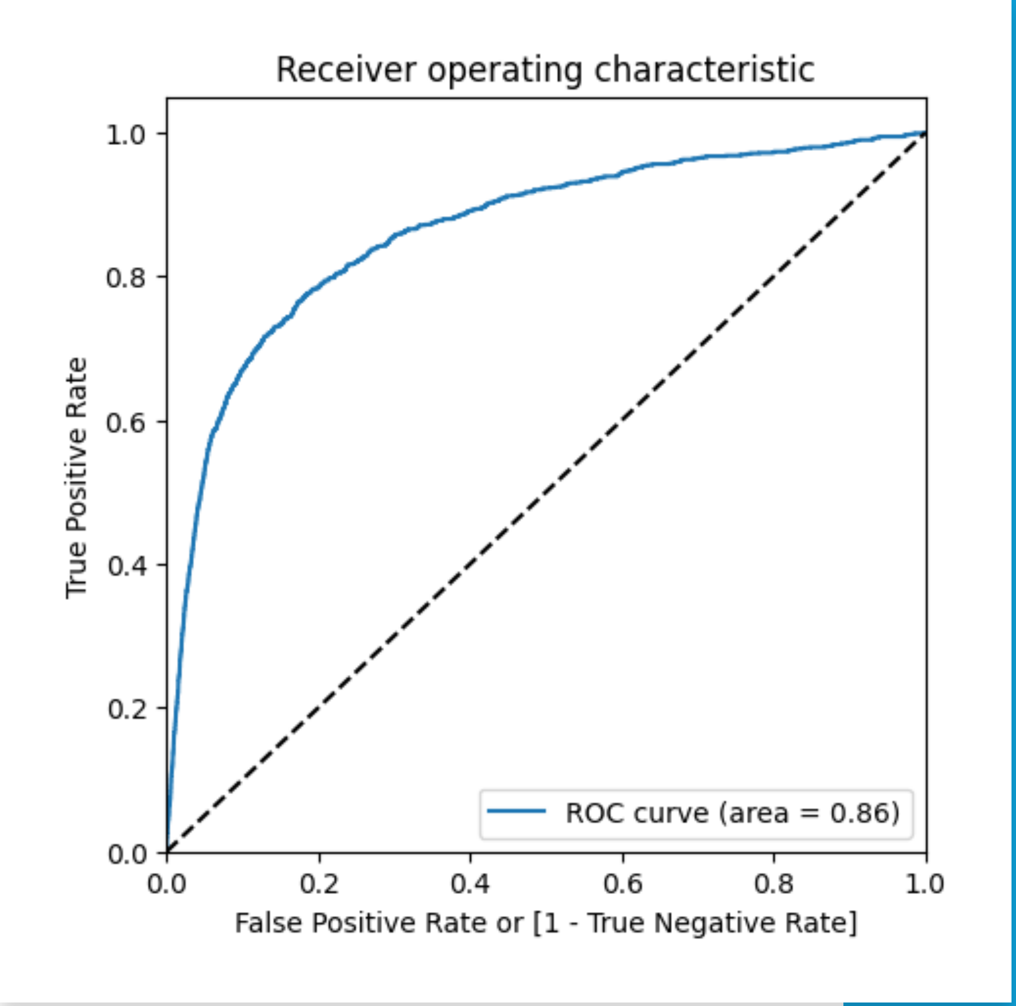
Confusion Matrix

We see a clearly it's a imbalance dataset

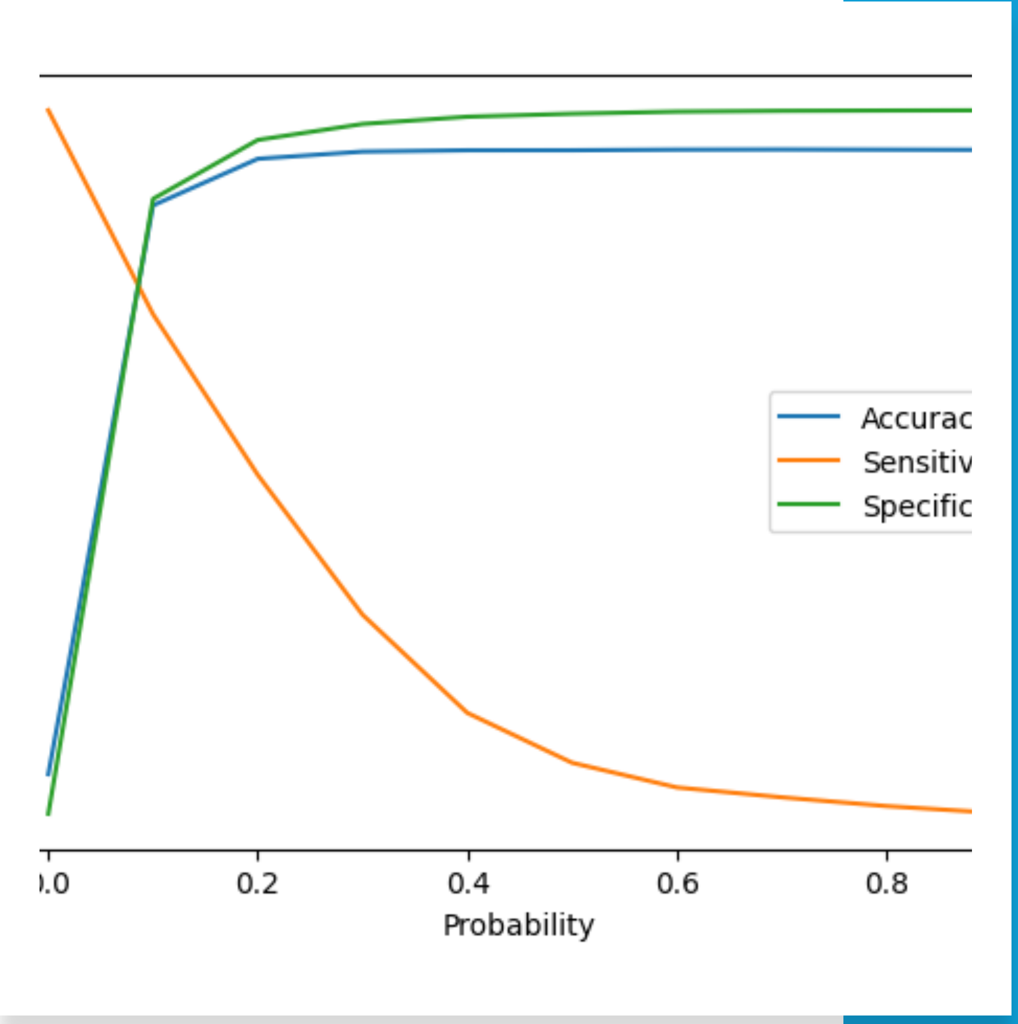
- Logistic Regression assumes a linear relationship between the features and the log-odds of the target variable.
- SMOTE creates synthetic samples by interpolating between existing minority class samples, which might not respect this linearity. The new samples could lie in regions of the feature space that do not accurately reflect the underlying data distribution.
- As our dataset has a high number of features relative to the number of samples, SMOTE can create synthetic samples in sparse regions of the feature space. This can lead to overfitting, especially for Logistic Regression, which might struggle to generalize well on unseen data.
- Logistic Regression can be sensitive to multicollinearity among features. If SMOTE introduces new synthetic samples that exacerbate this issue, it may affect the model's interpretability and performance.
- Hence will do GridSearchCV and SMOTE in the Decision Tree Classifier Model and get the best estimators.

Confusion	Predicted	
Actual	Not Churn	Churn
Not Churn	21415	105
Churn	1190	93

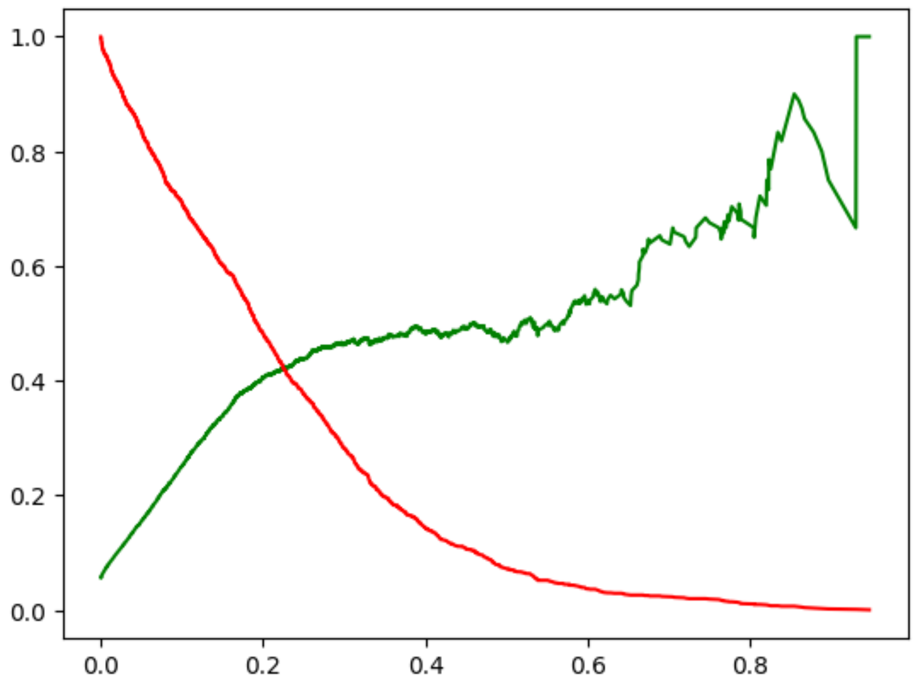
ROC Curve


Receiver operating characteristic
True Positive Rate
False Positive Rate or [1 - True Negative Rate]
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (area = 0.86)

Optimal Cut off

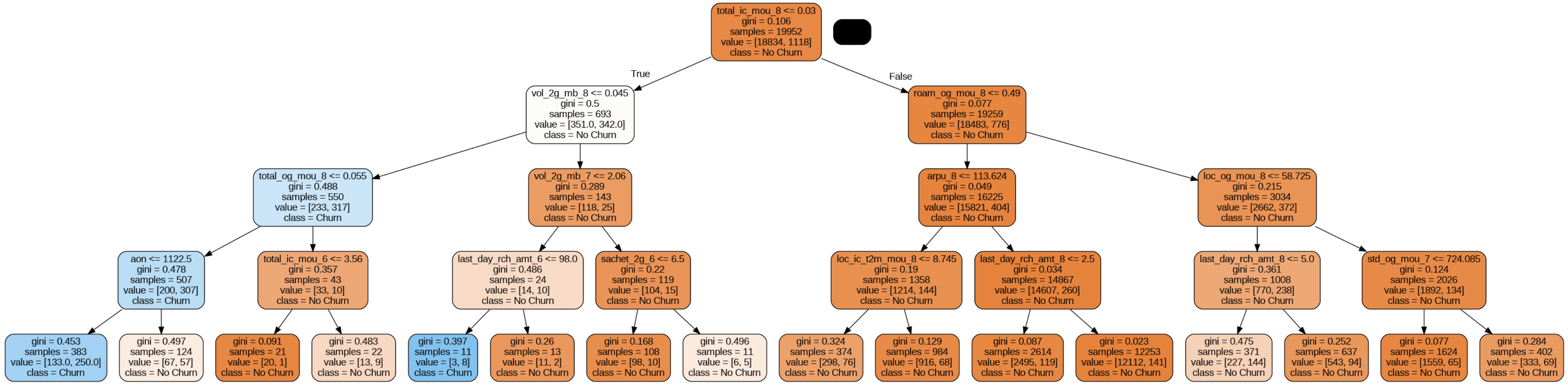


Precision and Recall Tradeoff



Decision Tree Model

With default parameters



[redacted]

total_ic_mou_8 <= 0.03
gini = 0.106
samples = 19952
value = [18834, 1118]
class = No Churn

True

False

vol_2g_mb_8 <= 0.045
gini = 0.5
samples = 693
value = [351.0, 342.0]
class = No Churn

roam_og_mou_8 <= 0.49
gini = 0.077
samples = 19259
value = [18483, 776]
class = No Churn

total_og_mou_8 <= 0.055
gini = 0.488
samples = 550
value = [233, 317]
class = Churn

vol_2g_mb_7 <= 2.06
gini = 0.289
samples = 143
value = [118, 25]
class = No Churn

arpu_8 <= 113.624
gini = 0.049
samples = 16225
value = [15821, 404]
class = No Churn

loc_og_mou_8 <= 58.725
gini = 0.215
samples = 3034
value = [2662, 372]
class = No Churn

aon <= 1122.5
gini = 0.478
samples = 507
value = [200, 307]
class = Churn

total_ic_mou_6 <= 3.56
gini = 0.357
samples = 43
value = [33, 10]
class = No Churn

last_day_rch_amt_6 <= 98.0
gini = 0.486
samples = 24
value = [14, 10]
class = No Churn

sachet_2g_6 <= 6.5
gini = 0.22
samples = 119
value = [104, 15]
class = No Churn

loc_ic_t2m_mou_8 <= 8.745
gini = 0.19
samples = 1358
value = [1214, 144]
class = No Churn

last_day_rch_amt_8 <= 2.5
gini = 0.034
samples = 14867
value = [14607, 260]
class = No Churn

last_day_rch_amt_8 <= 5.0
gini = 0.361
samples = 1008
value = [770, 238]
class = No Churn

std_og_mou_7 <= 724.085
gini = 0.124
samples = 2026
value = [1892, 134]
class = No Churn

gini = 0.453
samples = 383
value = [133.0, 250.0]
class = Churn

gini = 0.497
samples = 124
value = [67, 57]
class = No Churn

gini = 0.091
samples = 21
value = [20, 1]
class = No Churn

gini = 0.483
samples = 22
value = [13, 9]
class = No Churn

gini = 0.397
samples = 11
value = [3, 8]
class = Churn

gini = 0.26
samples = 13
value = [11, 2]
class = No Churn

gini = 0.168
samples = 108
value = [98, 10]
class = No Churn

gini = 0.496
samples = 11
value = [6, 5]
class = No Churn

gini = 0.324
samples = 374
value = [298, 76]
class = No Churn

gini = 0.129
samples = 984
value = [916, 68]
class = No Churn

gini = 0.087
samples = 2614
value = [2495, 119]
class = No Churn

gini = 0.023
samples = 12253
value = [12112, 141]
class = No Churn

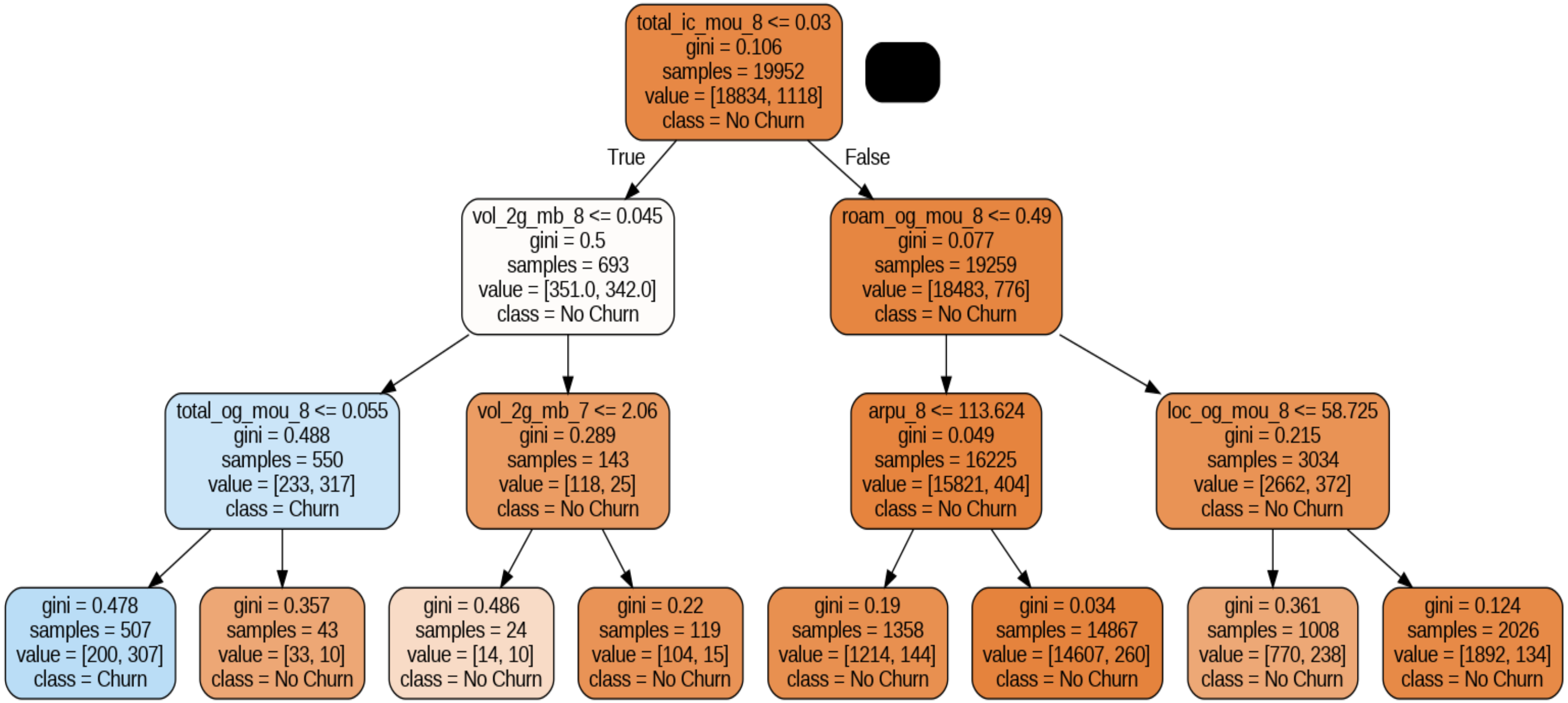
gini = 0.475
samples = 371
value = [227, 144]
class = No Churn

gini = 0.252
samples = 637
value = [543, 94]
class = No Churn

gini = 0.077
samples = 1624
value = [1559, 65]
class = No Churn

gini = 0.284
samples = 402
value = [333, 69]
class = No Churn

Hyper Parameter Tunning using GridSearchCV



total_ic_mou_8 <= 0.03
gini = 0.106
samples = 19952
value = [18834, 1118]
class = No Churn

[redacted]

True

False

vol_2g_mb_8 <= 0.045
gini = 0.5
samples = 693
value = [351.0, 342.0]
class = No Churn

roam_og_mou_8 <= 0.49
gini = 0.077
samples = 19259
value = [18483, 776]
class = No Churn

total_og_mou_8 <= 0.055
gini = 0.488
samples = 550
value = [233, 317]
class = Churn

vol_2g_mb_7 <= 2.06
gini = 0.289
samples = 143
value = [118, 25]
class = No Churn

arpu_8 <= 113.624
gini = 0.049
samples = 16225
value = [15821, 404]
class = No Churn

loc_og_mou_8 <= 58.725
gini = 0.215
samples = 3034
value = [2662, 372]
class = No Churn

gini = 0.478
samples = 507
value = [200, 307]
class = Churn

gini = 0.357
samples = 43
value = [33, 10]
class = No Churn

gini = 0.486
samples = 24
value = [14, 10]
class = No Churn

gini = 0.22
samples = 119
value = [104, 15]
class = No Churn

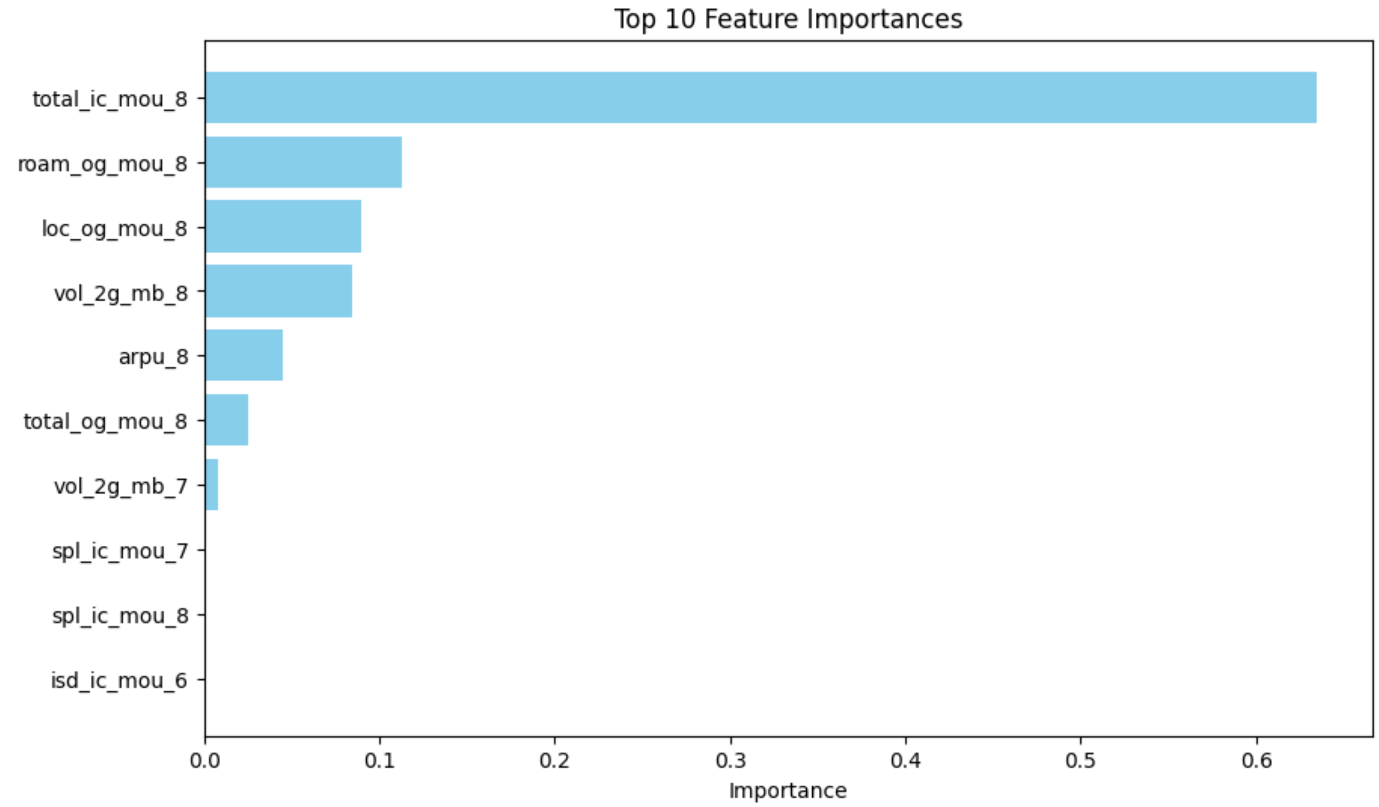
gini = 0.19
samples = 1358
value = [1214, 144]
class = No Churn

gini = 0.034
samples = 14867
value = [14607, 260]
class = No Churn

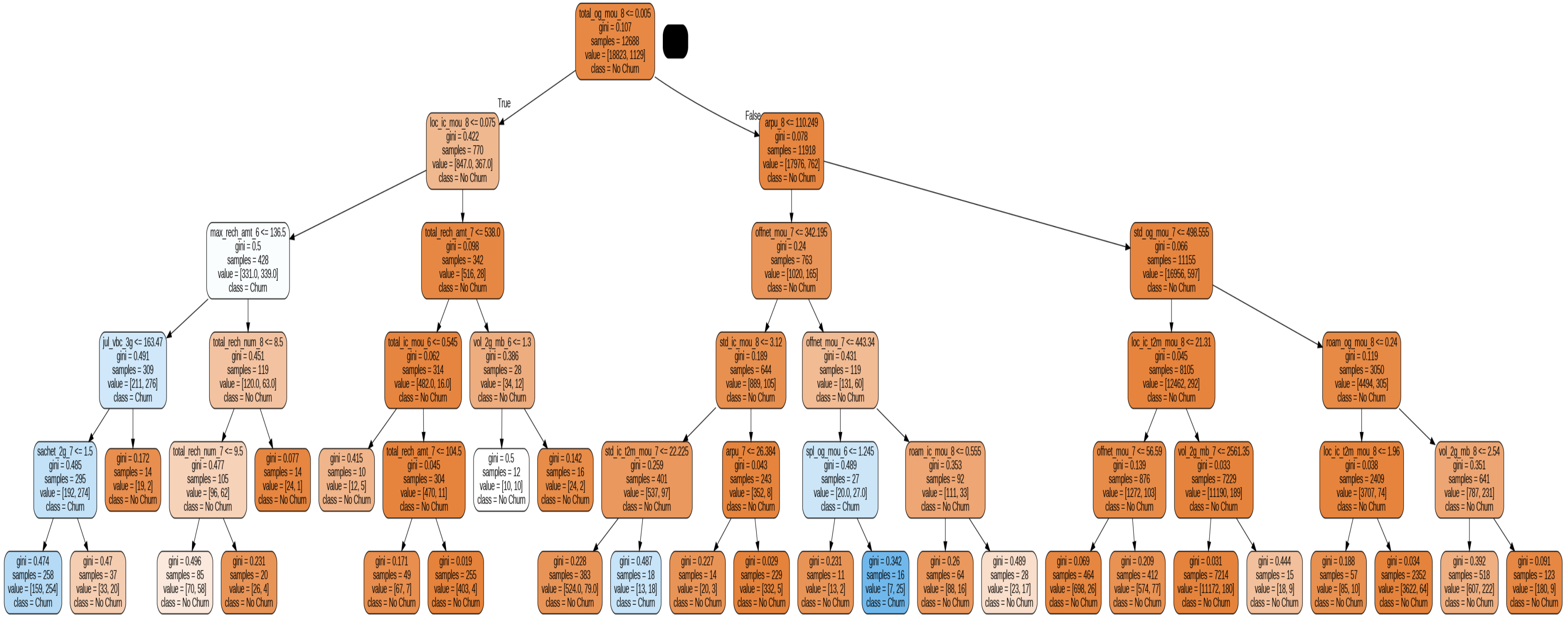
gini = 0.361
samples = 1008
value = [770, 238]
class = No Churn

gini = 0.124
samples = 2026
value = [1892, 134]
class = No Churn

Top 10 best features for predicting churn or not churn

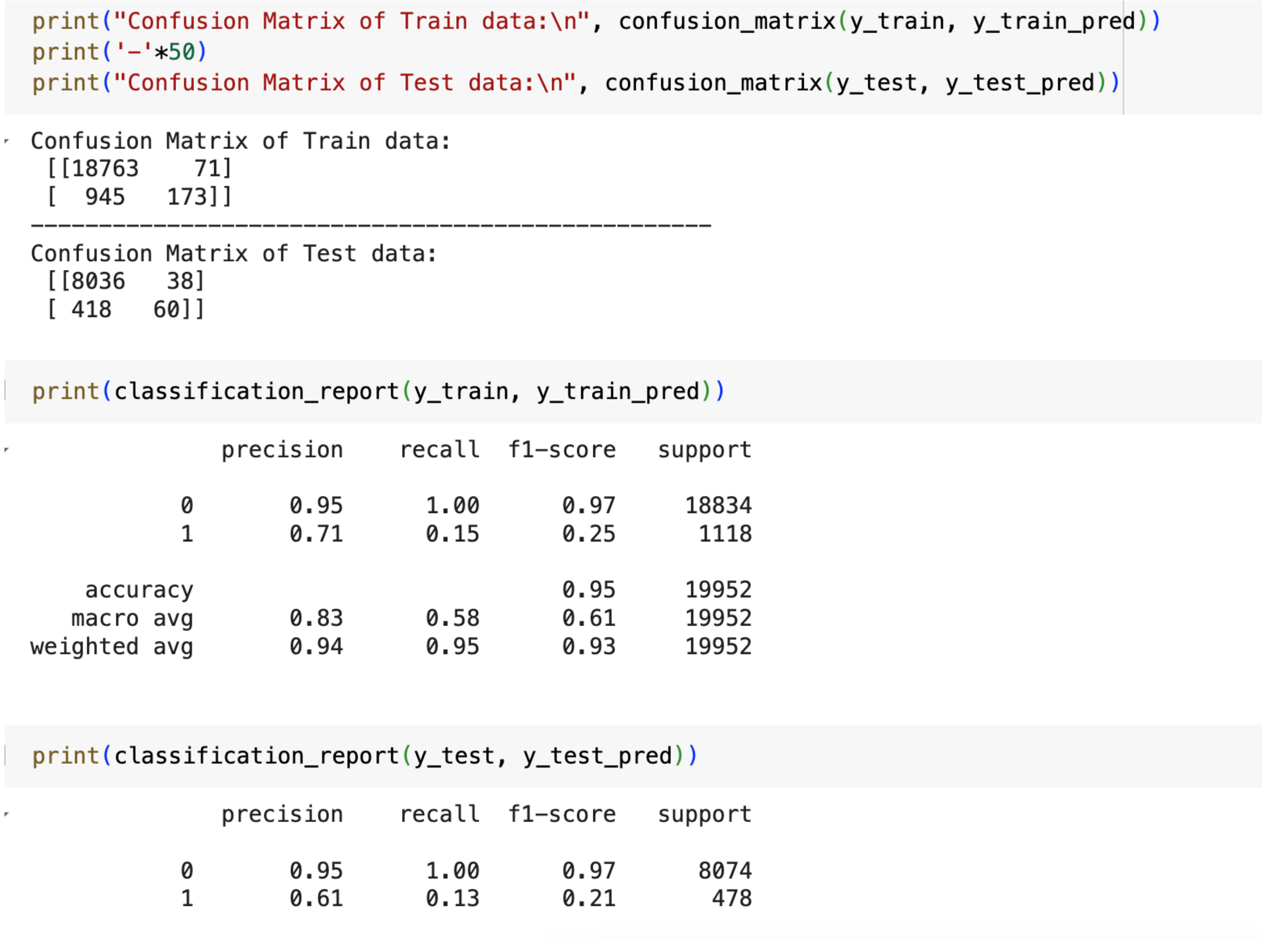


Random Forest Classifiers



Ensemble Model

- m1 = LogisticRegression()
- m2 = KNeighborsClassifier(5)
- m3 = DecisionTreeClassifier(random_state=42, max_depth=4)



```
print("Confusion Matrix of Train data:\n", confusion_matrix(y_train, y_train_pred))
print('-'*50)
print("Confusion Matrix of Test data:\n", confusion_matrix(y_test, y_test_pred))
```

```
Confusion Matrix of Train data:
 [[18763    71]
 [  945   173]]
--------------------------------------------------
Confusion Matrix of Test data:
 [[8036   38]
 [ 418   60]]
```

```
print(classification_report(y_train, y_train_pred))
```

```
              precision    recall  f1-score   support

           0       0.95      1.00      0.97     18834
           1       0.71      0.15      0.25      1118

    accuracy                           0.95     19952
   macro avg       0.83      0.58      0.61     19952
weighted avg       0.94      0.95      0.93     19952
```

```
print(classification_report(y_test, y_test_pred))
```

```
              precision    recall  f1-score   support

           0       0.95      1.00      0.97      8074
           1       0.61      0.13      0.21       478
```

Conclusion

Top 10 feature variables for business which will be very helpful in business making decision and give additional benefits/discount to customers who are likely to churn.

-
  - total_ic_mou_8 (Total incoming minutes of usage for 8th month)
- - roam_og_mou_8 (Roaming outgoing minutes of usage for 8th month)
- - loc_og_mou_8 (local outgoing minutes of usage for 8th month)
- - vol_2g_mb_8 (Mobile 2g internet usage volume (in MB) for 8th month)
- - arpu_8 (average revenue per user for 8th month)
- - vol_2g_mb_7 (Mobile 2g internet usage volume (in MB) for 7th month)
- - spl_ic_mou_7 (special incoming minutes of usage for 7th month)
- - spl_ic_mou_8 (special incoming minutes of usage for 8th month)
- - isd_ic_mou_6 (ISD incoming minutes of usage for 6th month)

From the final logistic regression using 20 feature variables we got:`

- - Recall Score= 71%
- - Accuracy = 86%
- - ROC AUC= 86%
- - Specificity= 87%

- Both logistic regression and the random forest classifier yield almost similar predictor variables for identifying customers at risk of churning.
- The telecom company should utilize the Top 10 parameters got through GridSearchCV to implement promotional offers targeted at these at-risk customers.
- We have applied all our models specifically to high-value customers, enabling business stakeholders to focus on the most significant predictor variables.
- Although we used various models and feature selection techniques, the resulting predictor variables are similar, with minor differences in their ranking.
- Therefore, we recommend that the telecom company prioritize these key feature variables, as they are strong indicators for effectively managing customer churn.

Thank you