

FINAL REPORT

Batch: DSBA Sep C 20



October 3, 2021

NAME: - Atul Kumar Singh

Table of Contents

[**1.** **Introduction** 2](#_Toc84703463)

[a. Brief introduction about the problem statement and the need of solving it. 2](#_Toc84703464)

[**2.** **EDA and Business Implication** 2](#_Toc84703465)

[a. Uni-variate / Bi-variate / Multi-variate analysis to understand relationship b/w variables. How your analysis is impacting the business? 2](#_Toc84703466)

[b. Both visual and non-visual understanding of the data. 6](#_Toc84703467)

[**3.** **Data Cleaning and Pre-processing** 6](#_Toc84703468)

[a. Approach used for identifying and treating missing values and outlier treatment (and why) 6](#_Toc84703469)

[b. Need for variable transformation (if any) 9](#_Toc84703470)

[c. Variables removed or added and why (if any) 9](#_Toc84703471)

[**4.** **Model building** 9](#_Toc84703472)

[a. Clear on why was a particular model(s) chosen. 9](#_Toc84703473)

[b. Effort to improve model performance. 13](#_Toc84703474)

[**5.** **Model validation** 14](#_Toc84703475)

[a. How was the model validated? Just accuracy, or anything else too? 14](#_Toc84703476)

[**6.** **Final interpretation / recommendation** 17](#_Toc84703477)

[a. Detailed recommendations for the management/client based on the analysis done. 17](#_Toc84703478)

[**7.** **Appendix** 18](#_Toc84703479)

[a. Model Hyper-Parameter Tuning values 18](#_Toc84703480)

[b. Performance Metrics of all models 19](#_Toc84703481)

[c. Regression coefficient Analysis 24](#_Toc84703482)

# **Introduction**

## Brief introduction about the problem statement and the need of solving it.

* Business is looking for a churn prediction model. Since in current scenario it is becoming difficult for them to retain customer. Here churning is a major problem for business because a single account is having multiple users, and if they lose an account they will be losing bunch of user at a time.
* So here we will be developing a churn model and will analyse the factors affecting churn. And will be recommending solution based on it.
* Also there is a constraint like we don’t have to give a lot of free stuff as it will be making a loss to business in revenue term.

# **EDA and Business Implication**

## Uni-variate / Bi-variate / Multi-variate analysis to understand relationship b/w variables. How your analysis is impacting the business?

* There are 11260 rows and 19 columns in data.
* There are some numerical columns which are coming as categorical, due to presence of special character.

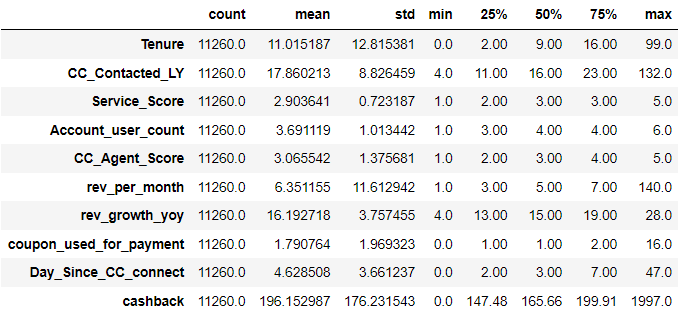


Figure 1: Summary of numerical variable

* From 5 point summary we can see that most of the data are right skewed.
* Tenure Analysis:-

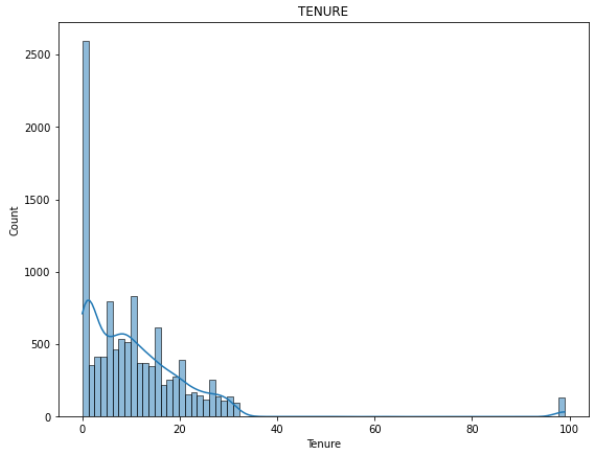
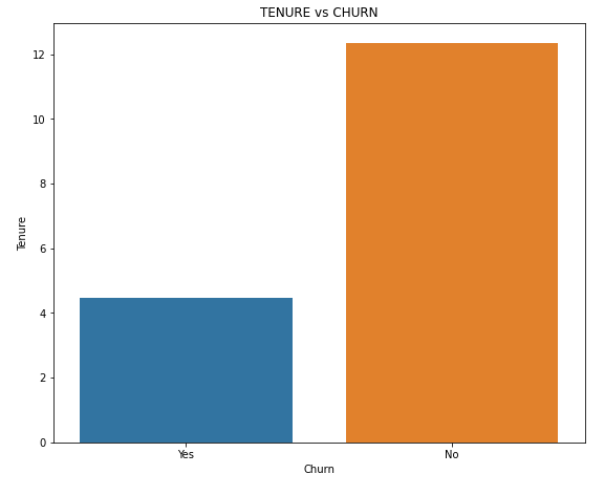
 

Figure 2: Tenure Figure 3: Tenure vs Churn

* From figure 2 we can see that most of the customers are for less period of time. And from figure 1 of summary we can also see that 75% of people are for less tenure.
* People who are churning are people who have less tenure same can be seen from figure 3.

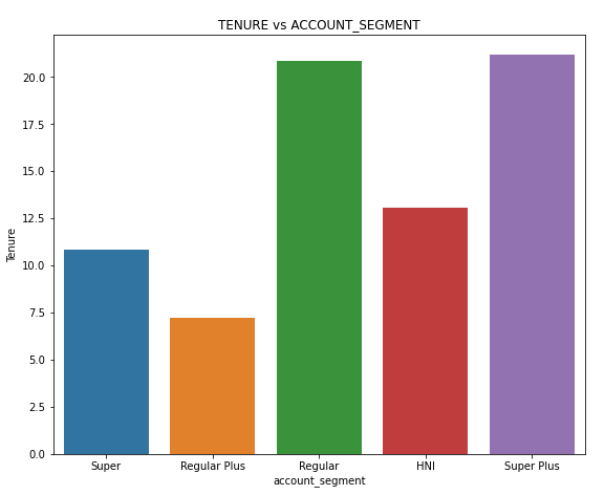
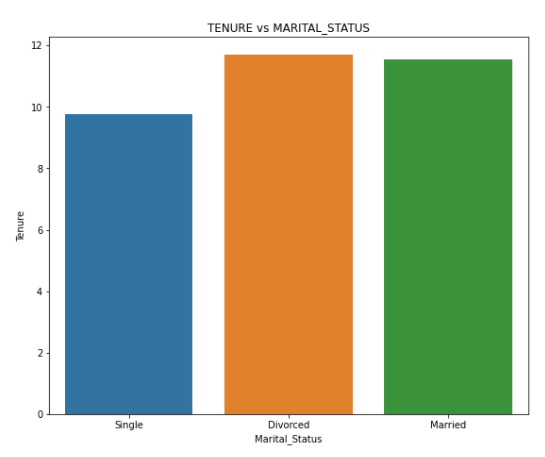
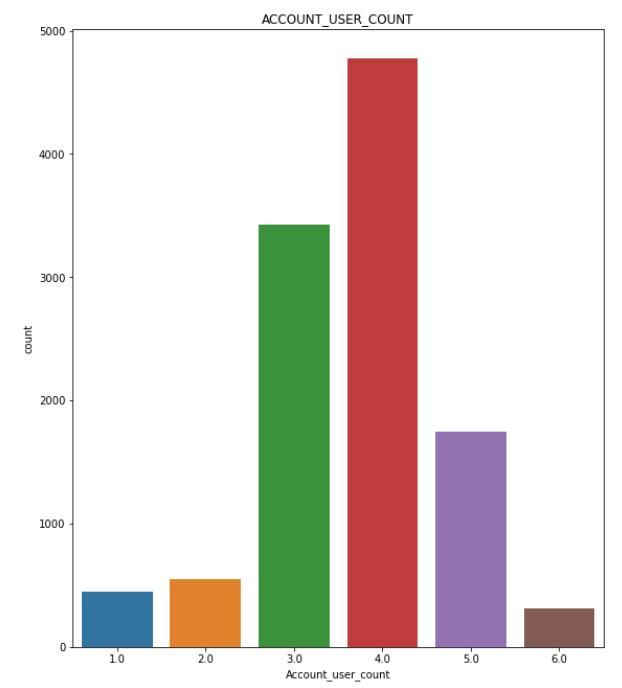
 

Figure 4: Tenure vs Account Segment Figure 5: Tenure vs Marital Status

* Regular Plus and Super account segment people have less tenure than other account segments same can be seen in figure 4.
* From figure 5 we can see that Single people have less tenure than divorced and married.
* Account Analysis:

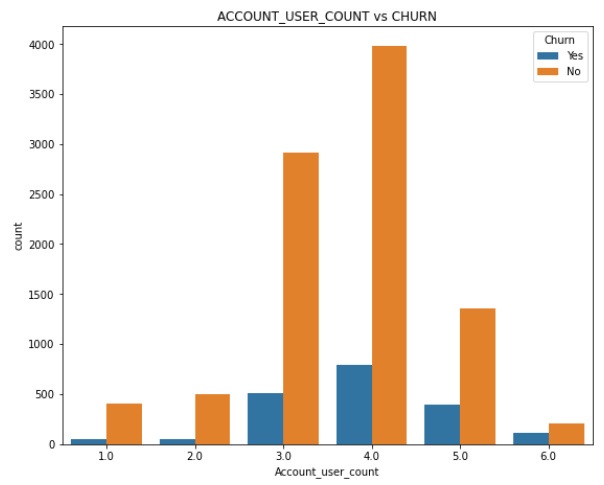


Figure 6: Account user count Figure 7: Account user count vs churn

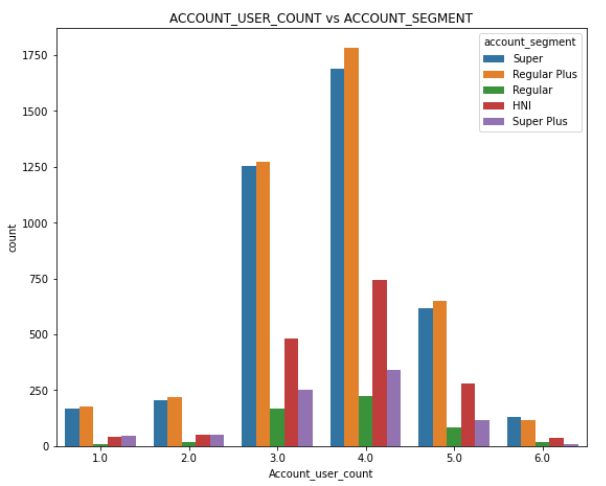
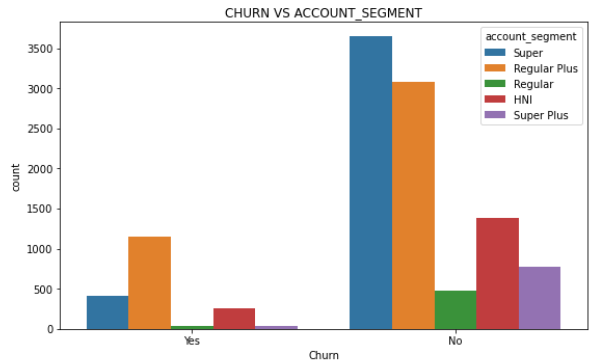
* From fig 6 we can conclude that most of the account have 4 users whereas there are few account who have 6 users.
* And if we check for churning rate in figure 7 most of the churners are from account having 4 users.

Figure 8: Account user count vs Account Segment Figure 9: Churn vs Account Segment

* Regular Plus and Super seem to be most popular among all accounts followed by HNI same can be seen in figure 8.
* And churn for these accounts are also highest among all same can be depicted in fig 9.
* Service and customer care connect analysis:

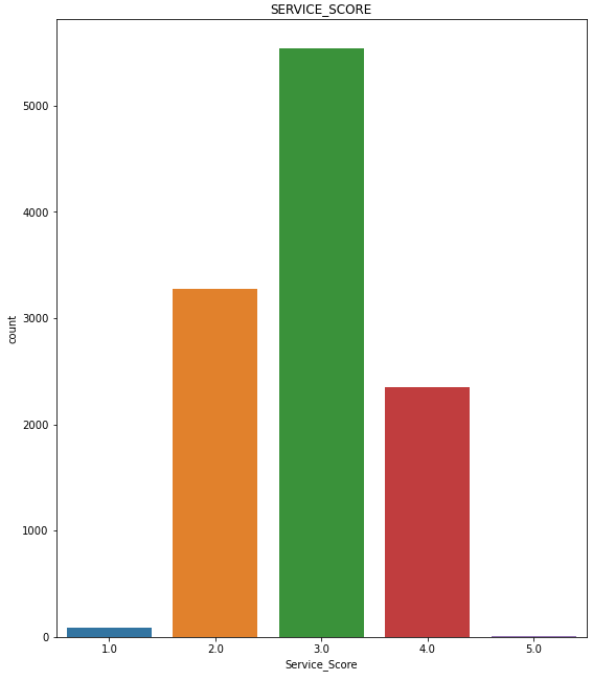
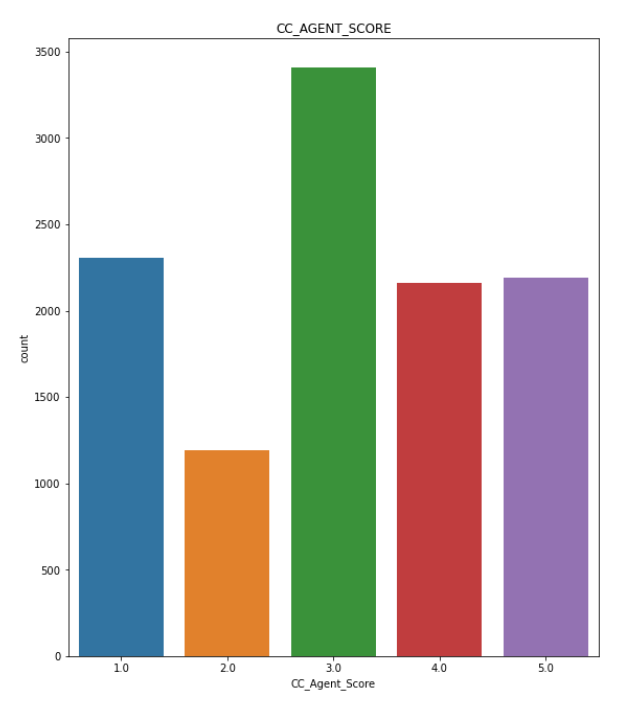


Figure 10: Service Score Figure 11: Agent Score

* From figure 10 and 11 we can see that both Service and Agent Score are either average or less than average.
* From data summary in fig 1 we can see that 50% of data is below average for agent score.

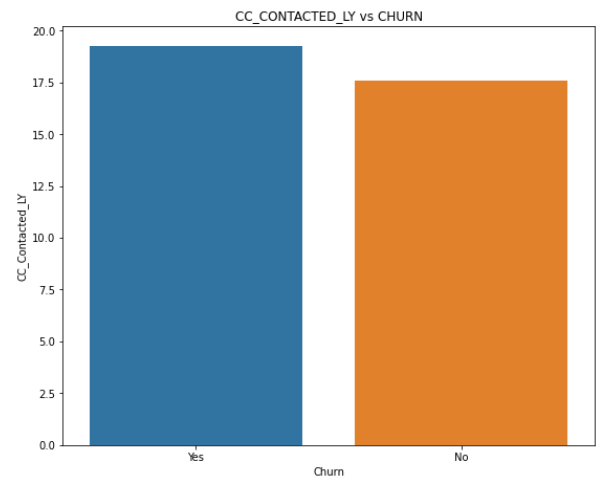
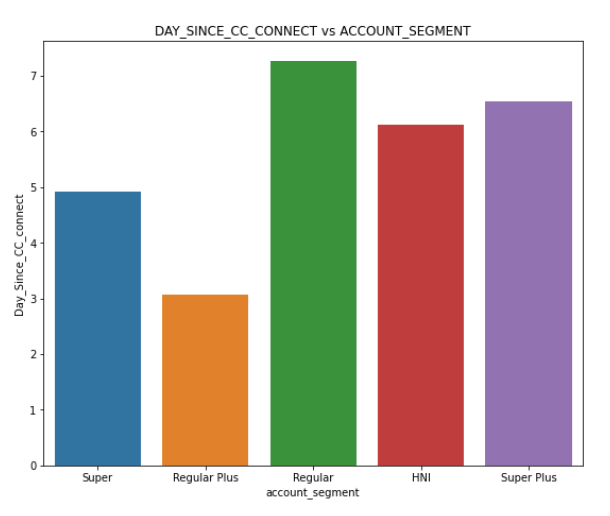


Figure 12:CC\_Contacted\_LY vs churn Figure 13: Day\_Since\_CC\_Connect vs Account Segment

* From figure 12 we can see that customer who contacted more times last year have churned more.
* Also our most subscribed plan i.e., Regular Plus and super both customers are contacting agent more frequently.
* Revenue Analysis:

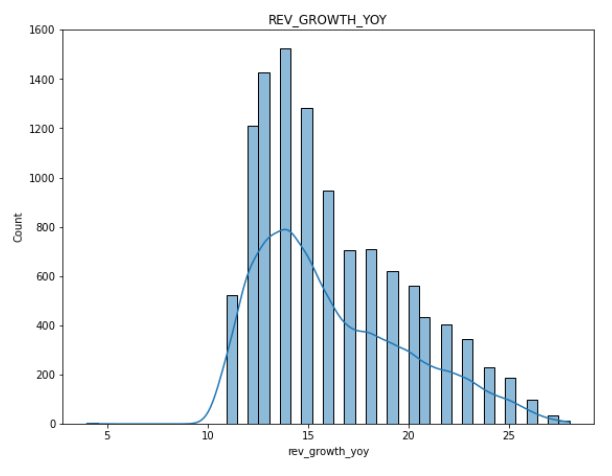
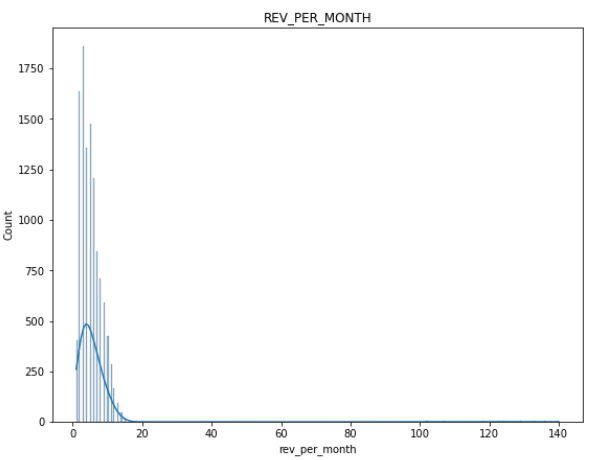


Figure 14: Revenue per month Figure 15: Rev Growth YOY

* From fig 14 and 15 we can see that most of the customers are generating lower revenue.

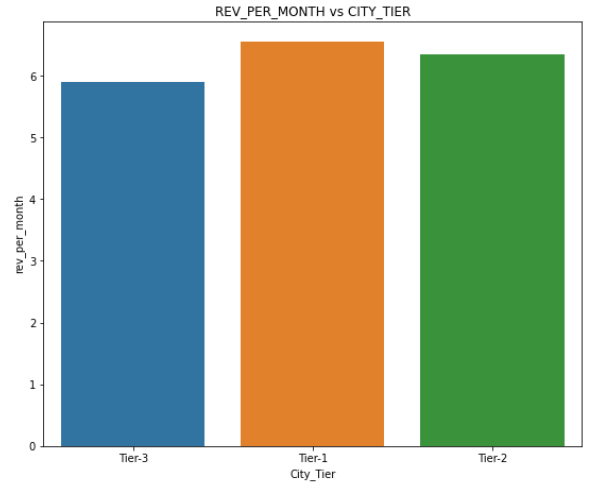
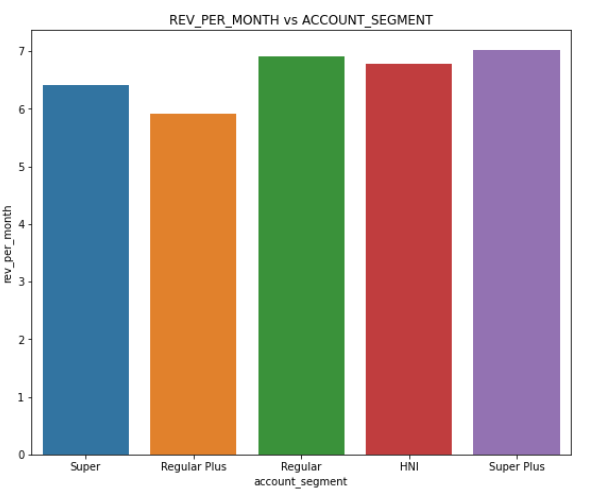
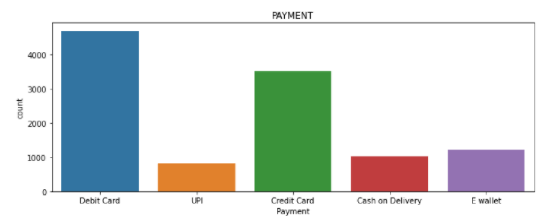


Figure 16: Rev\_per\_month vs Account\_Segment Figure 17: Rev\_per\_month vs City\_Tier

* From figure 16 we can say that our most subscribed plan i.e., Regular Plus and Super are generating lower revenue.
* Tier-1 cities are generating highest revenue per month followed by Tier-2 and Tier-3 cities.
* Payment Analysis:



* Payment via cards i.e., Debit and Credit cards are most preferred for payment. Whereas UPI is least.

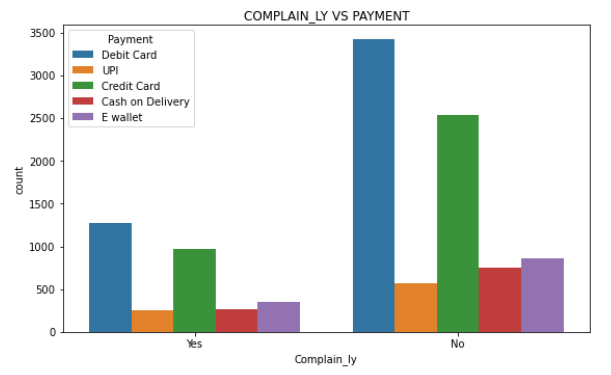
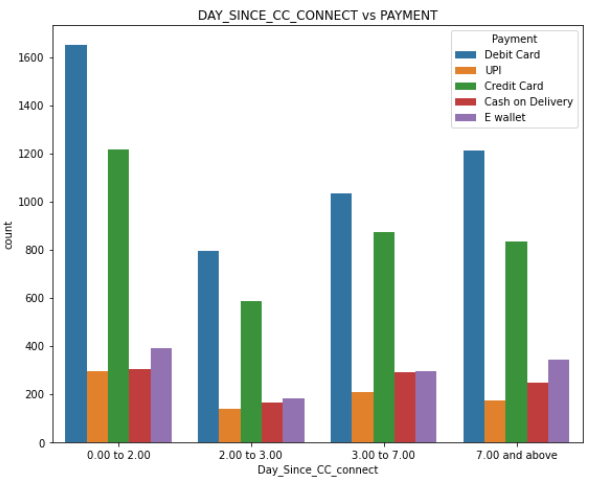


Figure 18: Complain\_LY vs Payment Figure 19: Day\_SInce\_CC\_Connect vs Payment

* From figure 18 & 19 we can see that most of the customer were contacting customer for card payment.
* And most of the customer recently also contacted for same.

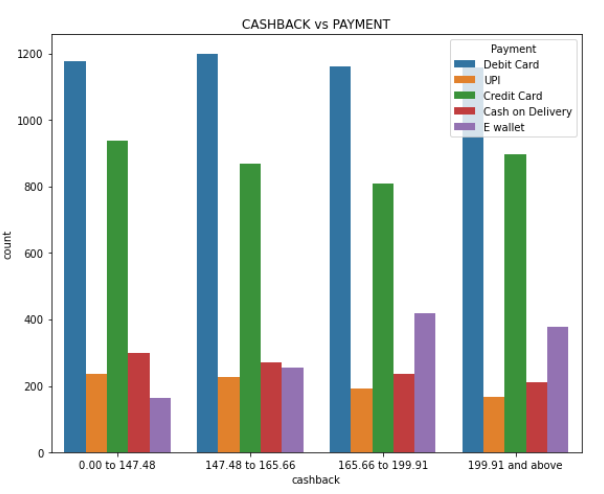
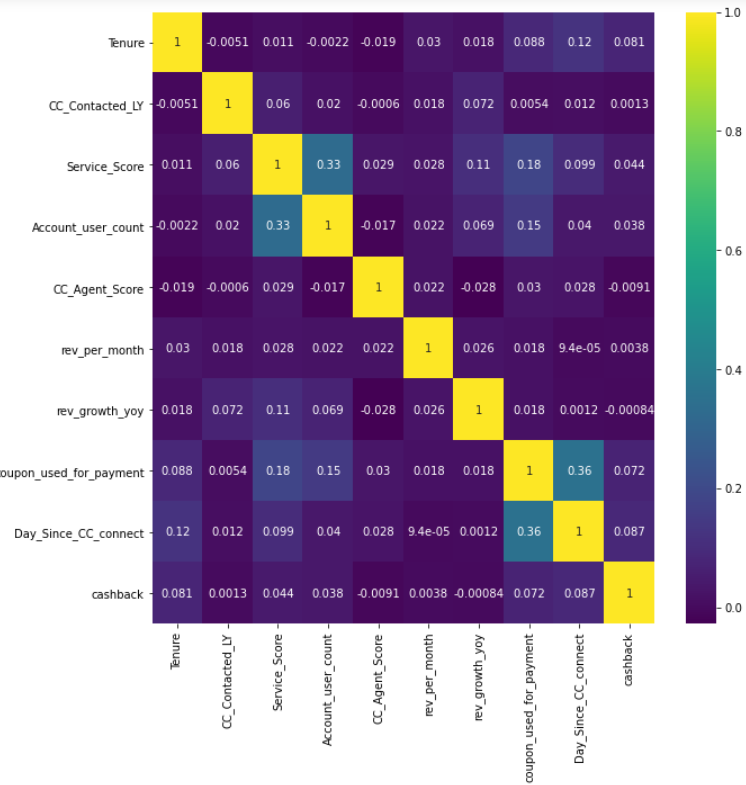


Figure 20: Cashback vs Payment

* Most of the cashback was awarded for payment done via card.
* Correlation Analysis:



* Most of the variables are weakly correlated.
* Except Service Score with user count and coupon used for payment with day since cc connect.

## Both visual and non-visual understanding of the data.

* There seem to various problem associated within the service and its resolution.
* We can see that for the problem customer contacted last year are still present in the system.
* From above correlation plot we can see that there is weak but positive correlation between day since cc connect and coupon used for payment and we can assume that people who are contacting for payment related issue are given some coupon to process their payment.
* Also we saw earlier there is problem with payment service.
* Our most subscribed plans are generating lower revenues.

# **Data Cleaning and Pre-processing**

## Approach used for identifying and treating missing values and outlier treatment (and why)

* As discussed in section 2.a there are certain variables which are of numerical type but they are coming as object.
* So on checking for special character below is count of unwanted variable for categorical columns in Fig 21.

|  |
| --- |
| TENURE : 116  PAYMENT : 0  GENDER : 0  ACCOUNT\_USER\_COUNT : 332  ACCOUNT\_SEGMENT : 0  MARITAL\_STATUS : 0  REV\_PER\_MONTH : 689  REV\_GROWTH\_YOY : 3  COUPON\_USED\_FOR\_PAYMENT : 3  DAY\_SINCE\_CC\_CONNECT : 1  CASHBACK : 2  LOGIN\_DEVICE : 539 |

Figure 21: Count of unwanted variable

* **To treat the value a list is created by using punctuations presented in string library.**
* **Then using lambda function special values present in the data was replaced by null.**
* **Below is info of data in fig 22, after treating special values and we can see that all the variables are fine now.**

|  |
| --- |
|  |

Figure 22: Info after treating values

* After that to treat the values first values is distributed in to parts i.e., categorical and numerical.
* Below is list of the values:

List of categorical values:

['CHURN', 'CITY\_TIER', 'PAYMENT', 'GENDER', 'ACCOUNT\_SEGMENT', 'MARITAL\_STAT US', ' COMPLAIN\_LY', 'LOGIN\_DEVICE']

List of numerical values:

['ACCOUNTID', 'TENURE', 'CC\_CONTACTED\_LY', 'SERVICE\_SCORE', 'ACCOUNT\_USER\_COUNT', 'CC\_AGENT\_SCORE', 'REV\_PER\_MONTH', 'REV\_GROWTH\_YOY', 'COUPON\_USED\_FOR\_PAYMENT', 'DAY\_SINCE\_CC\_CONNECT']

* To treat categorical values simple imputer function of sklearn.impute is used having most frequent as the strategy.
* To treat numerical values knnimputer function of sklearn.impute is used having 3 neighbours.
* Below is count of null values in fig 23 after treatment.

|  |
| --- |
|  |

Figure 23: Count of null value

* After treating values with knn imputer a problem arises where some field are treated as continuous variable but really they are not. For example service score can contain values 1-5.
* So to treat this first we need to round off the values to 0th place.
* And then for variable service score values which are 0 need to change to 1.
* Above approach is used to identify and treat missing value. We have used above approach because there are two types of variable in our column i.e., numerical and categorical.
* So if we treat numerical value by most common value it will be generating biasness in the data. That’s why KNN is used here.
* And for categorical value we have used simple imputer with most frequent values because it will help to synthetically not add other data.

|  |
| --- |
|  |

Figure 24: Outlier

* From figure 24 we can see that most of the variables have outlier and if we try to treat it we may lose some important variables. Hence, we will not be treating outliers here.

## Need for variable transformation (if any)

* Few variables were mapped correctly in the data which were having two values with same meaning.
* Like in field gender we have M and Male. S here M is marked as Male.
* Along with account segment and Login Device.

## Variables removed or added and why (if any)

* No variables were added or removed as all the variables had significant meaning.

# **Model building**

## Clear on why was a particular model(s) chosen.

* Data is divided into two parts X and y. Where X is input variables with all columns except output variable churn.
* After separating values in dependent and independent values. Variables are splitted using train test split function having 70-30 split, seed with 1 and stratify as y. Figure 25 shows distribution of train test.

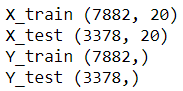


Figure 25: Train Test Split

* Below are different models build:
* Naïve Bayes :
  + Gaussian Naïve Bayes model is built and it is giving both train and test accuracy as 85 %.
* Logistic Regression
  + Model is built max iteration values as 10000, no penalty and including all jobs i.e., number of jobs as -1.
  + After that model is passed through grid search with cross validation as 10, scoring accuracy and parameter grid.
  + In parameter grid we are passing different solver like ‘newton-cg’, ’lbfgs’, ‘sag’ and ‘saga’ among these best solver will be chosen by grid search.
  + After running for 10 folds and 4 candidates we are getting best parameter of solver as lbfgs.
  + From the best parameter we are getting both train and test accuracy as 88%.
  + A [regression](#_Regression_coefficient_Analysis) Analysis is also done on the model and Tenure is coming as the most significant variable whereas, Cashback is coming as least significant variable.
* Decision Tree
* Default Decision tree classifier model is built.
* After that it is passed to Randomized Search cv with estimator decision tree, cross validation as 10, seed =1, all available job and some parameter distribution.
* Parameters passed in search have criterion, max\_depth, min\_samples\_split, min\_samples\_leaf, min\_impurity\_decrease.
* Figure 26 shows best parameter of Decision tree.

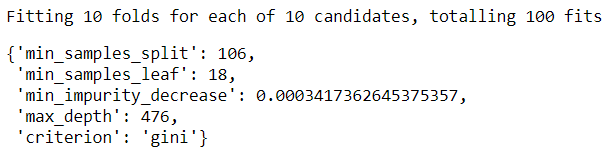


Figure 26: Decision Tree Best Parameter

* After using default parameter we are getting Train accuracy as 91% and Test Accuracy as 90%.
* Random Forest
* A default random forest is built with seed as 1.
* And then it is passed as estimator in Randomized search cv with other parameter like cross validation as 10 and all the jobs involved and parameter distribution.
* In Parameter distribution 5 parameters are used i.e., criterion, number of estimators, minimum samples leaf, max features and max samples.
* After Randomised search runs we find best parameters. And Figure 27 shows best parameter value.

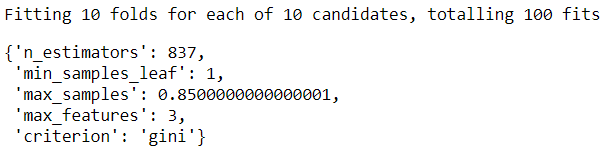


Figure 27: Random Forest Best Parameter

* After using above best parameter we are getting Train Accuracy as 100% and Test Accuracy as 97%.
* We are getting Tenure as most important variable and Payment\_UPI as least important. Figure 28 shows distribution of important variables.

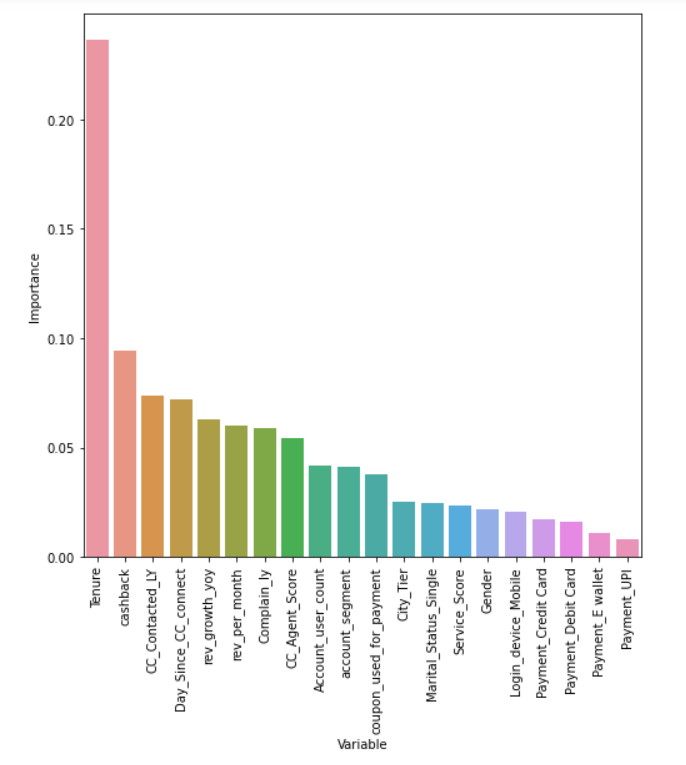


Figure 28: Random Forest Important Variable

* KNN Model
* To build model KNeighborsClassifier() is used from sklearn.
* Since it is a distance based algorithm so a pipeline is created which will first scale the value using Standard Scaler and then pass the value in model.
* Model is passed through GridsearchCV with scaled value in KNN as estimator, cross validation 10 and parameter grid.
* In parameter grid 3 parametrs are passed i.e., model\_KNN\_n\_neighbors, model\_KNN\_weights and KNN metrics.
* After running grid search cv we are getting best parameter. Figure 29 shows best parameters.

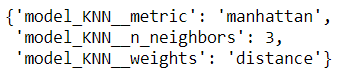


Figure 29: KNN Best Parameter

* After using best parameter we are getting Train Accuracy as 100% and Test Accuracy as 97%.
* Linear Discriminant Analysis
* Model is build using sklearn’s default LinearDiscriminantAnalysis() model.
* Model is passed as estimator to grid search cv with cross validation as 3 and parametric grid.
* In Parametric grid different solvers are passed.
* We get svd as best solver from Grid search for this problem. Figure 30 shows same.

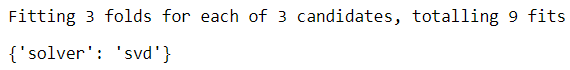


Figure 30: LDA Best Parameter

* By using above solver both Train and Test accuracy are coming as 87%.
* Adaptive Boosting
* Model is build using sklearn’s AdaBoostClassifer() model.
* Model is passed as estimator to Randomized search cv with cross validation as 10 and parameter distribution.
* In Parameter grid list of 2 variables passed number of estimator and learning rate.
* After running grid search we get best parameter. Figure 31 shows best parameter for Adaptive boosting.

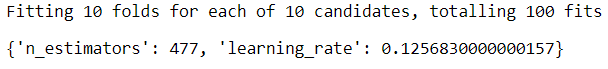


Figure 31: Adaptive model best parameter

* Model is giving both train and test accuracy as 90% for best parameter.
* We are getting Tenure as most important variable whereas Service Score as least important. Figure 32 shows distribution of important variables.

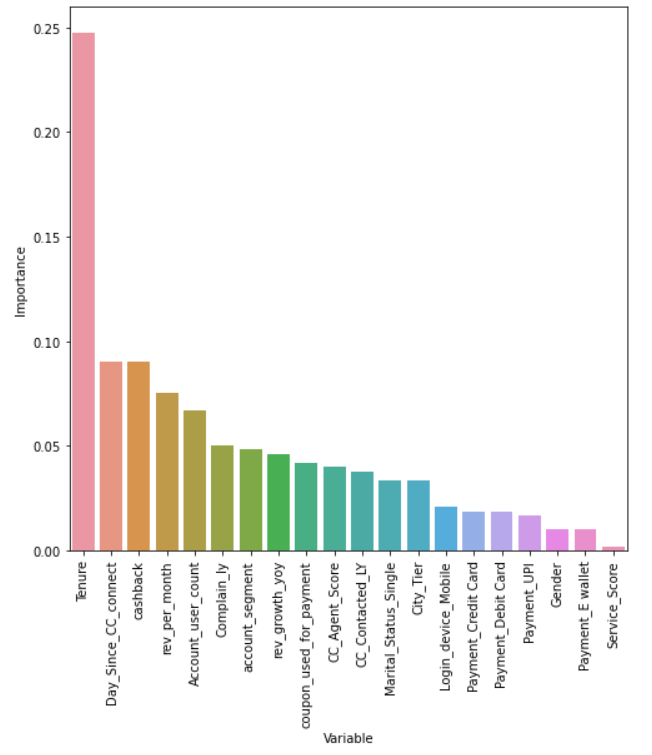


Figure 32: Adaptive Model Important Variables

* XG Boost Model
* Model is built using sklearn’s GradientBoostingClassifer() model.
* Model is passed as estimator to RandomizedSearchCV() with cross validation as 10 and parameter distribution.
* In parameter distribution 5 variables is passed i.e., learning rate, number of estimator, subsample, min sample split and min samples leaf.
* Figure 33 shows best parameter after running RandomSearchCV().

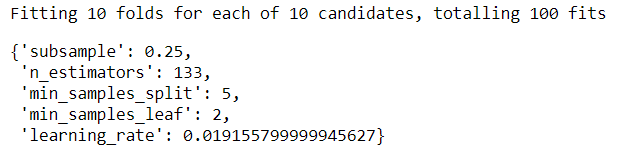


Figure 33: XG Boost Best Parameter

* With above best parameter model is giving both Train and Test accuracy as 90%.
* Tenure is coming as important variable whereas Payment\_DebitCard is coming as least important variable. Figure 34 shows distribution of important variables.

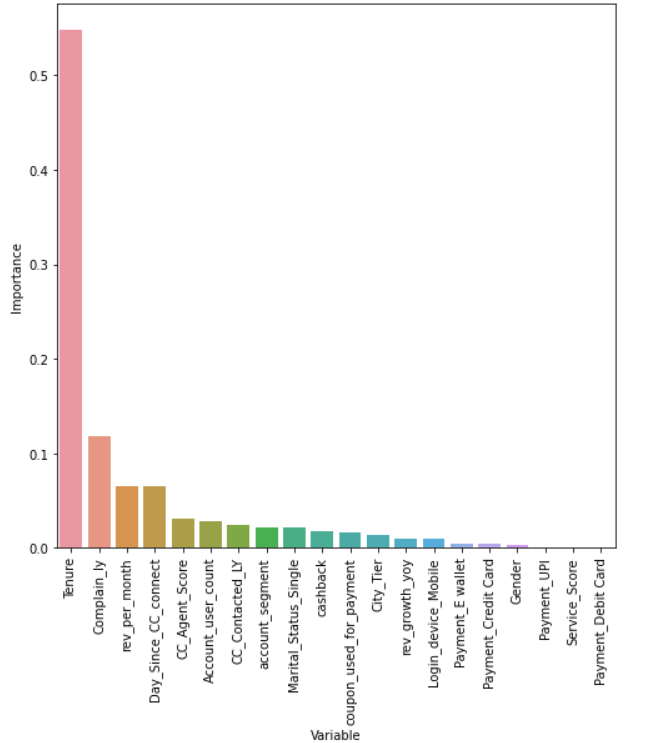


Figure 34: XGBoost Important Variables

[NOTE: - Parameter used during tuning can be found at Appendix-a](#_Model_Hyper-Parameter_Tuning)

## Effort to improve model performance.

* In order to improve model performance ensemble model were built.
* Decision Tree, Random Forest, KNN Model, Adaptive Model, XG Boost Model is selected to make ensemble model.
* Voting Classifier is chosen for first model with voting parameter as soft. Also same weights is assigned to Decision Tree, Random Forest and KNN i.e., 16 and Adaptive have 11 as weight whereas XG Boost has 12 as weight. Weight will help model to predict better.
* Figure 35 shows Voting Classifier parameter.

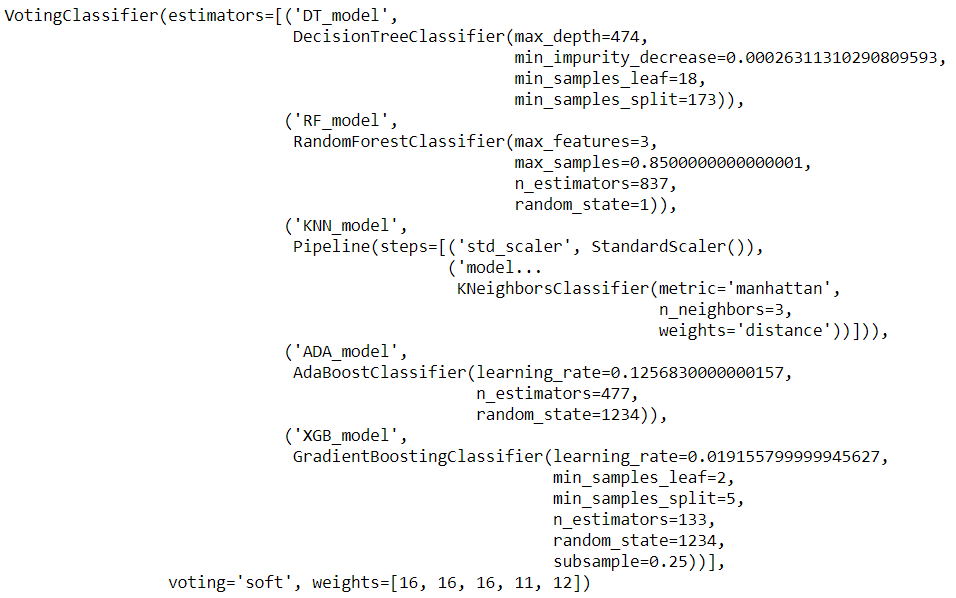


Figure 35: Voting Classifier Parameter

* Second Ensemble model is based on Stacking classifier.
* Logistic Regression is passed as final estimator and estimator list is being passed as estimator.
* Figure 36 shows parameter of Stacking Classifier.



Figure 36: Stacking Classifier Parameter

* Stacking Model is chosen above all model due to its better performance metric. Discussion on performance metric is done in next section.

# **Model validation**

## How was the model validated? Just accuracy, or anything else too?

* Model is validated based on different parameter figure 37 shows a quick comparison of all models.

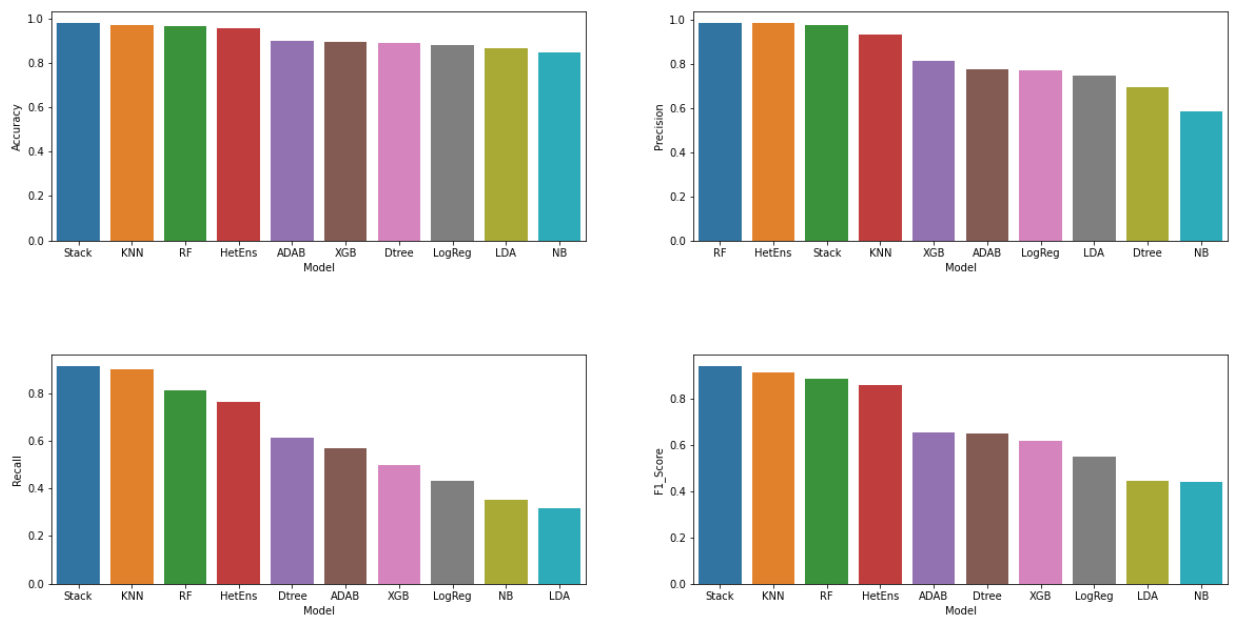


Figure 37: Model Comparison

* In terms of accuracy, recall and F1\_ score stack model is performing best.
* And random forest is performing best in terms of precision.
* Below is comparison of all model in tabular form on test set.

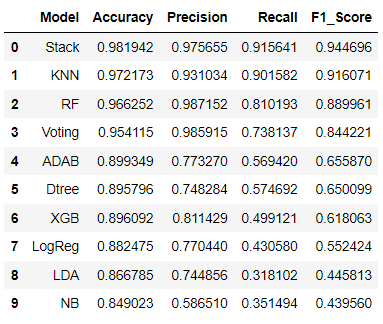


Figure 38: Tabular Model Comparison

* We are getting accuracy of above 84% with all the models.
* But since we are looking for churn model here accuracy will not be our parametric because, Accuracy is how many churners are predicted as churners and how many non-churners as non- churners.
* And it does not talk about the values which are not predicted right.
* So for that we will be using balance of precision and recall known as f1 score as our parameter.
* When F1 score is used as metric most of the model seem to underperform and have score less than 65%.
* Stack, KNN, RF and Voting Classifier came as our top 4 model with score above 84%.
* KNN and RF are having a decent F1 score of 0.91 and 0.88.
* Here Precision means out of all positive predicted how many where actual positive. So our stack model is able to predict 97% times positive values as positive.
* And recall means how many times are we able to recall positives in our prediction. So here we are getting 91% recall for stack model which means out of values predicted 91% times we are able to get positives.
* And since f1 score is harmonic mean of it we are getting 94% of f1 score.
* Below is performance metric of stack model.

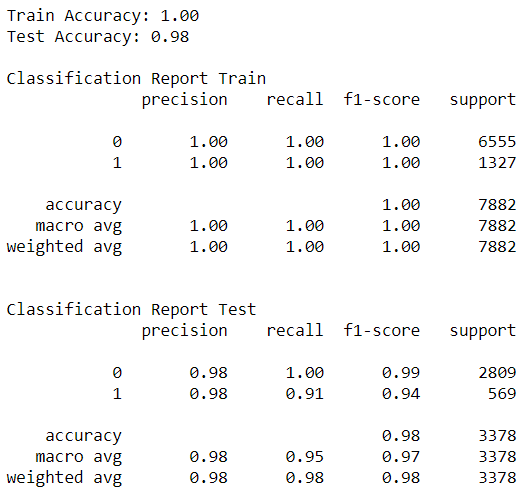


Figure 39: Stacking classifier report

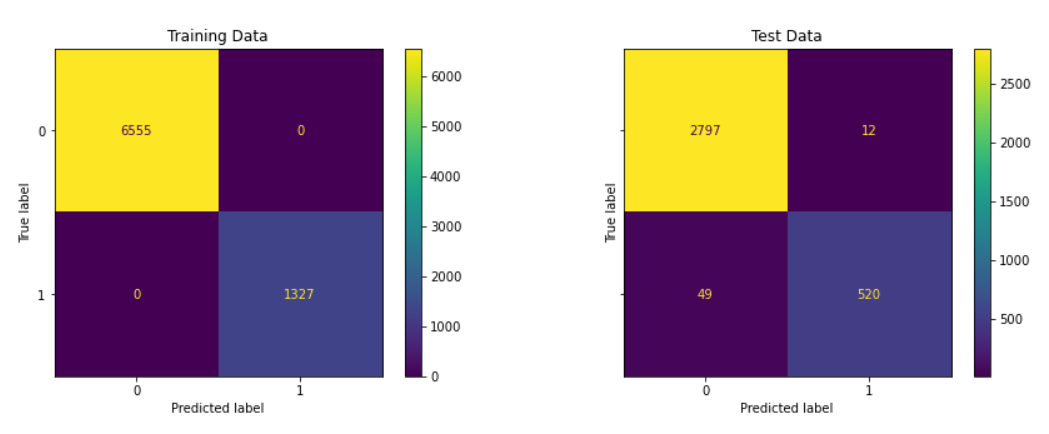


Figure 40: Stacking Classifier Confusion Matrix

* In confusion matrix also model seem to perform pretty well and we have reduced False Negative to 49 and also increased true positive from all other model.

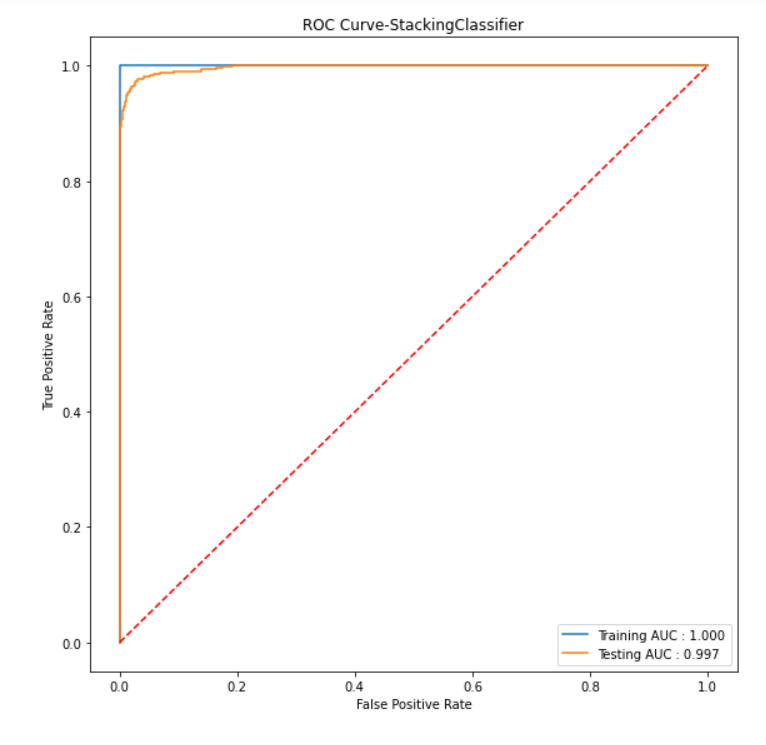


Figure 41: AUC and ROC

* Model is giving AUC score of 99.7% on test data.
* Comparison of all other model can be found in [appendix](#_Performance_Metrics_of).

# **Final interpretation / recommendation**

## Detailed recommendations for the management/client based on the analysis done.

* From regression analysis and EDA we came to know that Tenure is most important variable. So business need to focus on increasing tenure.
* Cashback is coming as least important so business need to stop focusing on cashbacks and they can offer some coupons but in limited amount.
* Service Score need to be increased by fixing the problem related to service.
* Business need to also limit users per account either to 3 or 4 and not more.
* Also customer are frequently connecting customer care either for payment related or service related problem and that need to be fixed.
* Quality of customer care is also average and below average. So, business also need to focus on increasing quality of customer care.
* Most subscribed plans are generating less revenue so business need to tackle it either by reducing user count or increasing plan cost.
* To ensure Tenure Company can introduce some offers like youth offers and other lucrative offers.

# **Appendix**

## Model Hyper-Parameter Tuning values

|  |  |
| --- | --- |
| Model | Tuning Parameter |
| Logistic Regression |  |
| Decision Tree |  |
| Random Forest |  |
| KNN Model |  |
| LDA |  |
| Adaptive Boosting |  |
| XG Boost |  |

## Performance Metrics of all models

* Classification Report:

|  |  |
| --- | --- |
| Model | Classification Report |
| Gaussian Naïve Bayes |  |
| Logistic Regression |  |
| Decision Tree |  |
| Random Forest |  |
| KNN Model |  |
| Linear Discriminant Analysis |  |
| Adaptive Boosting |  |
| XG Boost |  |

* Confusion Matrix

|  |  |
| --- | --- |
| Model | Confusion Matrix |
| Gaussian Naïve Bayes |  |
| Logistic Regression |  |
| Decision Tree |  |
| Random Forest |  |
| KNN Model |  |
| Linear Discriminant Analysis |  |
| Adaptive Boosting |  |
| XG Boost |  |

* ROC and AUC

|  |  |
| --- | --- |
| Model | ROC and AUC |
| Gaussian Naïve Bayes |  |
| Logistic Regression |  |
| Decision Tree |  |
| Random Forest |  |
| KNN Model |  |
| Linear Discriminant Analysis |  |
| Adaptive Boosting |  |
| XG Boost |  |

## Regression coefficient Analysis

