FLIP ROBO

MICRO CREDIT DEFAULTER

Submitted by:

Prachi Parmar

ACKNOWLEDGMENT

I researched about the Project details, Domain knowledge and some technical knowledge about solving imbalanced classification problems.

Domain Knowledge References:

https://www.sciencedirect.com/science/article/pii/S1877050919320277

https://ieeexplore.ieee.org/document/9240729

https://medium.com/analytics-vidhya/classification-model-for-loan-default-risk-prediction-98c2cc7ef1bf

Technical Knowledge References:

https://machinelearningmastery.com/combine-oversampling-and-undersampling-for-imbalanced-classification/#:~:text=Random%20oversampling%20involves%20randomly%20duplicating,training%20dataset%20does%20not%20matter.

https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/

INTRODUCTION

- ## Business Problem Framing

  We need to predict probability of a customer to repay the loan taken in next 5 days. Prediction of a Customer being a Defaulter i.e not repaying the loan label 0 or non-Defaulter meaning Loan has been repayed, label 1.

- ## Conceptual Background of the Domain Problem

  This kind of problems belong to Banking and Financial Services.

  Banking and Financial Service Companies deal with Transactions, investments, insurance, wealth management and loans.

- ## Review of Literature

  This project is created for a Microfinance company providing agricultural loans, business loans and etc to unbanked poor family living in remote areas with not much source of income.

  Our Problem belongs to Loans Domains, where there are s/w or models which can decide whether a newly enrolled customer will default or not.

  On Based on the Prediction from our model, we can save time and effort of manpower needed to look into the details of the customer to decide whether to provide them with credit loan or not.

  From this Defaulter Prediction Model, we can save the losses that occur due to the defaulter loans.

- ## Motivation for the Problem Undertaken

The unbanked poor families can be given opportunity to invest in their business that was previously a limitation for them. This could motivate them to change their futures and making an income. Opportunities should be given to everyone or the idea that there is an option available to take loans from bank for poor families .
This is a chance to bring change in the society and break the chain and give equal opportunity to all. This is a step in developing our country.

Analytical Problem Framing

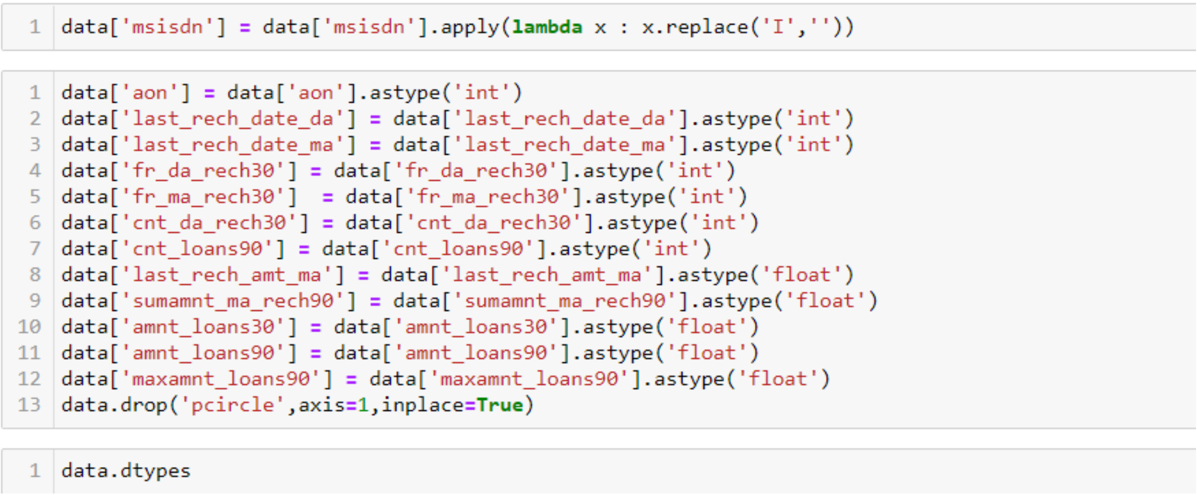
• Mathematical/ Analytical Modeling of the Problem

As the attributes of the Model like daily_descr30/90 i.e daily spend amount on a account, Average Balance of account or frequency of recharges on an amount, amount of loan taken from a account has a very wide range, from negative value to a high Value. For Eg, An account Balance can be 2,00,000 also and -500 also but average or most loan transaction account balance might lie between 0-50,000. Mathematically making negative values and extreme values like 2,00,000 outliers. But Realistically, these outliers are meaning full Data as we find relation that high account balance have less defaulter rate. These Extreme values that seem like outliers are meaningful data.

• Data Sources and their formats

Data was provided by the flipRobo technologies.
Some Data Attribute had Unrealistic Extreme values. For Eg, count of transaction/ Frequency of Recharges made will between the range of 0-200, but some Negative values and Extreme float values were found like 4,000,00 which needed to be treated.
And some dtypes of the Data needed to be changed from float to integer or inter to float. Refer the Photo below to see which columns needed dtypes changing.

For eg, msisdn was suppose to be phone number and it contained I in between which needed to change.
Aon feature, age of network should have been integer dtype as it is no of days which cannot be float. But it contained some extreme unrealistic values.



```
data['msisdn'] = data['msisdn'].apply(lambda x : x.replace('I',''))
```

```
data['aon'] = data['aon'].astype('int')
data['last_rech_date_da'] = data['last_rech_date_da'].astype('int')
data['last_rech_date_ma'] = data['last_rech_date_ma'].astype('int')
data['fr_da_rech30'] = data['fr_da_rech30'].astype('int')
data['fr_ma_rech30']  = data['fr_ma_rech30'].astype('int')
data['cnt_da_rech30'] = data['cnt_da_rech30'].astype('int')
data['cnt_loans90'] = data['cnt_loans90'].astype('int')
data['last_rech_amt_ma'] = data['last_rech_amt_ma'].astype('float')
data['sumamnt_ma_rech90'] = data['sumamnt_ma_rech90'].astype('float')
data['amnt_loans30'] = data['amnt_loans30'].astype('float')
data['amnt_loans90'] = data['amnt_loans90'].astype('float')
data['maxamnt_loans90'] = data['maxamnt_loans90'].astype('float')
data.drop('pcircle',axis=1,inplace=True)
```

```
data.dtypes
```

- **Data Preprocessing Done**

As Columns like Account Balance, Daily recharge amount are very widely spread date. In real Time, Account balance or features like this always have a Limit . For example, there is a limit in Spent amount, Account Balance Limit. Also, to keep a definite range of the Features Definite min/max limit needs to be set to decide our outliers as in some features outliers are meaningful data.

ASSUMPTIONS:

- The msisdn (Phone number) has least influence on whether the repayment will be done or not, and also I assumed that I in between the Number as separator and removed it.
- Ranges Assumed after visualizing the Data,
  1. For Aon, It cannot be greater than 2500 as most 99% values are below that level.
  2. For Fr_ma_rech, maximum value is assumed to be 250 as 99.5% data is below that range and above that range are just some unrealistic values.
  3. maxAmount_laons_30 it can only be in value of 6/12.
  4. Last Recharge Date MA,DA have some unrealistic Values and also as we checking the dataset for past 90 days history the Last recharge days should not be greater than 90 and cannot be Negative.

After Defining Boundaries for above Features, I replaced the unrealistic values to np.Nan as they are missing values and dropped the null values and encountered loss of 5.2%.

Data Cleaning was done by first removing the outliers, next step was removing the skewness with Power Transformer and it is not effected by presence of Outliers, Next I used PCA to determine no of attributes needed for estimating the Target variable (Defaulter/ Non Defaulter).

From PCA, I established 24 components were giving 95% + information about the Target variable.

Scaled PCA Data was next used for modelling the data.

- **Data Inputs- Logic- Output Relationships**

Rental30/90 , Avg account Balance 30/90 average is higher for non defaulters than defaulters category.
no relations found with target and medianprebal , medianamount.

SumAomunt 30/90 avg show relations with target variable. Non defaulters have total SumAmount is higher than Defaulters.

Count of Loans of Defaulter are not more than 20 in count (Major), while Non Defualters have wide range of count of loans below and above 20. Observation made is count of loans are more than 20 it is a non-defaulter loan transaction as per the visualization .

From the Amount of Loans Taken, we observe Defaulters donot have more than 120 in terms of amount of loans taken. We can assume that if laon amount is more than 120 , it might be Non defaulter loan transaction.

Max_Amount Loan 30/90 has 3 categoies 0/6/12 , 0 for new entries or first time loan takers. WE See that the max_ amount taken 0 category are Non Defaulters, which makes sense because To be a defaulter you need to take a loan , the loan _amount cannot be zero.

Last Recharge in no of days for non defaulters is lower as compared to Defaulters.

Daily Avg Amount Spend for Defaulter are very lower as compared to Non Defaulter.

- **State the set of assumptions (if any) related to the problem under consideration**

For Aon, It cannot be greater than 2500 as most 99% values are below that level.
For Fr_ma_rech, maximum value is assumed to be 250 as 99.5% data is below that range and above that range are just some unrealistic values.
Max amount of loan taken can be 0/6/12.
Last Recharge Date MA,DA have some unrealistic Values and also as we checking the dataset for past 90 days history the Last recharge days should not be greater than 90 and cannot be Negative.

- **Hardware and Software Requirements and Tools Used**

Python Notebook – jupyter for visualization, data cleaning, data modelling.

Model/s Development and Evaluation

- ## Identification of possible problem-solving approaches (methods)

This particular DataSet/Problem have a wide range with every feature like total loan amount can be as high and can have few high values and can be concentrated on various range high making high values of loan outliers but realistically seeing it is not an outliers, high loan amount gives the information that that loan transaction have less probability to default or not repaying. This is the Case with many features of this problem statement.

Here, outliers have to treated as meaningful data. Still, there are some extreme unrealistic values present in some columns which can be treated by assuming ranges manually for that columns.

Due to which, Data loss while treating actual misspelled values was 5.2%.

Data was a little skewed as distribution of Data had statistically considered outliers. It can be reduced via Power Transformer Scaling Technique as Min Max / Standard Scaler would be effected by outliers.

PCA was also used to remove columns that did not give much information about the Target Variable.

- ## Testing of Identified Approaches (Algorithms)

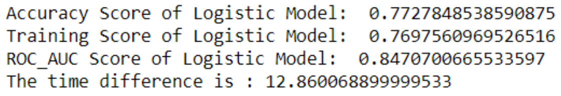
Test the Cleaned Data with logistic Regression, Decision Tree, Linear SVC, Random Forest, AdaBoost Algorithmns.

KNN and SVC not preferred due to time Constraints.

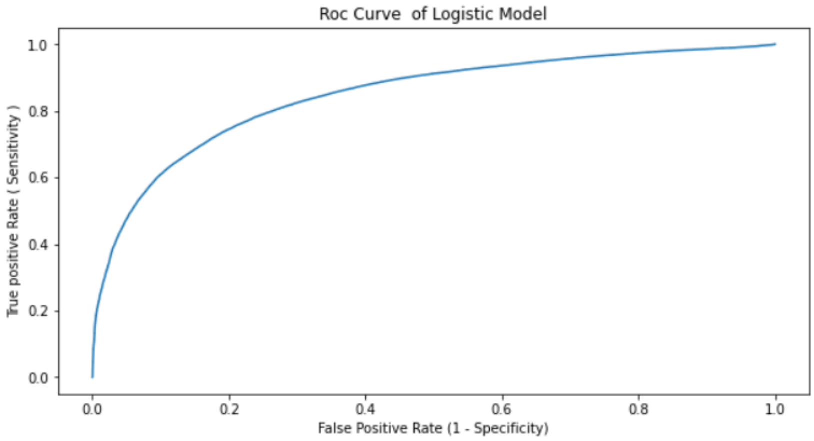
- **Run and Evaluate selected models**

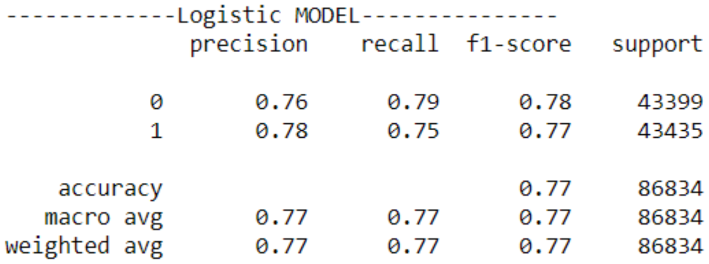
Logistic Regression: probability of target variable -1 if probability is >0.5 , else 0 if probability<0.5
Decision Tree: Every Feature is Split into nodes. Root Node being the most important feature influencing the Target.
Random Forest: bagging ensemble technique where many decision trees/ many same algorithms output is taken for vote and highest vote target is considered the Final Output .
Adaboost : Boosting ensemble Technique where each decision tree works in sequential manner, one giving the other model its predictions so that next model can reduce it.
LinearsVC

Logistic Regression MODEL: 77% (Accuracy Score) , Roc AUC SOCre (84%), cvScore: 76%

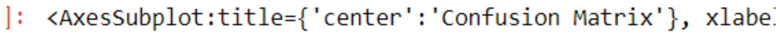


```
Accuracy Score of Logistic Model:  0.7727848538590875
Training Score of Logistic Model:  0.7697560969526516
ROC_AUC Score of Logistic Model:  0.8470700665533597
The time difference is : 12.860068899999533
```

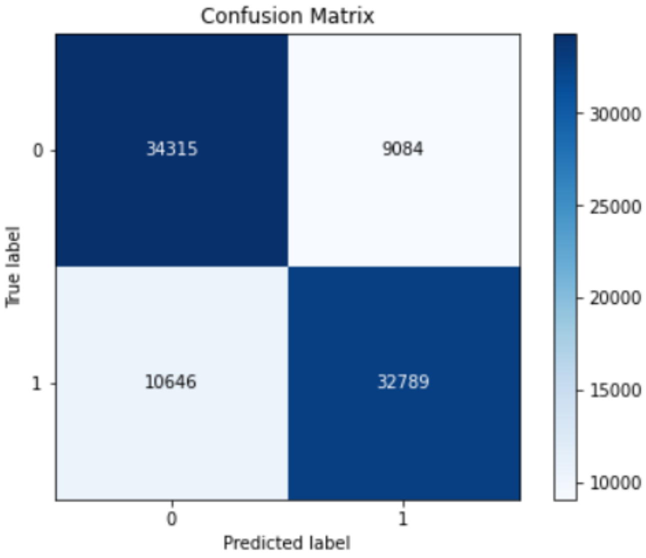




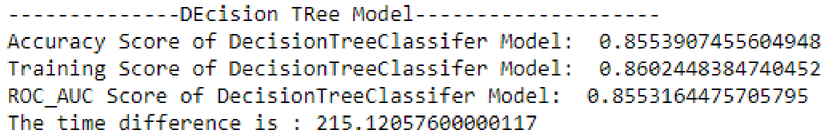
```
-------------Logistic MODEL--------------
              precision    recall  f1-score   support

           0       0.76      0.79      0.78     43399
           1       0.78      0.75      0.77     43435

    accuracy                           0.77     86834
   macro avg       0.77      0.77      0.77     86834
weighted avg       0.77      0.77      0.77     86834
```



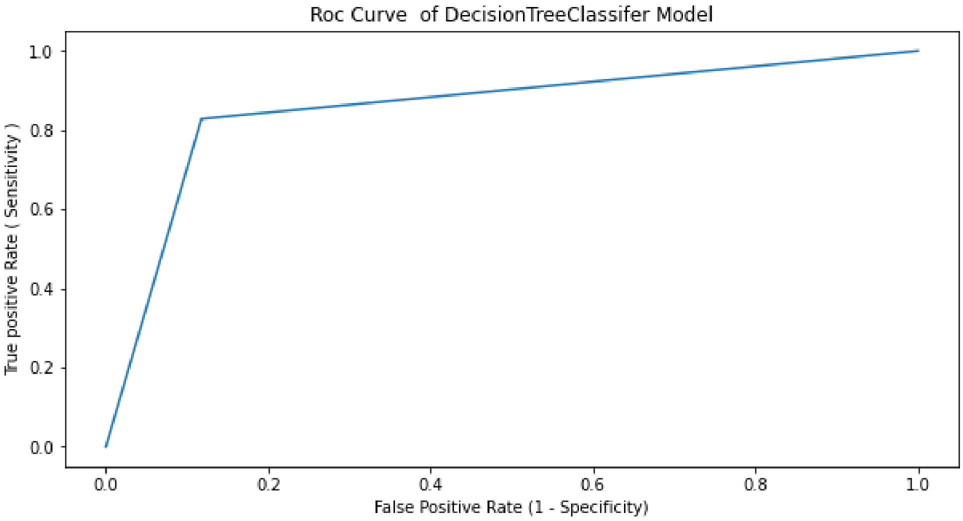
```
]: <AxesSubplot:title={'center':'Confusion Matrix'}, xlabe
```

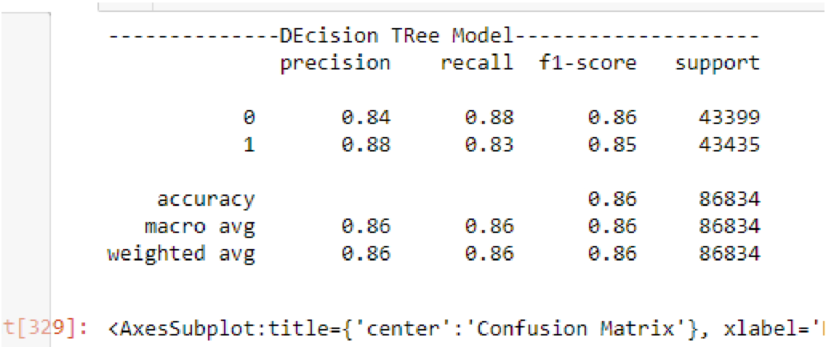


DECISIONTREE Model : ACC_SCORE: 85 , ROC_ACC_score: 85 , CV_Score:86



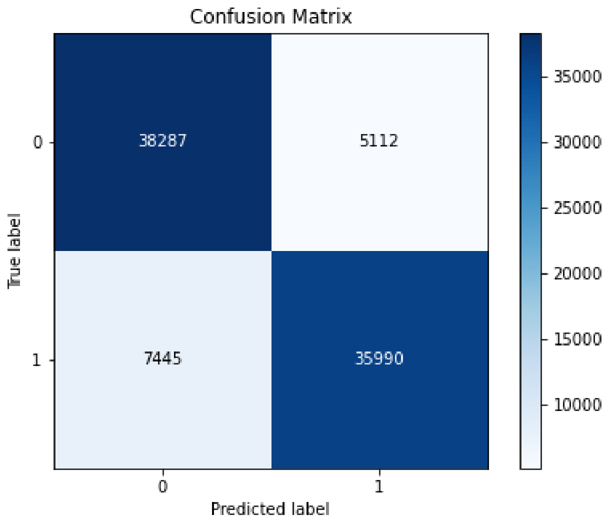
```
-------------DEcision TRee Model-------------------
Accuracy Score of DecisionTreeClassifer Model:  0.8553907455604948
Training Score of DecisionTreeClassifer Model:  0.8602448384740452
ROC_AUC Score of DecisionTreeClassifer Model:  0.8553164475705795
The time difference is : 215.12057600000117
```



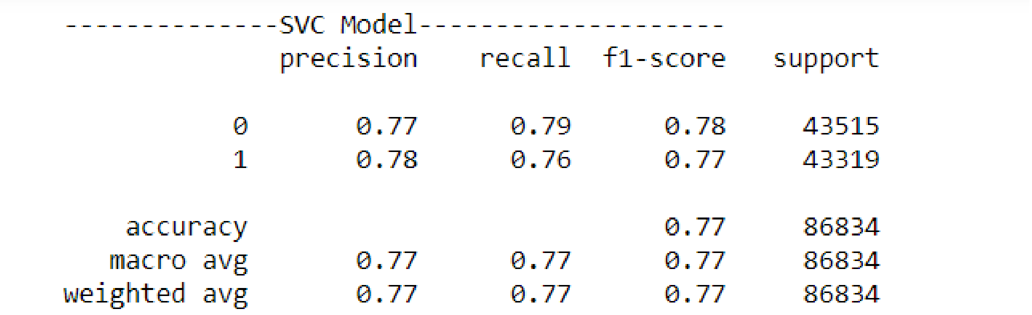


```
-------------DEcision TRee Model-------------------
              precision    recall  f1-score   support

           0       0.84      0.88      0.86     43399
           1       0.88      0.83      0.85     43435

    accuracy                           0.86     86834
   macro avg       0.86      0.86      0.86     86834
weighted avg       0.86      0.86      0.86     86834
```

```
t[329]: <AxesSubplot:title={'center':'Confusion Matrix'}, xlabel='
```

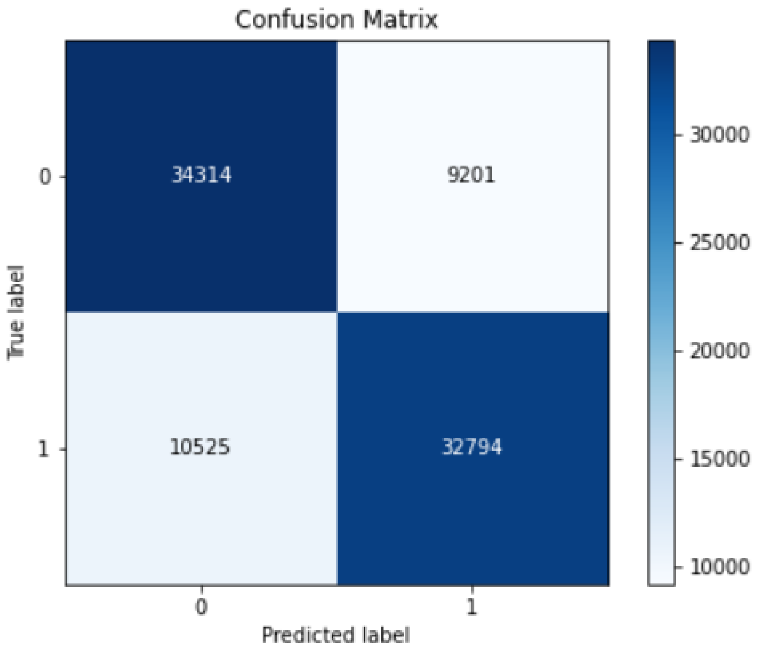


LINEAR SVC :



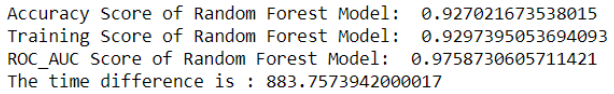
```
--------------SVC Model-------------------
              precision    recall  f1-score   support

           0       0.77      0.79      0.78     43515
           1       0.78      0.76      0.77     43319

    accuracy                           0.77     86834
   macro avg       0.77      0.77      0.77     86834
weighted avg       0.77      0.77      0.77     86834
```

```
ə]:  <AxesSubplot:title={'center':'Confusion Matrix'}, xlabel='Predi
```

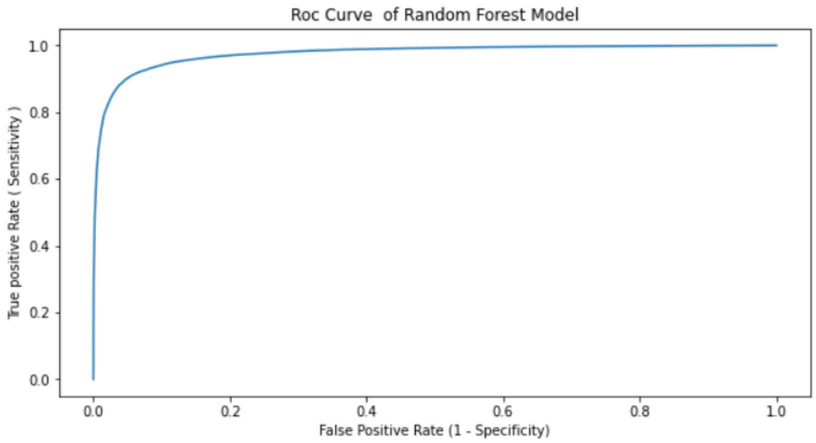


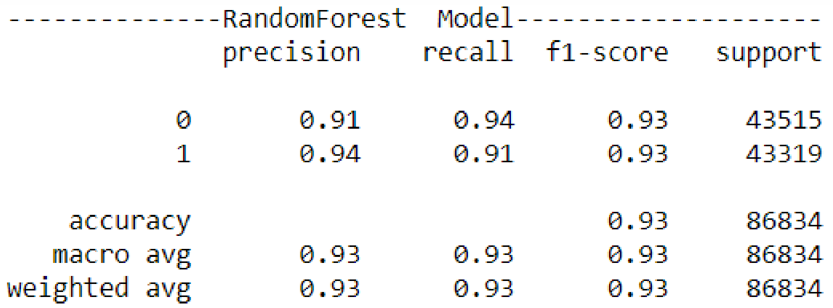
ACCURACY_score : 77%

RANDOM FOREST: : ACC_SCORE: 92 , ROC_ACC_score: 97 , CV_Score:92

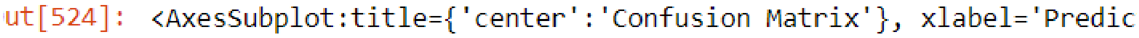


```
Accuracy Score of Random Forest Model:  0.927021673538015
Training Score of Random Forest Model:  0.9297395053694093
ROC_AUC Score of Random Forest Model:  0.9758730605711421
The time difference is : 883.7573942000017
```

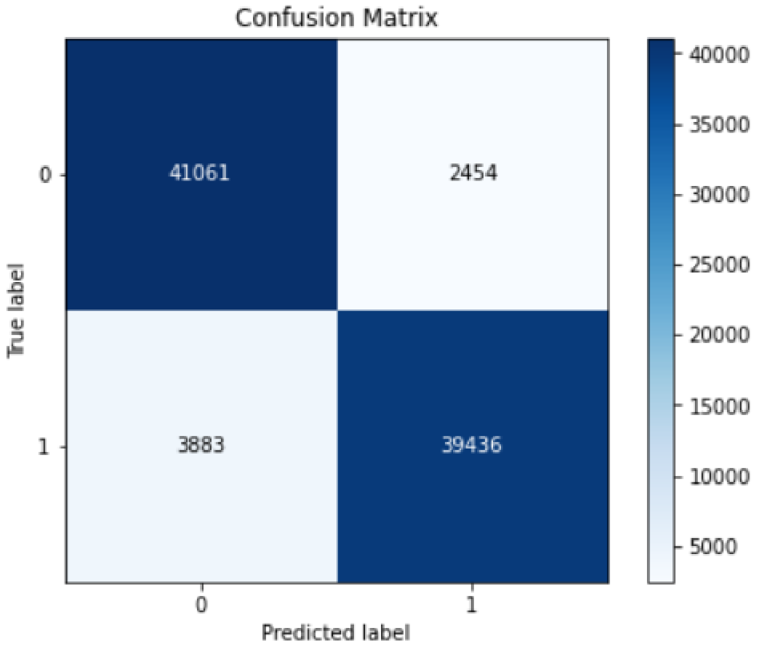




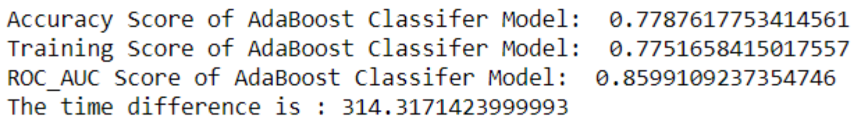
```
--------------RandomForest  Model-------------------
              precision    recall  f1-score   support

           0       0.91      0.94      0.93     43515
           1       0.94      0.91      0.93     43319

    accuracy                           0.93     86834
   macro avg       0.93      0.93      0.93     86834
weighted avg       0.93      0.93      0.93     86834
```



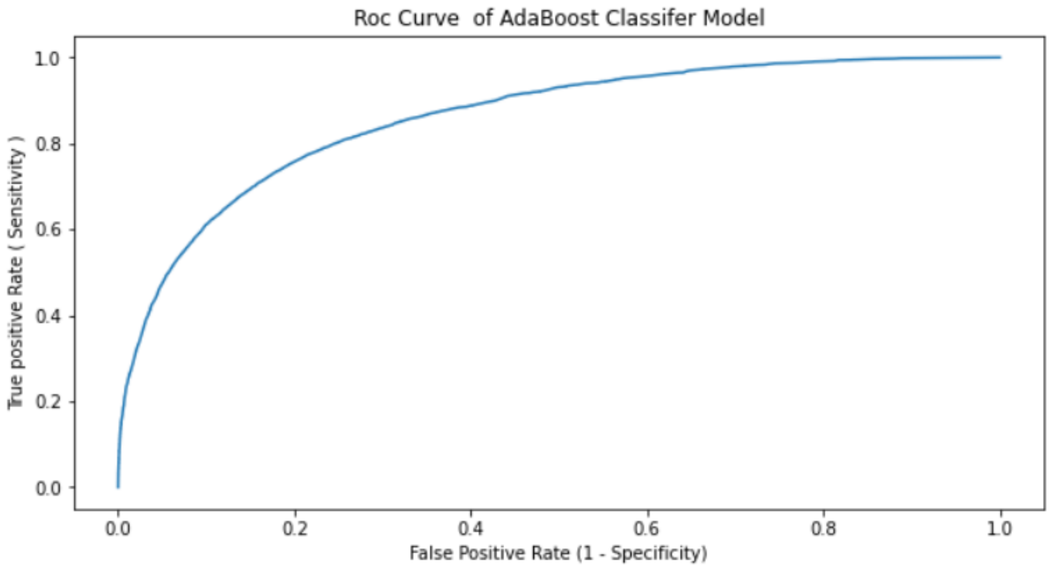
```
ut[524]:  <AxesSubplot:title={'center':'Confusion Matrix'}, xlabel='Predic
```

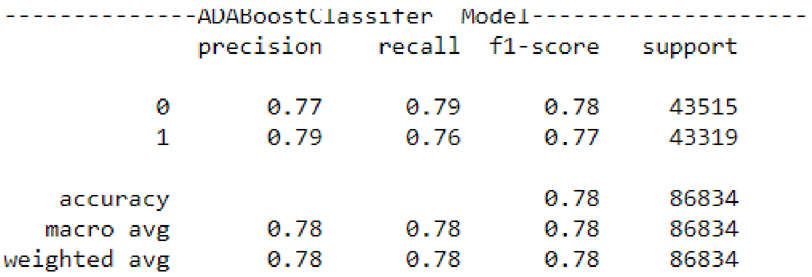


ADA BOOST : ACC_SCORE: 77 , ROC_ACC_score: 85 , CV_Score:77

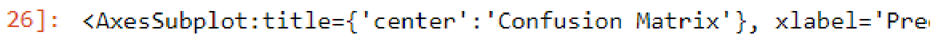


```
Accuracy Score of AdaBoost Classifer Model:  0.7787617753414561
Training Score of AdaBoost Classifer Model:  0.7751658415017557
ROC_AUC Score of AdaBoost Classifer Model:  0.8599109237354746
The time difference is : 314.3171423999993
```

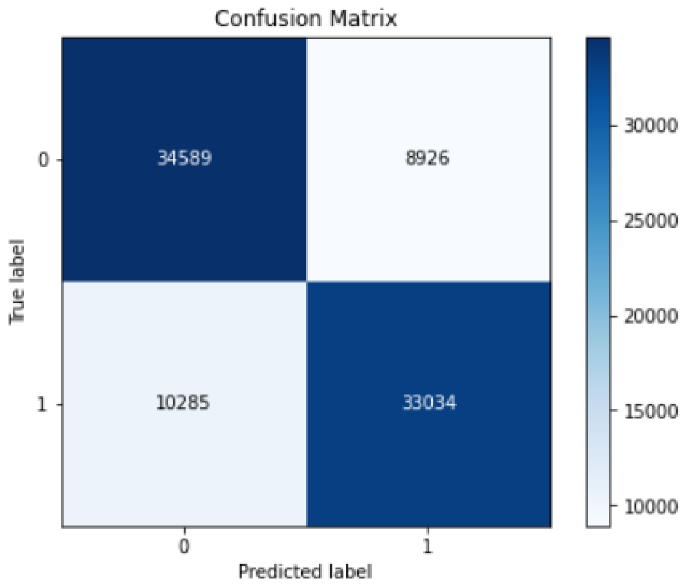




```
--------------ADABoostClassifer  Model--------------------
              precision    recall  f1-score   support

           0       0.77      0.79      0.78     43515
           1       0.79      0.76      0.77     43319

    accuracy                           0.78     86834
   macro avg       0.78      0.78      0.78     86834
weighted avg       0.78      0.78      0.78     86834
```

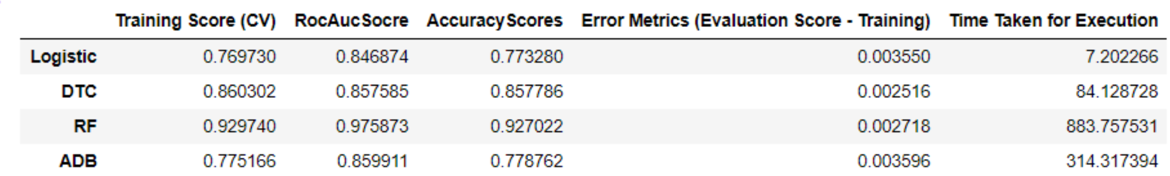


```
26]: <AxesSubplot:title={'center':'Confusion Matrix'}, xlabel='Pre
```



- **Key Metrics for success in solving problem under consideration**

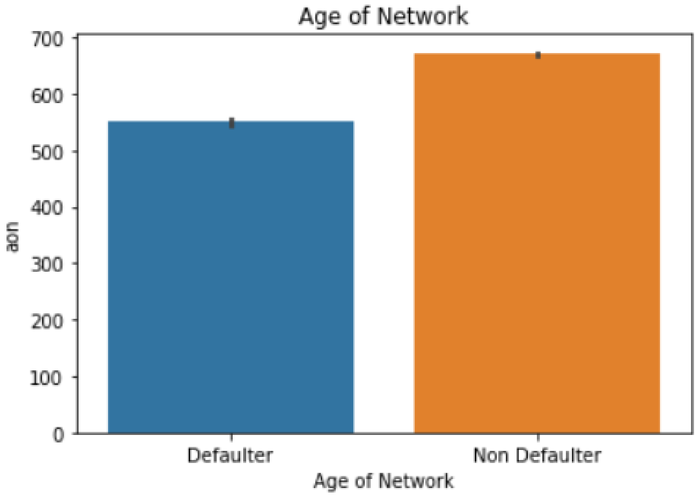
  Metrics used for evaluating the model are Accuracy_Score, Roc_AUC_Score, Precision, Recall, Confusion Matrix, F1-Score.
  Roc Curve.
  Error Metrics: AccuracyScore – TrainingScore(cross validation score)

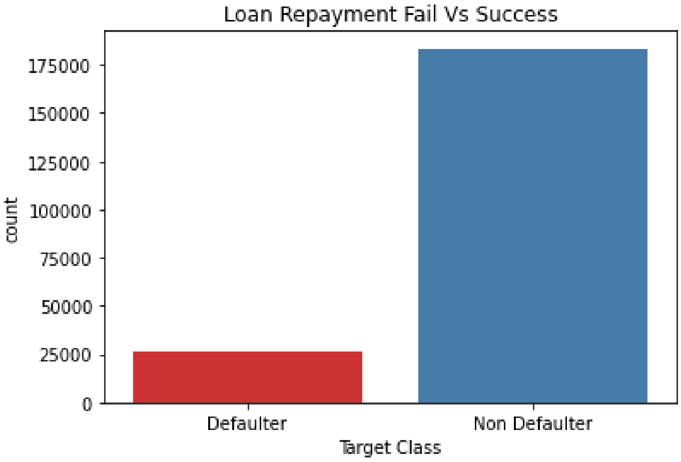


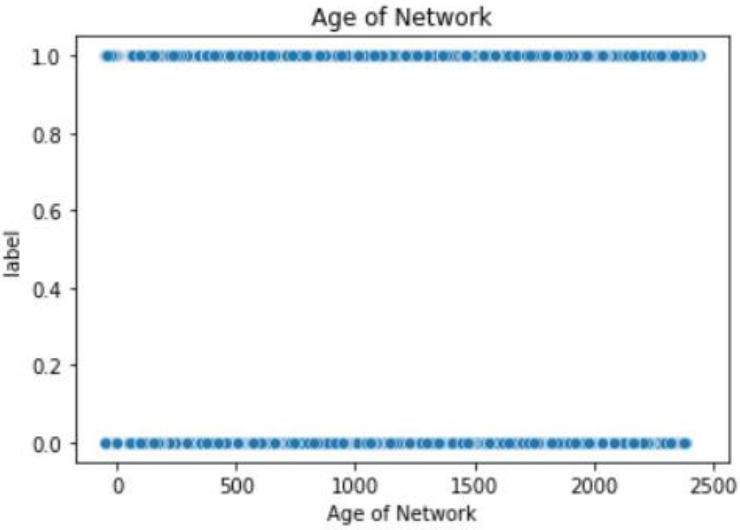
	Training Score (CV)	RocAucSocre	AccuracyScores	Error Metrics (Evaluation Score - Training)	Time Taken for Execution
Logistic	0.769730	0.846874	0.773280	0.003550	7.202266
DTC	0.860302	0.857585	0.857786	0.002516	84.128728
RF	0.929740	0.975873	0.927022	0.002718	883.757531
ADB	0.775166	0.859911	0.778762	0.003596	314.317394

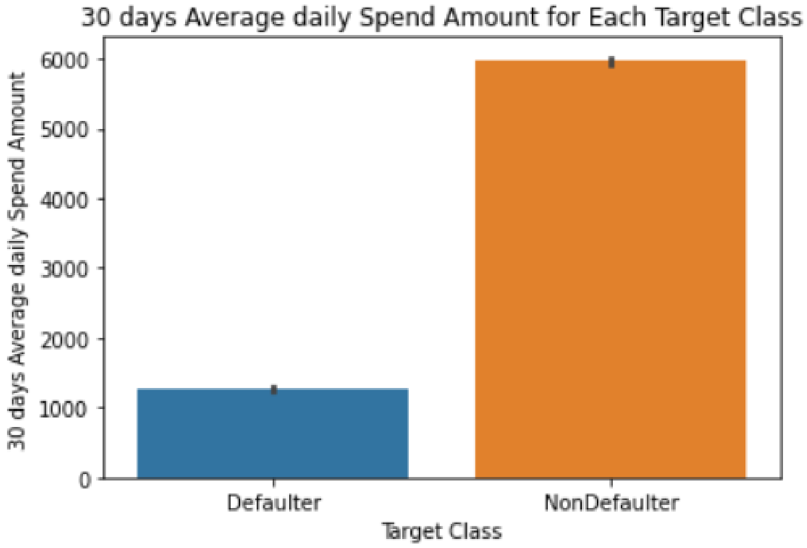
- **Visualizations**

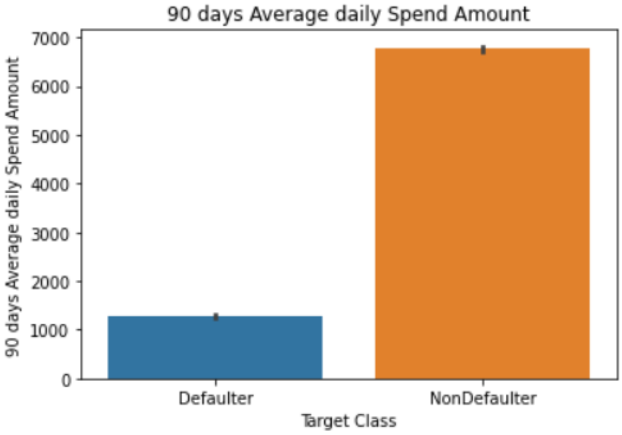
1. Countplots were used to check the Target Variable and its distribution – Imbalanced Classes.
2. Barplots and Scatterplots between each feature and Target Variable. Barplots helped in giving information of average value per Class . Scatterplots helped in visualizing the Spread of the values among the classes.
3. Heatmap to see correlation between the features and features and target. Heatmaps to check NULL values in the Data.
4. Boxplots to check statistically derived outliers and, median values of each feature.
5. Distribution plots to check the gaussian like structure for continuous features.
6. Joint plots to see relationship between features.

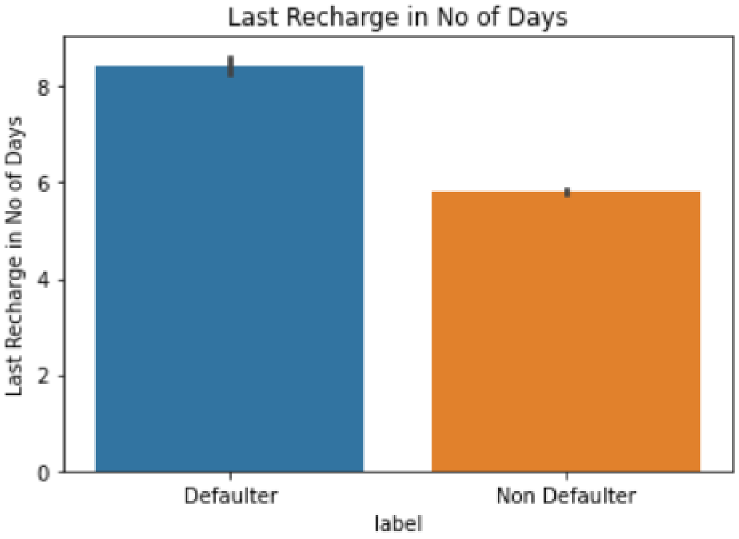

Age of Network
700
600
500
400
300
200
100
0
aon
Defaulter
Non Defaulter
Age of Network

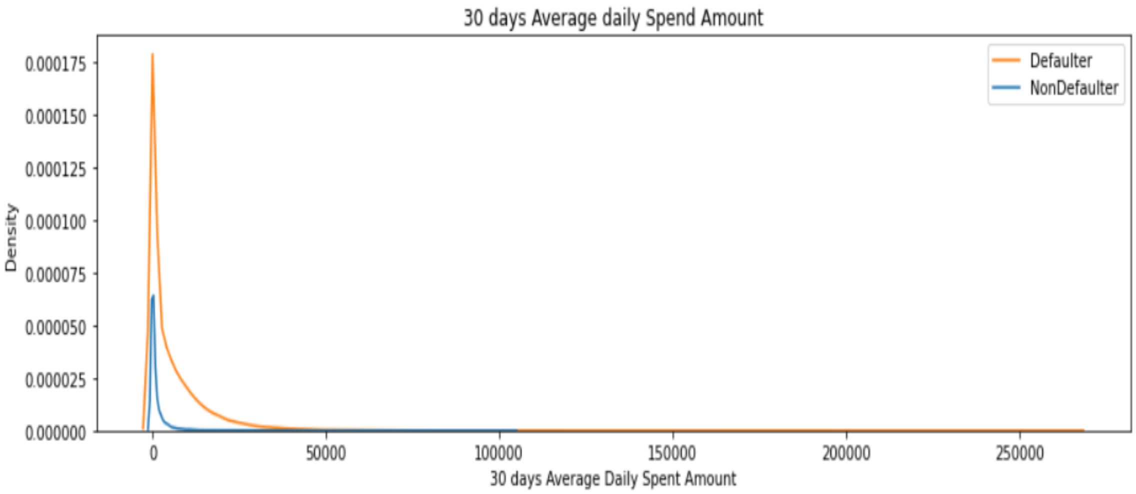

Loan Repayment Fail Vs Success
175000
150000
125000
100000
75000
50000
25000
0
count
Defaulter
Non Defaulter
Target Class

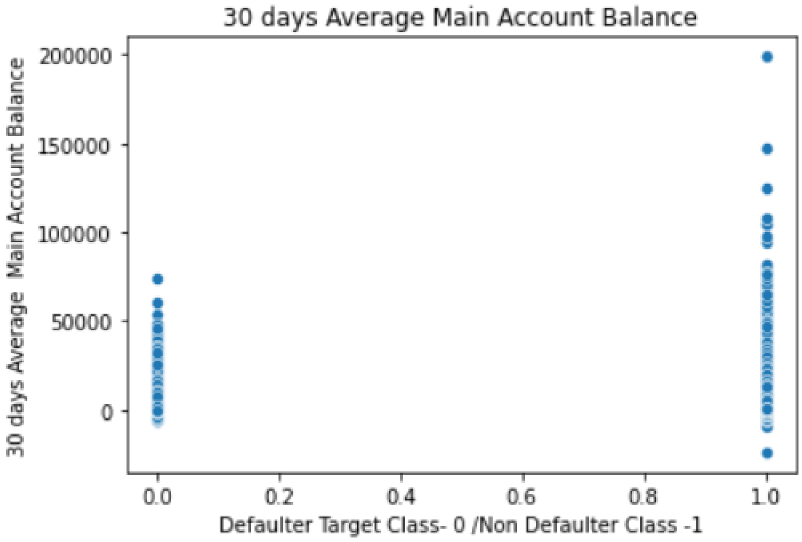

Age of Network
1.0
0.8
0.6
0.4
0.2
0.0
label
0
500
1000
1500
2000
2500
Age of Network

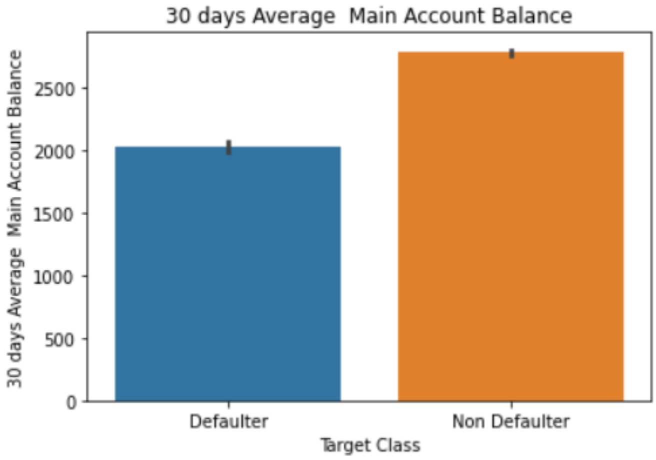

30 days Average daily Spend Amount for Each Target Class
30 days Average daily Spend Amount
0
1000
2000
3000
4000
5000
6000
Defaulter
NonDefaulter
Target Class

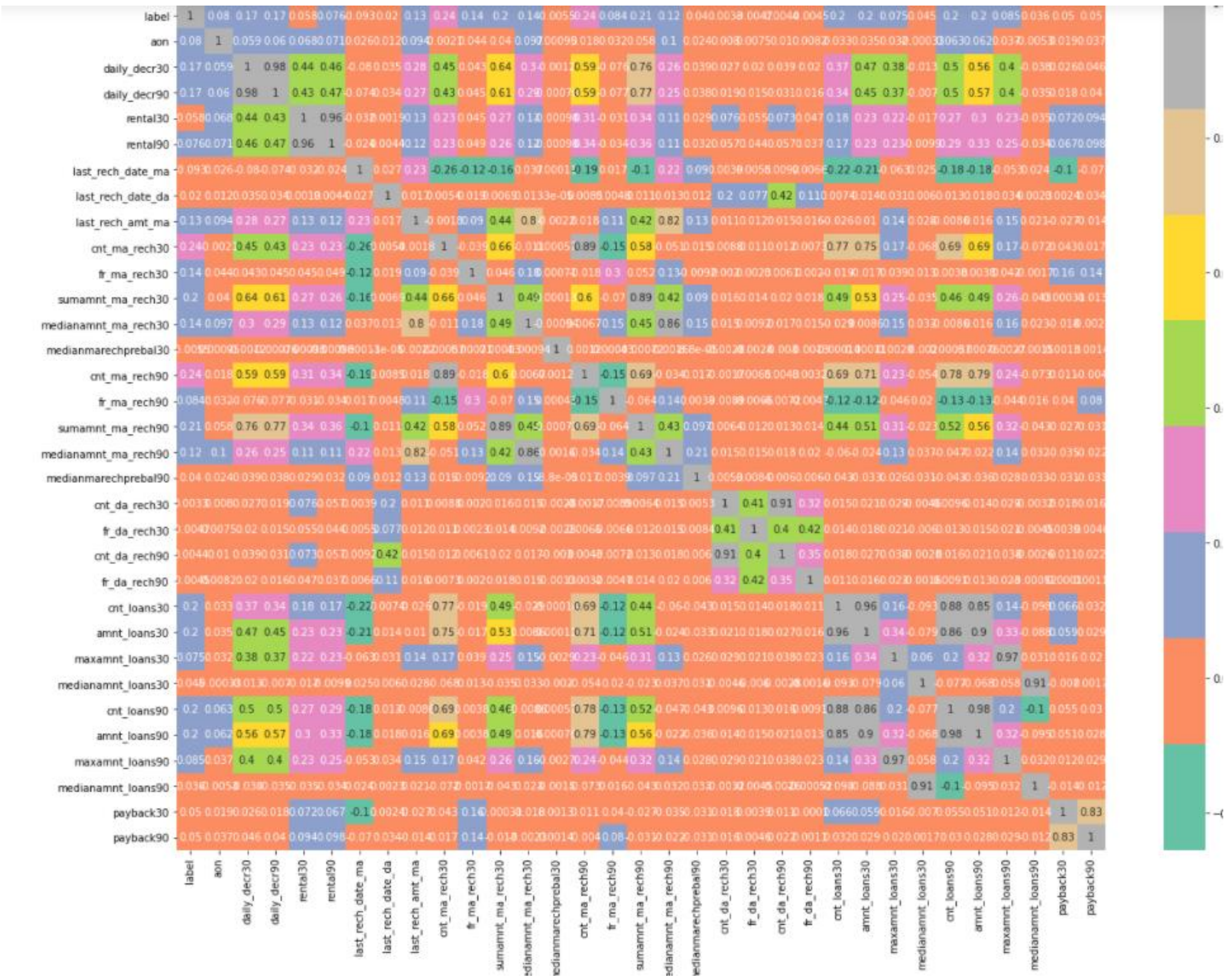

90 days Average daily Spend Amount
90 days Average daily Spend Amount
0
1000
2000
3000
4000
5000
6000
7000
Defaulter
NonDefaulter
Target Class


Last Recharge in No of Days
Last Recharge in No of Days
0
2
4
6
8
Defaulter
Non Defaulter
label


30 days Average daily Spend Amount
Defaulter
NonDefaulter
Density
0.000175
0.000150
0.000125
0.000100
0.000075
0.000050
0.000025
0.000000
0
50000
100000
150000
200000
250000
30 days Average Daily Spent Amount


30 days Average Main Account Balance
30 days Average Main Account Balance
200000
150000
100000
50000
0
0.0
0.2
0.4
0.6
0.8
1.0
Defaulter Target Class- 0 /Non Defaulter Class -1


30 days Average Main Account Balance
30 days Average Main Account Balance
2500
2000
1500
1000
500
0
Defaulter
Non Defaulter
Target Class



- **Interpretation of the Results**

  Statistically calculated outliers were not actually outliers but were meaningful data giving insights such that high account balance/high count of loans taken had a high probability to pay back the loan

  EDA of the Data gave various insights about the Data that Total Loan Amount, Last recharge Amount, Main Account balance, Daily Average Spend Amount count of loans were higher for non-defaulters than defaulters.
  AS our Data has meaning outliers, data is a little skewed and power transformer reduced and scaled the Data.

Modelling gave us insights that Bagging Technique works best for this kind of problem statements i.e instead of creating a strong model, combine results of many weak models and take vote of highest vote.

Random Forest Gives the highest model evaluation with Accuracy score of 92% i.e probability of correctly predicted data. And, ROC_AUC_SCORE of 97% i.e probability of correctly predicting A class w.r.t to other, false positive rate vs true positive rate of Target Variable 1 with respect to the other target Class.

CONCLUSION

- ## Key Findings and Conclusions of the Study

1. Target classes were highly imbalanced but oversampling can be used with SMOTHE to balance the classes.
2. Total Loan Amount, Last recharge Amount, Main Account balance, Daily Average Spend Amount count of loans were higher for non-defaulters than defaulters.
3. Last Recharge Date was higher was Defaulters.
   5. Median amount, prebalancemedianamount was less likely to influence or give less information about the Target Classes.
4. 5 Main Account information influenced more than Data Account Information as main account was found actively in use than Data account.
5. Many entries were first time loan takers and the obstacle is to predict the Defaulter/ non-defaulter on those loan transactions.
6. Best Model for this problem statement would be bagging techniques as in this type of problem voting would be more appropriate and random Forest would work better. The Random Forest gives highest accuracy for this Problem statement giving 92% Accuracy with metrics being roc_auc_score

- ## Learning Outcomes of the Study in respect of Data Science

Came across many challenges like, working with such high level of imbalanced classes.

Visualization took time as dataset size was large and no of features were large to examine the relations of each feature with the Target Variable.

Data Cleaning techniques as Here statistically performed outlier detection gave 15% of the data as outliers which was a high amount of Data Loss. Had to search ways and techniques to keep the Data Loss to minimum.

Also, IN this Case some outliers were meaning ful data giving insights eg, Very High Count of Loans, high account balance told us it has high probability to be a non defaulter.

But, Some Unrealistic extreme values were present which can be treated as missing values as the values were unrealistic, treated it as null values and droped those rows.

Total data loss was 5.2%.

- **Limitations of this work and Scope for Future Work**

Technical Limitation: Data highly imbalanced, as our evaluation of data, non defaulter prediction score is more than defaulter prediction score.

Scope of This problem : To decide whether a person repay or not was done by manpower by carefully checking its past histories and any open redlines (Threats). As this project is about giving loans to poor families for various personal and business purposes. This is a Big step in future as there is still High % of Below poverty line families in search of loans for business or personal purpose and this could give them/ or their business a chance to succeed. This is actually bring many % people above the poverty line if these loans used by them helps in their business ans business returns can motivate other poor families to take a chance.

LIMITATIONS: For this Project to succeed, proper domain explanation should be given in villages and town to educate people how to execute ideas. As ideas can come to anyone irrespective of their caste, colour and gender and society status. But, Below poverty line people do not always have the chance to execute it or they don’t know if a chance is available to them to do so. Education of the Domain of people getting credit loans is important among the poor families.