

Micro-Credit Defaulter Model

Submitted by:

ABHISHEK MURARKA

**ACKNOWLEDGMENT**

I would like to express my sincere gratitude to FlipRobo Technologies for supporting me throughout the internship and giving me the opportunity to explore the depth of Data Science by providing multiple projects like this, there are multiple people, organizations, youtubers, who guided me in this wonderful journey and few journals which helped me develop my models in this project. I would like to thank following people for the inspiration and help,

* FlipRobo Technologies
* DataTrained Team
* Krish Naik
* surampudi vnss siva Krishna
* Jason Schvach

Research papers that helped me in this project was as follows:

<https://www.mdpi.com/1911-8074/13/8/180/pdf>

<https://www.czso.cz/documents/10180/88506448/32019719q2_129_mandak_analyses.pdf/550e60c0-149c-42ce-9bfa-b449002b10c6?version=1.0>

<https://www.researchgate.net/publication/319993632_Use_of_machine_learning_techniques_in_the_prediction_of_credit_recovery>

**INTRODUCTION**

* Business Problem Framing

This project was highly motivated project as it includes the real time problem for Microfinance Institution (MFI), and to the poor families in remote areas with low income, MFI to provide micro-credit on mobile balances to be paid back in 5 days. The Consumer is believed to be defaulter if he deviates from the path of paying back the loaned amount within the time duration of 5 days. For the loan amount of 5 (in Indonesian Rupiah), payback amount should be 6 (in Indonesian Rupiah), while, for the loan amount of 10 (in Indonesian Rupiah), the payback amount should be 12 (in Indonesian Rupiah).

* Conceptual Background of the Domain Problem

Generally, Credit Scores plays a vital role for loan approvals, and is very important in today’s financial analysis for an individual, Most of the loan lending vendors rely heavily on it, so in our case users has 5 days’ time to pay back the loan or else they are listed as defaulters which will impact the loan the credit score heavily, so there are few thing to lookout in this dataset as users who are taking extensive loans, user who have most frequent recharges in their main account have a good chance of 100% payback rate, and user who never recharged their main account for them loan should have never been approved as there is high chance for single user or default user taking multiple connections in name or documents of the family members.

* Review of Literature

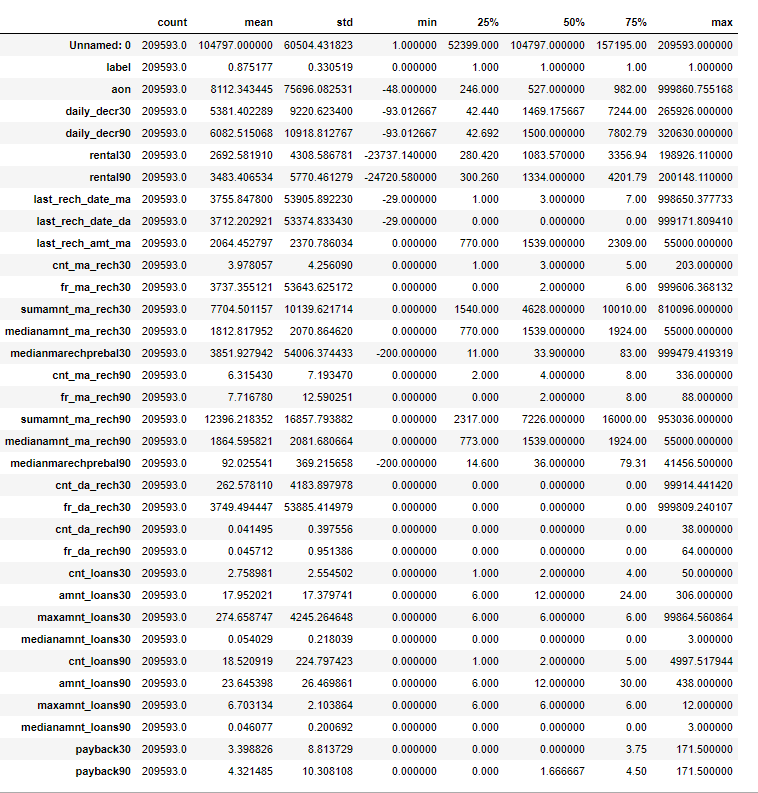
Logistic Regression, Decision Trees and Random Forest are evaluated for their ability to classify defaulters using several cross-validation approaches and the latter model performed best. When the default rate is below 2%, it is better to offer everyone a loan. For higher default rates, the model Substantially enhances profitability. The model quadruples the tolerable level of default rate for breaking even from 8% to 32%. Nonlinear classification models offer considerable potential for credit scoring, coping with higher levels of default and therefore allowing for larger volumes of customers.

* Motivation for the Problem Undertaken

This project was highly motivated project as it includes the real time problem for Microfinance Institution (MFI), and to the poor families in remote areas with low income, and it is related to financial sectors, as I believe that with growing technologies and Idea can make a difference, there are so much in the financial market to explore and analyse and with Data Science the financial world becomes more interesting.

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem



From the above statistical summary of the above part of the dataset, we can see that the min value is negative which is not even possible for most of the features like daily recharge and main account balance, and last recharge can't be negative.

We created multiple group based on min, 25% to 75%, above 75% and we compared it VS payback within 5 days.

We identified the outliers for features whose Z-score>3, and then did mean imputing and also applied cube root to bring the data closer to distribution.

We checked the correlation of the independent and dependent features and dropped the negative and less important features with the help of correlation matrix.

* Data Sources and their formats

**label :** Flag indicating whether the user paid back the credit amount within 5 days of issuing the loan{1:success, 0:failure}

**msisdn :** mobile number of user

**aon :** age on cellular network in days

**daily\_decr30:** Daily amount spent from main account, averaged over last 30 days (in Indonesian Rupiah)

**daily\_decr90:** Daily amount spent from main account, averaged over last 90 days (in Indonesian Rupiah)

**rental30:** Average main account balance over last 30 days

**rental90:** Average main account balance over last 90 days

**last\_rech\_date\_ma:** Number of days till last recharge of main account

**last\_rech\_date\_da:** Number of days till last recharge of data account

**last\_rech\_amt\_ma:** Amount of last recharge of main account (in Indonesian Rupiah)

**cnt\_ma\_rech30:** Number of times main account got recharged in last 30 days

**fr\_ma\_rech30:** Frequency of main account recharged in last 30 days

**sumamnt\_ma\_rech30:** Total amount of recharge in main account over last 30 days (in Indonesian Rupiah)

**medianamnt\_ma\_rech30:** Median of amount of recharges done in main account over last 30 days at user level (in Indonesian Rupiah)

**medianmarechprebal30:** Median of main account balance just before recharge in last 30 days at user level (in Indonesian Rupiah)

**cnt\_ma\_rech90:** Number of times main account got recharged in last 90 days

**fr\_ma\_rech90:** Frequency of main account recharged in last 90 days

**sumamnt\_ma\_rech90 :** Total amount of recharge in main account over last 90 days (in Indian Rupee)

**medianamnt\_ma\_rech90:** Median of amount of recharges done in main account over last 90 days at user level (in Indian Rupee)

**medianmarechprebal90:** Median of main account balance just before recharge in last 90 days at user level (in Indian Rupee)

**cnt\_da\_rech30:** Number of times data account got recharged in last 30 days

**fr\_da\_rech30:** Frequency of data account recharged in last 30 days

**cnt\_da\_rech90:** Number of times data account got recharged in last 90 days

**fr\_da\_rech90:** Frequency of data account recharged in last 90 days

**cnt\_loans30:** Number of loans taken by user in last 30 days

**amnt\_loans30:** Total amount of loans taken by user in last 30 days

**maxamnt\_loans30:** maximum amount of loan taken by the user in last 30 days

**medianamnt\_loans30:** Median of amounts of loan taken by the user in last 30 days

**cnt\_loans90:** Number of loans taken by user in last 90 days

**amnt\_loans90:** Total amount of loans taken by user in last 90 days

**maxamnt\_loans90:** maximum amount of loan taken by the user in last 90 days

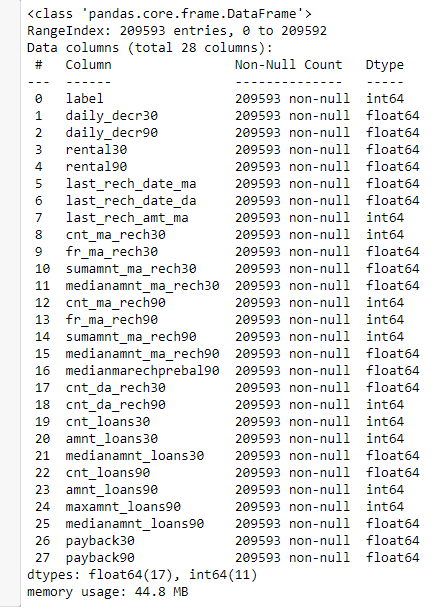
**medianamnt\_loans90:** Median of amounts of loan taken by the user in last 90 days

**payback30:** Average payback time in days over last 30 days

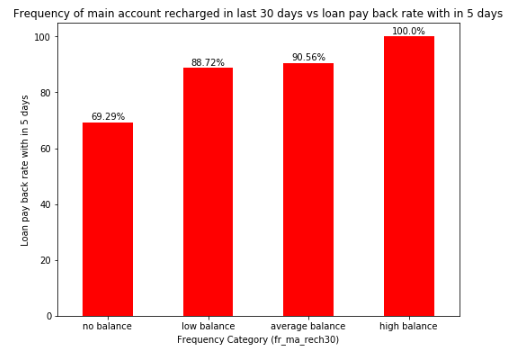
**payback90:** Average payback time in days over last 90 days

**pcircle:** telecom circle

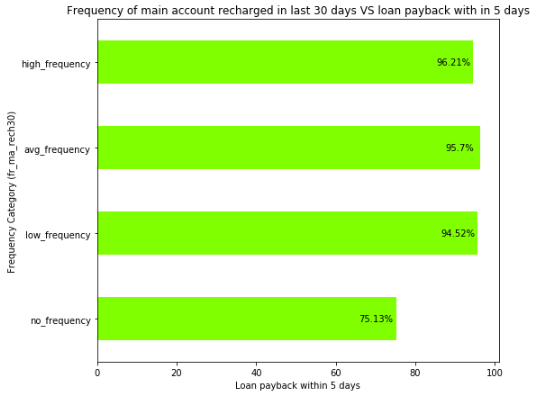
**pdate:** date



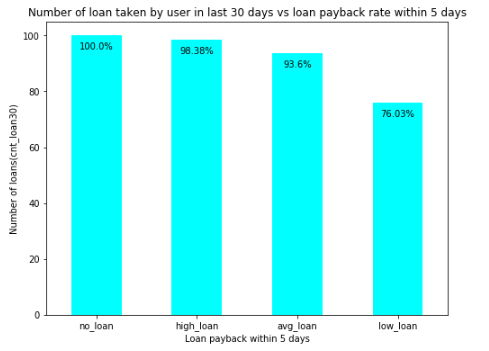
* Data Pre-processing Done
* We created multiple groups based on min, 25% to 75%, above 75% and we compared it VS payback within 5 days.
* We identified the outliers for features whose Z-score>3, and then did mean imputing and also applied cube root to bring the data closer to distribution.
* We checked the correlation of the independent and dependent features and dropped the negative and less important features with the help of correlation matrix.
* Applied SMOTETomek, to balance the dataset as the dataset was imbalanced dataset.
* Applied StandardScaler to our dependent features.
* Applied various machine learning model and compared it.
* Applied hyper tunning several models, but couldn’t achieve much better results.
* Saving final predictions in file.csv format
* Data Inputs- Logic- Output Relationships



From the above we can see that users with high balance always pays back the loan within 5 days and average and low category only 9% - 12% users failed to payback the loan within 5%, and users with zero balance around 30% users are not paying the loan back within 5 days.



From the above we can see there are no 100% rate in any frequency group to payback the loan within 5 days, and all average low and high frequency have atleast 6% to 4% users didn't payback the loan within 5 days. Comming to user have no frequency 25% users didn't payback the loan with in 5 days, till now we can see that users with no balance and no frequency are costing huge losses, company should implement some kind of strategies to reduce that like send SMS alerts for notification.



From the we can see that majority user who took high loans in last 30 days are more likely to payback within 5 days and 1.62% users failed to payback within 5 days, and among average loan user 7% users failed to pay back the loan within 5 days, and users with low loan have 24% didn't payback as expected might be defaulted.

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

Assumptions taken before the project was that user with low frequency or no frequency recharge shouldn’t have any loan, but they were provided with loan and 30% users defaulted from it, and after observation found that aon feature in statistical summary has outliers’ values which are impossible to be the age of the network.

* Hardware and Software Requirements and Tools Used

Hardware: 8GB RAM, 64-bit, i7 processor.

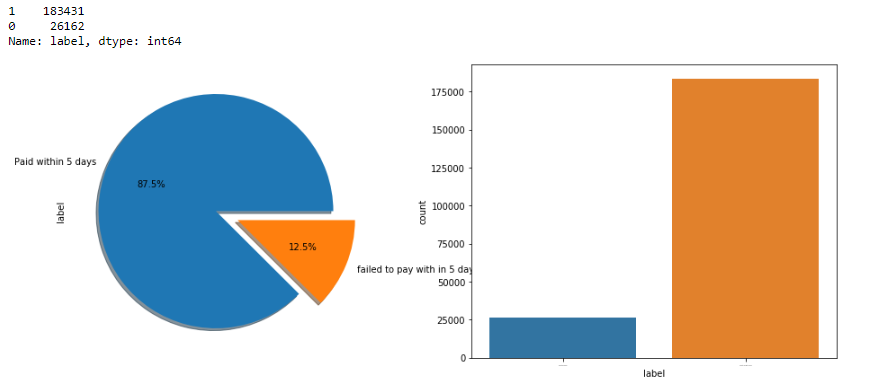
Software: Excel, Jupyter Notebook, python 3.6.

Libraries Used:-



**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)



From the above we can see that the data set is highly imbalanced dataset, so applied SMOTETomek to balance the dataset.

* Testing of Identified Approaches (Algorithms)

KNN = KNeighborsClassifier()

XGBC=xgb.XGBClassifier()

LR=LogisticRegression()

DT=DecisionTreeClassifier()

GNB=GaussianNB()

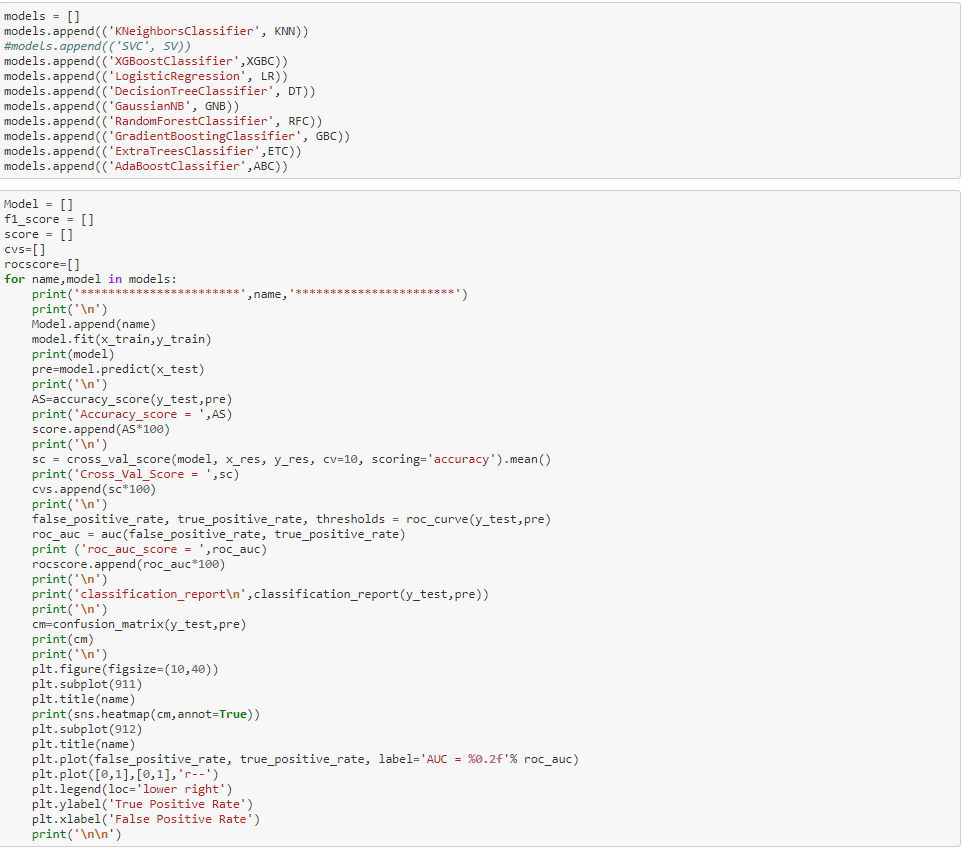
RFC=RandomForestClassifier()

GBC=GradientBoostingClassifier()

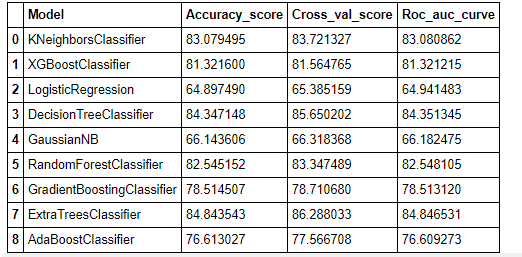
ABC=AdaBoostClassifier()

ETC=ExtraTreesClassifier()

* Run and Evaluate selected models

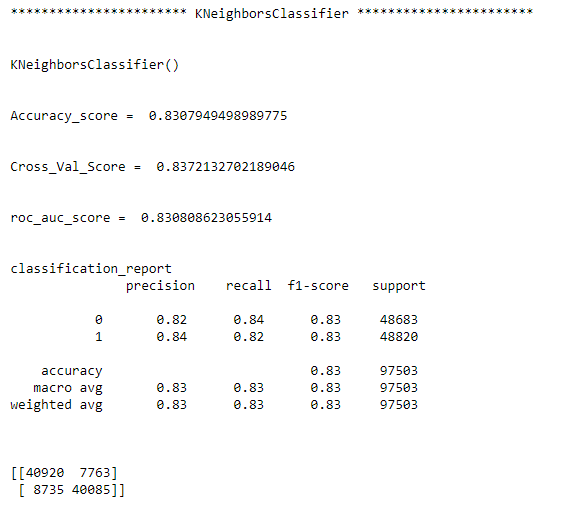


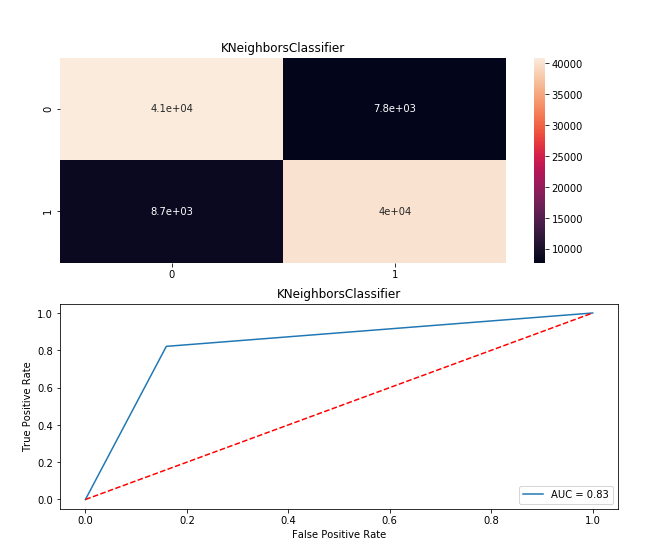
* Key Metrics for success in solving problem under consideration

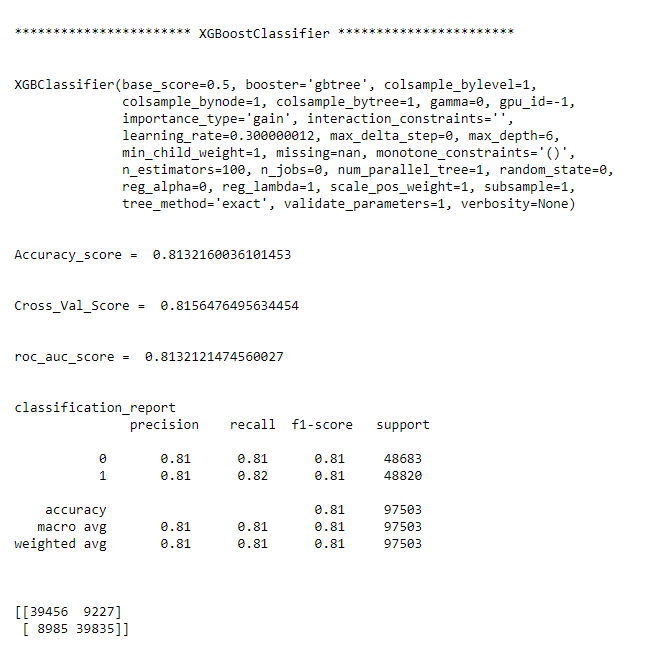


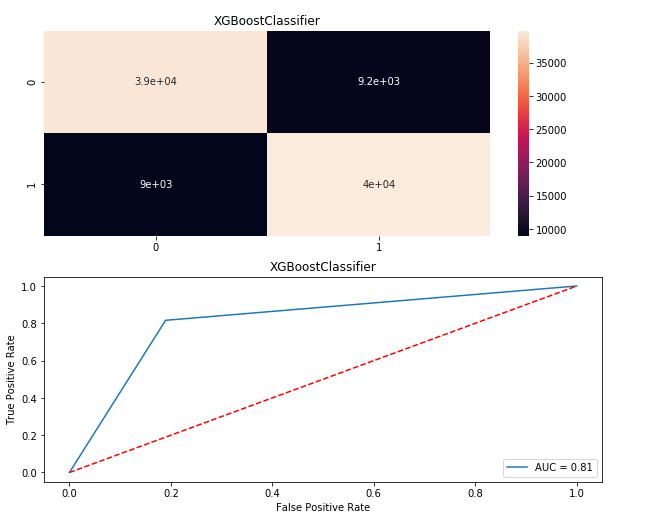
Key Metrices used were the Accuracy Score, Crossvalidation Score and AOC & ROC Curve as this was binary classification problem and we focus more on AOC & ROC curve metrices to observe True Positive Rate and False Positive Rare, for users who paid the loan and falsely marked as default and will their affect the credit score and we already talked about the importance of that in financial sector, and for the users who are marked falsely marked as paid but they didn’t, can affect the company revenue.

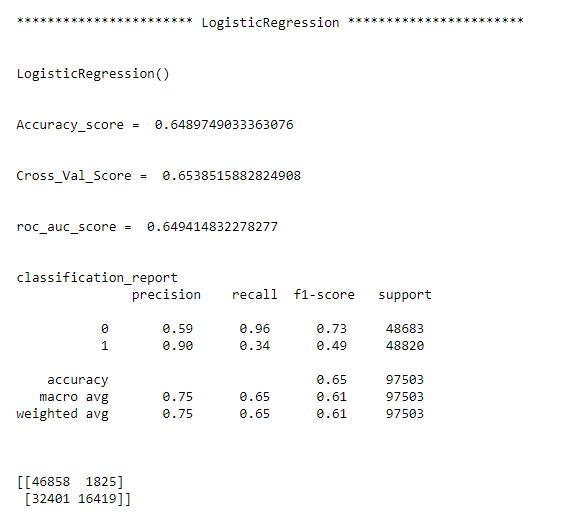
* Visualizations

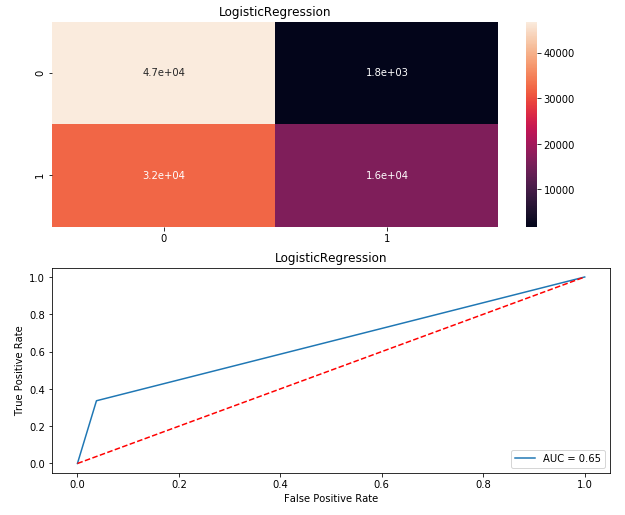


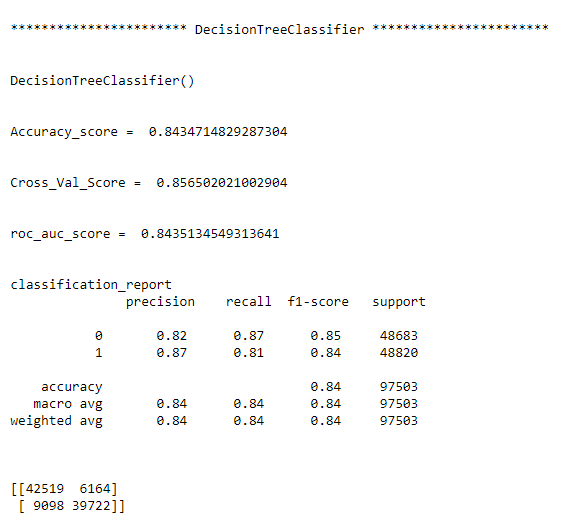


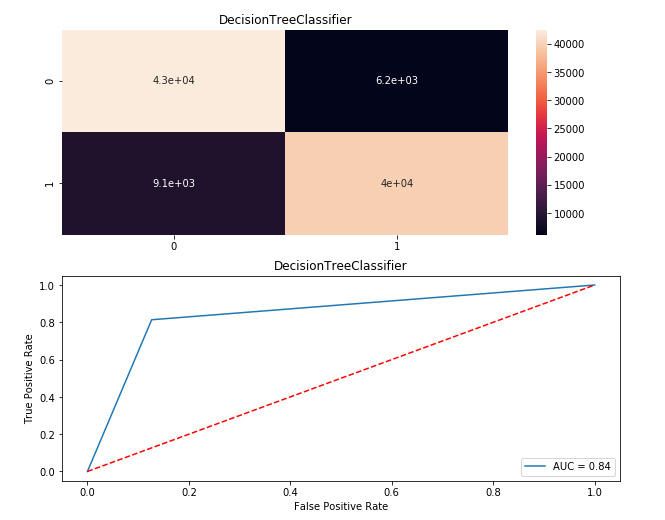












* Interpretation of the Results

From the above visualization and matrices found that the DecisionTree Classifier performed the best 84% AOC\_ROC\_SCORE, with precision recall score of 87, however the max score which we were able to achieve from dataset provided.

**CONCLUSION**

* Key Findings and Conclusions of the Study

From the whole evaluation we found that the MFIs have provided loan to the user who have no recharge or balance in their account which needs to be stopped as 30% defaulted user are from that type, and few high frequency loan takers and among users maintaining high balances are absorbed that 8% to 10% users are defaulted and some SMS altering notification before the deadlines can play a major role, in reducing the default rate.

* Limitations of this work and Scope for Future Work

Machine Learning Algorithms like SVC and LinearSVC took enormous amount of time to build the model.

One of the limitations to my approach in comparing the models is that as it run all the models on the same time, so if we want to make changes or implement any experiments, all models runs again unnecessarily, so need to define a function for evaluation metrices and pass each algorithm, so that we can manual tune and play with it more.

We could have experimented more with increasing the components of the PCA and could have tried opting RandomOverSampler instead of SMOTETomek, and we could have use log imputation over mean imputation, these might have affected the resulting matrices, but as hyperparameter tunning and SVC model building took most of the time.