

Micro-Credit-Loan Defaulter/Non Defaulter Model

Submitted by:

Neha Dixit

**ACKNOWLEDGMENT**

I would like to thanks to Flip Robo Technologies to give me a wonderful opportunity. This project is given by my SME Ms Swati Rustagi. I have referred below resources that helped and guided me in completion of this project as below:-

* <https://www.researchgate.net/publication/336800562_Credit_Card_Fraud_Detection_using_Machine_Learning_and_Data_Science>
* file:///C:/Users/Neha/Downloads/risks-09-00050-v2.pdf
* [www.towardsdatascience.com](http://www.towardsdatascience.com)
* https://medium.com/kitepython/handling-imbalanced-datasets-with-smote-in-python-a94090d031f0
* <https://www.google.com/search?q=micro+credit+loan+dataset+description+in+machine+learning&sxsrf=ALeKk03WCLzEw2jpqLe87IeIPApuqu4OJA%3A1622025412932&ei=xCSuYIWyOK6e4-EPx5G66A4&oq=micro+credit+loan+dataset+description+in++m&gs_lcp=Cgdnd3Mtd2l6EAMYADIFCCEQoAEyBQghEKABMgUIIRCgATIFCCEQoAE6BwgAEEcQsANQ9bgDWLrbA2DP5wNoAnACeACAAcUCiAGnEpIBBTItNi4ymAEAoAEBqgEHZ3dzLXdpesgBCMABAQ&sclient=gws-wiz>

**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

The project objective is to find out the defaulters (i.e., the users who don’t repay the loan within 5 days). Loan giving capacity will get decided based on below parameters-Daily amount spend & average main account balance in last 30 days, Frequency of recharge for data account & main account in 30/90 days, loan taken in last 90 days & payback time for last 30 days). Now, Using Different Mathematical and statistical tools. Many assumptions regarding the data are made and data Cleaning is done.

After the Data Cleaning part Model Training takes place in which different models like: KNN, Random Forest Classifier, Decision Tree Classifier Ada Boost Classifier, Gradient Boosting Classifier etc. models are used for the Training of the data.

* Motivation for the Problem Undertaken

In order to understand to whom loan to be given from lower income earning people and data from telecom industry clearly stats parameters to be taken into consideration to declare borrower as defaulter or not & amount limit also can be decide based on this.

If I talk about the poor families in remote areas with low income, and it is related to financial sectors, I believe that with growing technologies and Idea can make a difference, and I am really excited as there are so much in the financial market to explore and analyse with Data Science that makes the financial world more interesting.

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem

This problem is a classification problem. The dataset is in CSV format with 37 attributes (36 features and 1 target). The target variable is itself a statistical parameter. In given dataset “Label “column is our target. There are only two unique value in this column. The target variable is either 1 or 0 which means non defaulter and defaulter, respectively. For a loan amount of 5 payback amount should be 6, and for loan amount of 10 payback amount is 12. We must predict in terms of a probability for each loan transaction, whether the customer will be paying back the loaned amount within 5 days of insurance of loan.

* Data Sources and their formats

This Dataset is provided by Flip Robo Technologies CSV format. In this dataset, there are 209593 rows and 37 columns. The data description as given below:

|  |  |
| --- | --- |
| ***Column Name*** | ***Definition*** |
| 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 Indonasian Rupiah) |
| medianamnt\_ma\_rech90 | Median of amount of recharges done in main account over last 90 days at user level (in Indonasian Rupiah) |
| medianmarechprebal90 | Median of main account balance just before recharge in last 90 days at user level (in Indonasian Rupiah) |
| 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 |

# Check the data information

df.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 209593 entries, 0 to 209592

Data columns (total 37 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Unnamed: 0 209593 non-null int64

1 label 209593 non-null int64

2 msisdn 209593 non-null object

3 aon 209593 non-null float64

4 daily\_decr30 209593 non-null float64

5 daily\_decr90 209593 non-null float64

6 rental30 209593 non-null float64

7 rental90 209593 non-null float64

8 last\_rech\_date\_ma 209593 non-null float64

9 last\_rech\_date\_da 209593 non-null float64

10 last\_rech\_amt\_ma 209593 non-null int64

11 cnt\_ma\_rech30 209593 non-null int64

12 fr\_ma\_rech30 209593 non-null float64

13 sumamnt\_ma\_rech30 209593 non-null float64

14 medianamnt\_ma\_rech30 209593 non-null float64

15 medianmarechprebal30 209593 non-null float64

16 cnt\_ma\_rech90 209593 non-null int64

17 fr\_ma\_rech90 209593 non-null int64

18 sumamnt\_ma\_rech90 209593 non-null int64

19 medianamnt\_ma\_rech90 209593 non-null float64

20 medianmarechprebal90 209593 non-null float64

21 cnt\_da\_rech30 209593 non-null float64

22 fr\_da\_rech30 209593 non-null float64

23 cnt\_da\_rech90 209593 non-null int64

24 fr\_da\_rech90 209593 non-null int64

25 cnt\_loans30 209593 non-null int64

26 amnt\_loans30 209593 non-null int64

27 maxamnt\_loans30 209593 non-null float64

28 medianamnt\_loans30 209593 non-null float64

29 cnt\_loans90 209593 non-null float64

30 amnt\_loans90 209593 non-null int64

31 maxamnt\_loans90 209593 non-null int64

32 medianamnt\_loans90 209593 non-null float64

33 payback30 209593 non-null float64

34 payback90 209593 non-null float64

35 pcircle 209593 non-null object

36 pdate 209593 non-null object

dtypes: float64(21), int64(13), object(3)

memory usage: 59.2+ MB

Check the data type:

df.dtypes

Unnamed: 0 int64

label int64

msisdn object

aon float64

daily\_decr30 float64

daily\_decr90 float64

rental30 float64

rental90 float64

last\_rech\_date\_ma float64

last\_rech\_date\_da float64

last\_rech\_amt\_ma int64

cnt\_ma\_rech30 int64

fr\_ma\_rech30 float64

sumamnt\_ma\_rech30 float64

medianamnt\_ma\_rech30 float64

medianmarechprebal30 float64

cnt\_ma\_rech90 int64

fr\_ma\_rech90 int64

sumamnt\_ma\_rech90 int64

medianamnt\_ma\_rech90 float64

medianmarechprebal90 float64

cnt\_da\_rech30 float64

fr\_da\_rech30 float64

cnt\_da\_rech90 int64

fr\_da\_rech90 int64

cnt\_loans30 int64

amnt\_loans30 int64

maxamnt\_loans30 float64

medianamnt\_loans30 float64

cnt\_loans90 float64

amnt\_loans90 int64

maxamnt\_loans90 int64

medianamnt\_loans90 float64

payback30 float64

payback90 float64

pcircle object

pdate object

dtype: object

* Data Preprocessing Done
  + I checked the correlation of the independent and dependent features and f**rom the correlation table it is also clear that the features with time span of 30 and 90 days almost have the same correlation thus we can drop one for the same information.**
  + Some columns can’t have any negative value, so those columns were treated accordingly.
  + Outliers are treated manually for the features giving some important information, and then the threshold values were set to make the data free from outliers.
  + Applied LabelEncoder for object type data.
  + Applied various machine learning model and compared it.

Checking the null values

df.isnull().sum()

Unnamed: 0 0

label 0

msisdn 0

aon 0

daily\_decr30 0

daily\_decr90 0

rental30 0

rental90 0

last\_rech\_date\_ma 0

last\_rech\_date\_da 0

last\_rech\_amt\_ma 0

cnt\_ma\_rech30 0

fr\_ma\_rech30 0

sumamnt\_ma\_rech30 0

medianamnt\_ma\_rech30 0

medianmarechprebal30 0

cnt\_ma\_rech90 0

fr\_ma\_rech90 0

sumamnt\_ma\_rech90 0

medianamnt\_ma\_rech90 0

medianmarechprebal90 0

cnt\_da\_rech30 0

fr\_da\_rech30 0

cnt\_da\_rech90 0

fr\_da\_rech90 0

cnt\_loans30 0

amnt\_loans30 0

maxamnt\_loans30 0

medianamnt\_loans30 0

cnt\_loans90 0

amnt\_loans90 0

maxamnt\_loans90 0

medianamnt\_loans90 0

payback30 0

payback90 0

pcircle 0

pdate 0

dtype: int64

And we can find here that there is no null values in dataset

#finding unique number, max value, min value of each column in same time

for column in df.columns:

print(f"========Column: {column}========")

print(f"Number of unique values: {df[column].nunique()}")

print(f"Max: {df[column].max()}")

print(f"Min: {df[column].min()}")

========Column: label========

Number of unique values: 2

Max: 1

Min: 0

========Column: aon========

Number of unique values: 4507

Max: 999860.755167902

Min: -48.0

========Column: daily\_decr30========

Number of unique values: 147026

Max: 265926.0

Min: -93.0126666666667

========Column: daily\_decr90========

Number of unique values: 158670

Max: 320630.0

Min: -93.0126666666667

========Column: rental30========

Number of unique values: 132148

Max: 198926.11

Min: -23737.14

========Column: rental90========

Number of unique values: 141033

Max: 200148.11

Min: -24720.58

========Column: last\_rech\_date\_ma========

Number of unique values: 1186

Max: 998650.3777327021

Min: -29.0

========Column: last\_rech\_date\_da========

Number of unique values: 1174

Max: 999171.809409745

Min: -29.0

========Column: last\_rech\_amt\_ma========

Number of unique values: 70

Max: 55000

Min: 0

========Column: cnt\_ma\_rech30========

Number of unique values: 71

Max: 203

Min: 0

========Column: fr\_ma\_rech30========

Number of unique values: 1083

Max: 999606.368131936

Min: 0.0

========Column: sumamnt\_ma\_rech30========

Number of unique values: 15141

Max: 810096.0

Min: 0.0

========Column: medianamnt\_ma\_rech30========

Number of unique values: 510

Max: 55000.0

Min: 0.0

========Column: medianmarechprebal30========

Number of unique values: 30428

Max: 999479.4193189591

Min: -200.0

========Column: cnt\_ma\_rech90========

Number of unique values: 110

Max: 336

Min: 0

========Column: fr\_ma\_rech90========

Number of unique values: 89

Max: 88

Min: 0

========Column: sumamnt\_ma\_rech90========

Number of unique values: 31771

Max: 953036

Min: 0

========Column: medianamnt\_ma\_rech90========

Number of unique values: 608

Max: 55000.0

Min: 0.0

========Column: medianmarechprebal90========

Number of unique values: 29785

Max: 41456.5

Min: -200.0

========Column: cnt\_da\_rech30========

Number of unique values: 1066

Max: 99914.4414195325

Min: 0.0

========Column: fr\_da\_rech30========

Number of unique values: 1072

Max: 999809.240107425

Min: 0.0

========Column: cnt\_da\_rech90========

Number of unique values: 27

Max: 38

Min: 0

========Column: fr\_da\_rech90========

Number of unique values: 46

Max: 64

Min: 0

========Column: cnt\_loans30========

Number of unique values: 40

Max: 50

Min: 0

========Column: amnt\_loans30========

Number of unique values: 48

Max: 306

Min: 0

========Column: maxamnt\_loans30========

Number of unique values: 1050

Max: 99864.56086393449

Min: 0.0

========Column: medianamnt\_loans30========

Number of unique values: 6

Max: 3.0

Min: 0.0

========Column: cnt\_loans90========

Number of unique values: 1110

Max: 4997.51794431359

Min: 0.0

========Column: amnt\_loans90========

Number of unique values: 69

Max: 438

Min: 0

========Column: maxamnt\_loans90========

Number of unique values: 3

Max: 12

Min: 0

========Column: medianamnt\_loans90========

Number of unique values: 6

Max: 3.0

Min: 0.0

========Column: payback30========

Number of unique values: 1363

Max: 171.5

Min: 0.0

========Column: payback90========

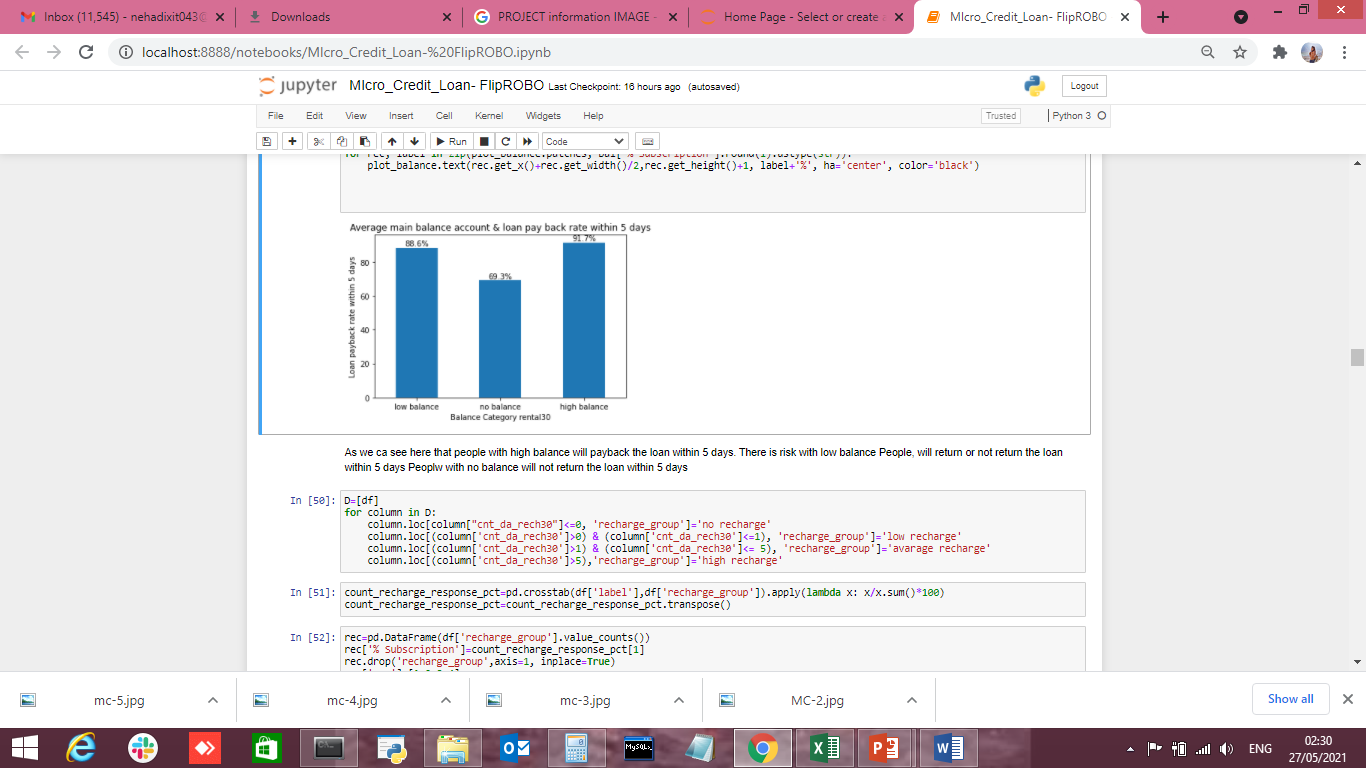
Number of unique values: 2381

Max: 171.5

Min: 0.0

* Data Inputs- Logic- Output Relationships

**I) Average main balance account vs loan pay back rate within 7 days**



1- We can differentiate the customers with main balance levels are paying back the loan within 5 days.

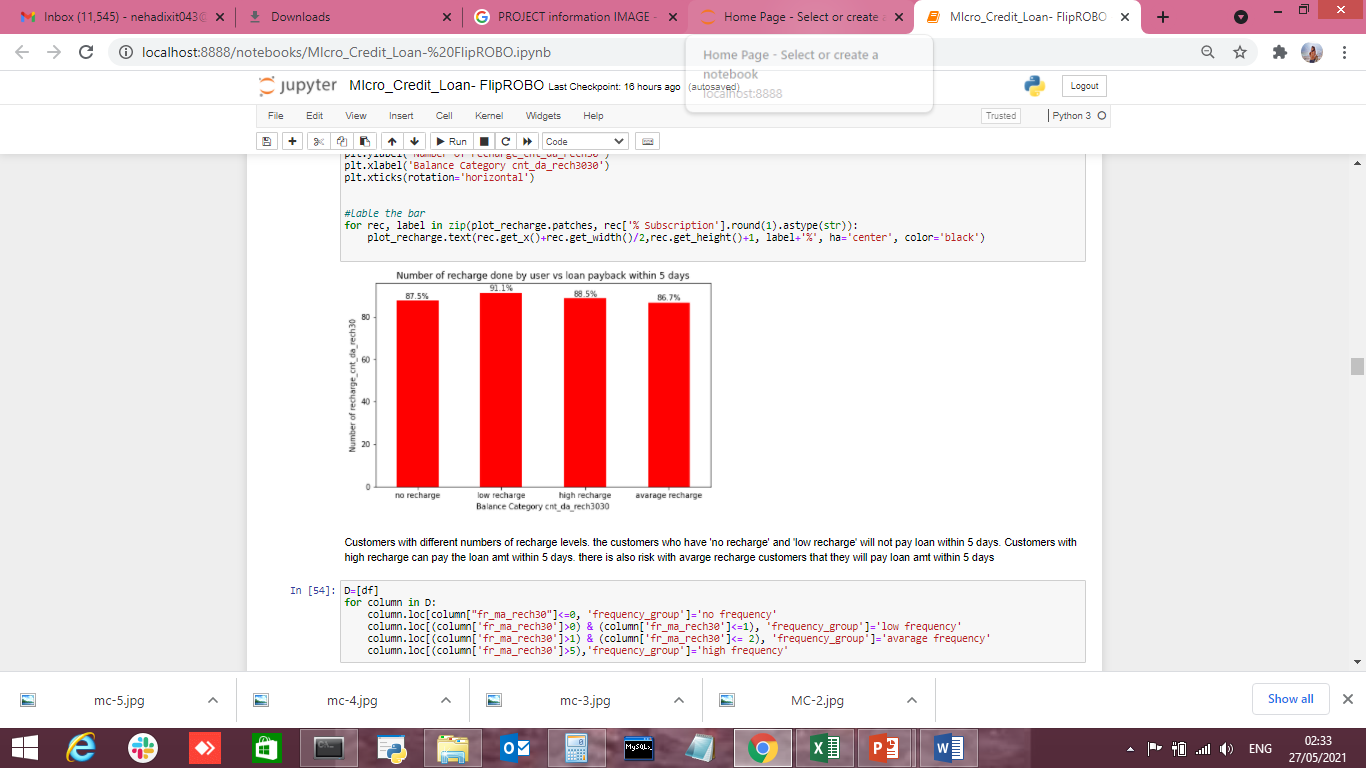
2- The high balance level people are with 92% rate i.e they are paying loan within 5 days.

3- The average and low balance people ,it is observed that around 10%-15% of people are not paying the loan within 5 days.

4- It is clearly shown that around 30% of low balance level people are not paying back the loan with in stipulated 5 days of time.

5- The 30% of people with no balance or negative balance people are creating a major loss to the company without paying back the loan within 5 days of time.

**II) Number of recharged in last 30 days vs loan pay back rate within 7 days.**

****

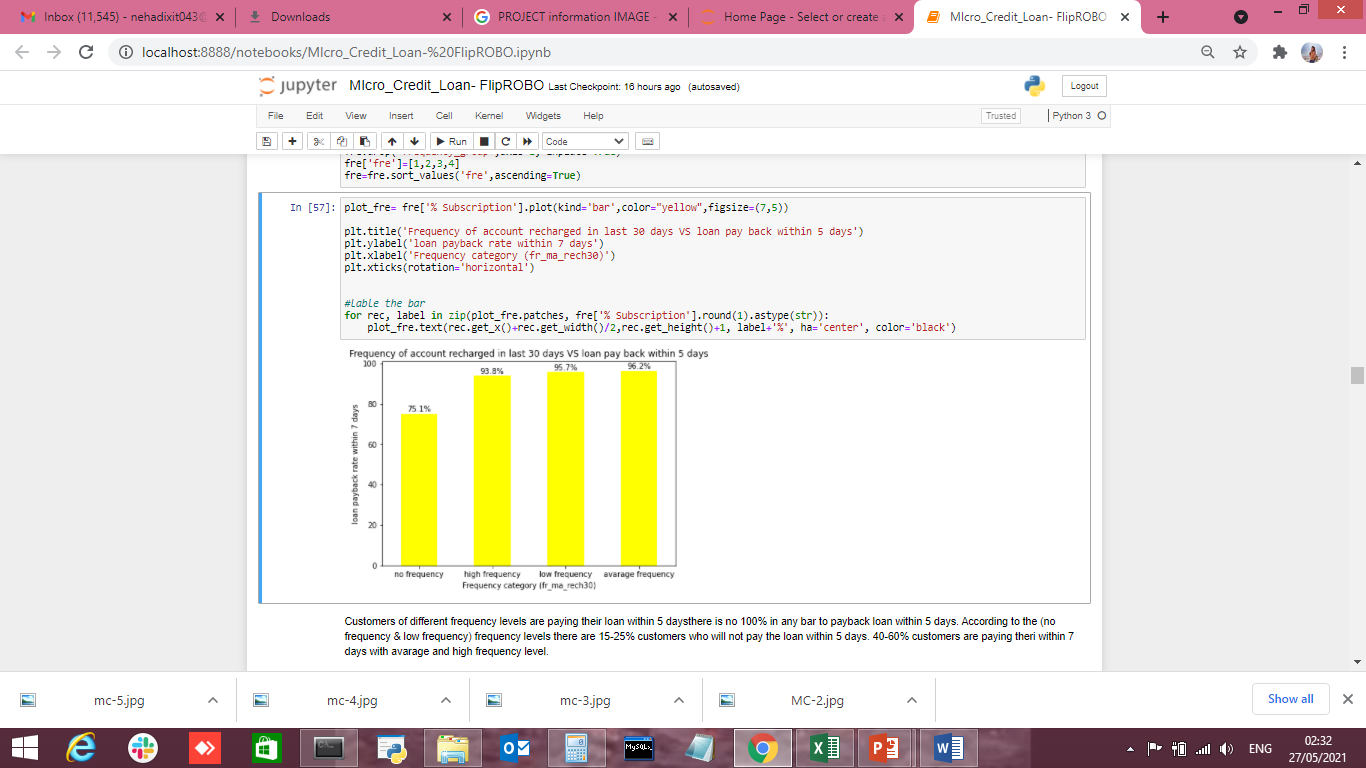
1- Customers with different recharge levels (main account recharge) are paying back the loan within 5 days.

2- Coming to the average and low & medium frequency people it is observed that around 5%-6% of people are not paying the loan within 5 days.

3- Coming to low frequency level people, it is observed that around 35% of people are not paying back the loan with in stipulated 5 days of time.

4- The 25% people who are not getting their main account recharge for 30 days creating a major loss to the company without paying back the loan within 5 days of time.

**III) Number of frequency account recharge in last 30 days vs loan pay back rate within 7 days.**

****

1- Customers of different frequency levels are paying their loan within 7 days.

2-there is no 100% in any bar to payback loan within 7 days.

3-According to the (no frequency & low frequency) frequency levels there are 15-25% customers who will not pay the loan within 7 days.

4-40-60% customers are paying their loan within 7 days with avarage and high frequency level.

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

I have not consider any pre-assumption ,project performance from beginning to end is based on data facts only.

* Hardware and Software Requirements and Tools Used

**Hardware Requirement-**Laptop with below configurations-

Windows Edition-Windows 8.1 Pro

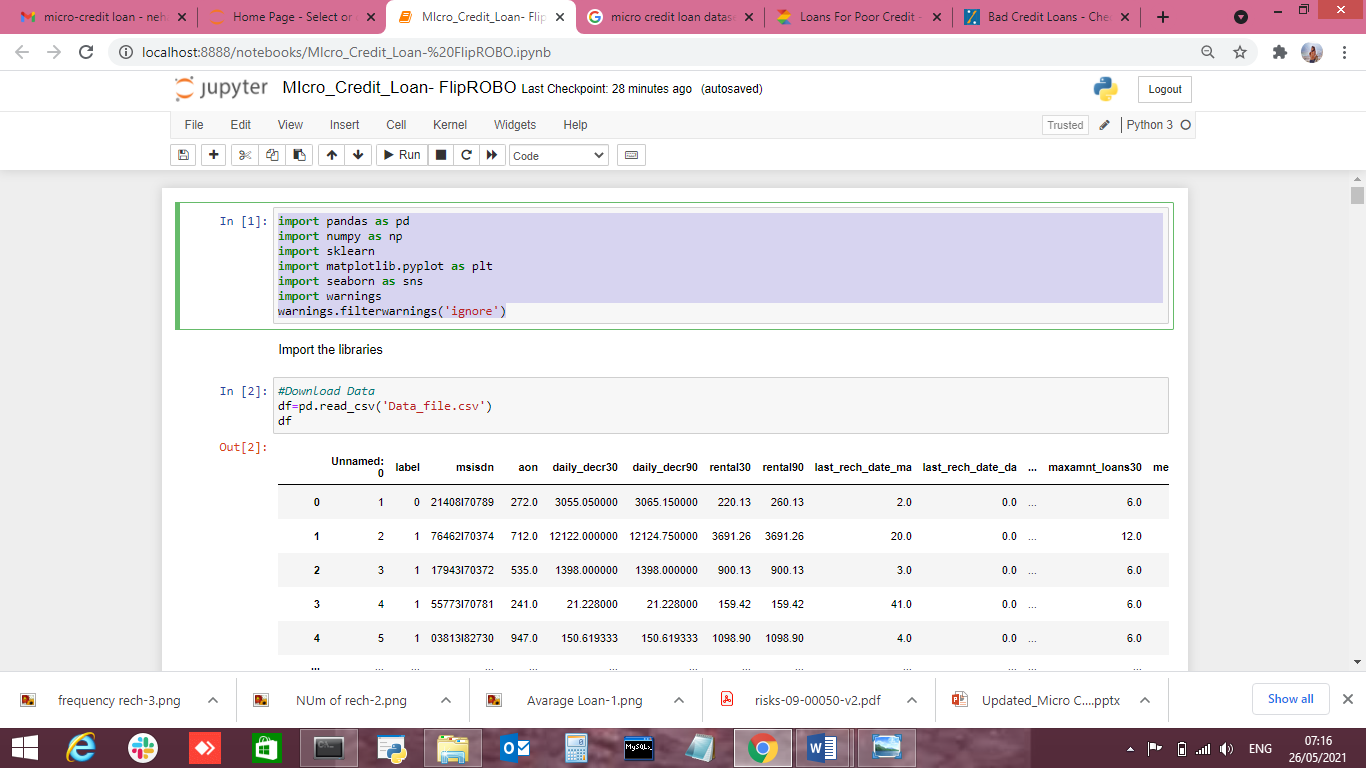
Processor-Intel(R) Core(TM) i3-5005U CPU @ 2.00GHz 2.00GHz

Installed memory RAM- 4 GB

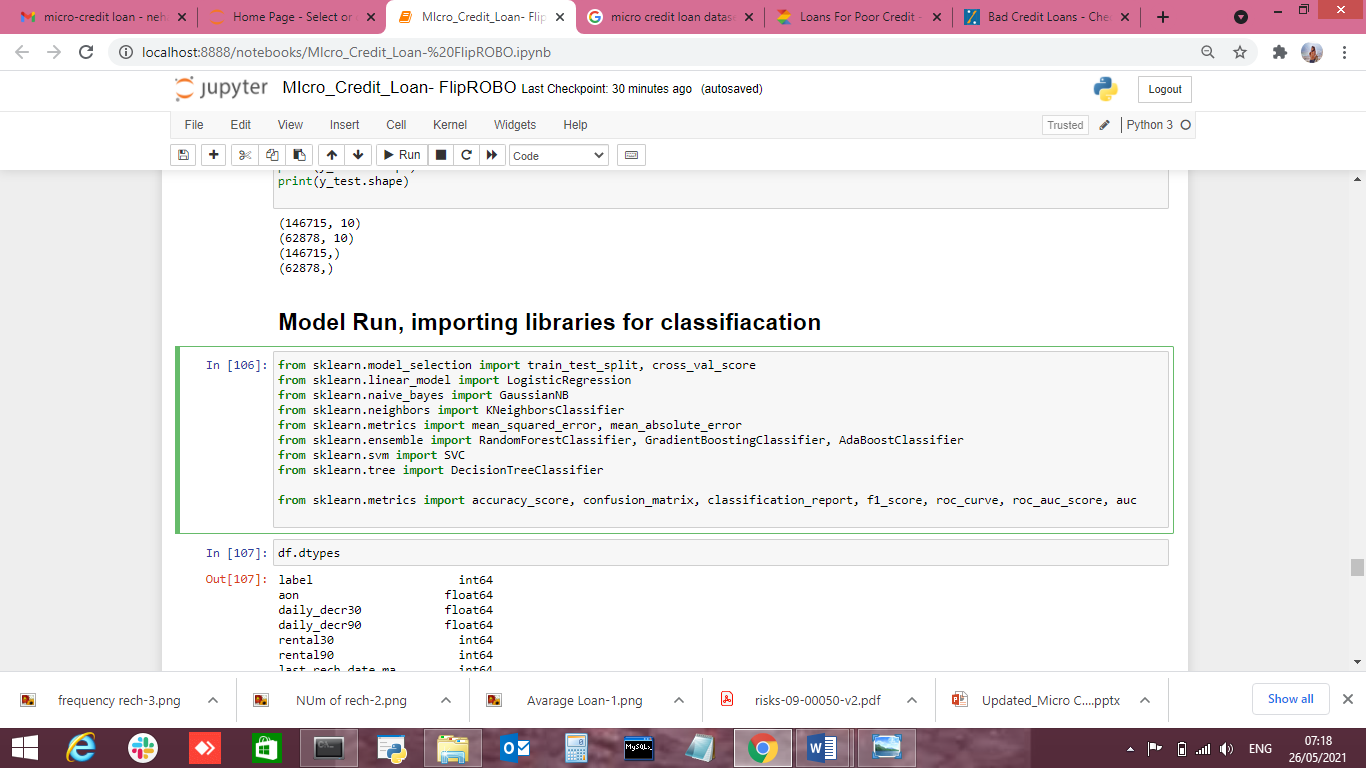
System Type-64 bit OS, x64 based processor

**Software Requirement-** Anaconda 4.9.2 , Python 3.8.5, Jupiter Notebook.

Libraries used:-



Libraries for Model Training



**Model/s Development and Evaluation**

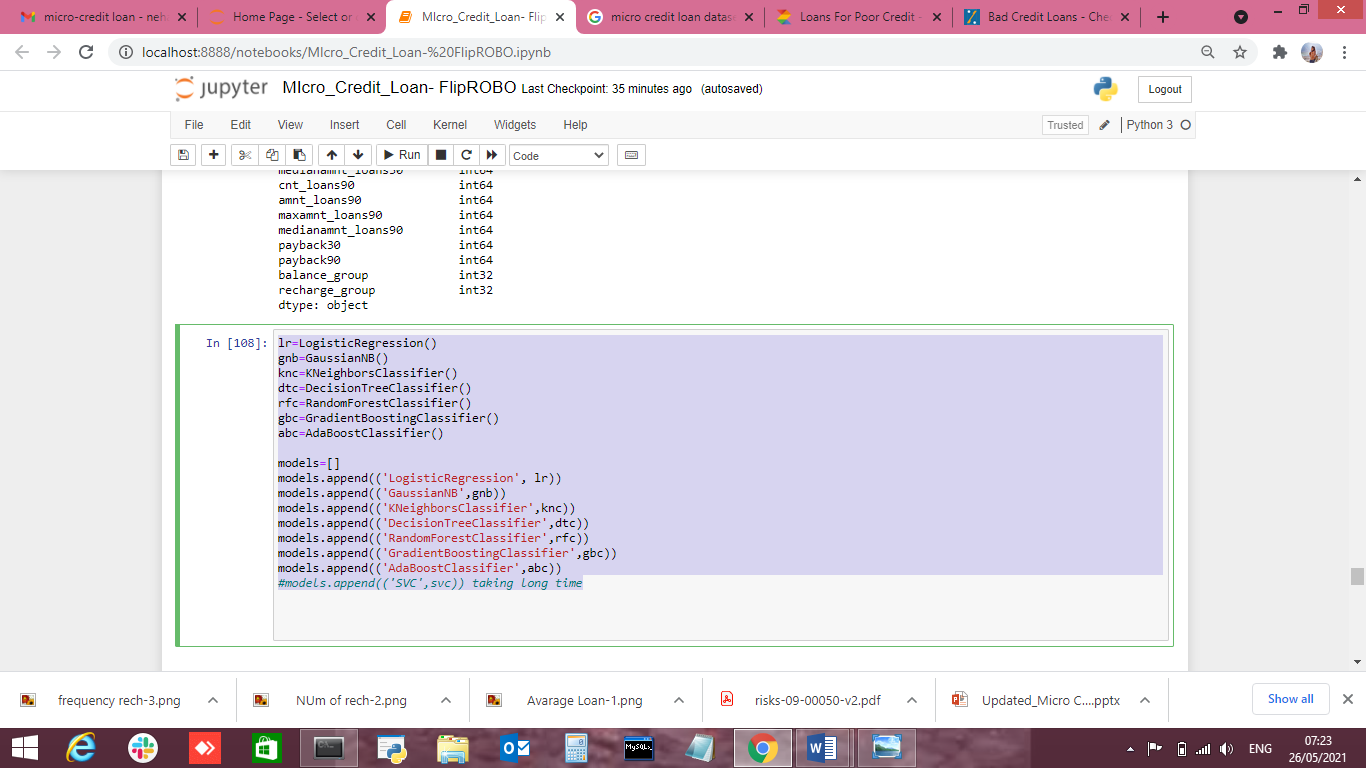
* Identification of possible problem-solving approaches (methods)
* **Analytical Approach –**Based on type of data by performing EDA I have decided which model to be used for this data.
* **Statistical Approach –** Data should be in scaled manner, it should not be distorted, for that all values using mean method due to continuous data numbers.
* Testing of Identified Approaches (Algorithms)

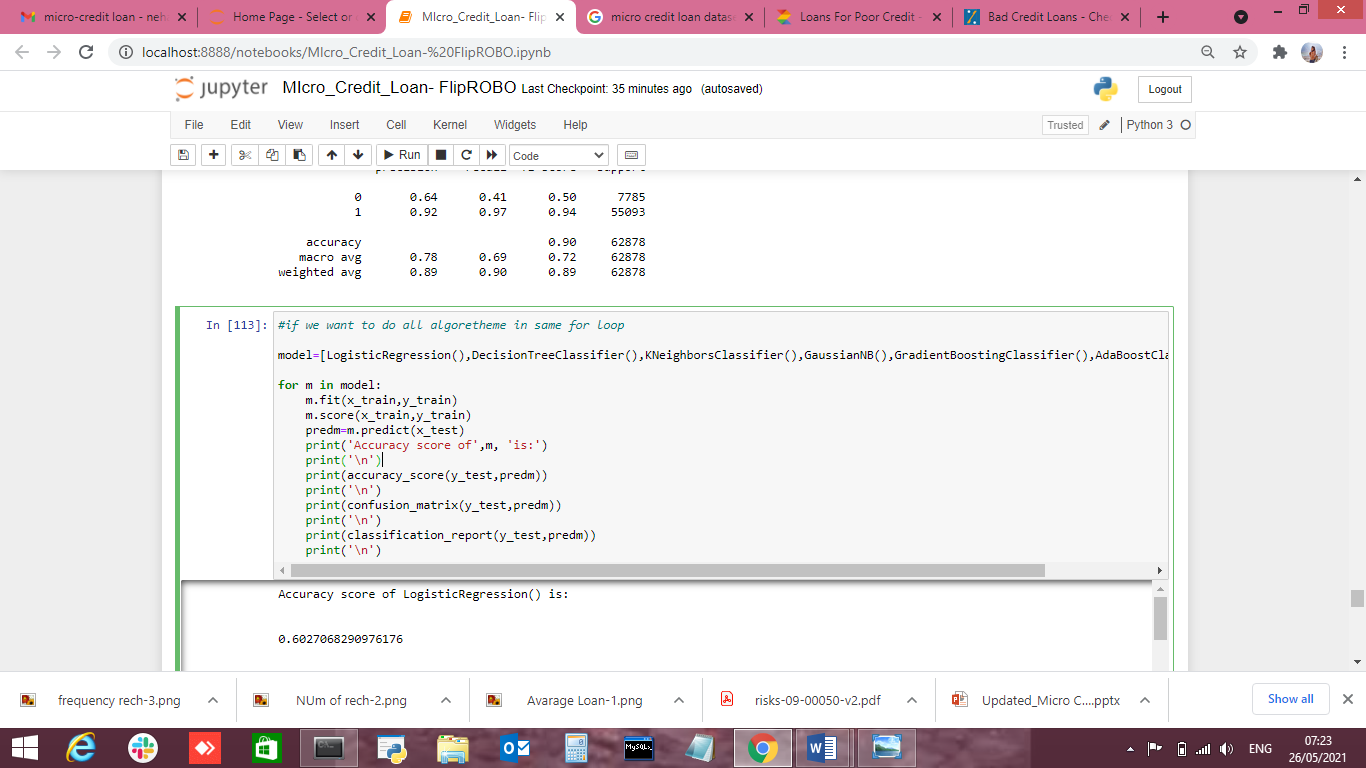
Below are classification algorithms used for the training and testing this dataset.

* lr=LogisticRegression()
* gnb=GaussianNB()
* knc=KNeighborsClassifier()
* dtc=DecisionTreeClassifier()
* rfc=RandomForestClassifier()
* gbc=GradientBoostingClassifier()
* abc=AdaBoostClassifier()
* models=[]
* models.append(('LogisticRegression', lr))
* models.append(('GaussianNB',gnb))
* models.append(('KNeighborsClassifier',knc))
* models.append(('DecisionTreeClassifier',dtc))
* models.append(('RandomForestClassifier',rfc))
* models.append(('GradientBoostingClassifier',gbc))
* models.append(('AdaBoostClassifier',abc))
* #models.append(('SVC',svc)) taking long time

Run and Evaluate selected models

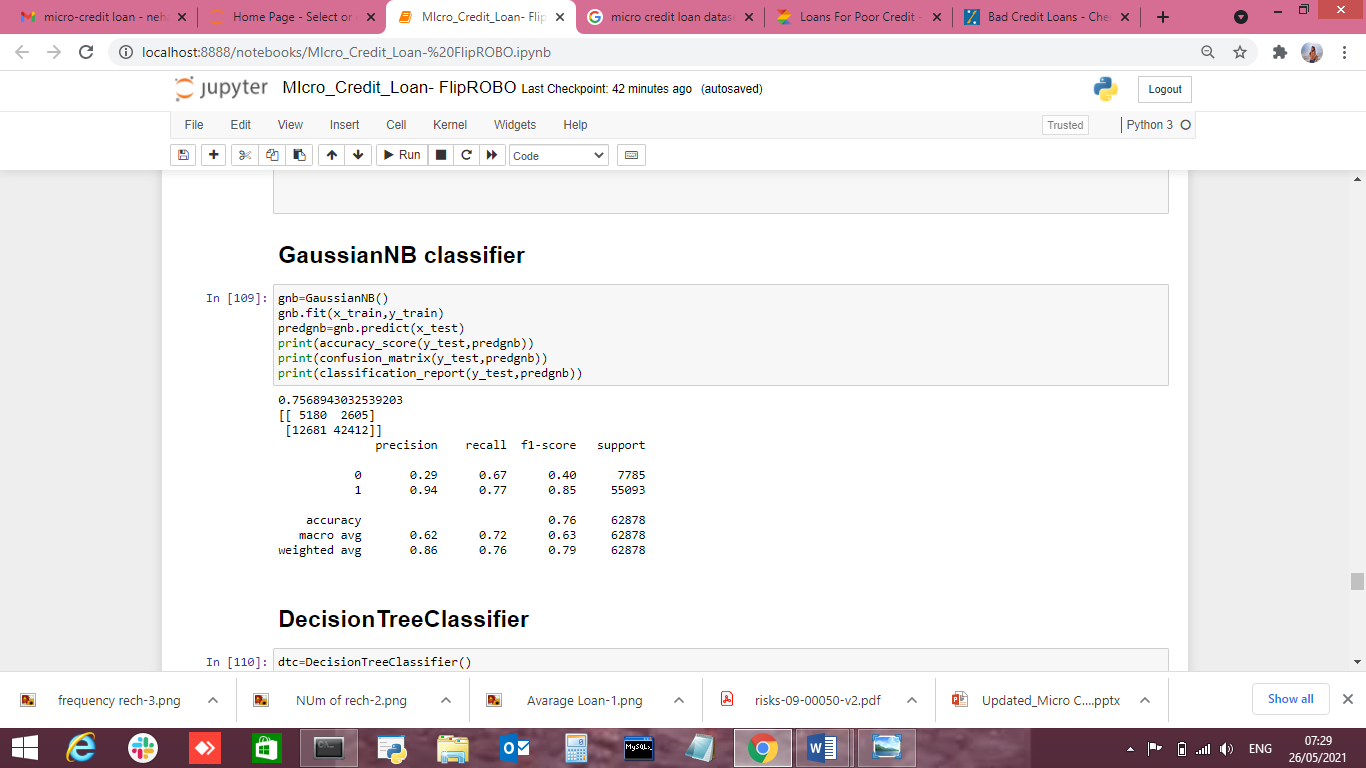
Pls find below matrix & their results also.

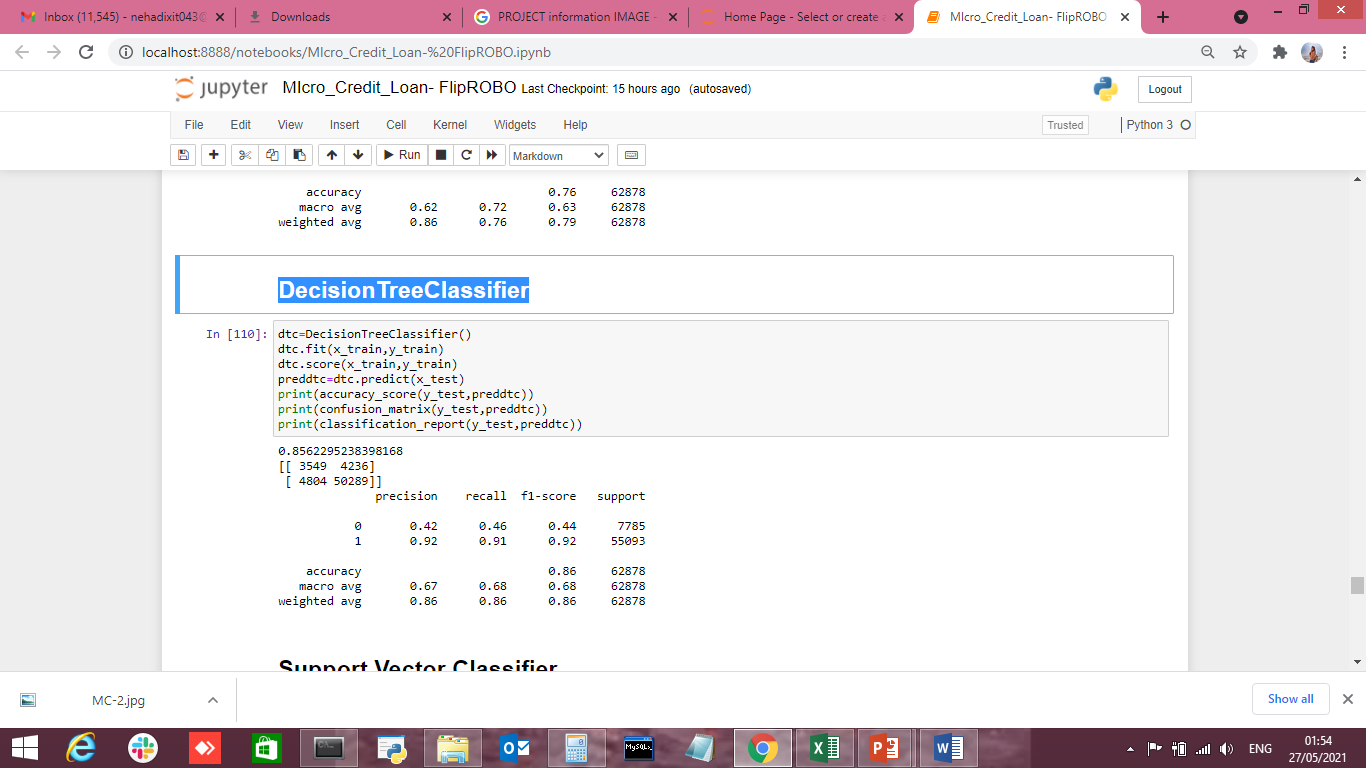


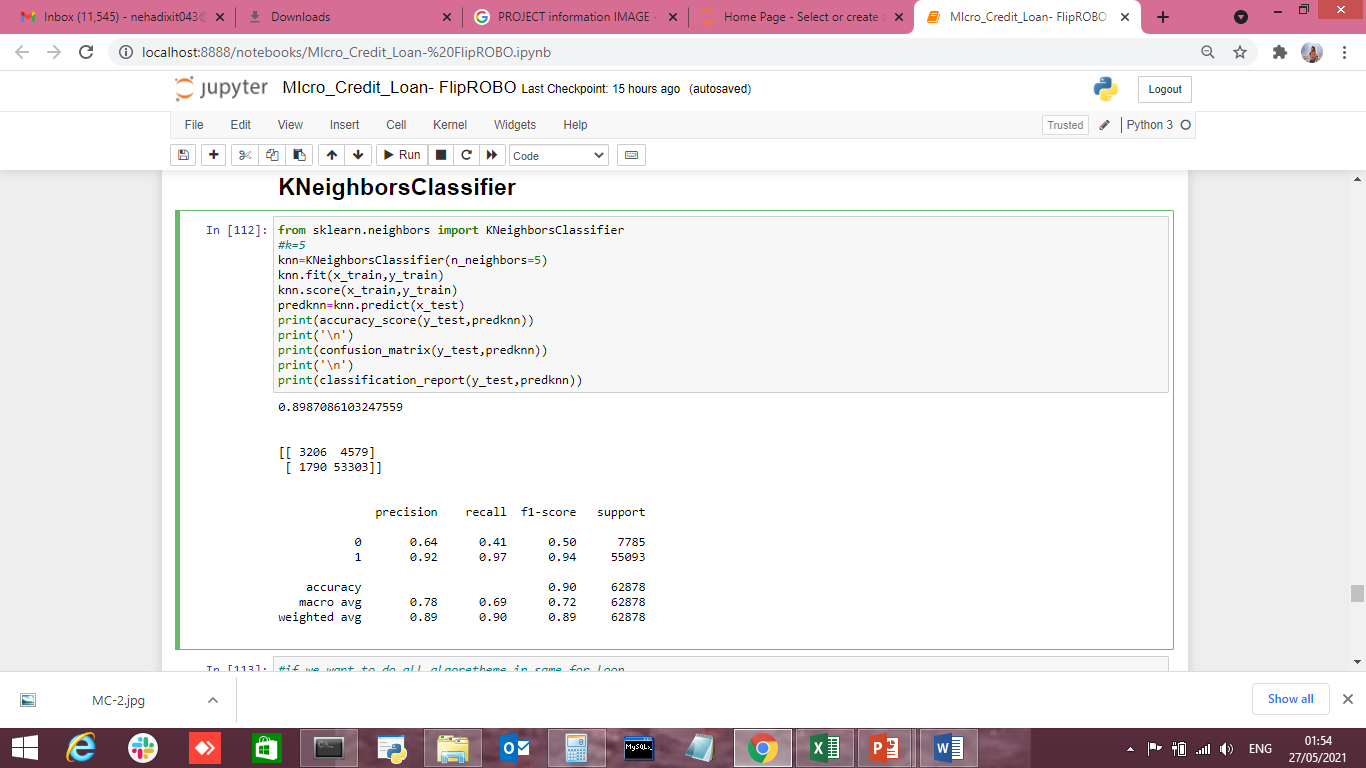


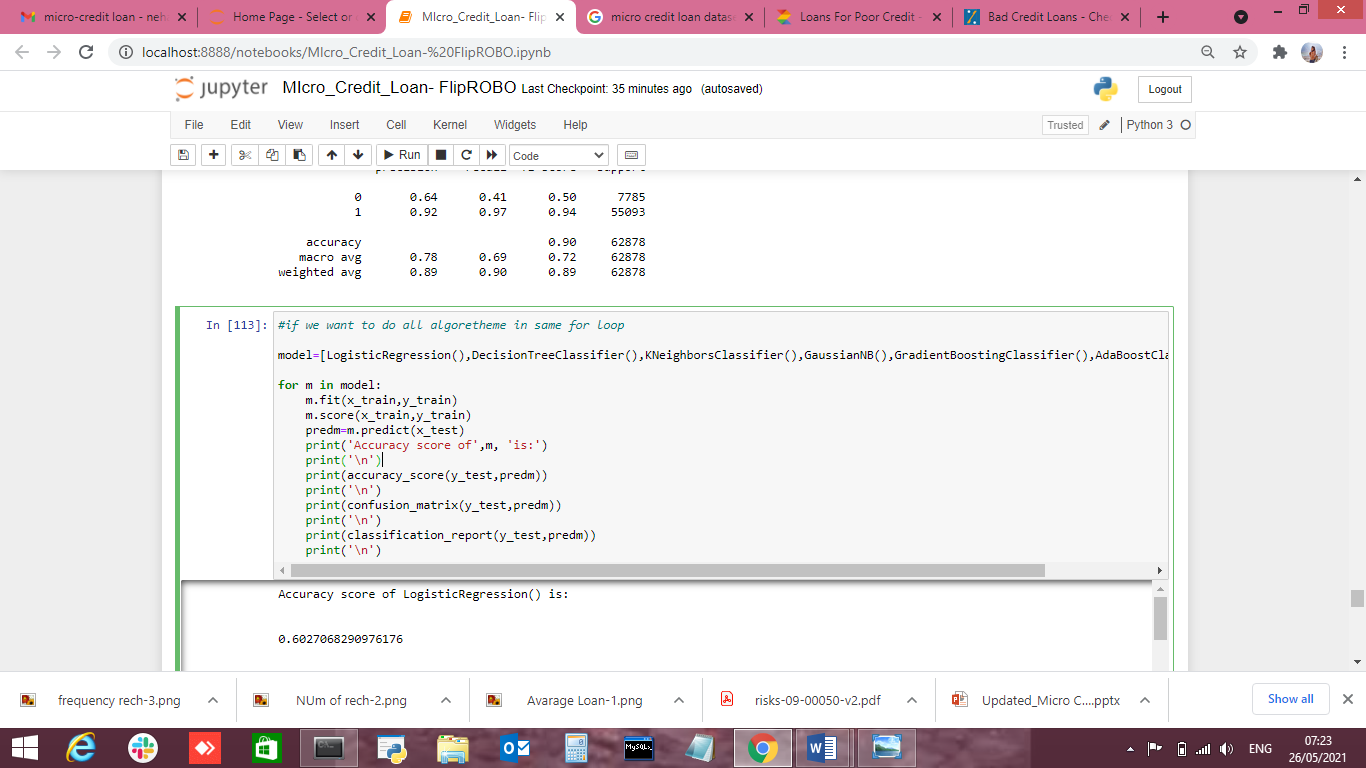
* Accuracy score of LogisticRegression() is:
* 0.6027068290976176
* [[ 6606 1179]
* [23802 31291]]
* precision recall f1-score support
* 0 0.22 0.85 0.35 7785
* 1 0.96 0.57 0.71 55093
* accuracy 0.60 62878
* macro avg 0.59 0.71 0.53 62878
* weighted avg 0.87 0.60 0.67 62878
* Accuracy score of DecisionTreeClassifier() is:
* 0.855879639937657
* [[ 3544 4241]
* [ 4821 50272]]
* precision recall f1-score support
* 0 0.42 0.46 0.44 7785
* 1 0.92 0.91 0.92 55093
* accuracy 0.86 62878
* macro avg 0.67 0.68 0.68 62878
* weighted avg 0.86 0.86 0.86 62878
* Accuracy score of KNeighborsClassifier() is:
* 0.8987404179522249
* [[ 3207 4578]
* [ 1789 53304]]
* precision recall f1-score support
* 0 0.64 0.41 0.50 7785
* 1 0.92 0.97 0.94 55093
* accuracy 0.90 62878
* macro avg 0.78 0.69 0.72 62878
* weighted avg 0.89 0.90 0.89 62878
* Accuracy score of GaussianNB() is:
* 0.7568943032539203
* [[ 5180 2605]
* [12681 42412]]
* precision recall f1-score support
* 0 0.29 0.67 0.40 7785
* 1 0.94 0.77 0.85 55093
* accuracy 0.76 62878
* macro avg 0.62 0.72 0.63 62878
* weighted avg 0.86 0.76 0.79 62878
* Accuracy score of GradientBoostingClassifier() is:
* 0.9036228887687268
* [[ 2455 5330]
* [ 730 54363]]
* precision recall f1-score support
* 0 0.77 0.32 0.45 7785
* 1 0.91 0.99 0.95 55093
* accuracy 0.90 62878
* macro avg 0.84 0.65 0.70 62878
* weighted avg 0.89 0.90 0.89 62878
* Accuracy score of AdaBoostClassifier() is:
* 0.8955914628327873
* [[ 1981 5804]
* [ 761 54332]]
* precision recall f1-score support
* 0 0.72 0.25 0.38 7785
* 1 0.90 0.99 0.94 55093
* accuracy 0.90 62878
* macro avg 0.81 0.62 0.66 62878
* weighted avg 0.88 0.90 0.87 62878
* Key Metrics for success in solving problem under consideration

Key Metrices used were the Accuracy Score and AUC & ROC Curve as this was binary classification problem and we focus more on AUC & 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.









Accuracy score of LogisticRegression() is:

0.6027068290976176

[[ 6606 1179]

[23802 31291]]

precision recall f1-score support

0 0.22 0.85 0.35 7785

1 0.96 0.57 0.71 55093

accuracy 0.60 62878

macro avg 0.59 0.71 0.53 62878

weighted avg 0.87 0.60 0.67 62878

Accuracy score of DecisionTreeClassifier() is:

0.855879639937657

[[ 3544 4241]

[ 4821 50272]]

precision recall f1-score support

0 0.42 0.46 0.44 7785

1 0.92 0.91 0.92 55093

accuracy 0.86 62878

macro avg 0.67 0.68 0.68 62878

weighted avg 0.86 0.86 0.86 62878

Accuracy score of KNeighborsClassifier() is:

0.8987404179522249

[[ 3207 4578]

[ 1789 53304]]

precision recall f1-score support

0 0.64 0.41 0.50 7785

1 0.92 0.97 0.94 55093

accuracy 0.90 62878

macro avg 0.78 0.69 0.72 62878

weighted avg 0.89 0.90 0.89 62878

Accuracy score of GaussianNB() is:

0.7568943032539203

[[ 5180 2605]

[12681 42412]]

precision recall f1-score support

0 0.29 0.67 0.40 7785

1 0.94 0.77 0.85 55093

accuracy 0.76 62878

macro avg 0.62 0.72 0.63 62878

weighted avg 0.86 0.76 0.79 62878

Accuracy score of GradientBoostingClassifier() is:

0.9036228887687268

[[ 2455 5330]

[ 730 54363]]

precision recall f1-score support

0 0.77 0.32 0.45 7785

1 0.91 0.99 0.95 55093

accuracy 0.90 62878

macro avg 0.84 0.65 0.70 62878

weighted avg 0.89 0.90 0.89 62878

Accuracy score of AdaBoostClassifier() is:

0.8955914628327873

[[ 1981 5804]

[ 761 54332]]

precision recall f1-score support

0 0.72 0.25 0.38 7785

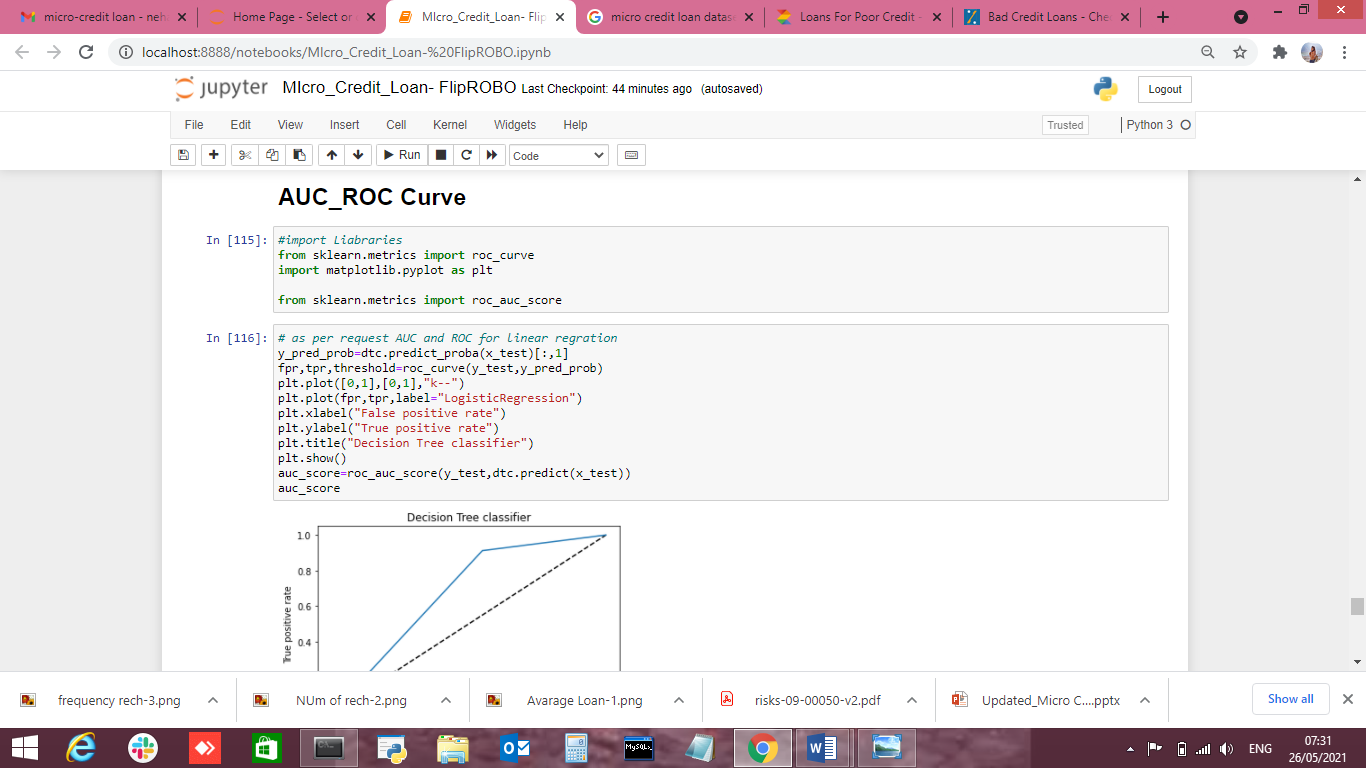
1 0.90 0.99 0.94 55093

accuracy 0.90 62878

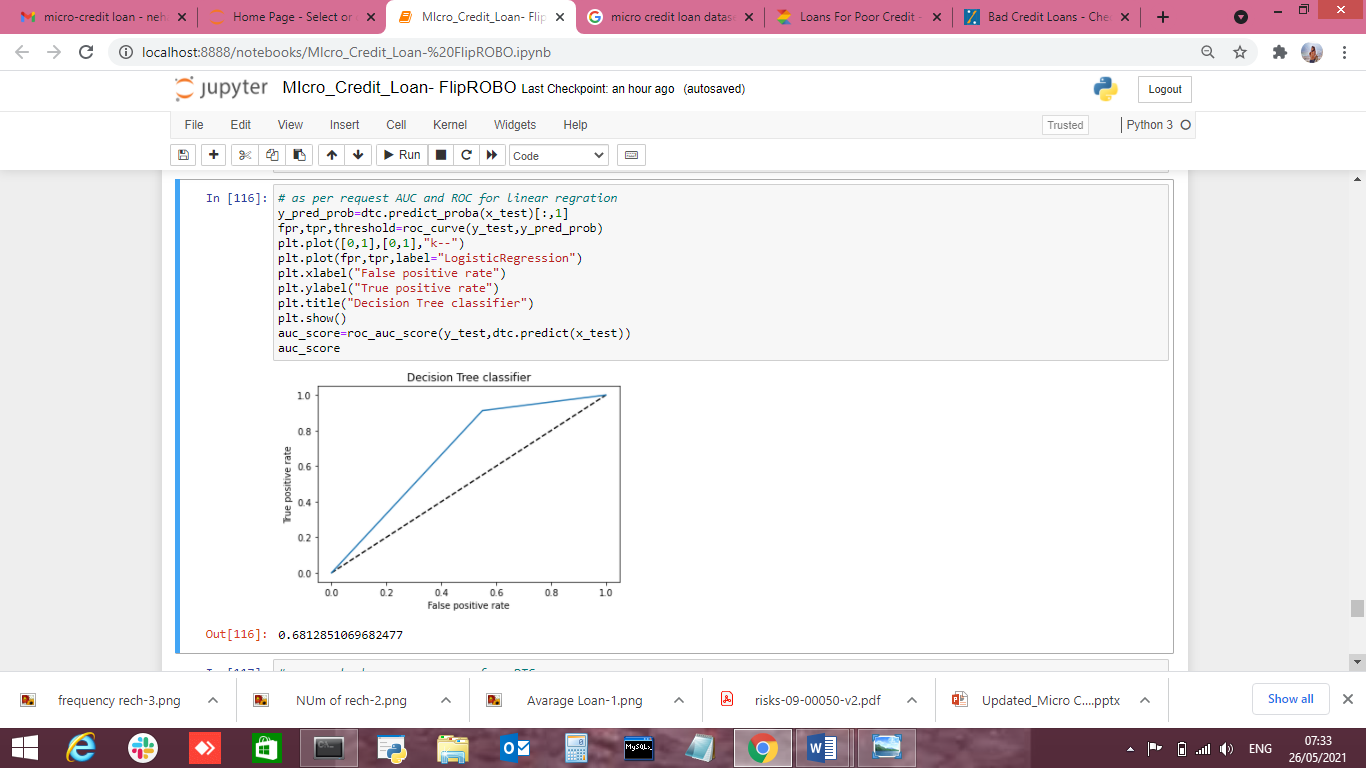
macro avg 0.81 0.62 0.66 62878

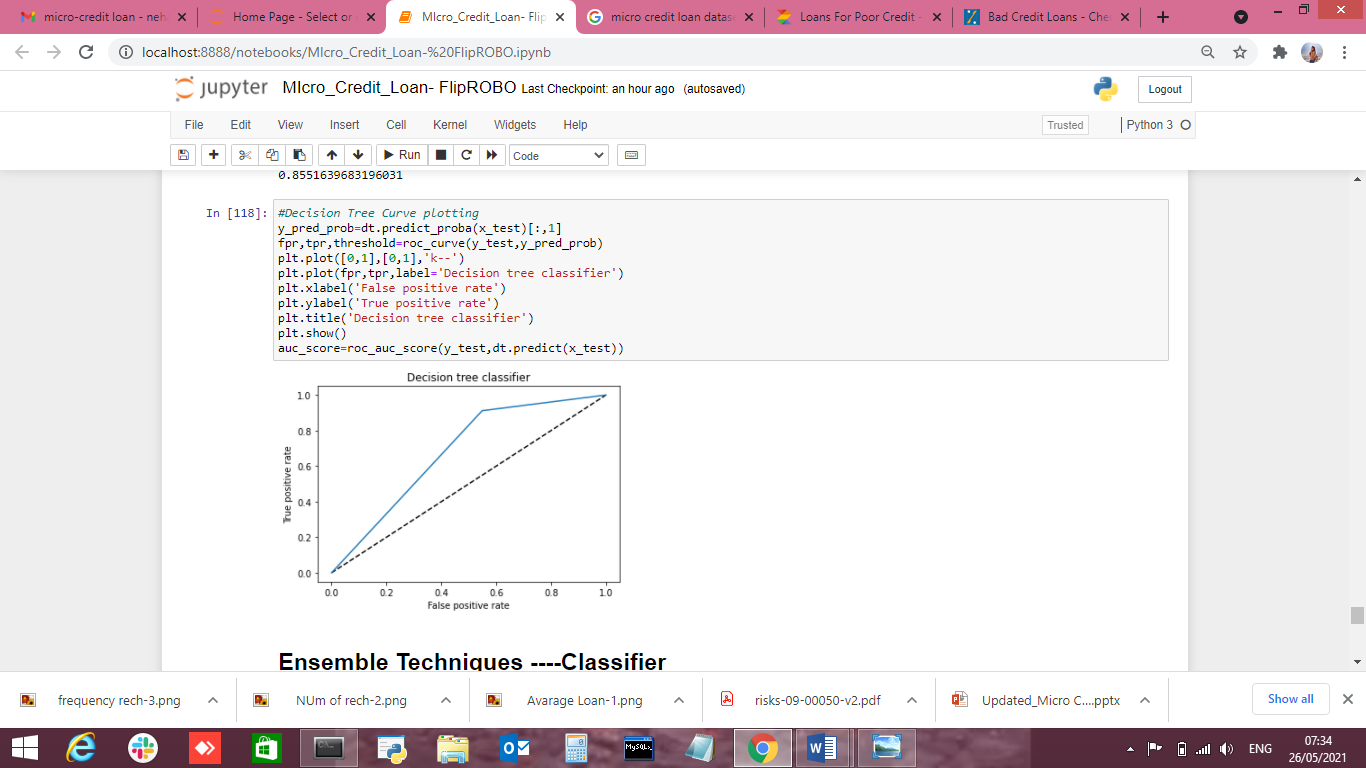
weighted avg 0.88 0.90 0.87 62878

AUC\_ROC Cirve



* Visualizations





* Interpretation of the Results

Data Pre-processing done by performing EDA (Exploratory Data Analysis), checking for best accuracy score.

We will save Gradient Boosting Classifier (GBC) as it's having 90% accuracy score.

**CONCLUSION**

* Key Findings and Conclusions of the Study

Conclusion-Loan giving capacity based on below parameters-Daily amount spend & average main account balance in last 30 days, Frequency of recharge for data account & main account in 30/90 days, loan taken in last 90 days & payback time for last 30 days.

Multi-Financial Institutions need to be taken into consideration for above parameters due to correlation & it is giving best score also.

* Learning Outcomes of the Study in respect of Data Science

This dataset is categorical in nature ,we can verify data by using read method & get stats related information for each column using describe method.

Visualizations and Data Cleaning part was very crucial as without the cleaning we were not able to judge the data effectively and won’t be able to remove the outliers thus adding into the errors.

As its categorical data, classification model best suits for this.

Check the prediction score using accuracy score & get AUC\_ROC Curve score.

Train data using classification models to get the best score & finalise best score giver model for this dataset.

Get the test score for same model.

Save file using joblib library.

* Limitations of this work and Scope for Future Work

Visualizations helped a lot in finding out those outliers values and helped in finding out the features having direct relation between the feature and the label.

Its always good to to have complete data while performing model but 7-8 % of data can be excluded based on performance impact.