

Micro-Credit Defaulter Model

Submitted by:

Naina Joshi

**ACKNOWLEDGMENT**

I would like to thanks to Flip Robo Technologies to give me a wonderful opportunity. This project is given by my SME Mr. Shubham Yadav. 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>
* [www.towardsdatascience.com](http://www.towardsdatascience.com)
* https://medium.com/kitepython/handling-imbalanced-datasets-with-smote-in-python-a94090d031f0

**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:

|  |  |
| --- | --- |
| ***Variable*** | ***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

* 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.**
  + There were data for 30 and 90 days, so considering data for 90 days is adding more information rather than then data of 30 days.
  + Some features can’t have any negative value, so those features 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 StandardScaler to our dependent features.
  + Applied various machine learning model and compared it.
* Data Inputs- Logic- Output Relationships

**I) Average main balance account vs loan pay back rate within 5 days**

Graphical user interface, application

Description automatically generated

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

2- The high balance level people are with 100% rate i.e they are paying loan within 5 days.

3- If talk about average and low balance people ,it is observed that around 10%-12% 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 five days of time.

**II) Frequency of main account recharged in last 30 days vs loan pay back rate within 5 days.**

**Graphical user interface, application

Description automatically generated**

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

2- There is no 100% rate in any of the frequency levels to pay back the loan within 5 days.

3- 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.

4- Coming to low frequency level people, it is observed that around 25% of people are not paying back the loan with in stipulated 5 days of time.

5- 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 five days of time.

**III) Number of loans taken by user in last 30 days vs loan pay back rate within 5 days**

**Graphical user interface, application

Description automatically generated**

1- Customers with different loans levels taken are paying back the loan within five days.

2- In the data set people not taken loans are labelled as ‘1’. So we should not consider the people with no loans labelled in the above graph.

3- Remaining levels, there is no 100% rate in any of the loan levels to pay back the loan within 5 days.

4- Coming to the high number of loan level people it is observed that around 25% of people are not paying the loan within 5 days.

5- Only 2% of the people from low number of loans category are not paying the loan within 5 days. This is followed by the people with medium number of loans having defaulters of 7% approximately.

**IV) 'Total amount of loans taken by user in last 30 days vs loan pay back rate within 5 days**

Graphical user interface, application

Description automatically generated

1- In the data set people not taken loans are labelled as ‘1’. So we should not consider the people with no loans labelled in the above graph.

2- Remaining levels, there is no 100% rate in any of the loan levels to pay back the loan within 5 days.

3-Coming to the low amount level people it is observed that around 25% of people are not paying the loan within 5 days.

4-Only 2% of the people taken high amount of loans are not paying the loan within 5 days. This is followed by the people with medium number of loans having defaulters of 7% approximately.

5- As 'msisdn', 'pcircle', 'pdate' features are not having much importance, we can ignore them. And also removing the extra columns created for the EDA part.

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 10 Pro

Processor-Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz

Installed RAM- 8 GB (7.83 GB Usable

System Type-64 bit OS

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

Libraries used:-

Application

Description automatically generated with medium confidence

Libraries for Model Training

Graphical user interface, application

Description automatically generated with medium confidence

**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 I have replace all null 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()
* #svc = SVC()
* dtc = DecisionTreeClassifier()
* rfc = RandomForestClassifier()
* gdc = GradientBoostingClassifier()
* ada = AdaBoostClassifier()
* Run and Evaluate selected models

Pls find below matrix & their results also.

Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

========== Logistic Regression ==========

LogisticRegression()

Accuracy score: 0.7124590476796335

Cross val score: nan

roc\_auc\_score = 0.7562213276759642

Confusion matrix:

[[ 6340 1445]

[16635 38458]]

Classification Report: precision recall f1-score support

0 0.28 0.81 0.41 7785

1 0.96 0.70 0.81 55093

accuracy 0.71 62878

macro avg 0.62 0.76 0.61 62878

weighted avg 0.88 0.71 0.76 62878

========== GaussianNb ==========

GaussianNB()

Accuracy score: 0.6355482044594294

Cross val score: 1.0

roc\_auc\_score = 0.7303109714605189

Confusion matrix:

[[ 6666 1119]

[21797 33296]]

Classification Report: precision recall f1-score support

0 0.23 0.86 0.37 7785

1 0.97 0.60 0.74 55093

accuracy 0.64 62878

macro avg 0.60 0.73 0.56 62878

weighted avg 0.88 0.64 0.70 62878

========== Kneighbors Classifier ==========

KNeighborsClassifier()

Accuracy score: 0.8664238684436528

Cross val score: 1.0

roc\_auc\_score = 0.9209616690530676

Confusion matrix:

[[ 7734 51]

[ 8348 46745]]

Classification Report: precision recall f1-score support

0 0.48 0.99 0.65 7785

1 1.00 0.85 0.92 55093

accuracy 0.87 62878

macro avg 0.74 0.92 0.78 62878

weighted avg 0.93 0.87 0.88 62878

========== DecisionTreeClassifier ==========

DecisionTreeClassifier()

Accuracy score: 0.9995705970291676

Cross val score: 1.0

roc\_auc\_score = 0.9997549597952553

Confusion matrix:

[[ 7785 0]

[ 27 55066]]

Classification Report: precision recall f1-score support

0 1.00 1.00 1.00 7785

1 1.00 1.00 1.00 55093

accuracy 1.00 62878

macro avg 1.00 1.00 1.00 62878

weighted avg 1.00 1.00 1.00 62878

========== Random Forest Classifier ==========

RandomForestClassifier()

Accuracy score: 0.9987117910875027

Cross val score: 1.0

roc\_auc\_score = 0.9992648793857659

Confusion matrix:

[[ 7785 0]

[ 81 55012]]

Classification Report: precision recall f1-score support

0 0.99 1.00 0.99 7785

1 1.00 1.00 1.00 55093

accuracy 1.00 62878

macro avg 0.99 1.00 1.00 62878

weighted avg 1.00 1.00 1.00 62878

========== GradientBoostinClassifier ==========

GradientBoostingClassifier()

Accuracy score: 0.7967015490314577

Cross val score: nan

roc\_auc\_score = 0.8062800040769504

Confusion matrix:

[[ 6376 1409]

[11374 43719]]

Classification Report: precision recall f1-score support

0 0.36 0.82 0.50 7785

1 0.97 0.79 0.87 55093

accuracy 0.80 62878

macro avg 0.66 0.81 0.69 62878

weighted avg 0.89 0.80 0.83 62878

========== AdaBoostClassifier ==========

AdaBoostClassifier()

Accuracy score: 0.7817837717484653

Cross val score: 1.0

roc\_auc\_score = 0.7915902684362721

Confusion matrix:

[[ 6264 1521]

[12200 42893]]

Classification Report: precision recall f1-score support

0 0.34 0.80 0.48 7785

1 0.97 0.78 0.86 55093

accuracy 0.78 62878

macro avg 0.65 0.79 0.67 62878

weighted avg 0.89 0.78 0.81 62878

* Key Metrics for success in solving problem under consideration

Key Metrices used were the Accuracy Score, Crossvalidation 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.

Graphical user interface, application

Description automatically generated

Text

Description automatically generated

DecisionTreeClassifier()

Accuracy\_score= 0.9995705970291676

Cross\_val\_score= 1.0

roc\_auc\_score= 0.9997549597952553

classification\_report

precision recall f1-score support

0 1.00 1.00 1.00 7785

1 1.00 1.00 1.00 55093

accuracy 1.00 62878

macro avg 1.00 1.00 1.00 62878

weighted avg 1.00 1.00 1.00 62878

Confusion Matrix

[[ 7785 0]

[ 27 55066]]

AxesSubplot(0.125,0.808774;0.62x0.0712264)

* Visualizations

Graphical user interface, application

Description automatically generated

* Interpretation of the Results

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

We will save Decision Tree Claasifiers (DTC) as it's having 100 % Accuracy & its cross val score & roc auc score is also almost 100%, or more accurate than Random Forest Model

**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 ROC-AUC 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.