Micro-Credit Defaulter Model

Submitted by:

Rohan V Borade

ACKNOWLEDGMENT

I have referred below resources that helped and guided me in completion of this project as below:

www.w3resource.com

www.towardsdatascience.com

www.stackoverflow.com

INTRODUCTION

- ## Business Problem Framing

This project is about providing loans (financial services) to low income populations by Micro-Financial Institution (MFI). MFI provide loan to Group Loans, Agricultural Loans, Individual Business Loans and so on. In Order to achieve this objective, MFI needs to decide criteria for customer selection.

- ## Conceptual Background of the Domain Problem

Banking domain knowledge is required to know about generic criteria for loan giving institutions, market risk and parameters to decide defaulter, interest charges, benefits, etc.

- ## Review of Literature

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.

- ## 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.

In every country poor population exists to some scale and financial services to be provided to them at affordable level of loan amount to uplift their financial situation, which may reduce the vulnerability factor.

Analytical Problem Framing

- ## Mathematical/ Analytical Modelling of the Problem

  Describe the mathematical, statistical and analytics modelling done during this project along with the proper justification.

- ## Data Sources and their formats

  I have use data sources, data provided by client in excel or csv format.

- ## Data Pre-processing Done

  Data which I received had many null values, zero values.

  Data containing multiple formats like floats, string, integers so I clean data by replacing zeros with NaN values & then replace NaN by mean method.

  Post this I used label encoder 2 sets one for float data & other for string data & converted all data into integers after that verified for non-null values.

- ## Data Inputs- Logic- Output Relationships

  Input data for feature list and target is in numeric format and hence classification model (K-nearest neighbours) best suits for this dataset.

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

  I have not considered 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)
Memory-4 GB
System Type-64-bit OS

Software Requirement- Anaconda 3.7 & above, Jupiter Notebook 6.

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.
  - Logistic Regression
  - Linear Discriminant Analysis
  - K-Neighbors Classifier
  - Decision Tree Classifier
  - Gaussian NB

- Run and evaluate selected models

Pls find below matrix & their results also.

```
In [51]: for i in range(1,1500):
             x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=i,test_size=0.20)

             lm.fit(x_train,y_train)
             pred_train=lm.predict(x_train)
             pred_test=lm.predict(x_test)

             if round(accuracy_score(y_train,pred_train)*100,1)==round(accuracy_score(y_test,pred_test)*100,1):
                 print("At random state",i,"The model performs very well")
                 print("At random state:-",i)
                 print("Training score is:-",accuracy_score(y_train,pred_train)*100)
                 print("Testing score is:-",accuracy_score(y_test,pred_test)*100)

At random state 4 The model performs very well
At random state:- 4
Training score is:- 87.88899889070458
Testing score is:- 87.86469142870774
At random state 8 The model performs very well
At random state:- 8
Training score is:- 87.90390877536171
Testing score is:- 87.85276366325533
At random state 10 The model performs very well
At random state:- 10
Training score is:- 87.91404749692856
Testing score is:- 87.86230587561727
At random state 11 The model performs very well
At random state:- 11
Training score is:- 87.92776459081313
Testing score is:- 87.89331806579355
At random state 13 The model performs very well
At random state:- 13
Training score is:- 87.90510156613428
```

```
In [58]: from sklearn.ensemble import RandomForestClassifier

In [59]: model_rf=RandomForestClassifier(n_estimators=100, criterion='gini', random_state = 100,max_depth=6, min_samples_leaf=8)

In [60]: model_rf.fit(x_train,y_train)
Out[60]: RandomForestClassifier(max_depth=6, min_samples_leaf=8, random_state=100)

In [61]: y_pred=model_rf.predict(x_test)

In [62]: model_rf.score(x_test,y_test)
Out[62]: 0.9051742646532599

In [63]: print(classification_report(y_test, y_pred, labels=[0,1]))
                   precision    recall  f1-score   support

                0       0.88      0.27      0.42      5199
                1       0.91      0.99      0.95     36720

         accuracy                           0.91     41919
        macro avg       0.89      0.63      0.68     41919
     weighted avg       0.90      0.91      0.88     41919
```

- # Key Metrics for success in solving problem under consideration

  Used cross validation matrix to overcome under-fitting /over-fitting this model by deciding number of folds.

- # Visualizations

  Mention all the plots made along with their pictures and what were the inferences and observations obtained from those. Describe them in detail.

  If different platforms were used, mention that as well.

- # Interpretation of the Results

  Visualisation shows outliers which need to be removed / corrected.

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

Modelling done based on type of data as this is categorical data, we have to go with multiple classification models & finalise the best score giving 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's 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.

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

Visualize the data using univariant / multi-variant analysis.

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 pickle/joblib library.

Find the prediction vs actual using distribution plot in order to get the perfect deviation if any.

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

Column with no impact/no correlation have excluded as it might have reduced the performance.

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