Micro-Credit Defaulter Model

Submitted by:

Venkateswara Reddy D

**ACKNOWLEDGMENT**

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

[www.w3resource.com](http://www.w3resource.com)

[www.towardsdatascience.com](http://www.towardsdatascience.com)

[www.stackoverflow.com](http://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, and 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 Modeling of the Problem

I have use describe function from pandas library to see all the statistical model of our dataset.

Image

This function gives all the statistics summary of our dataset that is mean, median, minimum and maximum values of our dataset. It also tells whether data is right skewed or left skewed based of the diference between mean and median.

From that part we also able to analyze that whether outliers are present in our dataset or not based on the difference between 75% and maximum value. The describe function also gives the counting of our data for the particular column.

* Data Sources and their formats

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

* Data Preprocessing 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 neighbors) 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-6 GB

System Type-64 bit OS

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

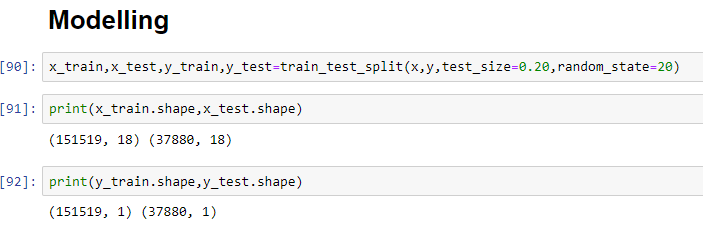
**Model/s Development and Evaluation**

* 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 methgod 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
* **Splitting the data into input and target features:**

Now when all categorical variables are transformed and all numerical features are normalized, we need to split our data into training and test sets. We split 80% to training and 20% for testing.



We have used Python Package because it is powerful and general purpose programming language.

NumPy—It is a math library to work with ndimensional arrays. It enables us to do computation effectively and regurarly. For working with arrays, dictionary, functions data type we need to know NumPy.

Pandas—It is high level Python library and easy to use for data importing , manipulation and data analysis.

Matplotlib—It is a plotting that provide 2D and 3D plotting.

Seaborn-- Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

SciPy—It is a collection of numerical algorithm and domain specific tool boxes including optimization, statistics and much more.

Scikit-learn—It is a collection of tools and algorithm for machine learning. It works with NumPy and SciPy and it is easy to implement machine learning models.





\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* KNeighborsClassifier \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

KNeighborsClassifier(algorithm='auto', leaf\_size=30, metric='minkowski',

metric\_params=None, n\_jobs=None, n\_neighbors=10, p=2,

weights='uniform')

Accuracy score = 0.8892291446673707

Cross\_val\_Score = 0.8895453824188507

roc\_auc\_score = 0.6785894907712662

classification\_report

precision recall f1-score support

0 0.54 0.40 0.46 4483

1 0.92 0.95 0.94 33397

accuracy 0.89 37880

macro avg 0.73 0.68 0.70 37880

weighted avg 0.88 0.89 0.88 37880

[[ 1805 2678]

[ 1518 31879]]

AxesSubplot(0.125,0.808774;0.62x0.0712264)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* LogisticRegression \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

LogisticRegression(C=1.0, class\_weight=None, dual=False, fit\_intercept=True,

intercept\_scaling=1, l1\_ratio=None, max\_iter=100,

multi\_class='warn', n\_jobs=None, penalty='l2',

random\_state=None, solver='warn', tol=0.0001, verbose=0,

warm\_start=False)

Accuracy score = 0.8904699049630412

Cross\_val\_Score = 0.8900205510455008

roc\_auc\_score = 0.5832149007174124

classification\_report

precision recall f1-score support

0 0.63 0.18 0.28 4483

1 0.90 0.99 0.94 33397

accuracy 0.89 37880

macro avg 0.76 0.58 0.61 37880

weighted avg 0.87 0.89 0.86 37880

[[ 810 3673]

[ 476 32921]]

AxesSubplot(0.125,0.808774;0.62x0.0712264)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* DecisionTreeClassifier \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

DecisionTreeClassifier(class\_weight=None, criterion='gini', max\_depth=None,

max\_features=None, max\_leaf\_nodes=None,

min\_impurity\_decrease=0.0, min\_impurity\_split=None,

min\_samples\_leaf=1, min\_samples\_split=2,

min\_weight\_fraction\_leaf=0.0, presort=False,

random\_state=20, splitter='best')

Accuracy score = 0.850026399155227

Cross\_val\_Score = 0.8521323175454454

roc\_auc\_score = 0.6544257338882021

classification\_report

precision recall f1-score support

0 0.37 0.40 0.39 4483

1 0.92 0.91 0.91 33397

accuracy 0.85 37880

macro avg 0.65 0.65 0.65 37880

weighted avg 0.85 0.85 0.85 37880

[[ 1785 2698]

[ 2983 30414]]

AxesSubplot(0.125,0.808774;0.62x0.0712264)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* GaussianNB \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

GaussianNB(priors=None, var\_smoothing=1e-09)

Accuracy score = 0.8665786694825766

Cross\_val\_Score = 0.8683942936382879

roc\_auc\_score = 0.6211328198964203

classification\_report

precision recall f1-score support

0 0.41 0.30 0.35 4483

1 0.91 0.94 0.93 33397

accuracy 0.87 37880

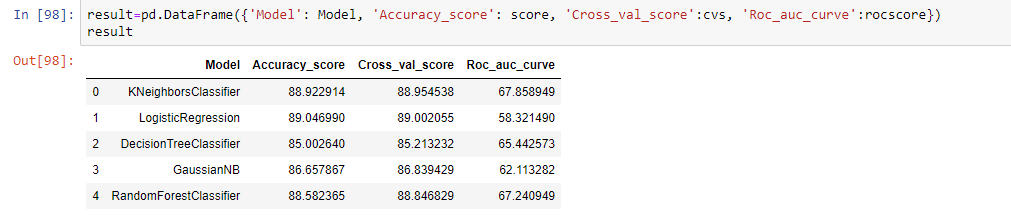
macro avg 0.66 0.62 0.64 37880

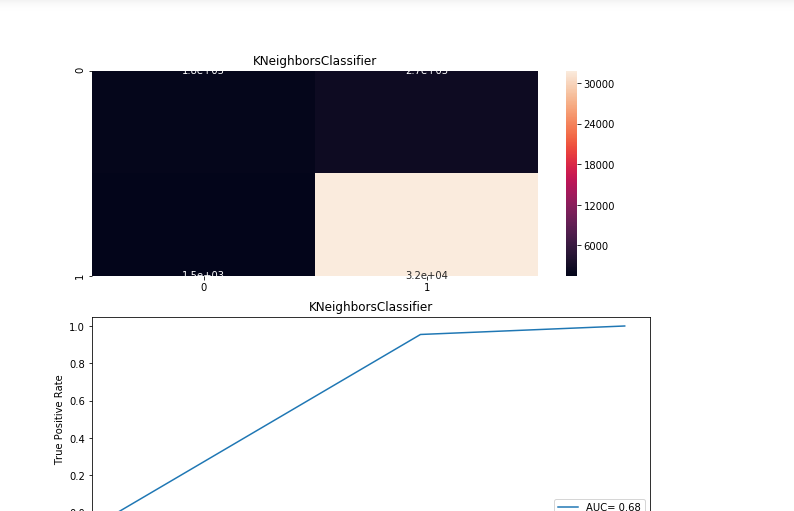
weighted avg 0.85 0.87 0.86 37880

[[ 1343 3140]

[ 1914 31483]]

AxesSubplot(0.125,0.808774;0.62x0.0712264)





* 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
* There are no null values in the dataset.
* The dataset is imbalanced. Label ‘1’ has approximately 86% records, while, label ‘0’ has approximately 14% records.
* maxamnt\_loans90 columns gives information about customers with no loan history.
* msisdn and aon features some values which might not be realistic. So drop the row which contain not realistic value
* There are some rows which is repeated means duplicate entries are present in our dataset.
* Sampling data gives the better precision and recall value along with auc score.
* The given dataset has only three months of data and it is also not the present year data.
* The collected data is only for one area circle.

* **Learning Outcomes of the Study in respect of Data Science**

In this project we learn how to build a machine learning model for classification based problem.

We also learn how to handle the imbalanced dataset for machine learning model. Because when we over sampling and use this over sampled data to build a ML model then it gives the better result.

The goal of any machine learning problem is to find a single model that will best predict our wanted outcome. Rather than making one model and hoping this model is the best/most accurate predictor we can make, ensemble methods take a myriad of models into account, and average those models to produce one final model.

On doing this project the biggest problem I have faced is that I am not able to use GridSearchCV. Because when I use GridSearchCV then my system takes too much time to give the result as our dataset is too large . So If I uses GridSearchCV then our result improves.

So based on all the learning and outcomes our Random Forest Classifier Model with over sampled data gives the best result so we save this model as our final model by using Joblib as a pickle file.

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

The dataset is belongs to only one area circle so our model trained for the data of only one area circle and also this data is too old it is from 2016 and contains only three months of data.

As Today I uses i5 processor computer so I am not able to use GridSearchCV and cross validation because my system takes too much time to give the result. If some how I uses GridSearchCV and Cross validate the model then our model result will be surely improved.

I would conclude the project report by hoping that now you have understood every step that is required to be done to build a machine learning model. We have built the classification model for classifying the labels that is which customers is in label 1 that is Non defaulter and which one is in label 0 that is defaulter customer and then evaluated it using different error metrics.