



Micro-Credit Defaulter Model

Submitted by: **NAROJ SANTHOSH KUMAR**

ACKNOWLEDGMENT

This includes mentioning of all the references, research papers, data sources, professionals and other resources that helped you and guided you in completion of the project.

References:

1. Python Data Analysis: Perform data collection, data processing, wrangling, visualization, and model building using Python, 3rd Edition. by Avinash Navlani (Author), Armando Fandango (Author), Ivan Idris (Author)
2. Data Scrap doubt clearing session provided by FlipRobo.
3. Notes and classes by Data trained Academy.

INTRODUCTION

- Business Problem Framing

(Describe the business problem and how this problem can be related to the real world.)

I am doing this analysis for the company Microfinance Institution (MFI), who lend facilities to unbanked poor families living in remote area. Low-income population are the majority in a country like Indonesia and denied to finance by leading banks and Financial institutions.

Our customer segment is the riskiest and more likely to default. Hence, it's important to identify the those who tend to default loan before approving.

This project to predict transactors and delinquent customers, help micro finance institution to reduce delinquency and thereby improve business.

- Conceptual Background of the Domain Problem

(Describe the domain related concepts that you think will be useful for better understanding of the project.)

The growth of new generation banks and small Micro finance institutions has been significant over last 2 decades. They have got attention by delivering microfinances to poor population.

Since they finance to the most unexplored segment ever, there is an exponential growth as well as hidden risk involved.

Under this circumstance, Machine Learning has its relevance by constantly monitoring repayment features of already funded customers and further lending will be based on Machine learning prediction.

- **Review of Literature**

(This is a comprehensive summary of the research done on the topic. The review should enumerate, describe, summarize, evaluate and clarify the research done.)

Poverty is a major problem in developing economies. Micro finance is the provision of financial service to poor with daily income. Only small fraction of people has access to financial instruments.

My study on Machine learning majorly focuses to identify customers who tend to default the micro credit facility. Payment features like Daily amount spent, Average main account balance, Average data account balance, No. of times the account is recharged, amount of recharge, No. of loans taken, Maximum loan amount and Average payback time are recorded for 30 days and 90 days period are the data input for evaluation.

Above said features were plotted to understand the general trend and noted observations, thereby deleted irrelevant features that doesn't contribute to the target and model was build.

- **Motivation for the Problem Undertaken**

(Describe your objective behind to make this project, this domain and what is the motivation behind.)

There is a growing interest in Micro Finance Institutions (MFI) as one of the avenues to enable low-income population to access financial services. With the growth of the sector in terms of both size and scope it needs more advanced technical regulations to sustain.

The motivation behind is the concern on sustainability of the growth rate and need for an optimized model for data evaluation.

Analytical Problem Framing

- **Mathematical/ Analytical Modeling of the Problem**

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

Every classification algorithm is built up with strong mathematical models and logic. The key challenge is to find out the appropriate algorithm for each application in machine learning.

I have use statistical models to find insights given set of data. Predictions are assessed differently depending on its type: Based on the data structure of Target variable, Model type was identified. I have used classification metrics like Confusion matrix, Classification report and Accuracy score. I also have used error measures such as True Positive, False Positive etc. for classification problem.

First and important step was to identify the model which was done by analyzing the target variable 'label'. The feature consists only values "1" and "0", which is ideal case of Binary classification and hence Classification algorithms were used. Best accuracy was achieved using Random Forest Classifier.

- Data Sources and their formats

(What are the data sources, their origins, their formats and other details that you find necessary? They can be described here. Provide a proper data description. You can also add a snapshot of the data.)

We have a Dataset of more than 2Lac sample/entries about customers who Loaned from their Mobile service provider and their repayment information.

Data Count : 209,593

Data Features 35

Data Types : int64, float64, object

Data sample:

Unnamed: 0	label	msisdn	aon	daily_decr30	daily_decr90	rental30	rental90	last_rech_date_ma	last_rech_date_da	...	maxamnt_loans30	med
0	1	0	21408170789	272.0	3055.050000	3065.150000	220.13	260.13	2.0	0.0	...	6.0
1	2	1	76462170374	712.0	12122.000000	12124.750000	3691.26	3691.26	20.0	0.0	...	12.0
2	3	1	17943170372	535.0	1398.000000	1398.000000	900.13	900.13	3.0	0.0	...	6.0

Data Columns and data types:

Unnamed: 0	int64	fr_ma_rech90	int64
label	int64	sumamnt_ma_rech90	int64
msisdn	object	medianamnt_ma_rech90	float64
aon	float64	medianmarechprebal90	float64
daily_decr30	float64	cnt_da_rech30	float64
daily_decr90	float64	fr_da_rech30	float64
rental30	float64	cnt_da_rech90	int64
rental90	float64	fr_da_rech90	int64
last_rech_date_ma	float64	cnt_loans30	int64
last_rech_date_da	float64	amnt_loans30	int64
last_rech_amt_ma	int64	maxamnt_loans30	float64
cnt_ma_rech30	int64	medianamnt_loans30	float64
fr_ma_rech30	float64	cnt_loans90	float64
sumamnt_ma_rech30	float64	amnt_loans90	int64
medianamnt_ma_rech30	float64	maxamnt_loans90	int64
medianmarechprebal30	float64	medianamnt_loans90	float64
cnt_ma_rech90	int64	payback30	float64
fr_ma_rech90	int64	payback90	float64
sumamnt_ma_rech90	int64	pcircle	object
medianamnt_ma_rech90	float64	pdate	object
medianmarechprebal90	float64	dtype: object	
cnt_da_rech30	float64		
fr_da_rech30	float64		
cnt_da_rech90	int64		

Except Telecom circle and Date all other columns are numeric datatypes.

There are no null values in the database.

- Data Preprocessing Done

(What were the steps followed for the cleaning of the data? What were the assumptions done and what were the next actions steps over that?)

- Checked for null values and identified no null values in dataset.
- Values of some of the columns were checked to get more insights and observed columns like 'unnamed', 'msisdn', 'pcircle' are irrelevant and hence removed the same.
- From total sample numbers 209593, and only 186243 are unique i.e. 23,350 duplicate mobile numbers. Retained the same as this consists of 11% of data and not more than 7-8% can be deleted.
- Outliers and skewness were checked.

- Hardware and Software Requirements and Tools Used

Listing down the hardware and software requirements along with the tools, libraries and packages used. Describe all the software tools used along with a detailed description of tasks done with those tools.

1. Lenovo flex corei5 laptop
2. Jupyter Notebook - The Jupyter Notebook is an interactive environment for running code in the browser. It is a great tool for exploratory data analysis and is widely used by data scientists.
3. MS PowerPoint - For preparing presentation of project.
4. MS word - For preparing report
5. Matplotlib - For data visualization to produce high quality plots, charts and graphs.
6. Pandas - Python library for data wrangling and analysis. It is built around a data structure called Data Frame.
7. NumPy - NumPy is one of the fundamental packages for scientific computing in Python.

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

I have used statistical models to find insights given set of data. Predictions are assessed differently depending on its type: Based on the data structure of Target variable, Model type was identified. I have used classification metrics like Confusion matrix, Classification report and Accuracy score. I also have used error measures such as True Positive, False Positive etc. for classification problem.

First and important step was to identify the model which was done by analyzing the target variable 'label'. The feature consists only values "1" and "0", which is ideal case of Binary classification and hence Classification algorithms were used. Best accuracy was achieved using Random Forest Classifier.

- Testing of Identified Approaches (Algorithms)

Build Models based on learning it was a supervised classification problem.

I built 3 models to evaluate performance of each of them:

- a. Logistic Regression
- b. Random forest Classifier
- c. Decision Tree Classifier

Since the data was imbalanced, I used metrics like precision, recall and ROC-AUC curve.

- Run and Evaluate selected models

Describing all the algorithms used along with the snapshot of their code and what were the results observed over different evaluation metrics.

1. LOGISTIC REGRESSION

```
# logistic regression object
lr = LogisticRegression(solver='lbfgs', max_iter=400)

# train the model on train set
lr.fit(x_train, Y_train)
lr.score(x_train, Y_train)

predictions = lr.predict(x_test)

# Remodelling of after sampling
lr1 = LogisticRegression(solver='lbfgs', max_iter=400)
lr1.fit(x_train_res, Y_train_res)
lr1.score(x_train_res, Y_train_res)
predictions = lr1.predict(x_test)
```

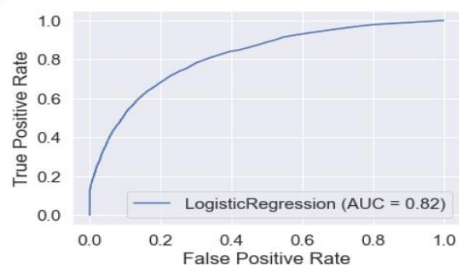
Confusion Matrix: [[5825 2077]
[13984 40992]]

Classification Report:		precision	recall	f1-score	support
0	0.29	0.74	0.42	0.52	7902
1	0.95	0.75	0.84	0.79	54976
accuracy			0.74	0.74	62878
macro avg	0.62	0.74	0.63	0.63	62878
weighted avg	0.87	0.74	0.78	0.78	62878

Accuracy score: 0.7445688476096568

ROC-AUC CURVE FOR LR

```
metrics.plot_roc_curve(lr1, x_test, Y_test)
plt.show()
```



2. DECISIONTREE CLASSIFIER

```
dtc=DecisionTreeClassifier()  
dtc.fit(x_train_res, Y_train_res)  
dtc.score(x_train_res, Y_train_res)  
prediction1 = dtc.predict(x_test)
```

```
print("Confusion Matrix:",confusion_matrix(Y_test, prediction1))  
print("Classification Report:",classification_report(Y_test, prediction1))  
print("Accuracy score:",accuracy_score(Y_test, prediction1))
```

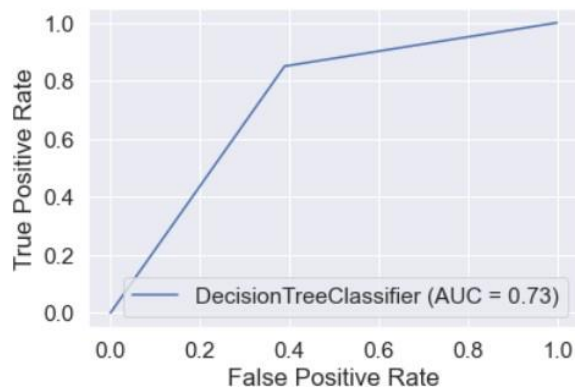
Confusion Matrix: [[4816 3086]
[8191 46785]]

Classification Report:		precision	recall	f1-score	support
0	0.37 0.61 0.46	0.46	7902		
1	0.94 0.85 0.89	0.89	54976		
accuracy		0.82	62878		
macro avg		0.65 0.73 0.68	62878		
weighted avg		0.87 0.82 0.84	62878		

Accuracy score: 0.8206526925156653

ROC-AUC CURVE FOR DTC

```
metrics.plot_roc_curve(dtc, x_test, Y_test)  
plt.show()
```



RANDOMFOREST CLASSIFIER

```
rfc=RandomForestClassifier()  
rfc.fit(x_train_res, Y_train_res)  
rfc.score(x_train_res, Y_train_res)  
prediction2 = rfc.predict(x_test)
```

```
print("Confusion Matrix:",confusion_matrix(Y_test, prediction2))  
print("Classification Report:",classification_report(Y_test, prediction2))  
print("Accuracy score:",accuracy_score(Y_test, prediction2))
```

Confusion Matrix: [[4938 2964]

[4368 50608]]

Classification Report: precision recall f1-score support

0	0.53	0.62	0.57	7902
---	------	------	------	------

1	0.94	0.92	0.93	54976
---	------	------	------	-------

accuracy			0.88	62878
----------	--	--	------	-------

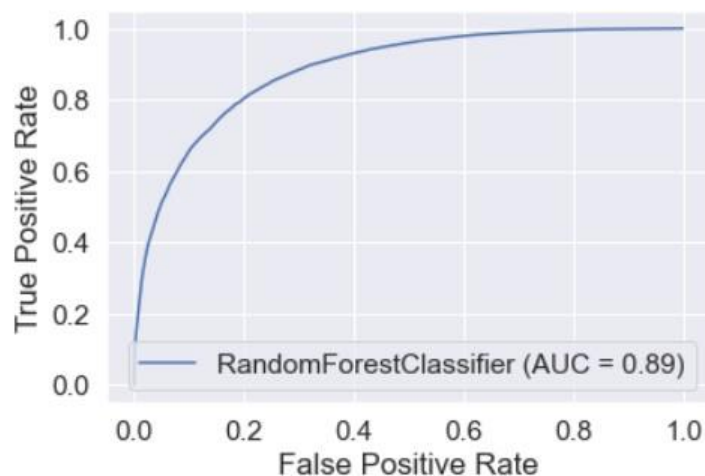
macro avg	0.74	0.77	0.75	62878
-----------	------	------	------	-------

weighted avg	0.89	0.88	0.89	62878
--------------	------	------	------	-------

Accuracy score: 0.8833932376984001

ROC-AUC CURVE FOR RFC

```
metrics.plot_roc_curve(rfc, x_test, Y_test)  
plt.show()
```



- Key Metrics for success in solving problem under consideration

The key metrics used:

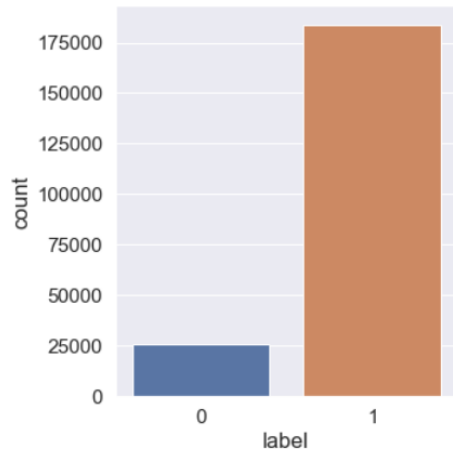
1. Confusion Metrics
2. Classification Report
3. Accuracy score

```
print("Confusion Matrix:",confusion_matrix(Y_test, prediction2))  
print("Classification Report:",classification_report(Y_test, prediction2))  
print("Accuracy score:",accuracy_score(Y_test, prediction2))
```

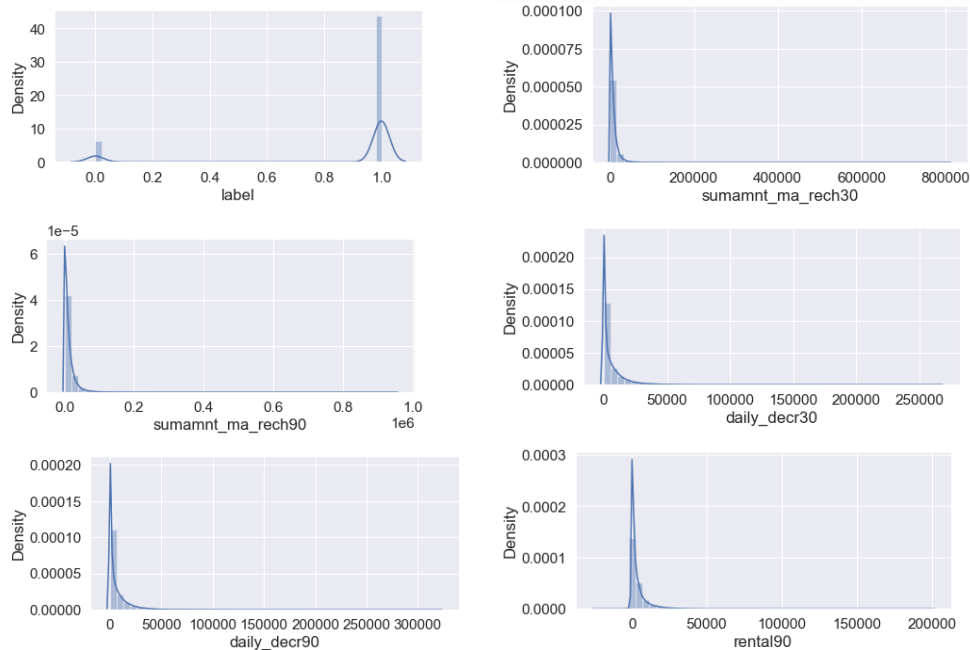
- Visualizations & Interpretation of the Results

Summary of what results were interpreted from the visualizations, preprocessing and modelling.

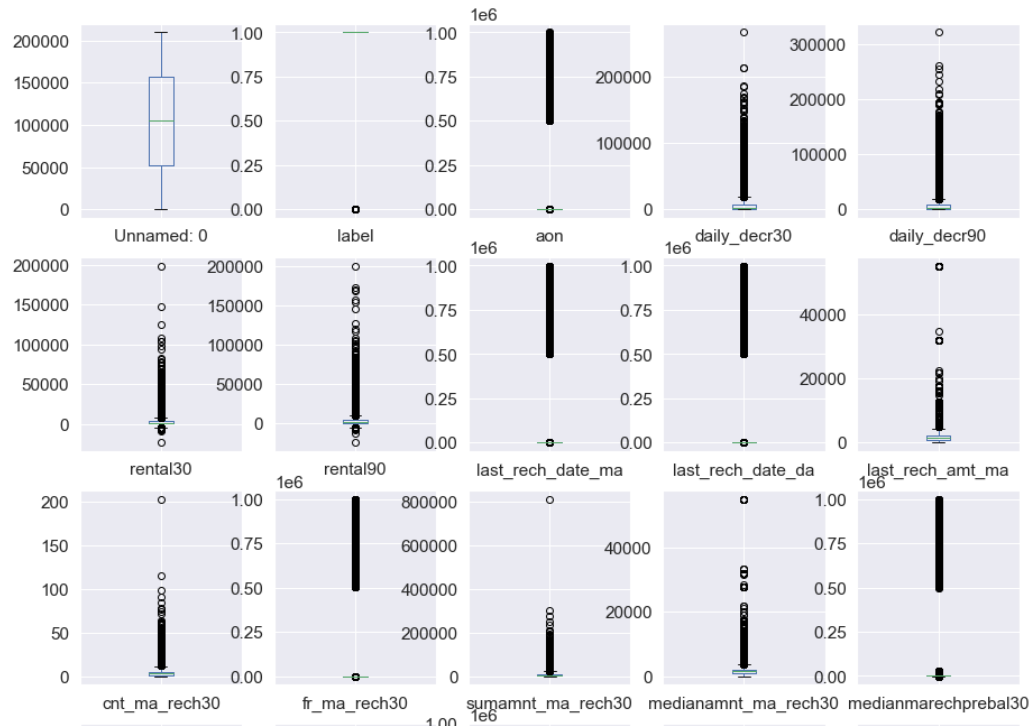
```
# Plotting target variable.  
sns.catplot(x='label', data=df, kind='count')  
plt.show()
```



It was observed that the dataset was imbalanced for the target feature(87.5% for Non-defaulters and 12.5% for Defaulters)

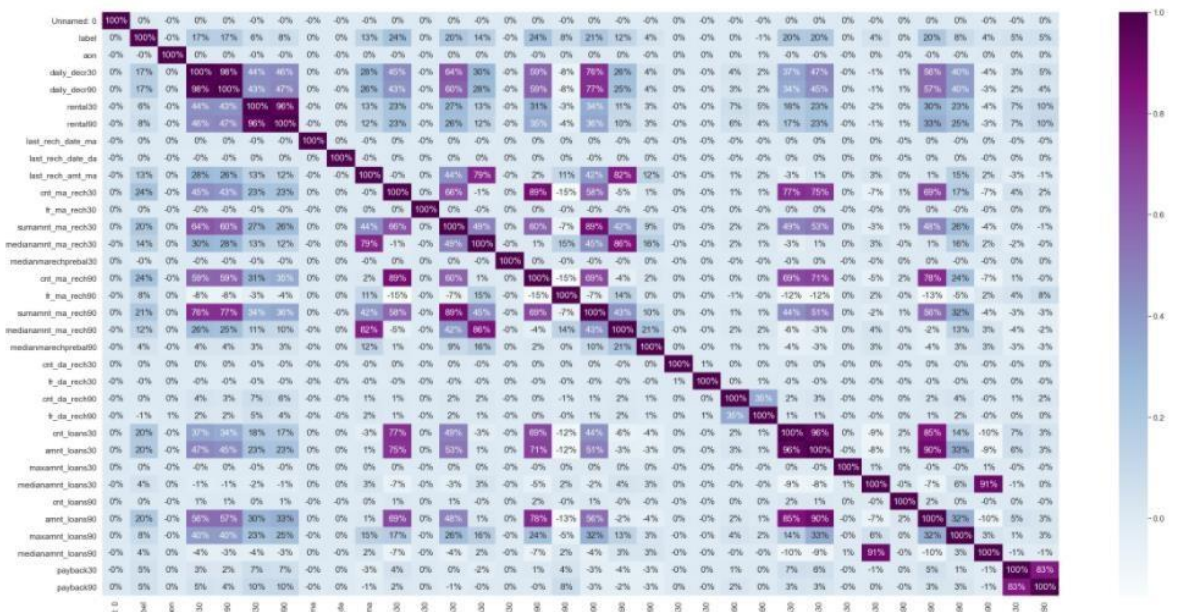


```
# Checking outliers with boxplot.
df.plot(kind='box', figsize=(15,30), layout=(8,5), sharex=False, subplots=True)
plt.show()
```



```
#Heatmap to know correlation between attributes.
correlations = df.corr()
plt.figure(figsize=(40,20))
sns.set(font_scale=1.4)
sns.heatmap(correlations, data=df, annot=True, fmt='.0%', cmap="BuPu")
```

<AxesSubplot:>



Analyzing the heatmap,

1. Target variable 'label' has no correlation more than 30% with any features.
2. 'label' has 24% correlation with count of main account got recharged in last 30 days and 90 days.

On visualizing data, there were two important insights I gathered.

- a. Imbalance of data
- b. Distribution was not normal

Data Normalization Since the data was not normal, I normalized all the features except the target variable.

Instead of under sampling the majority class. I chose to oversample the minority class using SMOTE.

CONCLUSION

- Key Findings and Conclusions of the Study

According to the performance metrics, Random Forest scores highest in accuracy. Also, the curve is tending towards the ideal shape. Hence, Random Forest looks like the best fit for this data.

Learning Outcomes of the Study in respect of Data Science

- Learnt about sampling technique and technique to perform oversampling with smote and installing smote as well.
- I have also learnt to perform normalization.
- Major challenge faced was during Feature reduction, I couldn't perform it properly and hence avoided the step.

- Limitations of this work and Scope for Future Work

Techniques can be followed to further,

1. Dimensionality reduction.
2. Better dealing with outliers.