



Micro-Credit Defaulter Model

Submitted by:

Vishal Pandey

ACKNOWLEDGMENT

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

I wish to express my sincere gratitude to Mr. Shubham Yadav, SME for providing me an opportunity to do my internship and project work in “FLIP ROBO”.

It gives me immense pleasure in presenting this project report on “Micro Credit Defaulter Model”. It has been my privilege to have a team of project guide who have assisted me from the commencement of this project. The success of this project is a result of sheer hard work, and determination put in by me with the help of You Tube videos, references taken from Kaggle.com, skikit-learn.org..

INTRODUCTION

Business Problem Framing

Loans default will cause huge loss for the banks, so they pay much attention on this issue and apply various methods to detect and predict default behaviors of their customers. In this blog, I am going to talk about the basic process of loan default prediction with machine learning algorithms.

The loan is one of the most important products of the banking. All the banks are trying to figure out effective business strategies to persuade customers to apply their loans. However, there are some customers behave negatively after their application are approved. To prevent this situation, banks have to find some methods to predict customers' behaviours. Machine learning algorithms have a pretty good performance on this purpose, which are widely-used by the banking. Here, I will work on loan behaviours prediction using machine learning models

Microcredit is a common form of micro finance that involves an extremely small loan given to an individual to help them become self-employed or grow a small business. These borrowers tend to be low-income individuals, especially from less developed countries. Microcredit is also known as "microlending" or "microloan."

The concept of microcredit was built on the idea that skilled people in under developed countries, who live outside of traditional banking and monetary systems, could gain entry into an economy through the assistance of a small loan. The people to whom such microcredit is offered may live in barter systems where no actual currency is exchanged

Conceptual Background of the Domain Problem

Today, micro-credit is widely accepted as a poverty-reduction tool, representing \$70 billion in outstanding loans and a global outreach of 200 million clients.

We are working with one such client that is in Telecom Industry. They are a fixed wireless telecommunications network provider. They have launched various products and have developed its business and organization based on the budget operator model, offering better products at Lower Prices to all value conscious customers through a strategy of disruptive innovation that focuses on the subscriber.

They understand the importance of communication and how it affects a person's life, thus, focusing on providing their services and products to low-income families and poor customers that can help them in the need of hour.

They are collaborating with an 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).

We can see there is huge population with no financial record is there to access all that remote areas to help them we need to come with more ground projects to help such population

In this whole scenario we encounter that there is a must use of Artificial Intelligence because as we can see that there is high variation of defaulters and non-defaulters.

By the use of AI we can analyse patterns of peoples who are taking micro credit by the help of their history of recharge, date of recharge, daily usage, there payment history of loans etc. After putting all these constrains in AI and modelling with different models we will come to a point where we can suggest a point of view what more improvement is needed or who all are the ones which will be defaulter and should be stop from credit facility.

Review of Literature

Micro credit has been the term which refers to the formal and informal arrangements of providing financial services to the poor for the upliftment. It is microfinance which over the past decades has changed the perception of poor from non-bankable to bankable and recommending various methodologies to provide financial services. Also, Microfinance over the years has not only tried to alleviate poverty across the world but also shown glimpses of sustaining themselves from profit earned in the process.

Asian Development Bank (ADB) defines microfinance as “the provision of a broad range of financial services as deposits, loans, money transfers, insurance to small enterprise and households, and providing credit in telecommunication services.

CGAP (2003) defines microfinance as “a credit methodology that employs effective collateral substitutes to deliver and recover short-term working capital loans to micro entrepreneurs.”

I went and felt that Microfinance Institution purpose is to fulfil the financial needs of the poor either through informal or flexible approach. There is no single model that fits in all the circumstances. Number of microfinance models emerged in different countries/states according to the suitability to their local conditions. Broadly, the microfinance delivery methods can be classified into six parts:

- Grameen Bank Model
- Joint Liability Group Model
- Individual Lending Model
- The Self-Help Group Model
- Village Banking Model
- Credit Unions and Cooperatives

We got to know after studying about micro finance that this facility is to curb down and help poor who are short form money. So, by the help of this they provide small loans to them and a deadline of paying back. But sometimes these low sound people become defaulter they cannot repay the loan taken so that is why these small amounts are given in loan because if they fail to repay then the micro finance company don't face big losses. As well as for more safety they are using Artificial Intelligence to speculate and understand the patterns where the person is failing to repay the credit taken. By varieties of model's data scientist and researcher are putting their expertise knowledge for minimizing and overcoming from this problem.

An attempt has been made in the available literature in the area of microfinance.

Motivation for the Problem Undertaken

The initiative of the company to provide Micro credit is very noble to help the low-income group of people but there are certain people who take advantage of this noble idea and don't bother to repay the money and become defaulter. So, it is necessary to stop this type of practice. Sometime people with good intension remain deprived of getting loan from the financial institution due to some dishonest people. Hence Machine Learning can be used to predict the defaulter and non-defaulter by using different parameters.

Analytical Problem Framing

Mathematical/ Analytical Modeling of the Problem

Starting with the dataset, when I looked through the statistical description, we come to see that most of the data are unbalanced. There is high standard deviation from the mean value. The difference between the third quartile and maximum value was huge in many cases which was quite abnormal. In some places the minimum values were negative which also seem to be abnormal in that case. It was found in some variables that, the maximum value was abnormally high which was replaced by a normal high number of that variable. The visualization also helped to identify the skewness present in the data. Those skewness were also corrected using power transformation. At last, after data pre-processing we come the model building section, where I used Logistic Regression, Gaussian NB and Random Forest Classifier, Decision Tree Classifier, ADA boostingClassifier and Gradient boosting classifier.

Data Sources and their formats

Our dataset is of micro credit defaulter which is a data taken from Indonesia telecom industry. The data is all about the credit given to low-income families for using telecom service. It consists of total 37 different columns which are as follows out of which our target variable column named as Label in it, we have entries of 0 and 1. 0 means defaulter and 1 means Non-defaulter. There are 209593 rows also present in it.

```
1 df=pd.read_csv("Data file.csv")
df
```

	Unnamed: 0	label	msisdn	aon	daily_decr30	daily_decr90	rental30	rental90	last_rech_date_ma	last_rech_date_da	...	maxamnt_loans30
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
3	4	1	55773170781	241.0	21.228000	21.228000	159.42	159.42	41.0	0.0	...	6.0
4	5	1	03813182730	947.0	150.619333	150.619333	1098.90	1098.90	4.0	0.0	...	6.0
...
209588	209589	1	22758185348	404.0	151.872333	151.872333	1089.19	1089.19	1.0	0.0	...	6.0
209589	209590	1	95583184455	1075.0	36.936000	36.936000	1728.36	1728.36	4.0	0.0	...	6.0
209590	209591	1	28556185350	1013.0	11843.111667	11904.350000	5861.83	8893.20	3.0	0.0	...	12.0
209591	209592	1	59712182733	1732.0	12488.228333	12574.370000	411.83	984.58	2.0	38.0	...	12.0
209592	209593	1	65061185339	1581.0	4489.362000	4534.820000	483.92	631.20	13.0	0.0	...	12.0

209593 rows x 37 columns

Data Pre-processing Done

After loading of dataset in jupyter, I first checked the shape of the dataset then I moved further in next step for checking any null values present in it. I analysed the data I came to know that in our dataset 'msisdn' which carry mobile numbers of persons is very big and for machine learning model it cannot be used as it will affect the overall model prediction so we came to the result to drop that column.

Now moving further, unnamed 0 column is of no use in the dataset as it gives no insights with our dataset so we will drop this column.

Next moving to column name msisdn, pcircle and pdate which are of object type and cannot give any help in best performance of model. Pcircle means a code given to certain areas which are not showing any relevance with population mobile credit taking.

Next is pdate first i checked taking only month and date from it and dropping of year because this dataset is for the year 2016. Then I checked doing visualization and came to know that it not giving any insights with other columns so i came to this position to drop pdate column.

I also found that there are columns which have negative values .

I checked by using describe function to know how is our dataset and came to observation that our target variable column 'label' is imbalanced

Most of the data in the dataset was full of outliers. Some of them were negatively whereas some are positively skewed. All the skewed data was corrected using power- transformation where ever applicable.

Data Inputs- Logic- Output Relationships

The input data provided, helps to understand the behaviour of the customer, their various transaction records, their frequency of transaction during a period of time etc, all these helps to predict the customer's intension toward the repayment of loan.

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

No as such assumption been done related to the circumstances.

Hardware and Software Requirements and Tools Used

Data Science task should be done with sophisticated machine with high end machine configuration. The machine which I'm currently using is powered by intel core i5 processor with 8GB of RAM. With this above-mentioned configuration, I managed to work with the data set in Jupyter Notebook which help us to write Python codes. As I'm using low configuration machine so it took more time then usual to execute codes. The library used for the assignment are Numpy, Pandas, Matplotlib, Seaborn, Scikit learn

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sn
```

```
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.metrics import r2_score
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
```

```
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import GridSearchCV
```

Identification of possible problem-solving approaches (methods)

The data set contain more than 2 lakh data with no null values related to the customer. The dataset is imbalanced. Label 1 has 87.5% of data whereas label 0 has approximately 12.5%. As I went through the dataset, I found lot of outliers and skewness are present in the dataset. The skewness was also reduced using power-transformation wherever applicable. There were certain columns which had least importance with our target variable, hence those were dropped. After data cleaning and data transformation, data visualization was done to represent data graphically. At last, the most important part was to build model for the data set.

```
from scipy.stats import zscore
```

```
z=np.abs(zscore(df))
new_df=df[(z<3).all(axis=1)]
```

```
new_df.skew()
```

```
: label                -2.086736
   aon                  0.972525
   daily_decr90         0.290555
   rental90             0.574243
   last_rech_date_ma     3.522300
   last_rech_date_da    10.489713
   last_rech_amt_ma      0.994621
   cnt_ma_rech30        1.141736
   fr_ma_rech30         3.677680
   sumamnt_ma_rech30    2.261745
   medianamnt_ma_rech30 2.114610
   medianmarechprebal30 0.890215
   cnt_ma_rech90        1.293288
   fr_ma_rech90         2.022999
   sumamnt_ma_rech90    1.904794
   medianamnt_ma_rech90 2.180151
   medianmarechprebal90 0.798798
   cnt_da_rech30        50.659837
   fr_da_rech30         52.349775
   cnt_da_rech90        6.820291
   fr_da_rech90         0.000000
   cnt_loans30          1.442689
   maxamnt_loans30      47.544210
   cnt_loans90          11.317192
   amnt_loans90         1.683847
   maxamnt_loans90      2.571450
   medianamnt_loans90   5.781663
   payback30           0.835882
   payback90           0.699226
   Month               0.430553
   Day                 0.201005
dtype: float64
```

```
from sklearn.preprocessing import power_transform
```

```
new_df[['aon', 'last_rech_date_ma', 'last_rech_date_da', 'cnt_ma_rech30', 'fr_ma_rech30', 'sumamnt_ma_rech30', 'medianamnt_ma_rech30', 'cnt_ma_rech90', 'fr_ma_rech90', 'sumamnt_ma_rech90', 'medianamnt_ma_rech90', 'cnt_da_rech30', 'fr_da_rech30', 'cnt_da_rech90', 'fr_da_rech90', 'cnt_loans30', 'maxamnt_loans30', 'cnt_loans90', 'amnt_loans90', 'maxamnt_loans90', 'medianamnt_loans90', 'payback30', 'payback90', 'Month', 'Day']]
```


After removing of outliers using zscore I checked my dataset shape as if how much of data It is been removed. We check this thing because the dataset is highly precious and we cannot lose a high percent of data.

After removing of outliers, I got to that the data is properly scaled so for scaling the values of columns we will be using MinMaxScaler from Sklearn. As it will scale down values of columns in a similar way

```
from sklearn.preprocessing import MinMaxScaler  
sc=MinMaxScaler()  
x=sc.fit_transform(x)
```

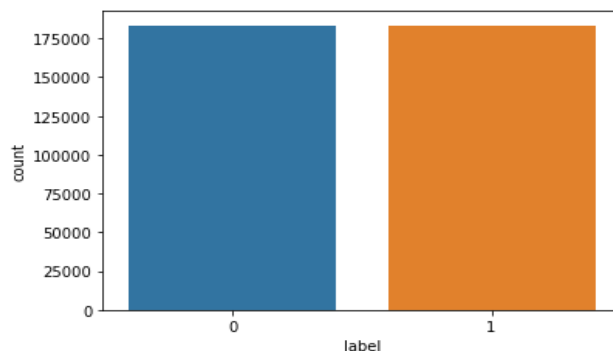
After this we checked that our target variable 'label' is having an imbalanced data which can lead to wrong model testing because our models predict the values which are in majority and tend down to ignore minority class, so we have to balance it by using SMOTE (synthetic minority oversampling technique) is one of the most commonly used oversampling methods to solve the imbalance problem

```
| from imblearn.over_sampling import SMOTE
```

```
| smt=SMOTE()  
| trainx,trainy=smt.fit_resample(dfx,dfy)
```

```
| sn.countplot(trainy)|
```

```
: <AxesSubplot:xlabel='label', ylabel='count'>
```



Testing of Identified Approaches (Algorithms)

Following are the algorithms used for the training and testing: - a. Logistic Regression b. Gaussian NB c. Random Forest Classifier. DecisionTreeClassifier

LogisticRegression

```
LR=LogisticRegression()

LR.fit(x_train,y_train)
LR_predicted=LR.predict(x_test)

print(accuracy_score(y_test,LR_predicted))
print(confusion_matrix(y_test,LR_predicted))
print(classification_report(y_test,LR_predicted))
print("Training accuracy::",LR.score(x_train,y_train))
print("Test accuracy::",LR.score(x_test,y_test))
```

0.7231583091880105
[[35296 14105]
 [13317 36335]]

	precision	recall	f1-score	support
0	0.73	0.71	0.72	49401
1	0.72	0.73	0.73	49652
accuracy			0.72	99053
macro avg	0.72	0.72	0.72	99053
weighted avg	0.72	0.72	0.72	99053

Training accuracy:: 0.7237732861853037
Test accuracy:: 0.7231583091880105

DecisionTreeClassifier

```
from sklearn.tree import DecisionTreeClassifier
dtc=DecisionTreeClassifier()

dtc=DecisionTreeClassifier()
dtc.fit(x_train,y_train)
predddtc=dtc.predict(x_test)
print(accuracy_score(y_test,predddtc))
print(classification_report(y_test,predddtc))
```

0.9037283070679333

	precision	recall	f1-score	support
0	0.90	0.91	0.90	49401
1	0.91	0.90	0.90	49652
accuracy			0.90	99053
macro avg	0.90	0.90	0.90	99053
weighted avg	0.90	0.90	0.90	99053

RandomForestClassifier

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.datasets import make_classification

rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
rfc.score(x_train,y_train)
predrfc=rfc.predict(x_test)
print(accuracy_score(y_test,predrfc))
print(classification_report(y_test,predrfc))
```

0.9445751264474574

	precision	recall	f1-score	support
0	0.95	0.93	0.94	49401
1	0.94	0.95	0.95	49652
accuracy			0.94	99053
macro avg	0.94	0.94	0.94	99053
weighted avg	0.94	0.94	0.94	99053

GaussianNB

```
from sklearn.naive_bayes import GaussianNB
```

```
gnb = GaussianNB()
gn_model1 = gnb.fit(x_train,y_train)
gn_pred1 = gnb.predict(x_test)
```

```
gn_CR1 = classification_report(y_test,gn_pred1)
gn_acc1 = accuracy_score(y_test,gn_pred1)

print('classification report ', '\n', gn_CR1)
print('accuracy score: ', '\n', gn_acc1)
```

```
classification report
              precision    recall  f1-score   support

     0               0.65       0.89       0.75       49401
     1               0.83       0.53       0.65       49652

 accuracy
macro avg       0.74       0.71       0.70       99053
weighted avg    0.74       0.71       0.70       99053

accuracy score:
0.7098321100824811
```

Cross validation

```
print(cross_val_score(dtc,x,y,cv=5).mean())
0.8840562429351294
```

```
print(cross_val_score(svc,x,y,cv=5).mean())
```

```
print(cross_val_score(rfc,x,y,cv=5).mean())
0.9207464040160307
```

```
print(cross_val_score(LR,y,cv=5).mean())
nan
```

```
print(cross_val_score(gnb,y,cv=5).mean())
nan
```

After passing different models in our train and test data we found the best model which is working for our dataset is Random Forest Classifier because from among all models it is giving me best precision, F1-score.

Key Metrics for success in solving problem under consideration

Passing GridSearchCV to our models we get best parameters. This is a time taking process for finding the best parameters because GridSearchCV is passed throughout the dataset with model for fetching out best params.

After we get our desired parameters for our models, we will be implementing it into our models for best accuracy score, precision, F1 score, ROC AUC curve, confusion matrix.

```

from sklearn.model_selection import GridSearchCV
import numpy as np

parameter={ 'criterion' : ["gini", "entropy"],
            'n_estimators':[200,50],
            'min_samples_split':[2,3],
            'random_state':[10]
          }

GCV=GridSearchCV(RandomForestClassifier(),parameter,cv=5)

GCV.fit(x_train,y_train)

GridSearchCV(cv=5, estimator=RandomForestClassifier(),
             param_grid={ 'criterion': ['gini', 'entropy'],
                           'n_estimators': [200, 50],
                           'min_samples_split': [2, 3],
                           'random_state': [10]})

GCV.best_params_

{'criterion': 'entropy',
 'min_samples_split': 2,
 'n_estimators': 200,
 'random_state': 10}

```

These are the best parameters

```

final_model=RandomForestClassifier(criterion='entropy')
final_model.fit(x_train,y_train)
pred=final_model.predict(x_test)
accuracy=accuracy_score(y_test,pred)
print (accuracy*100)

94.53121056404147

```

Plotting AOC RUC CURVE

```

from sklearn.metrics import roc_curve
import matplotlib.pyplot as plt
from sklearn.metrics import roc_auc_score

y_pred_prob=rfc.predict_proba(x_test)[:,-1]

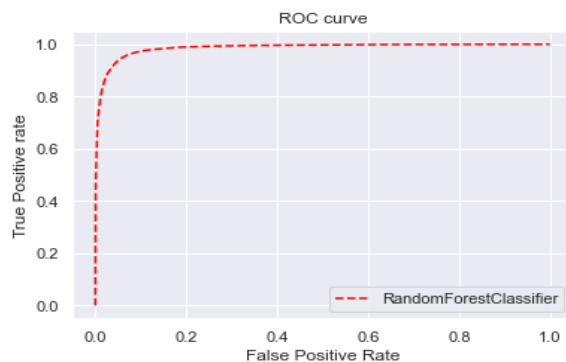
fpr,tpr,thresholds=roc_curve(y_test,y_pred_prob,pos_label=1)

sns.set_theme(style="darkgrid")
plt.plot(fpr, tpr, linestyle='--',color='red', label='RandomForestClassifier')

plt.title('ROC curve')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive rate')
plt.legend(loc='best')

```

]: <matplotlib.legend.Legend at 0x1b783ba9820>

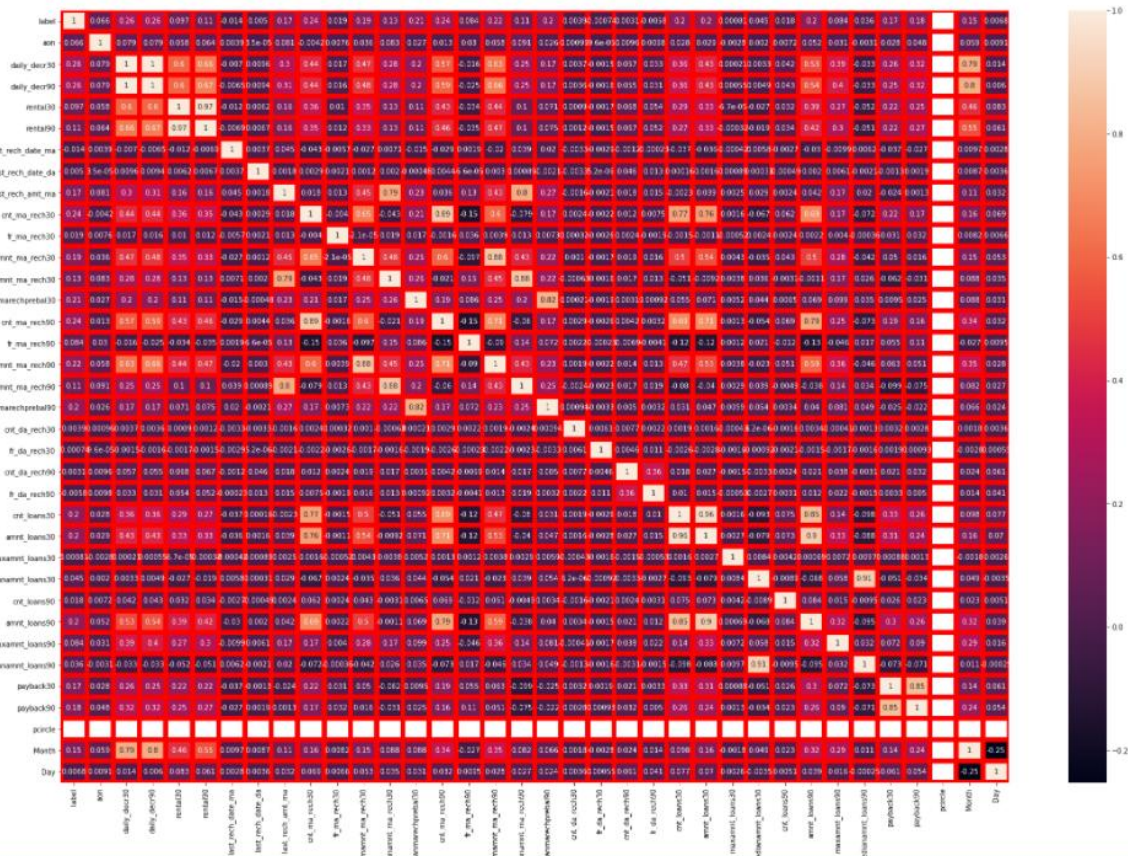


Visualizations

The plots used to visualize the data are :- a. Heatmap b. Count plot c. Dist plot d. Hist plot

```
plt.figure(figsize=[30,20])
sn.heatmap(cor,annot=True,linewidths=6,linecolor='r')
```

<AxesSubplot: >



Observation:

1-daily_decr30 and daily_decr90 features are highly correlated with each other.

2-rental30 and rental90 features are highly correlated with each other.

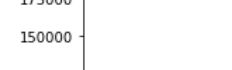
3-cnt_loans30 and amount_loans30 columns are highly correlated with each other.

4-amount_loans30 is also highly correlated with amount_loans90 column.

5-medianamnt_loans30 and medianamnt_loans90 is highly correlated with each other.

6-We have to drop one of the features which are highly correlated with other features. And if we don't do this then our model will face multicollinearity problem

```
sn.countplot(df["label"])
```



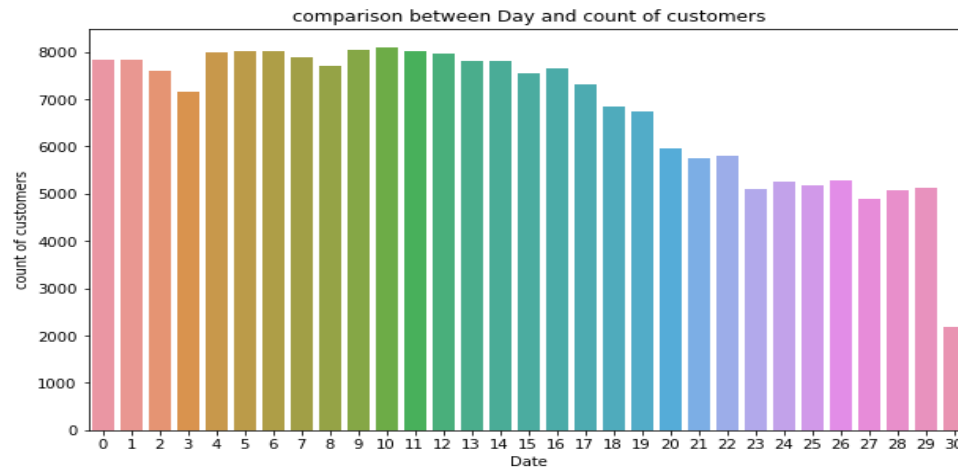
label	count
0	25000
1	180000

```
plt.figure(figsize=(18,15))
for i in enumerate(df.columns):
    plt.subplot(8,5,i[0]+1)
    sn.distplot(df[i[1]],color='g')
```

Checked the skewness of the each columns

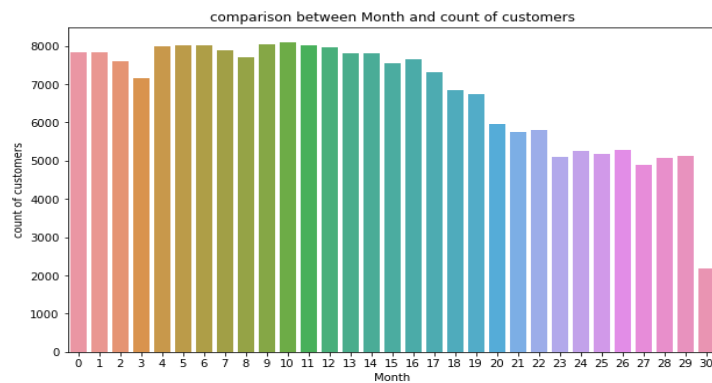
```
plt.figure(figsize=[10,6])
plt.title("comparison between Day and count of customers")
sn.countplot(df["Day"])
plt.xlabel("Date")
plt.ylabel("count of customers")
```

Text(0, 0.5, 'count of customers')



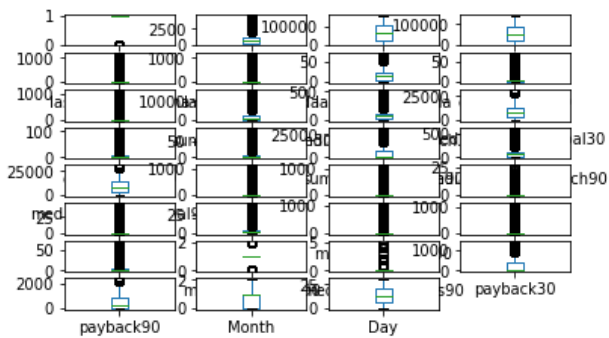
```
plt.figure(figsize=[10,6])
plt.title("comparison between Month and count of customers")
sn.countplot(df["Day"])
plt.xlabel("Month")
plt.ylabel("count of customers")
```

Text(0, 0.5, 'count of customers')



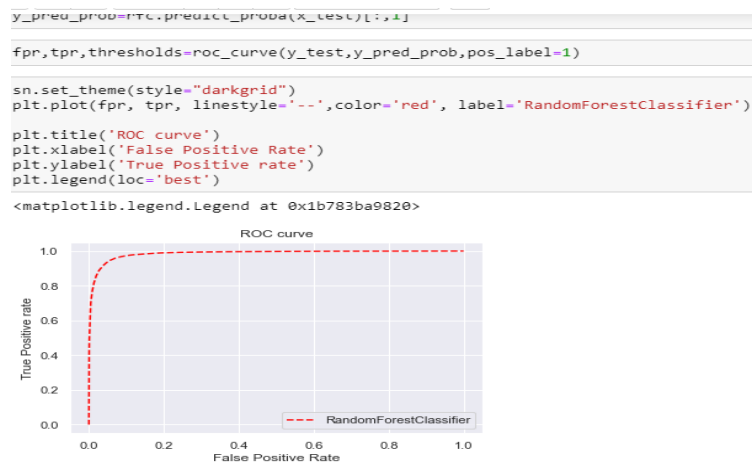
```
df.plot(kind="box", subplots=True, layout=(9,4))
```

label	AxesSubplot(0.125,0.808774;0.168478x0.0712264)
aon	AxesSubplot(0.327174,0.808774;0.168478x0.0712264)
daily_decr90	AxesSubplot(0.529348,0.808774;0.168478x0.0712264)
rental90	AxesSubplot(0.731522,0.808774;0.168478x0.0712264)
last_rech_date_ma	AxesSubplot(0.125,0.723302;0.168478x0.0712264)
last_rech_date_da	AxesSubplot(0.327174,0.723302;0.168478x0.0712264)
last_rech_amt_ma	AxesSubplot(0.529348,0.723302;0.168478x0.0712264)
cnt_ma_rech30	AxesSubplot(0.731522,0.723302;0.168478x0.0712264)
fr_ma_rech30	AxesSubplot(0.125,0.63783;0.168478x0.0712264)
sumamnt_ma_rech30	AxesSubplot(0.327174,0.63783;0.168478x0.0712264)
medianamnt_ma_rech30	AxesSubplot(0.529348,0.63783;0.168478x0.0712264)
medianamnt_rechprebal30	AxesSubplot(0.731522,0.63783;0.168478x0.0712264)
cnt_ma_rech90	AxesSubplot(0.125,0.552358;0.168478x0.0712264)
fr_ma_rech90	AxesSubplot(0.327174,0.552358;0.168478x0.0712264)
sumamnt_ma_rech90	AxesSubplot(0.529348,0.552358;0.168478x0.0712264)
medianamnt_ma_rech90	AxesSubplot(0.731522,0.552358;0.168478x0.0712264)
medianamnt_rechprebal90	AxesSubplot(0.125,0.466887;0.168478x0.0712264)
cnt_da_rech30	AxesSubplot(0.327174,0.466887;0.168478x0.0712264)
fr_da_rech30	AxesSubplot(0.529348,0.466887;0.168478x0.0712264)
cnt_da_rech90	AxesSubplot(0.731522,0.466887;0.168478x0.0712264)
fr_da_rech90	AxesSubplot(0.125,0.381415;0.168478x0.0712264)
cnt_loans30	AxesSubplot(0.327174,0.381415;0.168478x0.0712264)
maxamnt_loans30	AxesSubplot(0.529348,0.381415;0.168478x0.0712264)
cnt_loans90	AxesSubplot(0.731522,0.381415;0.168478x0.0712264)
amnt_loans90	AxesSubplot(0.125,0.295943;0.168478x0.0712264)
maxamnt_loans90	AxesSubplot(0.327174,0.295943;0.168478x0.0712264)
medianamnt_loans90	AxesSubplot(0.529348,0.295943;0.168478x0.0712264)
payback30	AxesSubplot(0.731522,0.295943;0.168478x0.0712264)
payback90	AxesSubplot(0.125,0.210472;0.168478x0.0712264)
Month	AxesSubplot(0.327174,0.210472;0.168478x0.0712264)
Day	AxesSubplot(0.529348,0.210472;0.168478x0.0712264)
dtype: object	



Checked the outliers of the columns

Interpretation of the Results



After performing all 4 models I found that will give best results I came to know that RandomForestClassifier is giving me best result in every criterion

From the dataset, it was clear that most of the customers are inclined to pay the loan as 95% of the customer repaid it and only 5% of the customers are defaulter.

CONCLUSION

Key Findings and Conclusions of the Study

Mostly, the customers have the intension of repaying. There are certain cases, when the customers have no intension of repayment but the number of such customers are few. With the model built, we can certainly determine customers having intension of repayment or not.

In past, micro credit and their institution were less as it could not cover the whole population who were facing issues in credit taking. There was no proper financial map planned to cover these low-income groups. As we know first this was practiced by economist Muhammad Yunus. This system started in Bangladesh in 1976, with a group of women borrowing \$27 to finance the group's own small businesses. The women repaid the loan and were able to sustain the business. Now various financial institutions, banks, NGO's are coming up with great credit facilities in many sectors such as agriculture, small-scale business, telecom service etc.

Role of Government

MFI's should be controlled and monitored by government and some laws against them.

- The ministry must enforce strong policies, strategies, laws and regulations that enhances introduction of enough microfinance institutions to influence companions in financial sectors in order to lower interest rate and other cost of borrowing.

- The ministry should take up regular monitoring and evaluation in MFIs on credit compliance and loan disbursement.

- The ministry must reinforce effective loan application and processing procedures to Influence timely loans disbursement for instance from the current two-week period to a minimum of three days so that SMEs can access such facilities that require capital to accomplish.

Recommendations to MFI's

MFI's should ensure that they promote timely disbursement of loans. Timely access to capital is very crucial to the growth of sme's because the challenge of capital underlies most of the challenges rural SMEs face.

This micro credit facility is very good for the upliftment of the society and give the needy ones an opportunity to stand and carry on their life with the help of such small loan provided by them. It is still in growing phase there must be more capital and respected sectors for recording and providing these facilities to humongous low-income population.

Learning Outcomes of the Study in respect of Data Science

The dataset was full of outliers, skewness and unbalanced data which was the biggest challenge to overcome. Hence data cleaning was very important to get proper prediction. I have used Logistic Regression, Gaussian NB and Random Forest Classifier. Among the three algorithms Random Forest Classifier gave the best outcome. As the dataset was unbalanced, the other algorithm may overfit and can come out with wrong prediction whereas Random forest can control overfitting and give best prediction.

During visualization I came to know that the dataset is huge which cannot be plotted by using histograms as the label we getting messed up. It was a bit difficult so I went on internet to know what all available options I have after that i tried some techniques but then after a lot research, I used univariate analysis by showing value counts of our target variable 'label'. After that I went doing bivariate analysis using catplot, factor plot, boxplot & distplot in visualizing the data which showed me how data is spread and weather columns are skewed our not and it also showed me the negative values present in many columns. I used boxplot for showing the outliers in the dataset.

During data cleaning first thing was the dataset was having negative values to make it right I used abs function which made my negative values to positive.

I saw that there were 4 columns which were of no use for our machine learning model those columns name was msisdn, pdate, pcircle, Unnamed: 0. So I came to decision to drop these 4 columns for dataset.

For removing outliers, I used zscore function, for removing of skewness I went for power transform function and after that for scaling down my dataset columns values in respect to each other I used MinMaxScaler function.

At last, when I was splitting my target variable from training dataset, I found that my target variable is imbalanced so without balancing my target variable all my models will result in mess which will be of no use. If I would talk about problem faced during this whole process was that GridSearchCV which is a long-time taking function although it gives a best result to our models' performance. But to find parameters of one single model it took average 4 to 5 hr time to run. So, you cannot hit and try every other model twice or thrice it is a time-consuming process. I can say our client will not give us leverage of time we have to give him what he wants in a certain period of time.

Limitations of this work and Scope for Future Work

The solution can be applied to the customer having a transaction history but the model may not perform well with customer having new profile and no transaction history. Nevertheless, the model will perform well with customer having transaction history and can predict whether a person will be a defaulter or non-defaulter. Hence, we can say that this statistical model will be helpful in future for the prediction of micro credit defaulter and non-defaulter customer.

As we are in living in the world where small thing is done there is a record of it which is in data to understand there comes the role of data scientist. We worked on micro credit defaulter model which is of telecom industry based in Indonesia. we used this data for our machine learning, Artificial Intelligence. For more improvement I would say the data should be properly organised and maintained.

And as we are in the learning stage yes it can be more improved by performing different techniques and models

THANK YOU