Micro Credit Defaulter Model

A Step-by-Step guide to Build a Machine Learning Model to predict Loan defaulter

Submitted By:

Razni Nazeem

30.08.2022

FlipROBO Technologies

FLIP ROBO

ACKNOWLEDGEMENT

I would like to express my special gratitude to the “Flip Robo” team, who has given me this opportunity to deal with a beautiful dataset and it has helped me to improve my analyzing skills. Also, I want to express my huge gratitude to Ms. Khushboo Garg (SME FlipRobo), who has helped me overcome difficulties within this project and others.

I would also like to thank various websites like stackoverflow, Kaggle, medium and towardsDatascience for helping me resolve any issues I face during my project.

A huge thanks to my academic team “Data trained” who has helped me grow from a non-Coder to what I am Now. Lastly, I would like to extend my Heartfelt thanks to my Husband and kids because without their support this project would not have been successful. And thank you to many other persons who have helped me directly or indirectly to complete the project.

CONTENTS

1. INTRODUCTION

1.1 Business Problem Framing:

A Microfinance Institution (MFI) is an organization that offers financial services to low income populations. MFS becomes very useful when targeting especially the unbanked poor families living in remote areas with not much sources of income. The Microfinance services (MFS) provided by MFI are Group Loans, Agricultural Loans, Individual Business Loans and so on.

Many microfinance institutions (MFI), experts and donors are supporting the idea of using mobile financial services (MFS) which they feel are more convenient and efficient, and cost saving, than the traditional high-touch model used since long for the purpose of delivering microfinance services. Though, the MFI industry is primarily focusing on low income families and are very useful in such areas, the implementation of MFS has been uneven with both significant challenges and successes.

Today, microfinance 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.

1.2 Conceptual Background of the Domain Problem:

Telecom Industries 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 have to build a model which can be used to 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. In this case, Label '1' indicates that the loan has been payed i.e. Non-defaulter, while, Label '0' indicates that the loan has not been payed i.e. defaulter.

1.3 Review of Literature

Microfinance in India has seen incredible growth in the last two decades in terms of the number of Microfinance customers and institutions offering microfinance. An attempt has been made in this report to review the available literature in microfinance. Approaches to microfinance, issues related to measuring social impact versus profitability of MFIs, issue of sustainability, variables impacting sustainability, effect of regulations of profitability and impact assessment of MFIs have been summarized in the below report. We hope that the below report of literature will provide a platform for further research and help the industry to combine theory and practice to take microfinance forward and contribute to alleviating the poor from poverty. The findings and recommendations of existing research work can help in identifying appropriate ML algorithms for credit assessment specifically for rural borrowers. This study can also help fintech startups, banking and non-banking financial institutions in India to develop more financially inclusive products.

1.4 Motivation for the Problem Undertaken:

I have to model the micro credit defaulters with the available independent variables. This model will then be used by the management to understand how the customer is considered as defaulter or non-defaulter based on the independent variables. They can accordingly manipulate the strategy of the firm and concentrate on areas that will yield high returns. Further, the model will be a good way for the management to understand whether the customer will be paying back the loaned amount within 5 days of insurance of loan. The relationship between predicting defaulter and the economy is an important motivating factor for predicting micro credit defaulter model. Furthermore, studies have indicated that analysis of social media data can also help in building predictive models for rural settings where there has been a penetration of mobile applications and online banking applications. Lastly, some research methodologies have also explored deep learning models such as artificial neural networks (ANN) for micro-credit score prediction.

2. Analytical Problem Framing

2.1 Mathematical/Analytical Modeling of the Problem

In this particular problem I had label as my target column and it was having two classes Label '1' indicates that the loan has been payed i.e., non-defaulter, while Label '0' indicates that the loan has not been payed i.e., defaulter. So clearly it is a binary classification problem, and I must use all classification algorithms while building the model. There were no null values in the dataset. Also, I observed some unnecessary entries in some of the columns like in some columns I found more than 90% zero values, so I decided to drop those columns. If I keep those columns as it is, it will create high skewness in the model. To get better insight on the features I have used plotting like distribution plot, bar plot and count plot. With these plotting I was able to understand the relation between the features in better manner. Also, I found outliers and skewness in the dataset, so I removed outliers using percentile method and I removed skewness using yeo-Johnson method. I have used all the classification algorithms while building model then tunned the best model and saved the best model. At last, I have predicted the label using saved model.

2.2 Data Sources and their formats:

The data was collected for my internship company – Flip Robo technologies in excel format. The sample data is provided to us from our client database. It is hereby given to us for this exercise. To improve the selection of customers for the credit, the client wants some predictions that could help them in further investment and improvement in selection of customers.

Also, my dataset was having 209593 rows and 36 columns including target. In this particular dataset, I have object, float, and integer types of data. The information about features is as follows.

Features Information:

1. label : Flag indicating whether the user paid back the credit amount within 5 days of issuing the loan{1:success, 0:failure}
2. msisdn : mobile number of user
3. aon : age on cellular network in days
4. daily_decr30 : Daily amount spent from main account, averaged over last 30 days (in Indonesian Rupiah)
5. daily_decr90 : Daily amount spent from main account, averaged over last 90 days (in Indonesian Rupiah)
6. rental30 : Average main account balance over last 30 days
7. rental90 : Average main account balance over last 90 days
8. last_rech_date_ma : Number of days till last recharge of main account
9. last_rech_date_da: Number of days till last recharge of data account
10. last_rech_amt_ma : Amount of last recharge of main account (in Indonesian Rupiah)
11. cnt_ma_rech30 : Number of times main account got recharged in last 30 days
12. fr_ma_rech30 : Frequency of main account recharged in last 30 days
13. sumamnt_ma_rech30 : Total amount of recharge in main account over last 30 days (in Indonesian Rupiah)
14. medianamnt_ma_rech30 : Median of amount of recharges done in main account over last 30 days at user level (in Indonesian Rupiah)
15. medianmarechprebal30 : Median of main account balance just before recharge in last 30 days at user level (in Indonesian Rupiah)
16. cnt_ma_rech90 : Number of times main account got recharged in last 90 days
17. fr_ma_rech90 : Frequency of main account recharged in last 90 days
18. sumamnt_ma_rech90 : Total amount of recharge in main account over last 90 days
19. medianamnt_ma_rech90 : Median of amount of recharges done in main account over last 90 days at user level (in Indonasian Rupiah)
20. medianmarechprebal90 : Median of main account balance just before recharge in last 90 days at user level (in Indonasian Rupiah)
21. cnt_da_rech30 : Number of times data account got recharged in last 30 days
22. fr_da_rech30: Frequency of data account recharged in last 30 days
23. cnt_da_rech90 : Number of times data account got recharged in last 90 days
24. fr_da_rech90 : Frequency of data account recharged in last 90 days
25. cnt_loans30 : Number of loans taken by user in last 30 days
26. amnt_loans30: Total amount of loans taken by user in last 30 days
27. maxamnt_loans30 : maximum amount of loan taken by the user in last 30 days
28. medianamnt_loans30 : Median of amounts of loan taken by the user in last 30 days
29. cnt_loans90 : Number of loans taken by user in last 90 days
30. amnt_loans90 : Total amount of loans taken by user in last 90 days
31. maxamnt_loans90 : maximum amount of loan taken by the user in last 90 days
32. medianamnt_loans90 : Median of amounts of loan taken by the user in last 90 days
33. payback30 : Average payback time in days over last 30 days
34. payback90 : Average payback time in days over last 90 days
35. pcircle : telecom circle
36. pdate : date

2.3 Data Processing done

* Imported necessary Libraries and loaded the dataset.

* Statistical Analysis done like Shape, info, nunique, value_counts.

* Dropped columns with more than 90% of Zero values.

* No Null values in the dataset.

* Dropped column- Unnamed:0, msisdn and pcircle which had all unique or 1 unique value.

* Feature extraction done in pdate and extracted pday, pmonth and pyear.

* Converted few columns with negative values to positive values.

* Checked Correlation between Variables and Feature

* Outliers removed, Encoded categorical columns.

* Skewness removed, removed columns with Multicollinearity issue.

* Using Min-Max Scaler, normalized the data.

2.4 Data Inputs- Logic- Output Relationships

- Since I had all numerical columns, I have plotted dist plot to see the distribution of each column data.
- I have used box plot for each pair of categorical features that shows the relation between label and independent features. Also we can observe whether the person pays back the loan within the date based on features.
- In maximum features relation with target, I observed Non-defaulter count is high compared to defaulters.

3 Hardware and Software Requirements and Tools Used

* Hardware: Processor- Core i5 and above, RAM- 8GB or above, SSD- 250 or above

* Software: Anaconda

* Libraries Used mentioned before.

There are 3 sets of libraries used.

- Basic Libraries for Data Analysis and Visualization
- Libraries for Data Cleaning and Feature Engineering (Data preprocessing)
- Libraries for Building the ML Models

Basic Libraries used are:

```
#importing required libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import datetime as dt

import warnings
warnings.filterwarnings('ignore')
```

Libraries for Data Cleaning and Feature Engineering (Data preprocessing):

```
#for Outliers removal, z-score is used
from scipy.stats import zscore

#for Skewness removal, Poer transformer is used
from sklearn.preprocessing import PowerTransformer

#for rncoding Ordinal encoder is used
from sklearn.preprocessing import OrdinalEncoder

#for normalizing, standard scaler is used
from sklearn.preprocessing import StandardScaler

#for checking multicollinearity, VIF Factor is used
from statsmodels.stats.outliers_influence import variance_inflation_factor
```

Libraries for Building the ML Models:

```
#Importing regression libraries

from sklearn.linear_model import LinearRegression, Lasso, Ridge
from sklearn.neighbors import KNeighborsRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.linear_model import ElasticNet
from sklearn.svm import SVR
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import GradientBoostingRegressor
from xgboost import XGBRegressor

from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import r2_score, mean_absolute_error, mean_squared_error
```

These are the libraries used in my Jupyter notebook for Prediction.

3. MODEL DEVELOPMENT AND EVALUATION

3.1 Identification of possible problem-solving approaches

To remove outliers, I have used percentile method. And to remove skewness I have used yeo-Johnson method. We have dropped all the unnecessary columns in the dataset according to our understanding. Use of Pearson's correlation coefficient to check the correlation between dependent and independent features. Also, I have used Normalization to scale the data. After scaling we must balance the target column using oversampling. Then followed by model building with all Classification algorithms. I have used oversampling (SMOTE) to get rid of data imbalancing. The balanced output looks like this.

```
In [59]: # Checking the value counts again

         y.value_counts()

Out[59]: 0    181388
         1    181388
         Name: label, dtype: int64
```

3.2 Testing of Identified Approaches (Algorithms)

Since label was my target and it was a classification column with 0-defaulter and 1-Non-defaulter, so this particular problem was Classification problem. And I have used all Classification algorithms to build my model. By looking into the difference of accuracy score and cross validation score I found RandomForestClassifier as a best model with least difference. Also to get the best model we have to run through multiple models and to avoid the confusion of overfitting we have go through cross validation. Below are the list of classification algorithms I have used in my project.

- **Decision Tree Model**
- **Random Forest Classifier**
- **KNN Classifier**
- **XGB Classifier**

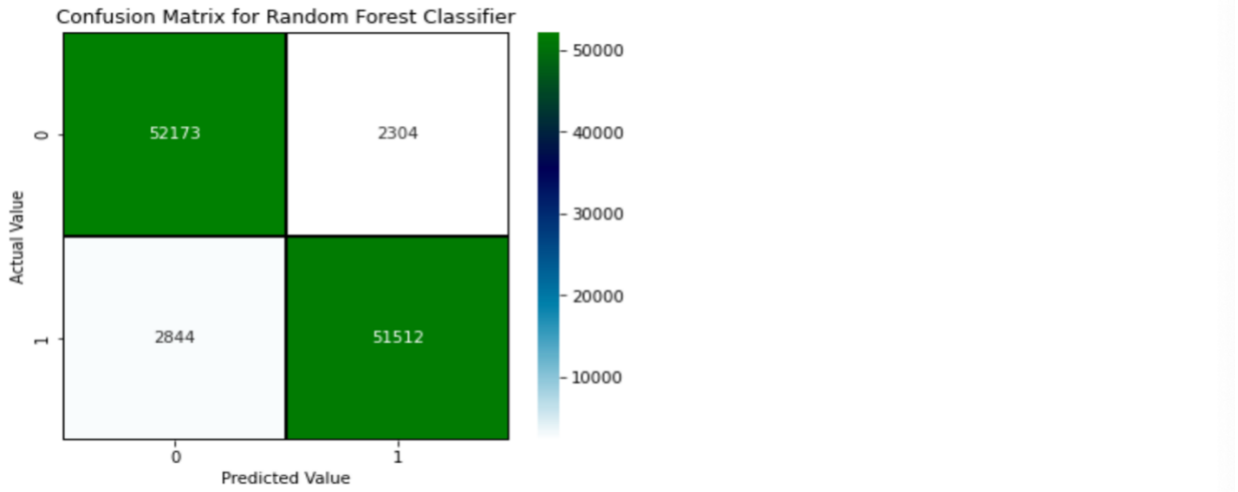
3.3 Run and evaluate selected models

A. Decision Tree Classifier Model:

I found the best random state which yields the best score and then fit the model. Accuracy score for Decision Tree Model was 91.57%. I have tuned with different parameters and the score has decreased to 80.86%. The CV Score of Decision tree model without tuned parameters are 91.14%. The Confusion Matrix of actual and predicted values using Decision Tree model is:

B. Random Forest Regressor (Best Model):

I first found the best random state and fit the model. I got a score of 95.26 % and a CV Score of 94.92%. Hence, I selected the random forest as the best model and saved the model. While performing the fitting of the final model, my score was improved, and it became **95.59%**.



The confusion Matrix shows 2844 False Negatives and 2304 False Positives. But that's comparatively better than other models.

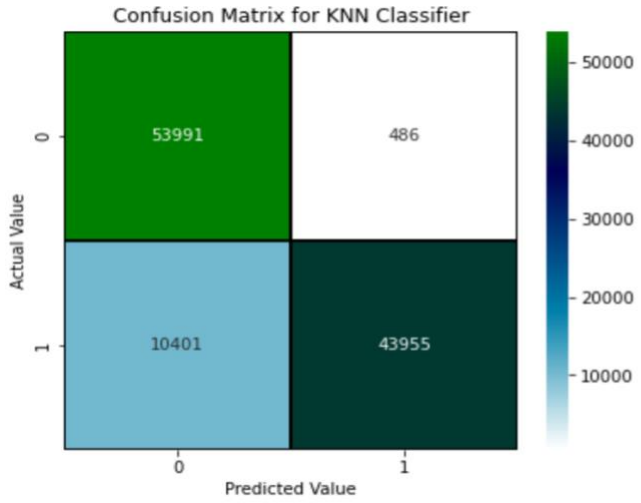
The CV Score:

```
score=cross_val_score(rf,x,y,cv=5)
score_d=score.mean()
print("Cross_Val_Score of RF:",score_d)
```

```
Cross_Val_Score of RF: 0.9492939788403951
```

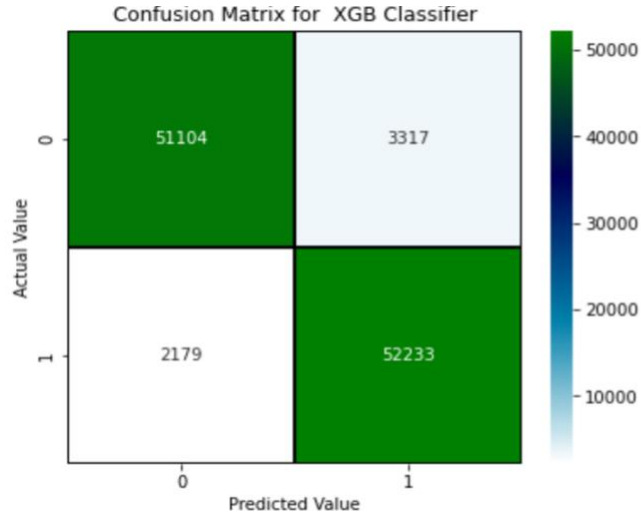
C. KNN Classifier:

For KNN Model, the Accuracy score is 89.99%. The CV Score for the KNN Model is 90.40%. Confusion matrix of the KNN Model is:



D. XGB Classifier:

The Accuracy score of XGB Model is 94.95% and CV Score of 93.63%.

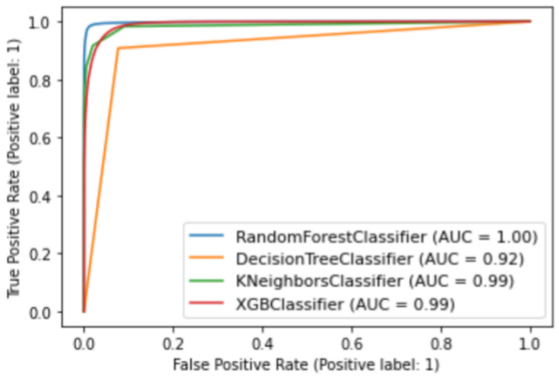



```
score=cross_val_score(XGB,x,y,cv=5)
score_g=score.mean()
print("Cross_Val_Score of XGB Classifier:",score_g)
```

```
Cross_Val_Score of XGB Classifier: 0.9363688597735809
```

AUC_ROC Curve:

```
In [79]: # Plotting AUC-ROC Curve for all the models used here

from sklearn.metrics import plot_roc_curve

disp = plot_roc_curve(rf,x_test,y_test)
plot_roc_curve(dtc,x_test, y_test, ax=disp.ax_)
plot_roc_curve(knn, x_test, y_test, ax=disp.ax_)
plot_roc_curve(XGB, x_test, y_test, ax=disp.ax_)

plt.legend(prop={'size':11}, loc='lower right')
plt.show()
```



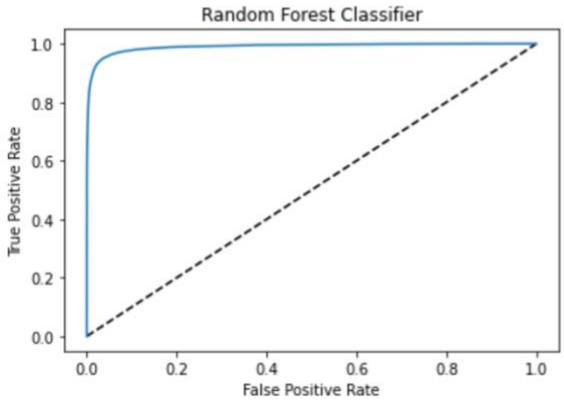
From all above 3 plots we can clearly see that random forest is the best model.

We can see clearly that AUC-ROC curve for random Forest is perfect and so I finally decided to take random forest as my final Model.

AUC-ROC Curve of Final Model:

Our AUC-ROC Curve of the final model after hyper parameter tuning is below. I have taken FPR, TPR and Thresholds and then found the AUC_ROC Curve for the same.

```
In [87]: plt.plot([0,1],[0,1],'k--')
plt.plot(fpr,tpr)
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Random Forest Classifier')
plt.show()
```



Saving the Model:

```
import joblib
joblib.dump(Final_Model,'Micro Credit RF Model.pkl')
```

```
['Micro Credit RF Model.pkl']
```

Predictions:

```
# Loading the saved model
model=joblib.load("Micro Credit RF Model.pkl")

#Prediction
prediction = model.predict(x_test)
prediction
```

```
array([1, 1, 0, ..., 0, 1, 0])
```

```
#saving as dataframe

base = pd.DataFrame()
base["actual"] = y_test
base["predictions"] = prediction
base
```

	actual	predictions
163460	1	1
8928	1	1
248038	0	0
137081	0	0
310285	0	0
...	...	...
245188	0	0
168885	1	1
219511	0	0
189838	1	1
350006	0	0

108833 rows x 2 columns

```
#Adding another column of thier difference.

base['difference']=base['actual']-base['predictions']

#If 0 then actual and predicted are same. else its different

print(base['difference'].value_counts())
```

```
 0    104043
-1      2491
 1      2299
Name: difference, dtype: int64
```

```
a=(4790/104043)*100
a
```

```
4.603865709370164
```

As per our Model, only 4% of the result is wrong and rest all are true values.

3.4 Key Metrics for success in solving problem under consideration

I have used the following metrics for evaluation:

Precision can be seen as a measure of quality, higher precision means that an algorithm returns more relevant results than irrelevant ones.

Recall is used as a measure of quantity and high recall means that an algorithm returns most of the relevant results.

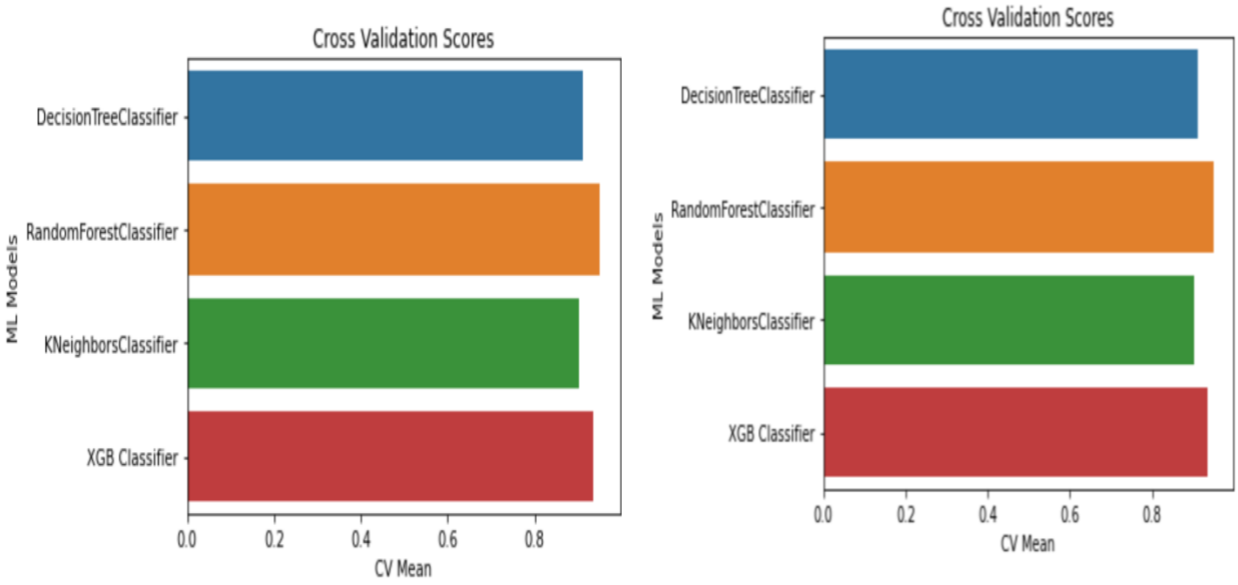
Accuracy score is used when the True Positives and True negatives are more important. Accuracy can be used when the class distribution is similar.

F1-score is used when the False Negatives and False Positives are crucial. While F1-score is a better metric when there are imbalanced classes.

Cross_val_score: To run cross-validation on multiple metrics and also to return train scores, fit times and score times. Get predictions from each split of cross-validation for diagnostic purposes. Make a scorer from a performance metric or loss function.

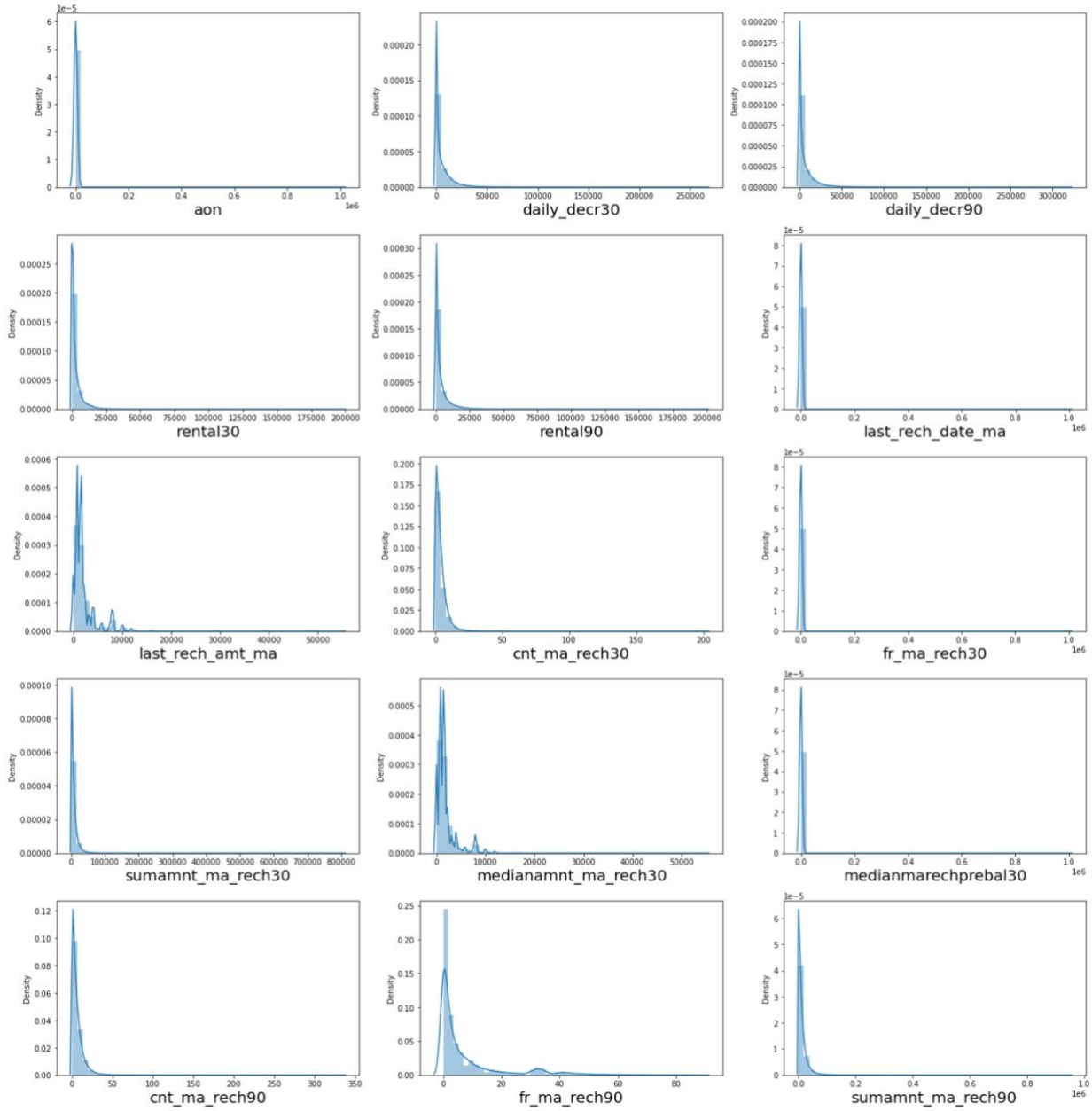
AUC_ROC _score: ROC curve. It is a plot of the false positive rate (x-axis) versus the true positive rate (y-axis) for a number of different candidate threshold values between 0.0 and 1.0

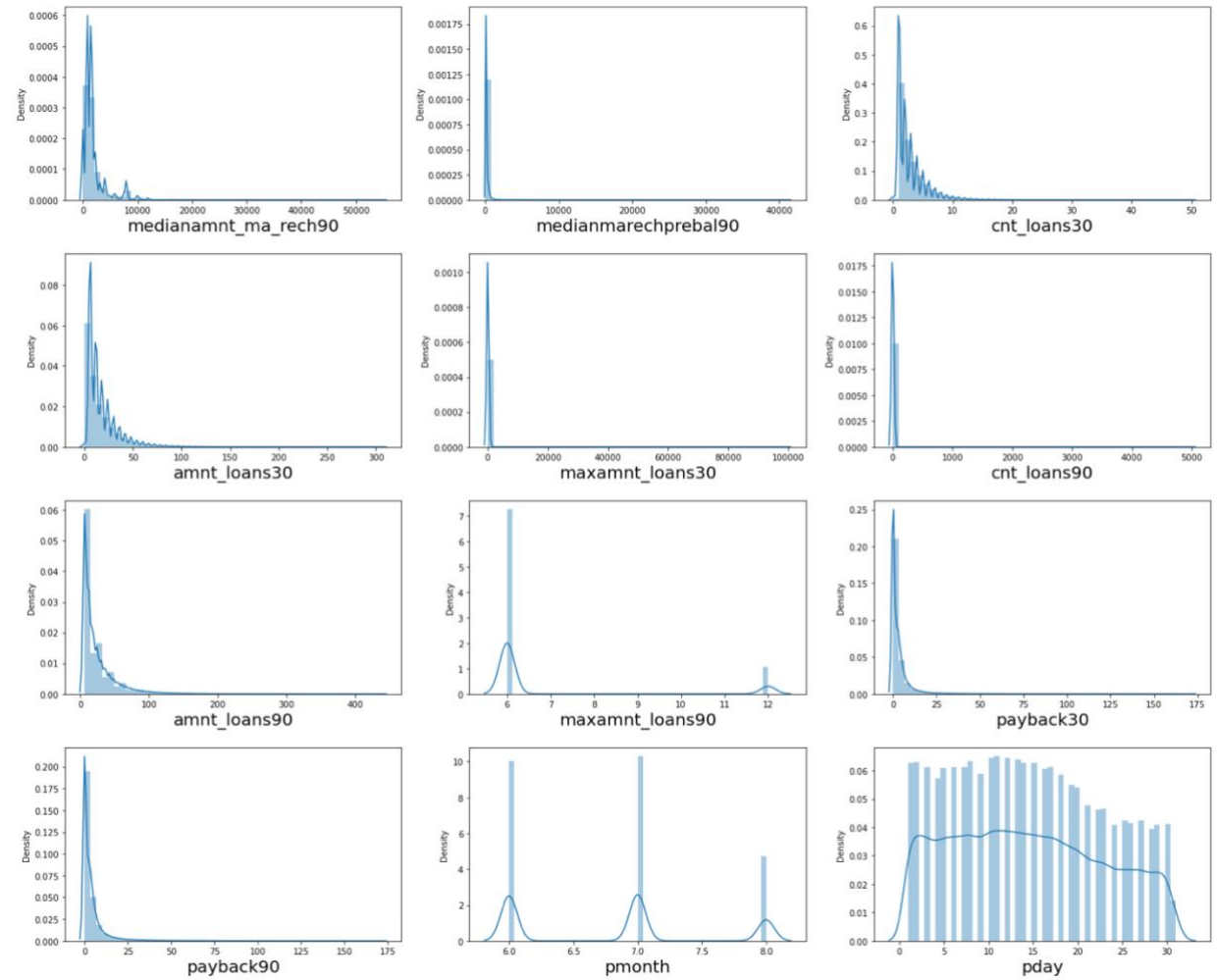
I have used accuracy score since I have balanced my data using SMOTE Technique.



3.4 Visualizations

- **Univariate analysis:**



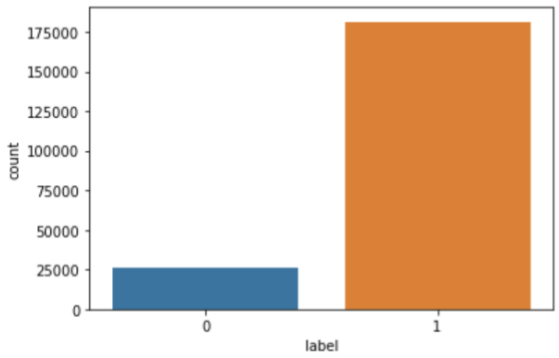


Observations:

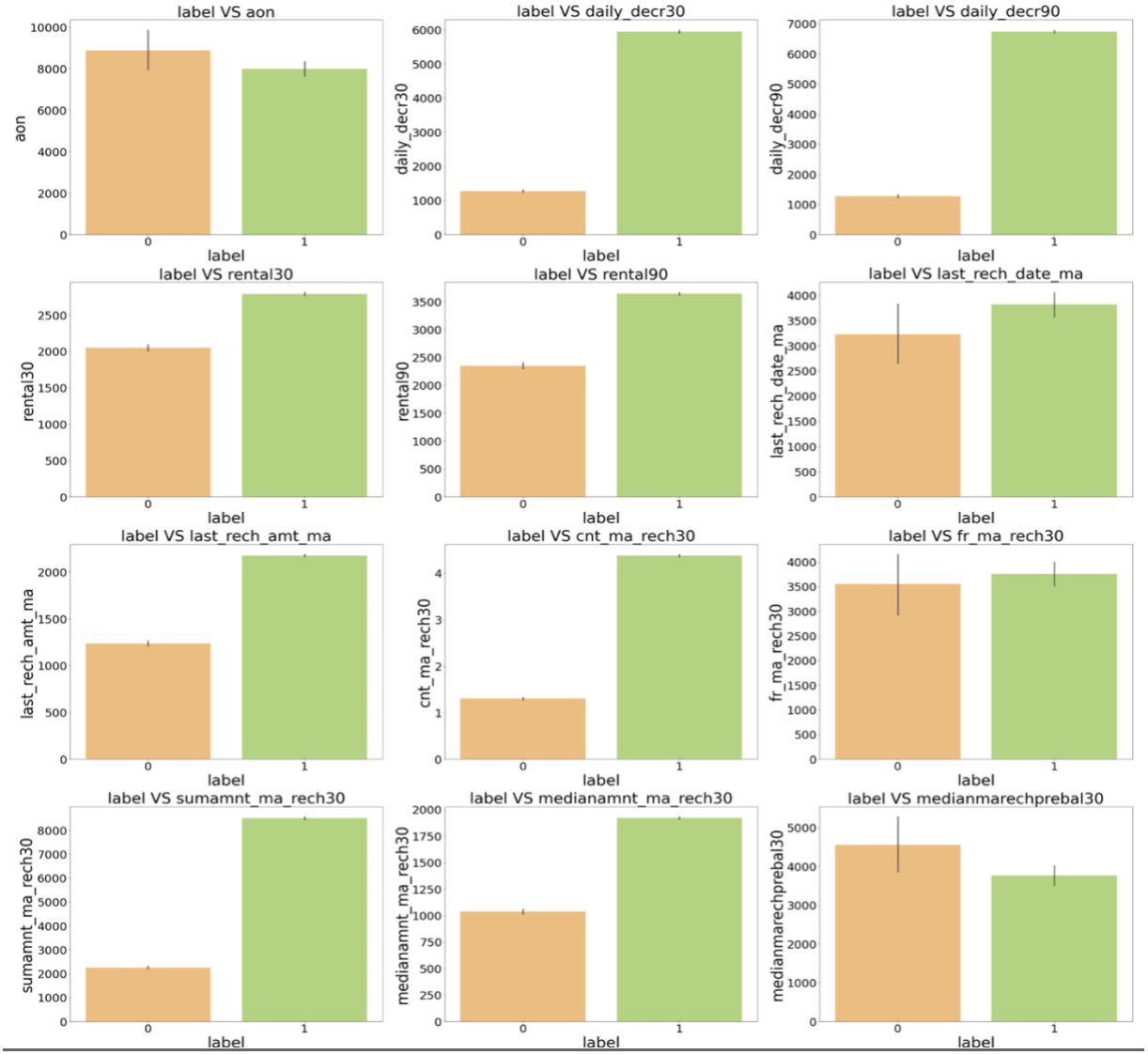
- Almost all columns are right skewed.
- We have symmetrical distribution for the pday and pmonth columns.
- Count plot of Target column

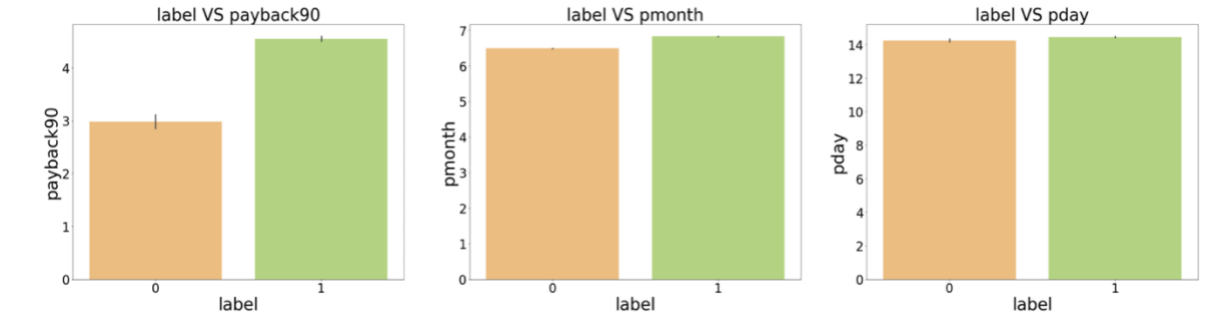
```
#count plot for target column

sns.countplot(df['label'])
```

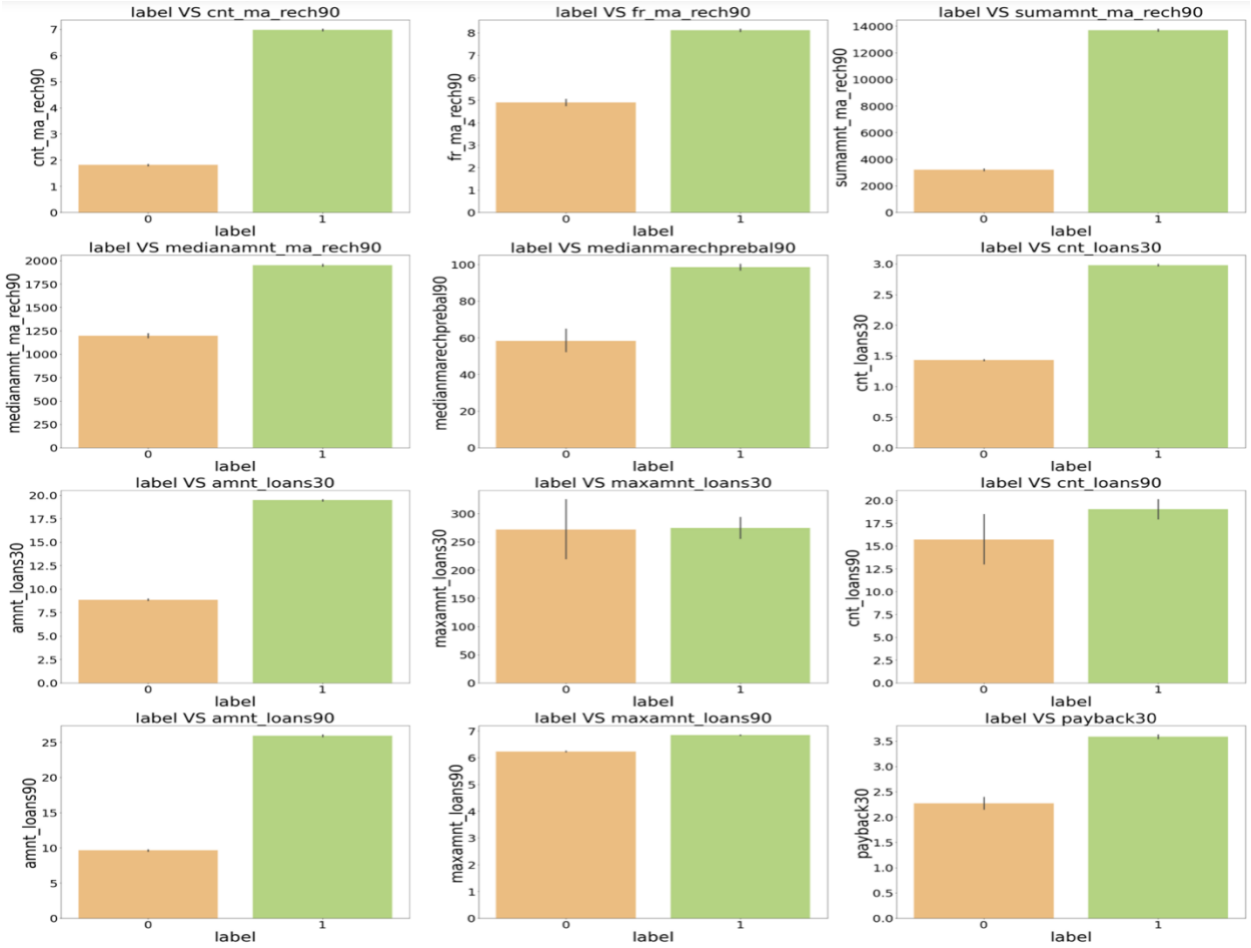
```
<AxesSubplot:xlabel='label', ylabel='count'>
```



- **Bivariate analysis:**







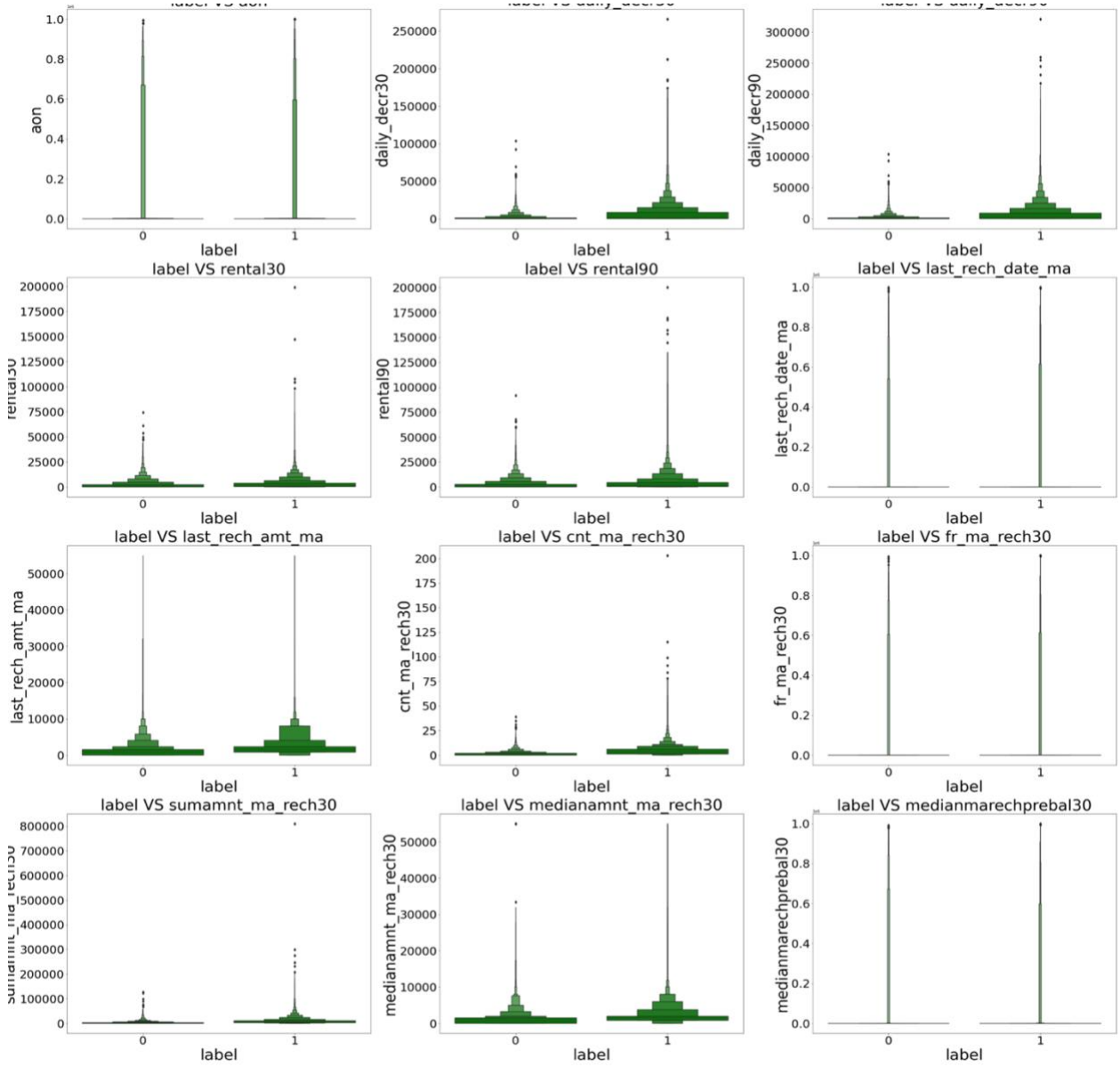
Observations:

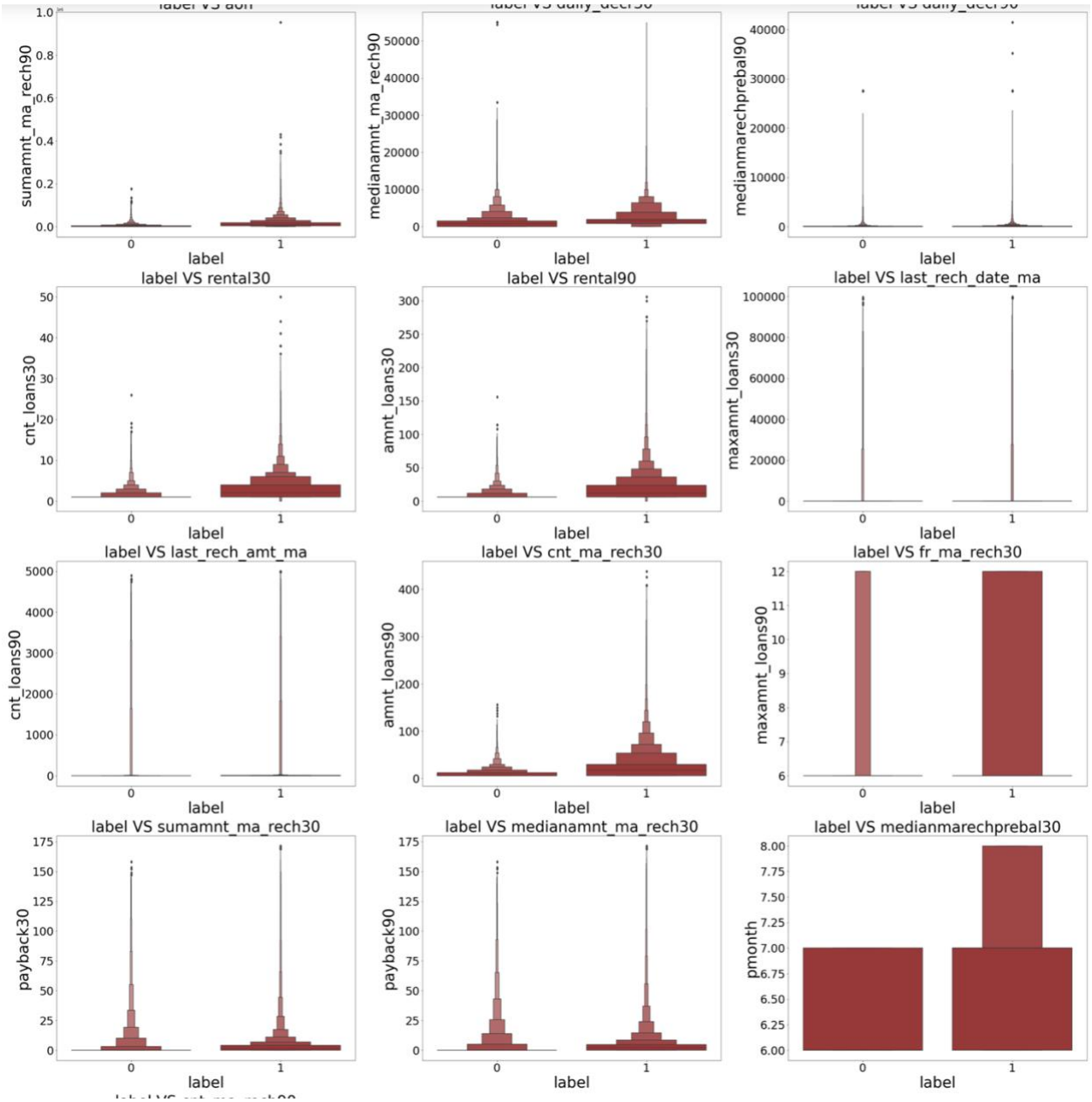
- 1.Customers with high value of Age on cellular network in days(aon) are maximum defaulters (who have not paid there loan amount 0).
- 2.Customers with high value of Daily amount spent from main account, averaged over last 30 days (in Indonesian Rupiah) (daily_decr30) are maximum non-defaulters (who have paid there loan amount-1).
- 3.Customers with high value of Daily amount spent from main account, averaged over last 90 days (in Indonesian Rupiah)(daily_decr90) are maximum Non-defaulters(who have paid there loan amount-1).
- 4.Customers with high value of Average main account balance over last 30 days(rental30) are maximum Non-defaulters(who have paid there loan amount-1).
- 5.Customers with high value of Average main account balance over last 90 days(rental90) are maximum Non-defaulters(who have paid there loan amount-1).
- 6.Customers with high Number of days till last recharge of main account(last_rech_date_ma) are maximum Non-defaulters(who have paid there loan amount-1).
- 7.Customers with high value of Amount of last recharge of main account (in Indonesian Rupiah)(last_rech_amt_ma) are maximum Non-defaulters(who have paid there loan amount-1).

- 8.Customers with high value of Number of times main account got recharged in last 30 days(cnt_ma_rech30) are maximum Non-defaulters(who have paid there loan amount-1).
- 9.Customers with high value of Frequency of main account recharged in last 30 days(fr_ma_rech30) are maximum Non-defaulters(who have paid there loan amount-1) and also the count is high for defaulters comparitively Non-defaulters are more in number.
- 10.Customers with high value of Total amount of recharge in main account over last 30 days (in Indonesian Rupiah)(sumamnt_ma_rech30) are maximum Non-defaulters(who have paid there loan amount-1).
- 11.Customers with high value of Median of amount of recharges done in main account over last 30 days at user level (in Indonesian Rupiah)(medianamnt_ma_rech30) are maximum Non-defaulters(who have paid there loan amount-1).
- 12.Customers with high value of Median of main account balance just before recharge in last 30 days at user level (in Indonesian Rupiah)(medianmarechprebal30) are maximum defaulters(who have not paid there loan amount-0).
- 13.Customers with high value of Number of times main account got recharged in last 90 days(cnt_ma_rech90) are maximum Non-defaulters(who have paid there loan amount-1).
- 14.Customers with high value of Frequency of main account recharged in last 90 days(fr_ma_rech90) are maximum Non-defaulters(who have paid there loan amount-1).
- 15.Customers with high value of Total amount of recharge in main account over last 90 days (in Indonasian Rupiah)(sumamnt_ma_rech90) are maximum Non-defaulters(who have paid there loan amount-1).
- 16.Customers with high value of Median of amount of recharges done in main account over last 90 days at user level (in Indonasian Rupiah)(medianamnt_ma_rech90) are maximum Non-defaulters(who have paid there loan amount-1).
- 17.Customers with high value of Median of main account balance just before recharge in last 90 days at user level (in Indonasian Rupiah)(medianmarechprebal90) are maximum Non-defaulters(who have paid there loan amount-1).
- 18.Customers with high value of Number of loans taken by user in last 30 days(cnt_loans30) are maximum Non-defaulters(who have paid there loan amount-1).
- 19.Customers with high value of Total amount of loans taken by user in last 30 days(amnt_loans30) are maximum Non-defaulters(who have paid there loan amount-1).
- 20.Customers with high value of maximum amount of loan taken by the user in last 30 days(maxamnt_loans30) are maximum Non-defaulters(who have paid there loan amount-1).
- 21.Customers with high value of Number of loans taken by user in last 90 days(cnt_loans90) are maximum Non-defaulters(who have paid there loan amount-1).
- 22.Customers with high value of Total amount of loans taken by user in last 90 days(amnt_loans90) are maximum Non-defaulters(who have paid there loan amount-1).

- 23.Customers with high value of maximum amount of loan taken by the user in last 90 days(maxamnt_loans90) are maximum Non-defaulters(who have paid there loan amount-1).
- 24.Customers with high value of Average payback time in days over last 30 days(payback30) are maximum Non-defaulters(who have paid there loan amount-1).
- 25.Customers with high value of Average payback time in days over last 90 days(payback90) are maximum Non-defaulters(who have paid there loan amount-1).
- 26.In between 6th and 7th month maximum customers both defualters and Non-defaulters have paid there loan amount.
- 27.Below 14th of each month all the customers have paid there loan amount.

- **Boxenplot of columns with Target:**





Observations:

- There are no specific pattern of defaulter and non-defaulter.
- As the max_amnt increases the number of defaulters decreases.
- Number of defaulters are very less compared to non-defaulters in all columns.
- A few columns show neutral pattern for both defaulter and non-defaulter.
- Pmonth is distributed more between June and July and the defaulters are 0 after mid-july.

CONCLUSION

1. Key Findings and Conclusions of the Study

In this project report, we have used machine learning algorithms to predict the micro credit defaulters. We have mentioned the step by step procedure to analyse the dataset and finding the correlation between the features. Thus we can select the features which are correlated to each other and are independent in nature. These feature set were then given as an input to four algorithms and a hyper parameter tunning was done to the best model and the accuracy has been improved. Hence we calculated the performance of each model using different performance metrics and compared them based on these metrics. Then we have also saved the best model and predicted the label.

2. Learning Outcomes of the Study in respect of Data Science

Improvement in computing technology has made it possible to examine social information that cannot previously be captured, processed, and analyzed. The power of visualization has helped us in understanding the data by graphical representation. Data cleaning is one of the most important steps to remove missing values and to replace them with respective mean, median or mode. I found that the dataset was quite interesting to handle as it contains all types of data in it.

To conclude, the application of machine learning in micro credit is still at an early stage. We hope this study has moved a small step ahead in providing some methodological and empirical contributions to crediting institutes, and presenting an alternative approach to the valuation of defaulters. Future direction of research may consider incorporating additional micro credit transaction data from a larger economical background with more features.

3. Limitations of this work and Scope for Future Work

- Length of the data is huge and hard to handle.
- There were so many outliers present in the dataset.
- There was a lot of skewness present in the dataset which will again affect the model as we must transform it.
- This study did not use all advanced algorithms but only a few simple classification algorithms to a few advanced ones.
- There were a few columns with more Zero values. I must remove those columns.
- There was multicollinearity in the dataset, but we cannot lose any more data so did not remove it.

Even after all these Limitations and drawbacks, my model tends to perform well with an accuracy of 95.59% with Random Forest model and a CV Score of 94.92%.

THANK YOU