**MICRO\_CREDIT DEFAULTER PROJECT**

Prepared by: Shashi Sahu

Data Science Intern at Flip Robo Technologies



SME Name:

Mohd Kashif Sir

**Acknowledgment**

It is my most profound pleasure and gratification to present this report. Working on this project was an incredible experience that gave me very informative knowledge of data analysis. All the required information and dataset are provided by Flip Robo Technologies (Bangalore) which helped me to complete the project. I want to thank my SME Mohd Kashif Sir for giving the dataset and instructions to perform the complete case study process.

**INTRODUCTION**

**Problem Statement:**

## 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.

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

## Conceptual Background of the Domain Problem

MFS 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).

## Review of Literature

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.

## Motivation for the Problem Undertaken

We understand the importance of communication and how it effects a person’s life and lack of communication can cause lot of uncertain problems so we want to work in order to bridge this gap between people.

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.

**Analytical Problem Framing**

Mathematical/ Analytical Modelling of the Problem:

We first look into the statistics of data shown in fig 1.



Fig 1 Statistical description of data

From this statistical analysis we make some of the interpretations that,

1. Maximum standard deviation is observed in aon column.
2. In the columns aon, daily\_decr30, daily\_decr90, rental30, rental90, last\_rech\_date\_ma, last\_rech\_date\_da, maxamnt\_loans30, cnt\_loans90, amnt\_loans90 mean is considerably greater than median so the columns are positively skewed.
3. In the columns label, month median is greater than mean so the columns are negatively skewed.

4. In the columns aon, daily\_decr30, daily\_decr90, rental30, rental90, last\_rech\_date\_ma, last\_rech\_date\_da, maxamnt\_loans30, cnt\_loans90, payback30, payback90 there is huge difference present between 75th perecentile and maximum so outliers are present here.

We look for the skewness present in data shown in fig 2,

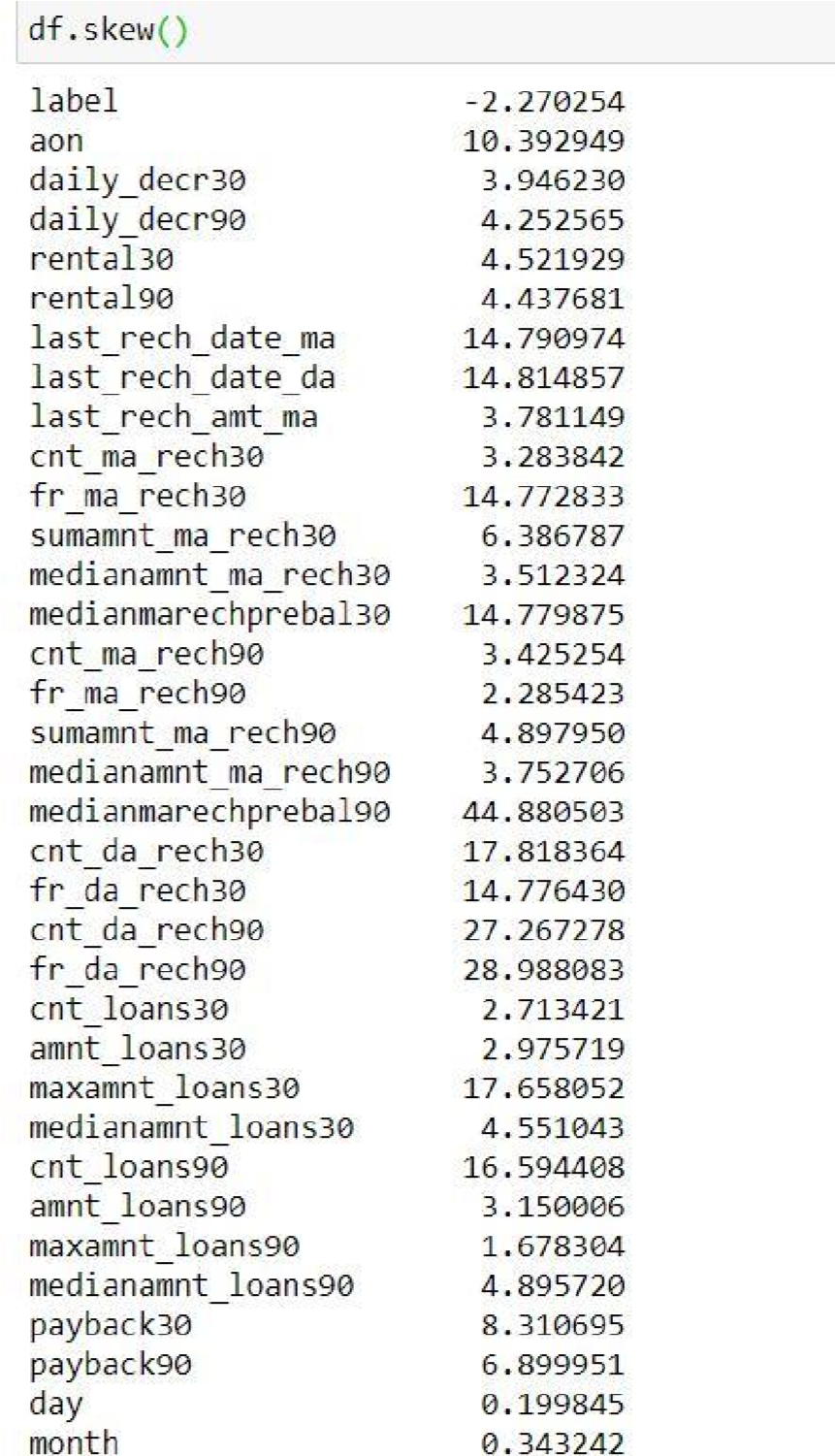


Fig 2 skewness in data

We observe skewness in the data due to outliers so we remove the 7-8% outliers through zscore method by keeping standard deviation 5 and treat the rest outliers through winsorization technique. Now the skewness observed is shown in fig 3.

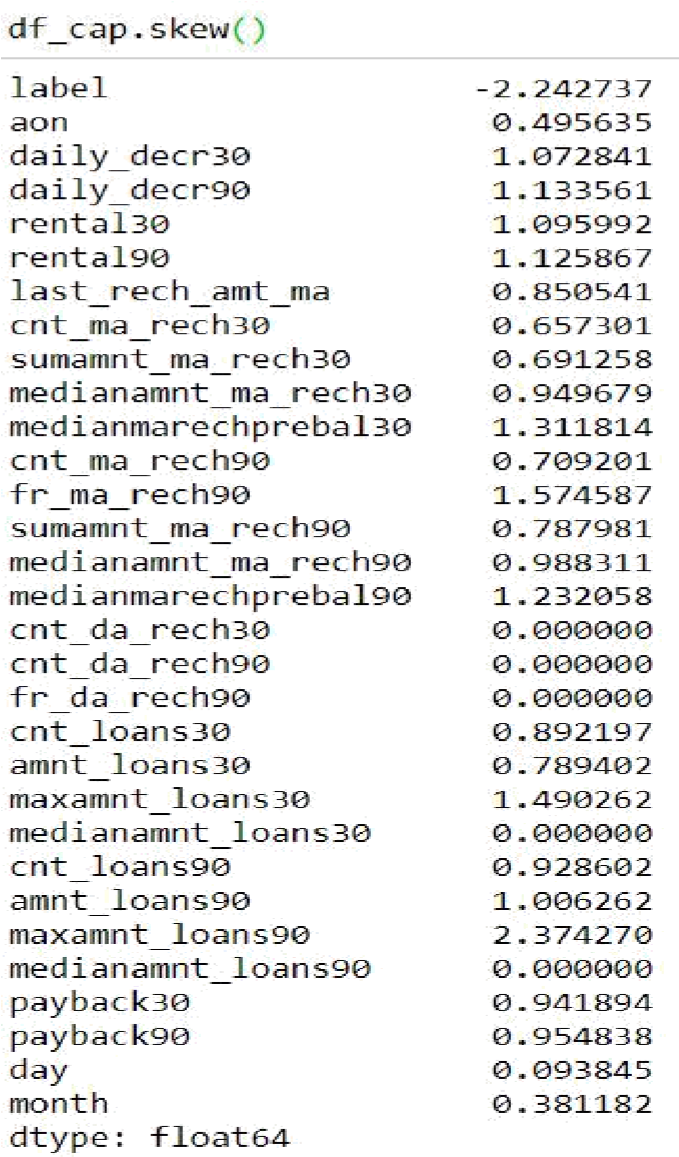


Fig3: Skewness observed after treating outliers through winsorization

## Data Sources and their formats

The variable features of this problem statement are:-

Variable: Definition -> comment

* label : Flag indicating whether the user paid back the credit amount within 5 days of issuing the loan{1:success, 0:failure}
* msisdn : mobile number of user
* aon : age on cellular network in days

* daily\_decr30 : Daily amount spent from main account, averaged over last 30 days (in Indonesian Rupiah)
* daily\_decr90 : Daily amount spent from main account, averaged over last 90 days (in Indonesian Rupiah)
* rental30 : Average main account balance over last 30 days
* rental90 : Average main account balance over last 90 days
* last\_rech\_date\_ma : Number of days till last recharge of main account
* last\_rech\_date\_da: Number of days till last recharge of data account
* last\_rech\_amt\_ma : Amount of last recharge of main account (in Indonesian Rupiah)
* cnt\_ma\_rech30 : Number of times main account got recharged in last 30 days
* fr\_ma\_rech30 : Frequency of main account recharged in last 30 days
* sumamnt\_ma\_rech30 : Total amount of recharge in main account over last 30 days (in Indonesian Rupiah)
* medianamnt\_ma\_rech30 : Median of amount of recharges done in main account over last 30 days at user level (in Indonesian Rupiah)
* medianmarechprebal30 : Median of main account balance just before recharge in last 30 days at user level (in Indonesian Rupiah)
* cnt\_ma\_rech90 : Number of times main account got recharged in last 90 days
* fr\_ma\_rech90 : Frequency of main account recharged in last 90 days
* sumamnt\_ma\_rech90 : Total amount of recharge in main account over last 90 days (in Indonasian Rupiah)
* medianamnt\_ma\_rech90 : Median of amount of recharges done in main account over last 90 days at user level (in Indonasian Rupiah)
* medianmarechprebal90 : Median of main account balance just before recharge in last 90 days at user level (in Indonasian Rupiah)
* cnt\_da\_rech30 : Number of times data account got recharged in last 30 days
* fr\_da\_rech30: Frequency of data account recharged in last 30 days
* cnt\_da\_rech90 : Number of times data account got recharged in last 90 days
* fr\_da\_rech90 : Frequency of data account recharged in last 90 days
* cnt\_loans30 : Number of loans taken by user in last 30 days
* amnt\_loans30: Total amount of loans taken by user in last 30 days
* maxamnt\_loans30 : maximum amount of loan taken by the user in last 30 days
* medianamnt\_loans30 : Median of amounts of loan taken by the user in last 30 days
* cnt\_loans90 : Number of loans taken by user in last 90 days
* amnt\_loans90 : Total amount of loans taken by user in last 90 days
* maxamnt\_loans90 : maximum amount of loan taken by the user in last 90 days
* medianamnt\_loans90 : Median of amounts of loan taken by the user in last 90 days
* payback30 : Average payback time in days over last 30 days
* payback90 : Average payback time in days over last 90 days
* pcircle : telecom circle
* pdate : date

The data types of features are shown in fig 4,

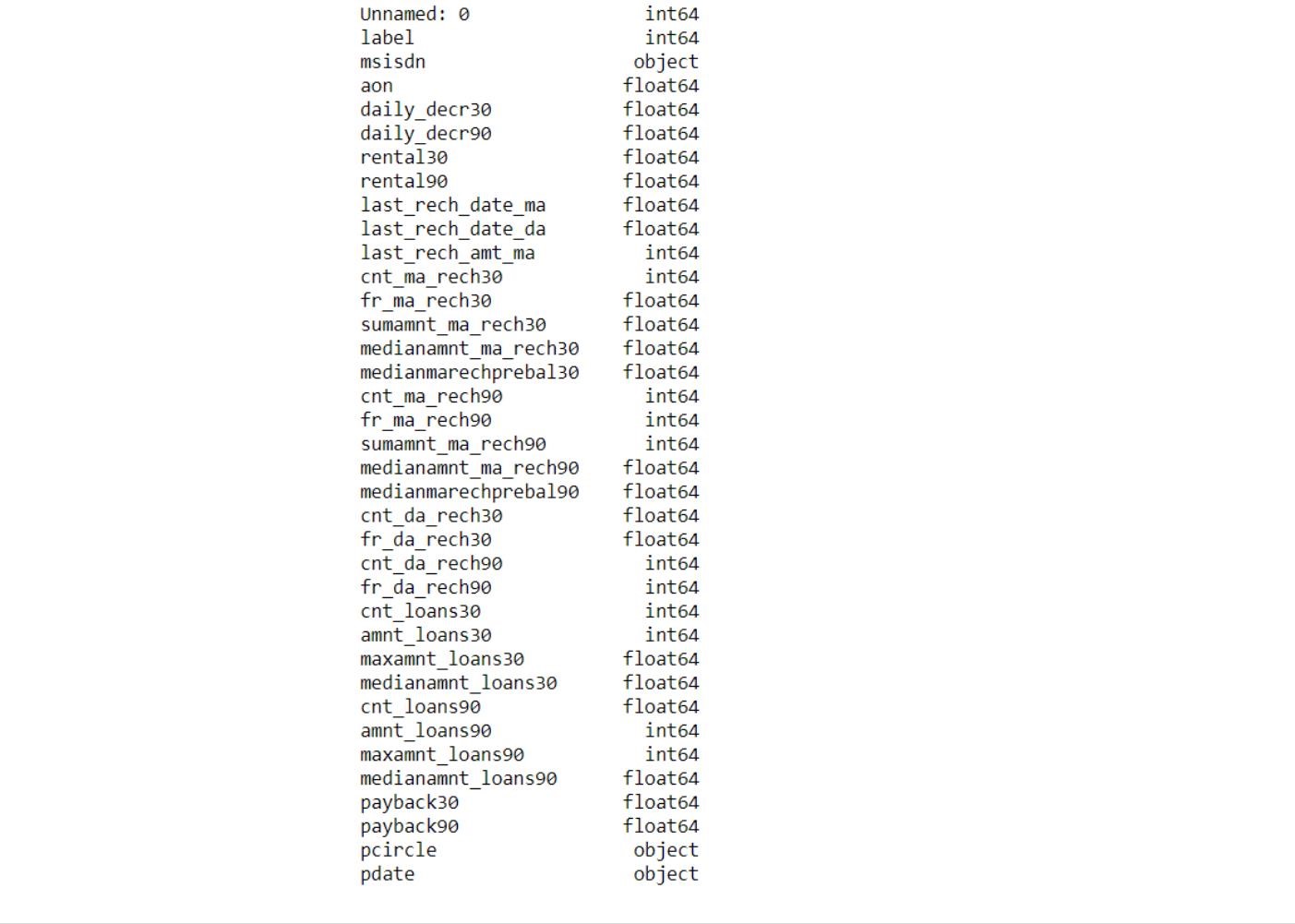


Fig 4 Data types of features

* Data Pre-processing Done

We first done data cleaning. In data cleaning we have done feature extraction, we extracted the features day and month from pdate column as shown in fig 5,



Fig 5 Feature extraction

AS we can see we extracted day and month from pdate column, we won’t be needing year as there is only one unique value of year present in the dataset i.e 2016 as shown in fig 6.

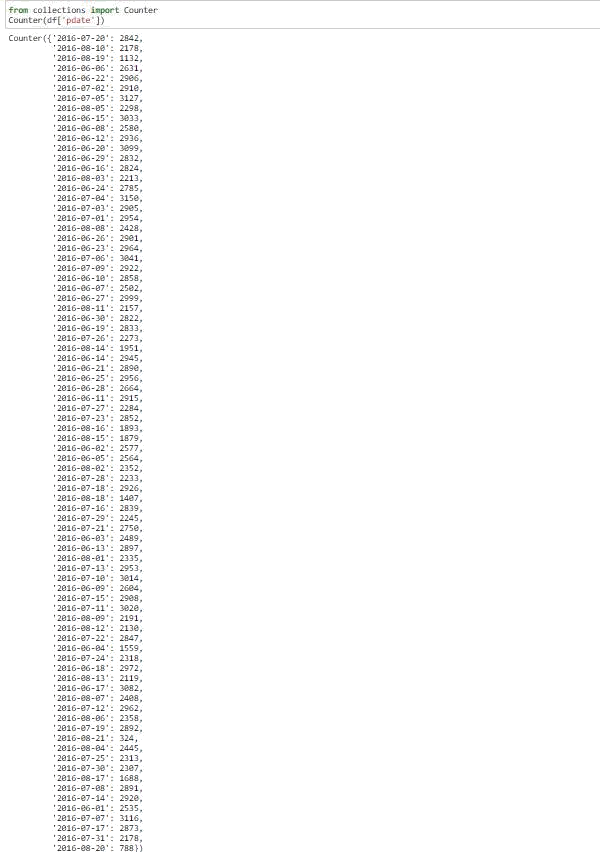


Fig 6: Unique values of pdate column

We then explored categorical variables as shown in fig 6.

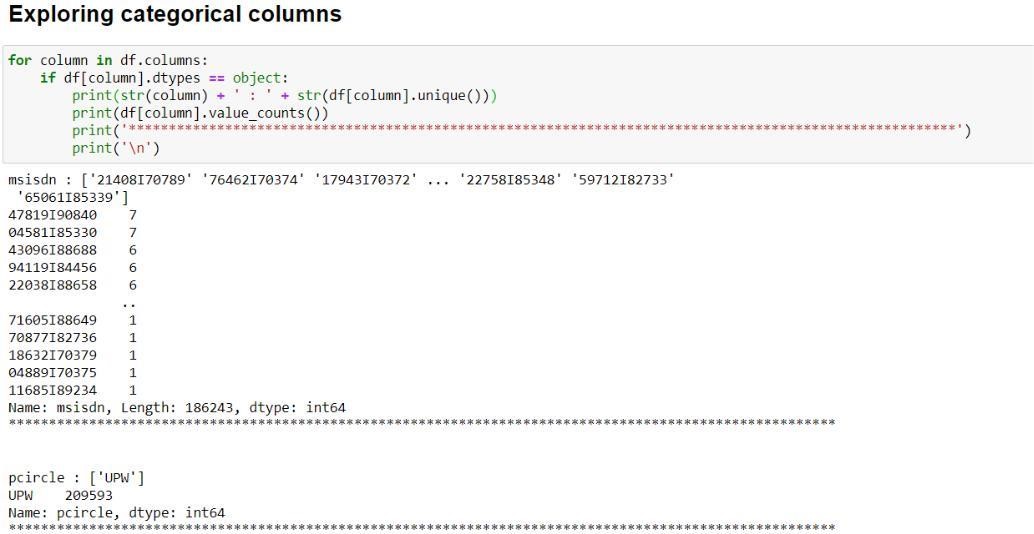


Fig 6: Exploring categorical variables

We observed that there is only one unique value present in pcircle column which is ‘UPW’ so will be dropping this column. Then we observed that column msisdn was present in categorical column so we encode it to numbers using label encoder as shown in fig 7, to check it’s correlation with other feature variables and target variable.

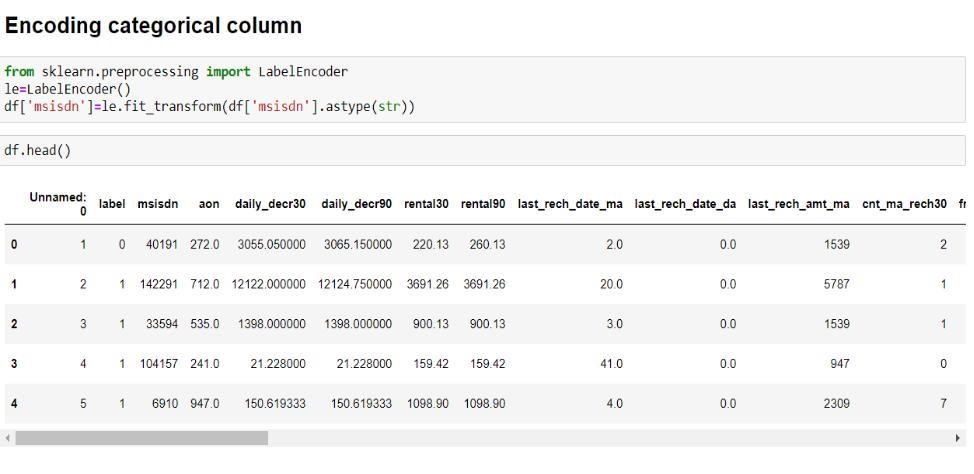


Fig 7: Encoding column msisdn

Then we checked the correlation with the help of heatmap as shown in fig 8,

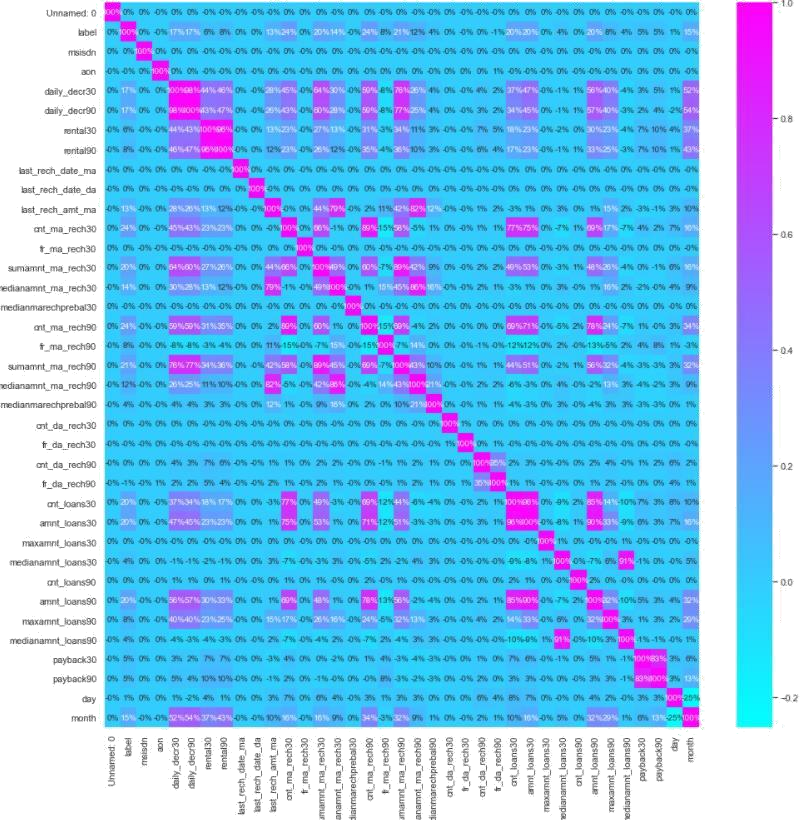


Fig 9: Heatmap of correlation

While checking the heatmap of correlation we observed that there exists multicollinearity in between columns.

We also observed that no correlation was present in unnamed: 0, msisdn, last\_rechdate\_ma, last\_rechdate\_da columns so we will be dropping these columns.

We then removed the outliers from the dataset through zscore method.

13 | P a g e

* **Data Inputs- Logic- Output Relationships**

Here we check the correlation between all our feature variables with target variable label as shown in fig 10.

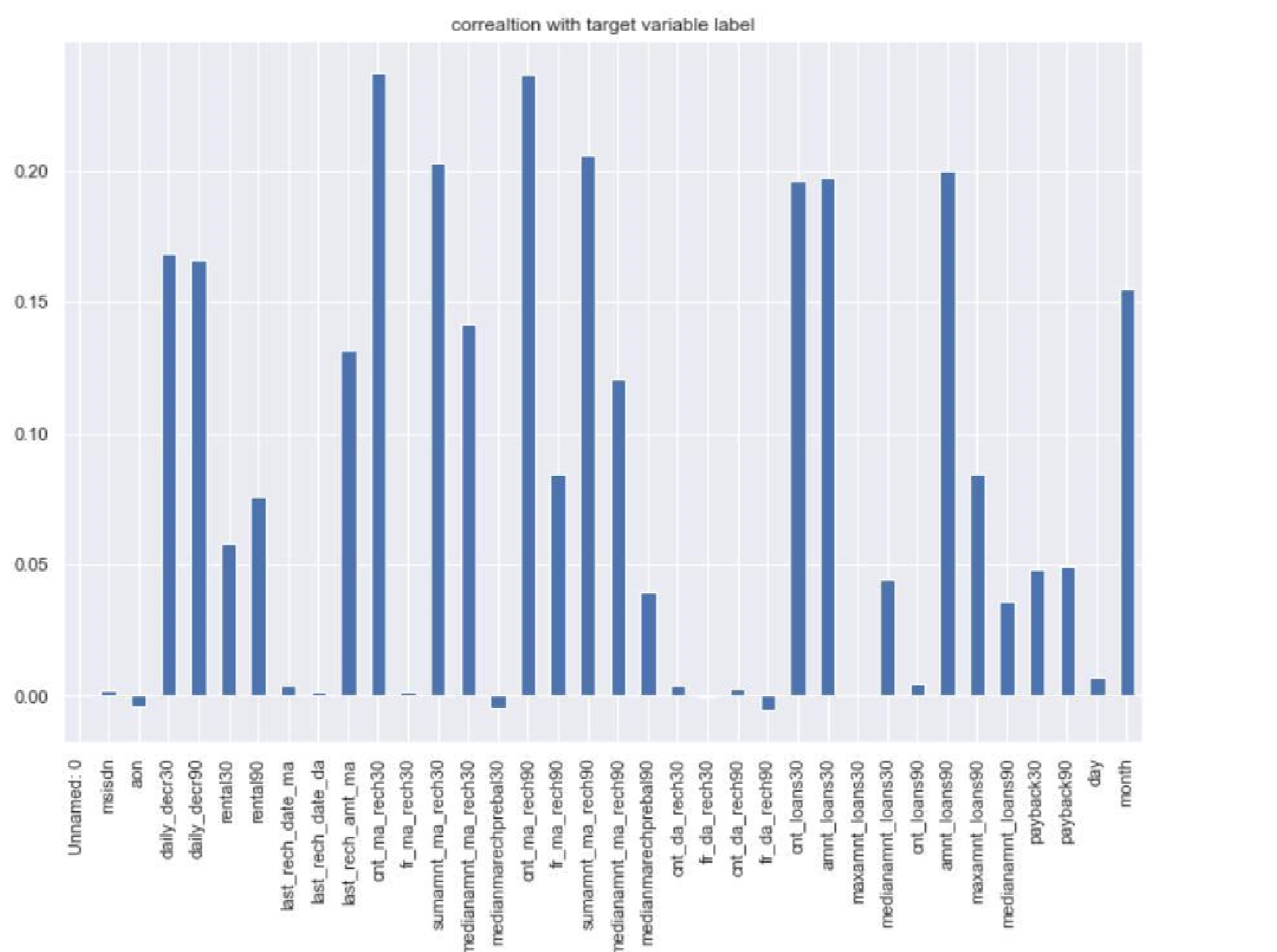


Fig 10 correlation with target variable label

* We observe that the columns cnt\_ma\_rech30 and cnt\_ma\_rech90 are highly positively correlated with label this means as the cnt\_ma\_rech30 and cnt\_ma\_rech90 are increasing the probability of cutomer being nonfraudulent is also increasing.

* We also observe that the columns aon, medianmarechprebal30 and fr\_da\_rech90 are negatively correlated with label this means as the aon, medianmarechprebal30 and fr\_da\_rech90 are increasing the probability

of customer being fraudulent is also increasing.

* Set of assumptions related to the problem under consideration

By looking into the target variable label we assumed that it was a classification type of problem.

We observed multicollinearity in between columns so we assumed that we will be using Principal Component Analysis (PCA).

We also observed that only one single unique value was present in pcircle and in year in pdate column and in Unnamed: 0 all the numbers were unique without any correlation so we assumed that we will be dropping these columns.

* **Hardware and Software Requirements and Tools Used**

This project was done on laptop with i5 processor with quad cores and eight threads with 8GB of ram and Intel(R) Core(TM) i3-7020U on Anaconda, jupyter notebook.

The tools, libraries and packages we used for accomplishing this project are pandas, numpy, matplotlib, seaborn, scipy stats, sklearn.decomposition pca, sklearn standardscaler, collections counter, imblearn SmoteTomek,

GridSearchCV, joblib.

* Through pandas library we loaded our csv file ‘Data file’ into dataframe and performed data manipulation and analysis. Through pandas library we converted pdate column to datetime format from which we were able

to extract day and month column.

* With the help of numpy we worked with arrays.

* With the help of matplotlib and seaborn we did plot various graphs and figures and done data visualization.
* With scipy stats we treated outliers through winsorization technique.

* With sklearn.decomposition’s pca package we reduced the number of feature variables from 34 to 7 by plotting scrre plot with their Eigenvalues and chose the number of columns on the basis of their nodes.

* With sklearn’s standardscaler package we scaled all the feature variables onto single scale.

* With collection’s counter package we were able to display all the unique values of the pdate column.

* Through imblearn’s SmoteTomek package we were able to handle the imbalanced data by increasing the number of fraudulent transactions on relevant data points.

* Through GridSearchCV we were able to find the right parameters for

hyper parameter tuning.

* Through joblib we saved our model in csv format.

# Model/s Development and Evaluation

• Identification of possible problem-solving approaches (methods)

We first converted all our categorical variables to numeric variables with the help of label encoder to check out the correlation between them and dropped the columns which we felt were unnecessary.

We observed skewness in data so we tried to remove the skewness through treating outliers with winsorization technique as shown in fig 3.

The data was imbalanced so through imblearn’s SmoteTomek package we were able to handle the imbalanced data by increasing the number of fraudulent transactions on relevant data points.

The data was improper scaled so we scaled the feature vaariables on a single scale using sklearn’s StandardScaler package.

There were too many (37) feature variables in the data so we reduced it to 7 with the help of Principal Component Analysis(PCA) by plotting Eigenvalues and taking the number of nodes as our number of feature variables.

* **Testing of Identified Approaches (Algorithms)**

The algorithms we used for the training and testing are as follows: -

* Extreme gradient boosting classifier
* Decision tree classifier
* KNeighbors classifier
* Logistic Regression
* GaussianNB
* Random forest classifier
* Ada boost classifier
* GradientBoostingClassifier
* Bagging classifier
* Extra trees classifier



16 | P a g e

* **Run and Evaluate selected models**

The algorithms we used are shown in fig 11,

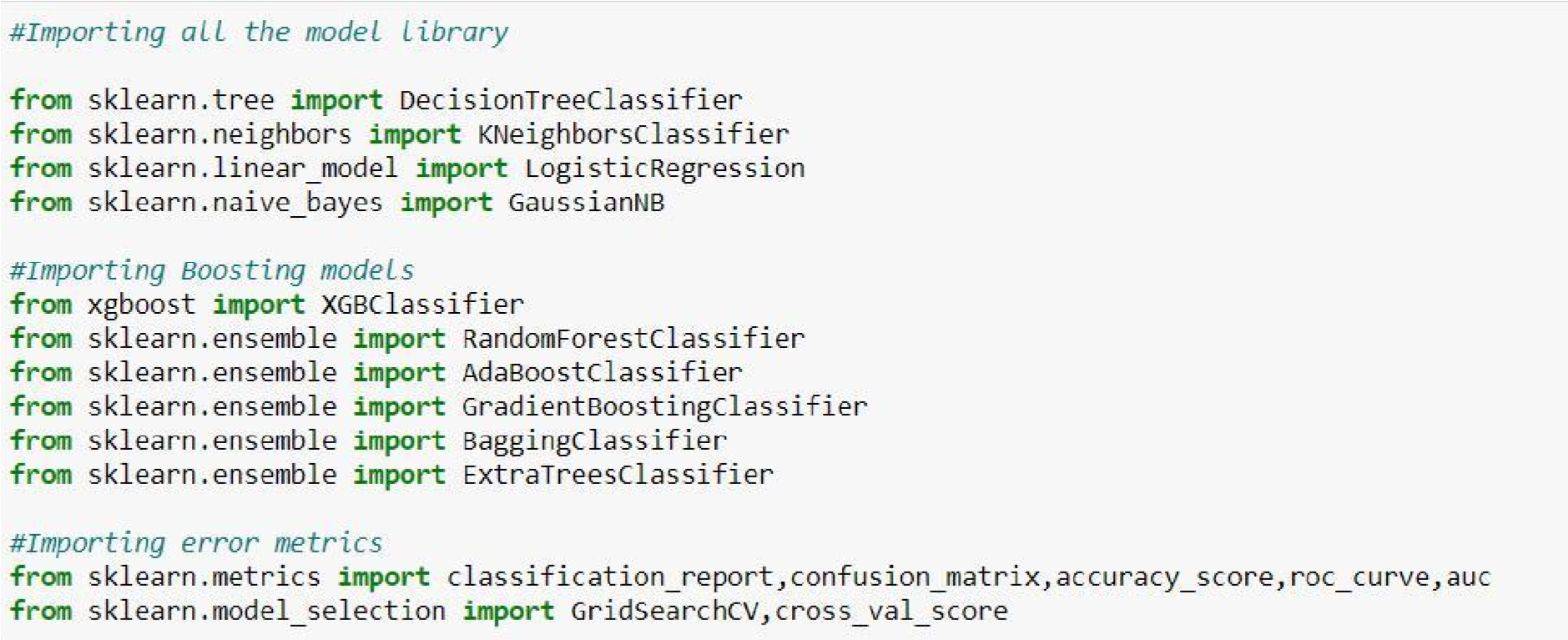


Fig 11: Algorithms used

The results observed over different evaluation metrics are shown in fig 12,

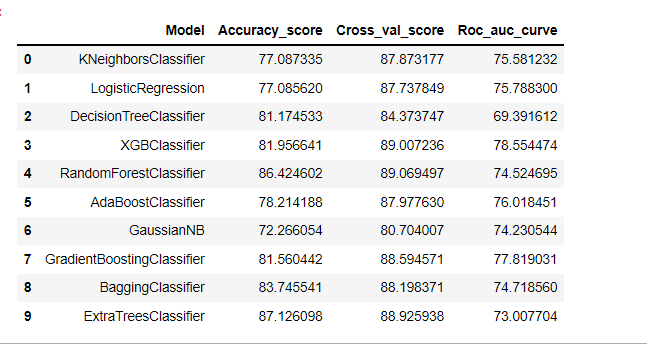


Fig 12 Results observed

* **Key Metrics for success in solving problem under consideration**

Accuracy is not an appropriate measure of model performance here and we used the metric AREA UNDER ROC CURVE to evaluate models performance because high roc\_score which mean high recall, which means the model does well by not classifying legit transactions as fraudulent.

* **Visualizations:**
* **Count Plot of label**:-

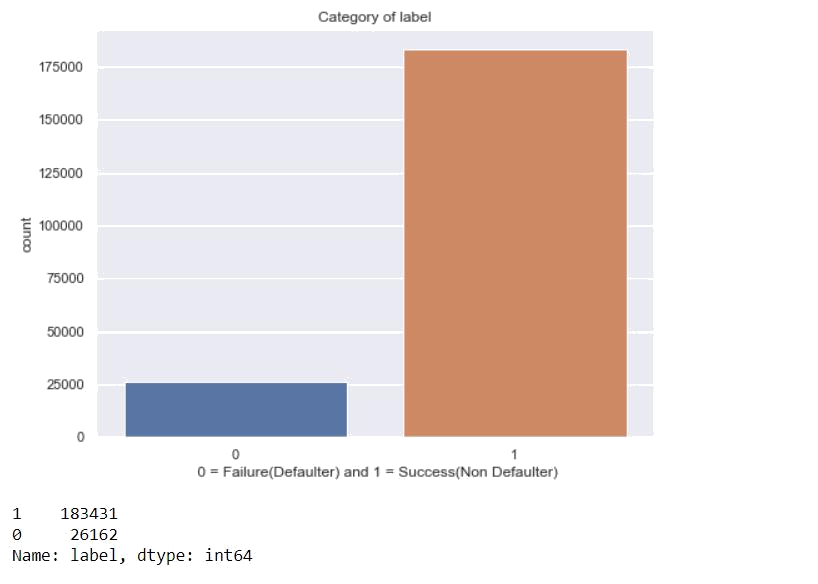


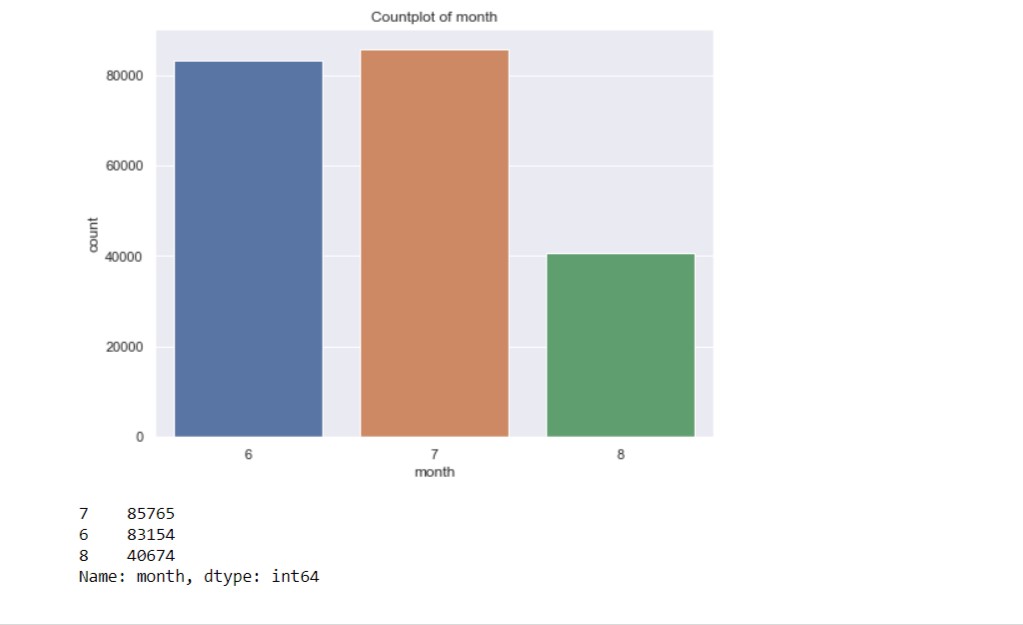
Fig 13: Count plot of label

**Observation:**

1. We observe 183431 number of Non defaulters whereas 26162 number of

Defaulters.

1. We observe that this is a very imbalanced data set.



* **Count Plot of month**:-

Fig 14: Count plot of month

**Observation:**

Maximum (85765) number of users has taken credit on 7th month of the year.

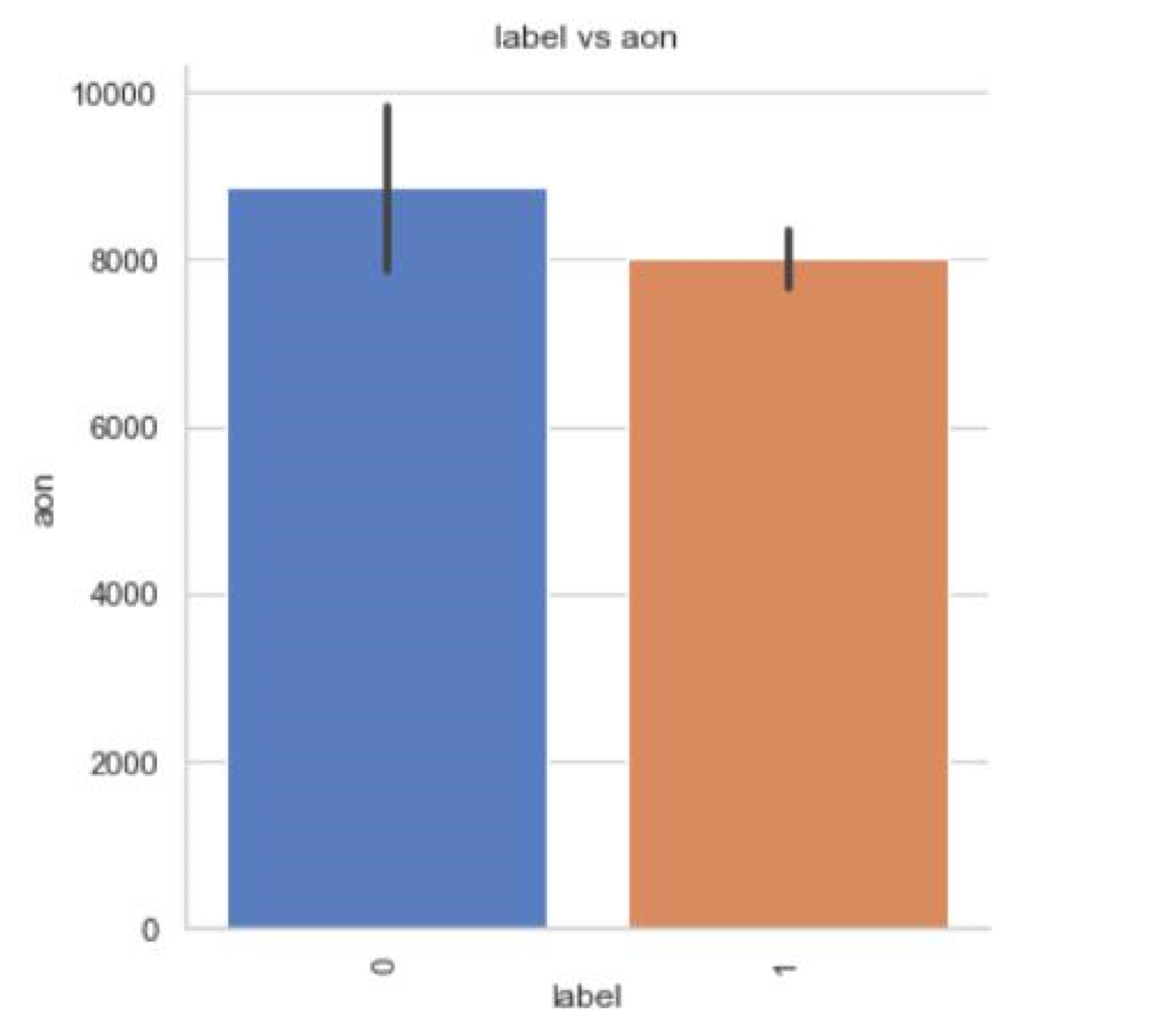
* **Checking column aon with label:-** 

Fig 15 label vs aon

**Observation:**

If the aon is high then the number of defaulters are also high.

* **Checking column cnt\_ma\_rech30 with label:-**

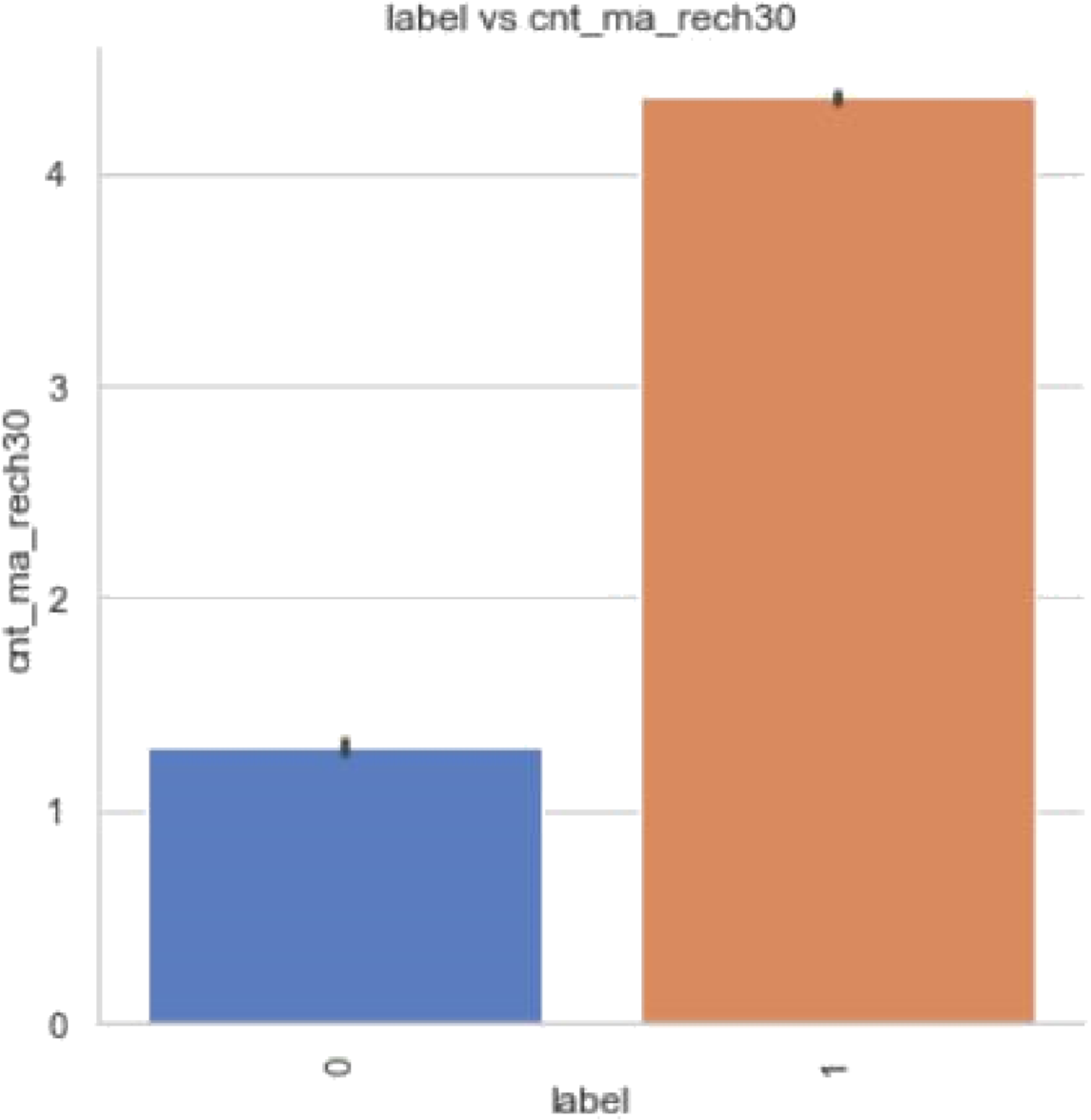


Fig 16: label vs cnt\_ma\_rech30

**Observation:**

If the number of times main account got recharged in last 30 days (cnt\_ma\_rech30) is more then there is less chance of defaulter of the account.

* **Checking the column sumamnt\_ma\_rech30 with label: -**

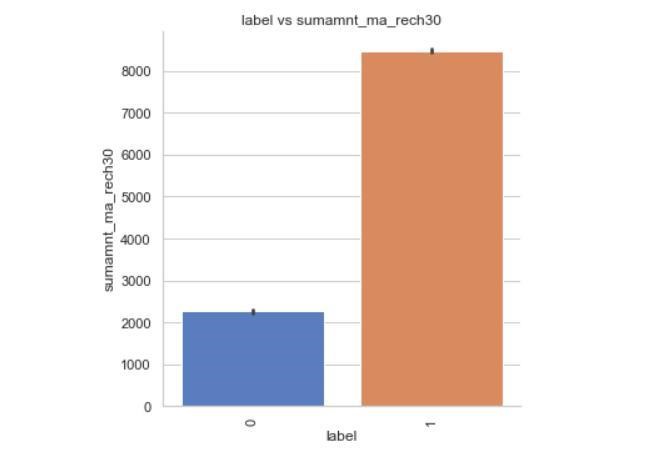


Fig 17 label vs sumamnt\_ma\_rech30

**Observation:**

If Total amount of recharge in main account over last 30 day (sumamnt\_ma\_rech30) is more, then the chances of default are less.

* **Checking cnt\_ma\_rech30 and cnt\_ma\_rech90 with label:-**

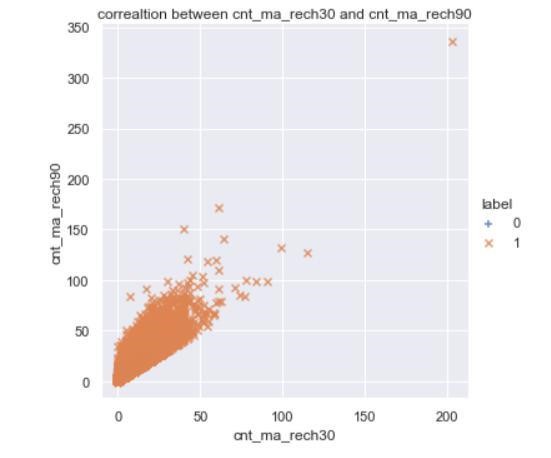


Fig 18 Scatter plot between cnt\_ma\_rech30 and cnt\_ma\_rech90 with respect to label

**Observation:**

As cnt\_ma\_rech30 and cnt\_ma\_rech90 are increasing then the number ofnon-defaulters are also increasing

* **Checking sumamnt\_ma\_rech90 and amnt\_loans90 with label:-**

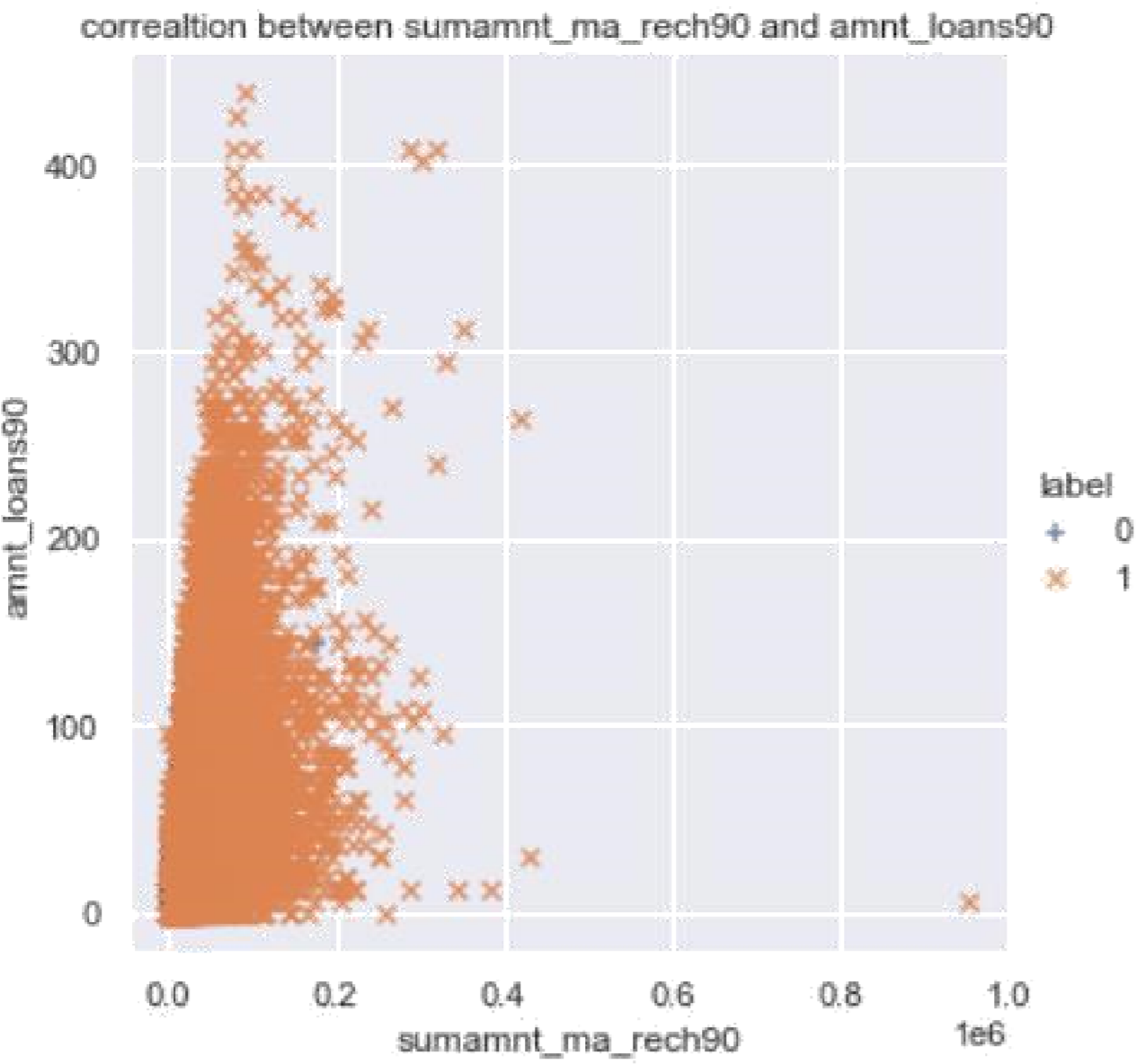


Fig 19 Scatter plot between sumamnt\_ma\_rech90 and amnt\_loans90 with respect to label

**Observation:**

As sumamnt\_rech90 and amnt\_loans30 are increasing the number of non- defaulters are also increasing.

* **Interpretation of the Results**

From the visualization we interpreted that the data was very imbalanced and the target variable was highly positively correlated with the columns cnt\_ma\_rech30 and cnt\_ma\_ma\_rech90.

From the pre-processing we interpreted that data was improper scaled, there were hidden features present in the data which needed to be extracted.

From the modelling we interpreted that XGBClassifier works best with respect to our model with roc\_score 0.94 as shown in fig 20



Fig 20 AUC\_ROC curve using XGBClassifier

# CONCLUSION

* **Key Findings and Conclusions of the Study**

In this project we have tried to show how to deal with unbalanced datasets like the Micro-Credit Defaulter where the instances of fraudulent cases is few compared to the instances of non-fraudulent cases. We have argued why accuracy is not an appropriate measure of model performance here and used the metric AREA UNDER ROC CURVE to evaluate how method of SmoteTomek technique can lead to better model training.

The best score of 0.94 was achieved using the best parameters of XGBClassifier through GridSearchCV though both random forest and gradient boosting models performed well too.

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

* This project has demonstrated the importance of sampling effectively, modelling and predicting data with an imbalanced dataset.
* Through different powerful tools of visualization we were able to analyse and interpret different hidden insights about the data.

* Through data cleaning we were able to remove unnecessary columns and outliers from our dataset due to which our model would have suffered from overfitting or underfitting.

The few challenges while working on this project were:-

* Improper scaling
* Too many features
* Hidden features
* Imbalanced data
* Skewed data due to outliers

The data was improper scaled so we scaled it to a single scale using sklearns’s package StandardScaler.

There were too many(37) features present in the data so we applied Principal Component Analysis(PCA) and found out the Eigenvalues and on the basis of number of nodes we were able to reduce our features upto 7 columns.

There were hidden features present in pdate column so we converted the column in datetime format in order to extract day and month column by doing feature extraction.

The data was imbalanced so we handled the unbalanced data through SmoteTomek technique by creating more number of fraudulent cases on relevant data points.

The columns were skewed due to presence of outliers which we handled through winsorization technique.

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

While we couldn’t reach out goal of 100% accuracy in fraud detection, we did end up creating a system that can with enough time and data get very close to that goal. As with any project there is also a room for improvement here. The very nature of this project allows for multiple algorithms to be integrated together as modules and their results can be combined to increase the accuracy of the final result. This model can further be improved with the addition of more algorithms into it. However, the output of these algorithms needs to be in the same format as the others. Once that condition is satisfied, the modules are easy to add as done in the code. This provides a great degree of modularity and versatility to the project.