

NAME OF THE PROJECT

***Micro-credit Defaulters***

Submitted by:

Manisha Kumari

# Analytical Problem Framing

# Mathematical/ Analytical Modeling of the Problem

the micro credit defaulters dataset that contains various features and information about it. Using the data in form of ‘read\_csv’ function provided by the Pandas package, which can import the data into our python environment. After importing the data, I have used the ‘head’ function to get a glimpse of our dataset.

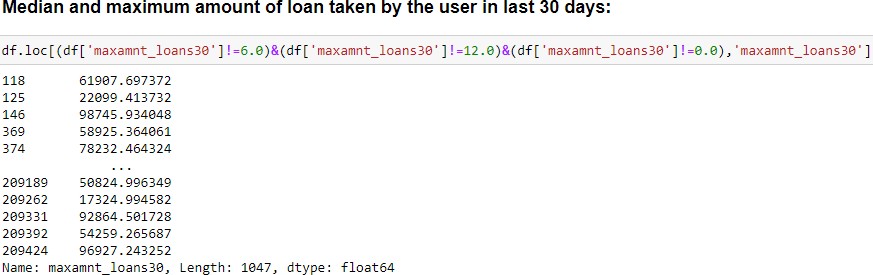
In this label is used as my target column and it was having two classes Label ‘1’ indicates that the loan has been paid i.e. Non-defaulter, whereas Label ‘0’ indicates that the loan

has not been paid i.e. defaulter. It’s clarify the binary classification problem, classification of algorithms for building model. There is no null values in the dataset and observed some unnecessary entries in some columns like in some columns it found more than 90% , zero values so dropped those columns. Those columns will create high skewness in the model.

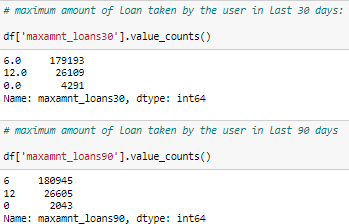
To get better insight on the features uses plotting function like distribution plot, bar plot and count plot. With these plotting it is able to understand the relation between the features in better manner. Also outliers and skewness found in the dataset so it is removed outliers using percentile method and skewness using yeo-johnson method. classification algorithms while building model then tuned the best and saved the best model. Lastly predicted the label using saved model.

## Data Pre-processing Done

In order to get a better understanding of the data, we plotted a histogram of the data. We noticed that the dataset had many outliers, so removing outliers using percentile method and skewness using yeo-johnson method. however, there were many data points that did not conform to this. This is because accident history and condition can have a significant effect of defaulter or non- defaulter, we pruned our dataset to standard deviations around the mean in order to remove outliers. We converted the Make, Model and State into one-hot vectors



Maximum loans in 30 & 90 days



Data Inputs- Logic- Output Relationships

Since all data has numerical columns and plotted dist plot to see the distribution of each column data. So box plot is used 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.

Exploratory Data Analysis (*EDA*)

This section shows the exploration done on the dataset, which is what motivated the use of the algorithm. The following are the questions explored in this project and for the sake of writing I will only show some of the visuals here while I will provide the codes that shows the full visualization of all the questions explored.

Is there a significant relationship between Non-defaulter & defaulter? It was used to check for this and we can see that there is a relationship between Label ‘1’ as Non-defaulter, whereas Label ‘0’ indicates that the loan has not been paid i.e. defaulter.

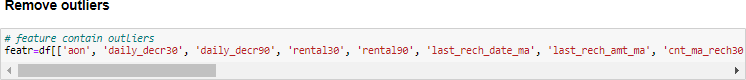
Dataset is imbalanced. Label ‘1’ has approximately 87.5% records, while, label ‘0’ has approximately 12.5% records. Need to balance.

There are two primary phases in the system:

Training phase: The system is trained by using the data in the data set and fits a model (line/curve) based on the algorithm chosen accordingly.

Testing phase: the system is provided with the inputs and is tested for its working. The accuracy is checked. And therefore, the data that is used to train the model or test it, has to be appropriate. The system is designed to detect and

predict and hence appropriate algorithms must be used to do the two different tasks. Before the algorithms are selected for further use, different algorithms were compared for its accuracy. The well-suited one for the task was chosen.

Data cleaning:

Steps :

* Importing the required packages into our python environment * Importing the data to do some EDA on it

* Dataset having 209593 rows and 36 columns including target.

* Data Visualization

* Feature Selection & Data Split

* Modelling the data using the algorithms

* Evaluating the built model using the evaluation metrics

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

* + Finally, we conclude which model is best suitable for the given case by evaluating each of them using the evaluation metrics provided by the scikit-learn package. This 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

Technological stack, algorithms, and shortcomings of the project which led to build this project.

**Model/s Development and Evaluation**

Identification of possible problem-solving approaches (methods)

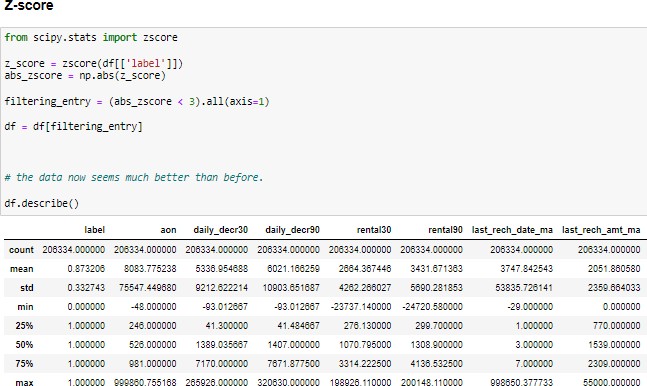
The factors need to be found which can impact the micro credit. This can be done by analysing the various factors and the stores the respondent prefers. This will be done by checking each of the factors impacts the respondents in decision making.

Machine Learning Algorithms:

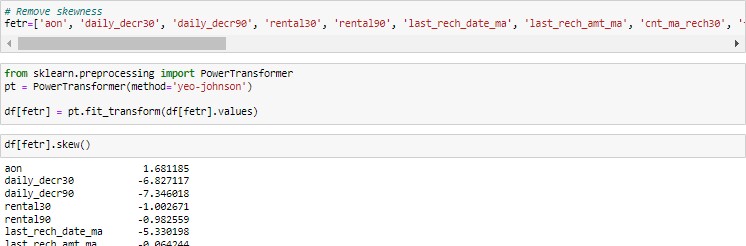
Machine learning-based systems are growing in popularity in research applications in most disciplines. Considerable decision-making knowledge from data has been acquired in the broad area of machine learning, in which decision-making tree-based ensemble

techniques are recognized for supervised classification problems. Thus, classification is an essential form of data analysis in data mining that formulates models while describing significant data.

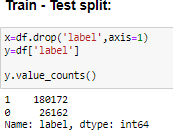
**Testing of Identified Approaches (Algorithms)**

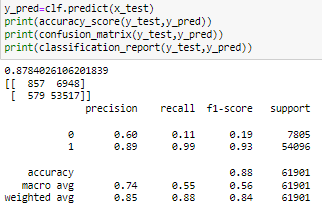
We utilized several classic and state-of-the-art methods, including ensemble learning techniques, with a 90% - 10% split for the training and test data. To reduce the time required for training, we used over 20 thousand examples in our dataset with 209593 rows and 36 columns.

Skewness:



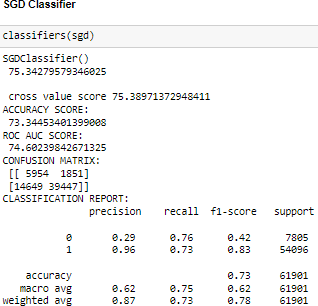
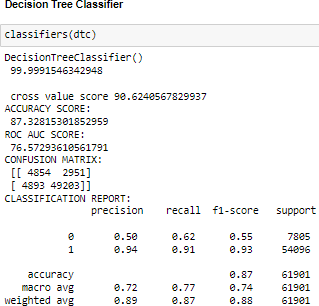
* + Random Forest Classifier, Decision Tree Classifier, AdaBoost Classifier, GradientBoosting Classifier, Bagging Classifier, XGB Classifier, SGD Classifier were our baseline methods. For most of the model implementations, the open-source Scikit-Learn package was used.

**Accuracy\_score of train-test:**

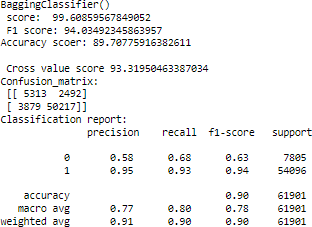
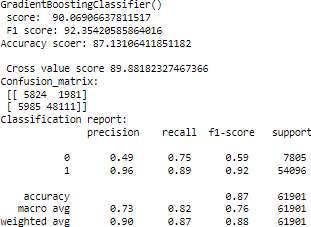


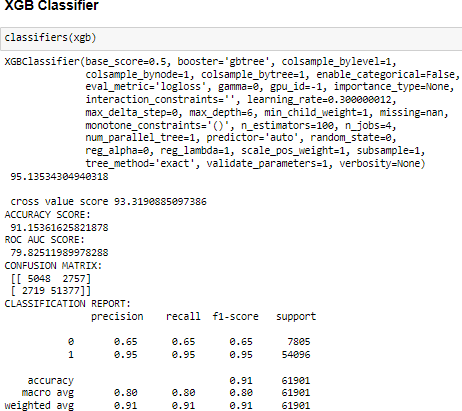
## Run and Evaluate selected models

* + Our primary packages for this project are going to be pandas for data processing, NumPy to work with arrays, matplotlib & seaborn for data visualizations, and finally scikit-learn for building an evaluating our ML model.

Models:

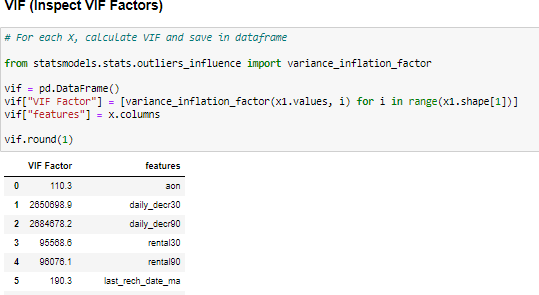
|  |  |
| --- | --- |
| **Random Forest Classifier** | **Ada-Boost Classifier** |
| **Gradient Boosting Classifier** | **Bagging Classifier** |





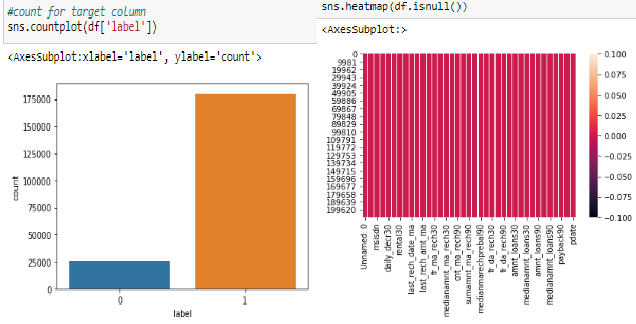
### Key Metrics for success in solving problem under consideration

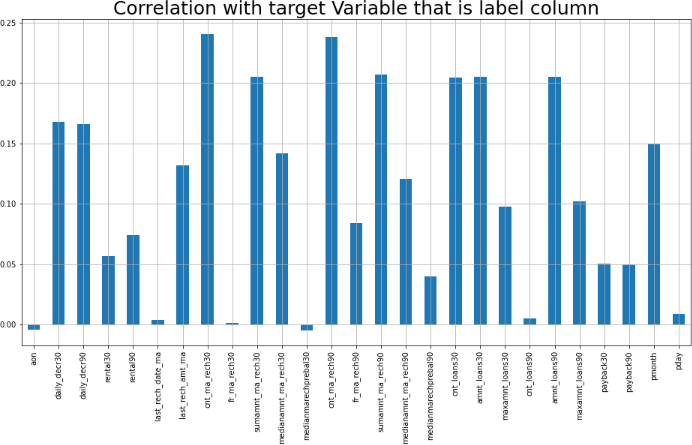
* + Using the sklearn.metrics calculated Adjusted R2 squared ,Mean Absolute Error (MAE),Mean Squared Error (MSE),Root Mean Squared Error (RMSE)
    - Precision can be seen as a measure of quality, higher precision means that an algorithm returns more relevant results than irrelevant ones.
    - Recall 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

Using Hyper-parameter : model parameters are estimated from data automatically and model hyper-parameters are set manually and are used in processes to help estimate model and Grid search is a basic method for hyper-parameter tuning. It performs an exhaustive search on the hyper-parameter set specified by users.

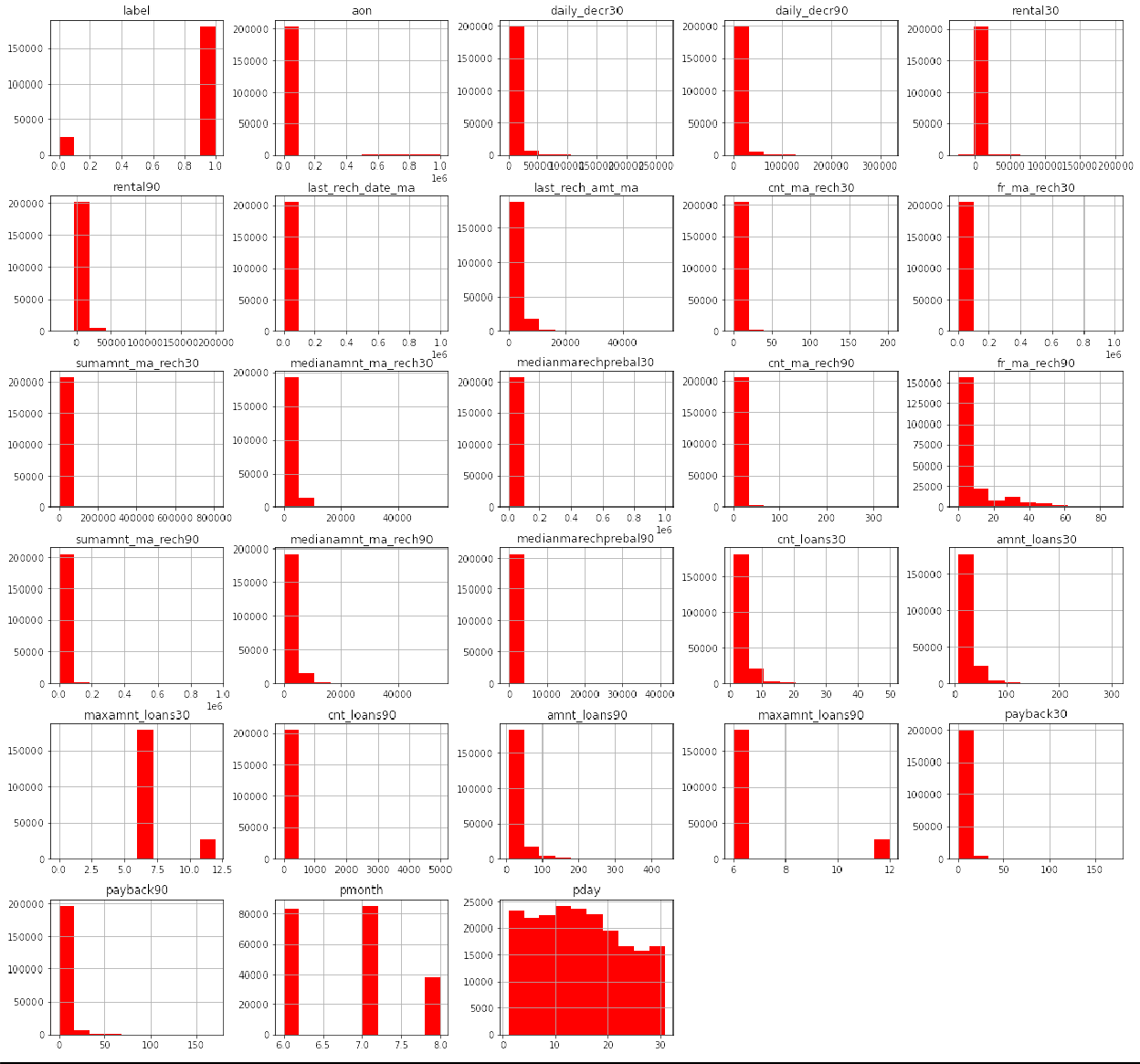
## Visualizations

* + As the value counts observation I find imbalance dataset in which defaulter values is less and Non defaulter values is high. About to 15% and 85% respectively
  + Dataset is imbalanced. Label ‘1’ has approximately 87.5% records, while, label ‘0’ has approximately 12.5% records. Need to balance.

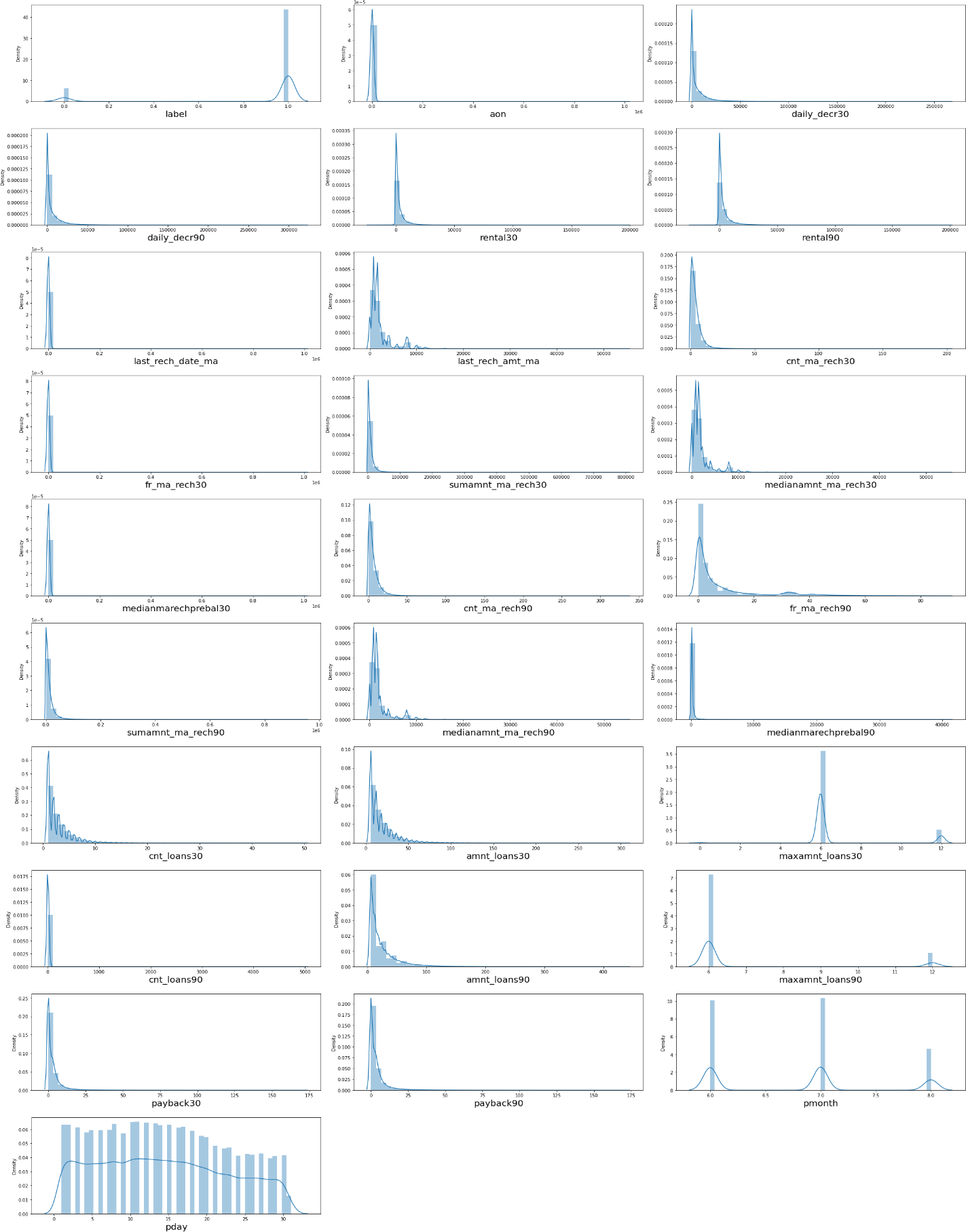




#### Plotting the Histogram

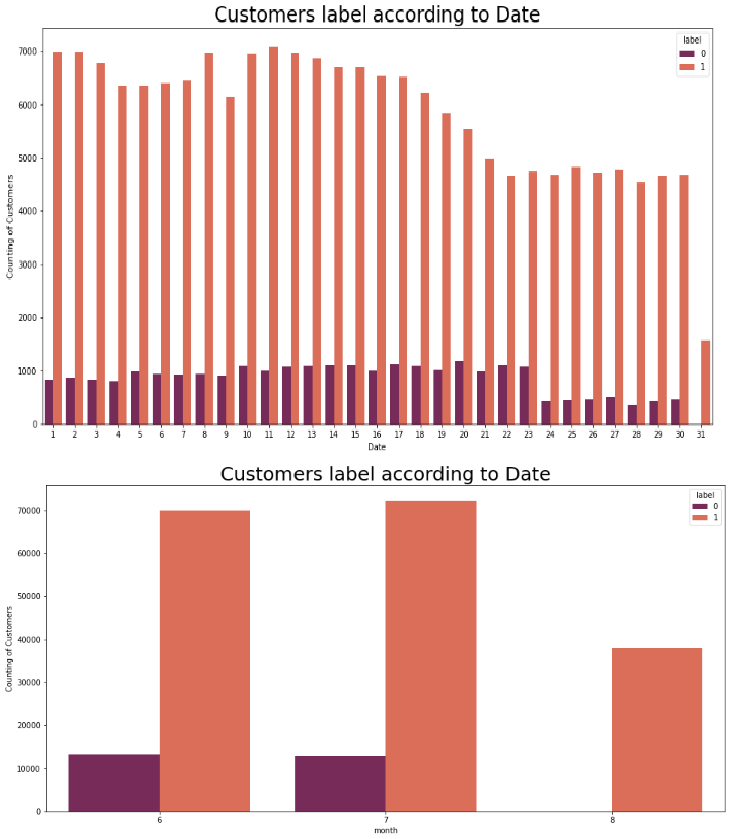


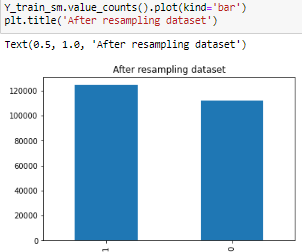
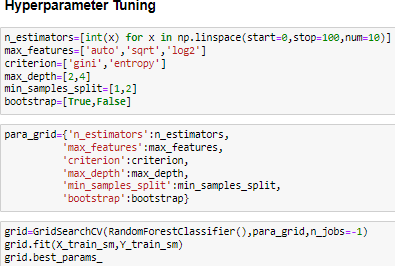
* To remove outliers I have used percentile method. And to remove skewness I have used yeo-johnson method. We have dropped all the unnecessary columns in the dataset according to our understanding. Use of Pearson’s correlation coefficient to check the correlation between dependent and independent features. Also I have used Normalization to scale the data. After scaling we have to balance the target column using oversampling. Then followed by model building with all Classification algorithms. I have used oversampling (SMOTETomek) to get rid of data unbalancing.

Bar plots to see the relation of numerical feature with target and 2 types of plots for numerical columns like distribution plot for univariate and bar plot for bivariate analysis.

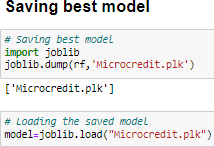
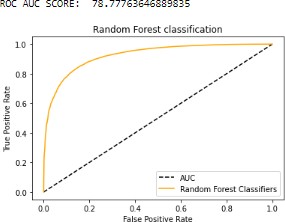
Outliers in most of the columns so we have to treat them using suitable methods.

Loan count as per date and month (Non-defaulter and Defaulter):





#### Roc & Auc

* + Present the receiver operating characteristic (ROC) curves and their respective areas under the curve (AUCs). ROC curves and AUCs are used to measure the quality of a classifier’s output; thus, they measure how correctly a classifier has been tuned. Movement along the ROC curve is typically a trade-off between the classifier’s sensitivity (true positive rate (TPR)) and specificity (TNR), and the steeper the curve, the better. For the ROC curve, sensitivity increases as we move up, and specificity decreases as we move right. The ROC curve along a 45\_ angle

## Interpretation of the Results

* + - In this research, two experiments were performed, the first experiment was validating and filtering data using all the variables available in the dataset after pre-processing, while the second experiment was conducted using most important variables and the goal of this is to be able to improve the model's performance using fewer variables.
    - Requirement of train and test and building of many models to get accuracy of the model.
    - There are multiple of matric which decide the best fit model like as : R-squared ,RMSE value, VIF, CDF & PDF Z-score , Roc & Auc and etc.
    - Database helped in making perfect model and will help in understanding Indonesian micro finance services (MFS) And use multiple metrics like F1\_score, precision, recall and accuracy\_score which will help to decide the best model.
    - Random forest Classifier as the best model with 91.18% accuracy\_score..
    - Lastly predicted wheather the loan is paid back or not using saved model. It was good!! that was able to get the predictions near to actual values.

# CONCLUSION



## Key Findings and Conclusions of the Study

This research evaluated individuals’ credit risk performance in a micro-finance environment using machine learning and deep learning techniques. While traditional methods utilizing models such as linear regression are commonly adopted to estimate reasonable accuracy nowadays, these models have been succeeded by extensive employment of machine and deep learning models that have been broadly applied and produce prediction outcomes with greater precision. Using real data, we compared the various machine learning algorithms’ accuracy by performing detailed experimental analysis while classifying individuals’ requesting a loan into three classes, namely, good, average, and poor.

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 analyze 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 tuning was done to the best model and the accuracy has been improved.

Calculated the performance of each model using different performance metrics and compared them based on these metrics. Then we have also saved the best fit model and predicted the label. This is interesting that predicted and actual values were almost same.

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

* + Dataset is imbalanced. Label ‘1’ has approximately 87.5% records, while, label ‘0’ has approximately 12.5% records, and defaulter are higher.
  + This 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

## Limitations of this work and Scope for Future Work

* + The length of the dataset it is very huge and hard to handle.
  + Number of outliers and skewness these two will reduce our model accuracy.
  + Also, we have tried best to deal with outliers, skewness and zero values. So it looks quite good that we have achieved a accuracy of 91.32% even after dealing all these drawbacks.
  + This study will not cover all Classification algorithms instead, it is focused on the chosen algorithm, starting from the basic assembling techniques to the advanced ones.

