**MICRO FINANCE DEFAULTER PROJECT**

Submitted by:

SANYA DUBEY

**ACKNOWLEDGMENT**

I, Sanya Dubey- the intern Data Science of FlipRobo is overwhelmed in all humbleness and gratefulness to acknowledge my deep gratitude to all those who have helped me put my ideas to perfection and have assigned tasks, well above the level of simplicity and into something concrete and unique I, whole heartedly thank my SME Mr. Nitin Mishra for having faith in me, selecting me to be a part of his valuable project and for constantly motivating me to do better. They were and are always there to show me the right track when needed help. With help of their brilliant guidance and encouragement, I was able to complete my tasks properly and were up to the mark in all the tasks assigned. During the process, I got a chance to see the stronger side of my technical and non-technical aspects and also strengthen my concepts.

**INTRODUCTION**

MicroFinancerefers to all financial products and services developed for those excluded from traditional banking channel as it encourages social and banking inclusion, by enabling socially vulnerable people to benefit from productive loans, savings solutions and more. Microfinance allows people to take on reasonable small business loans safely, and in a manner that is consistent with ethical lending practices. Thus, today, it is widely accepted as a poverty-reduction tool, representing $70 billion in outstanding loans and a global outreach of 200 million clients.

The MicroFinance Services(MFS) is very beneficial while targeting people living in the remote areas by providing them financial services such as Group Loans, Agricultural Loans, Individual Business Loans and so on.

For delivering the microfinance services (MFS), comparable microfinance institutions (MFI) donors and experts are in favor of using MFS that are more convenient, efficient and cost saving than the traditional high-touch model. 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. According to many researchers and policy makers, microfinance encourages entrepreneurship, increases income generating activity thus reducing poverty, empowers the poor (especially women in developing countries), increases access to health and education, and builds social capital among poor and vulnerable communities. On working with one such client of telecom industry, we acknowledged that they are a fixed wireless telecommunications network provider. MFI has improved their business by launching various products and organizations and they also provide better products at low costs.

In this project, our aim is to distinguish the defaulters. The data is of a Telecom Industry, which 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 also working on micro credit on mobile balances which is to be paid back in 5 days. The scheme is such that, the customer is considered a defaulter if he doesn’t pay back the loaned amount within the time period of 5 days. If the loan amount is 5, the payback amount is 6 and if the loaned amount is 10, the payback amount will be 12.

For the better investment and improvement in selection of customers for the goodwill of the MFI, some predictions is to be made about the selections of customers for benefits.

DATA SET

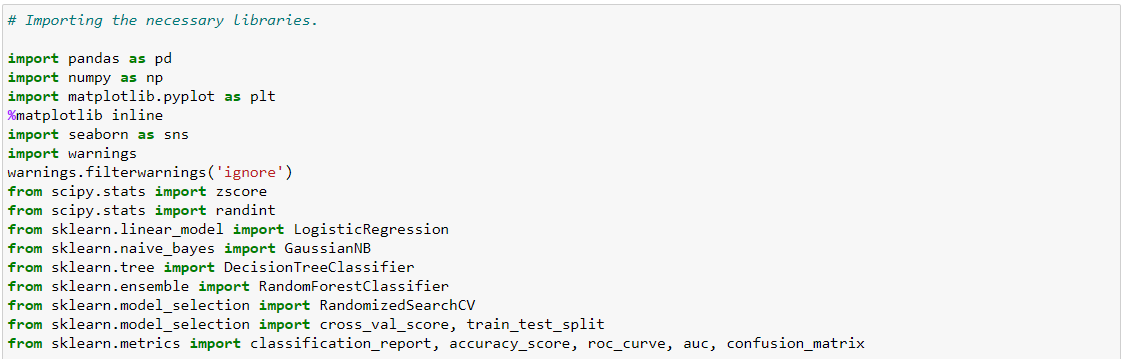
This dataset contains the information about the details of the customer to predict whether the customer is a defaulter or not.

This dataset contains 209593 rows and 37 columns including the target variable which is ‘Label’. The whole dataset is in numeric form except ‘pcircle’ and ‘pdate’ columns which are of object datatype. The data variable used in the present data is describes as below:-

|  |  |  |
| --- | --- | --- |
| 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 | Unsure of given definition |
| rental90 | Average main account balance over last 90 days | Unsure of given definition |
| 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 | Unsure of given definition |
| 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 | Unsure of given definition |
| 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 | There are only two options: 5 & 10 Rs., for which the user needs to pay back 6 & 12 Rs. respectively |
| 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 code in this project is written in Python 3.8 - Anaconda3. To begin the project, importing essential libraries are extremely initial and the very base step as it provides fast, expressive, and flexible data structures to easily work with.

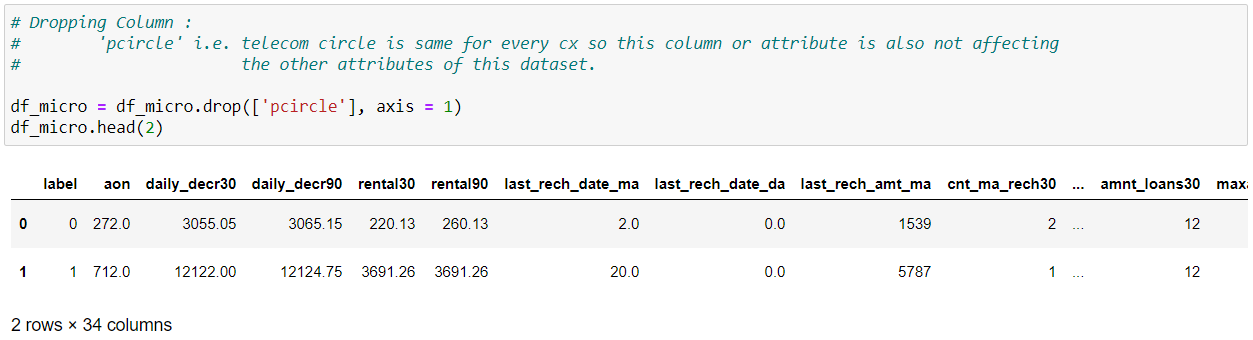
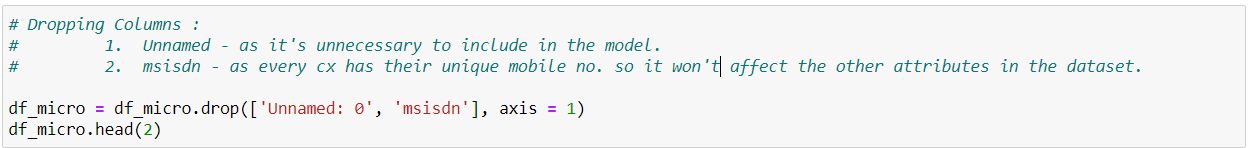
The libraries includes:



**ANALYTICAL PROBLEM FRAMING/METHODS**

**Dropping the columns**

Due to the wide range of given data, it is extremely fruitful to clean, shape and set the data in the most suitable form. Dropping unnecessary columns declines the chances of producing errors.



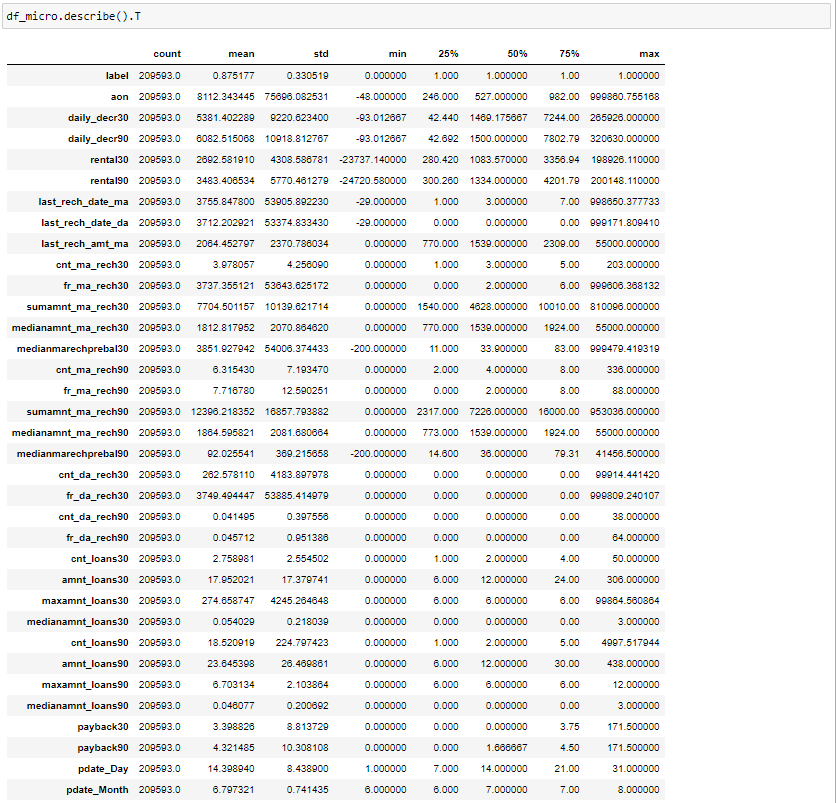
After dropping the columns and checking the shape again, only 34 attributes are present in the model including the target variable. Checking the value count shows there are no null values present in this dataset.

**Checking the missing values**

Null values are checked using heatmap and found that the dataset dosen’t contain any missing values.

**Analyzing the statistics**

Summarystatisticssummarize and provide information about your sample data. It tells you something about the values in your data set. It is calculated by reporting the mean along with a measure of variability (*standard deviation(s)* or *standard error of the mean*).



The statistical summary of the dataset depicts:-

1. The mean of most of the attributes is greater than its median.

2. There is a bit difference between the values of mean and median of the attributes.

3. The minimum values of attributes except for pdate\_Day and pdate\_Month is 0 and in negative value.

4. The range of the attributes merely differs from each other as major attributes are of high range and some ranges from 0-88.

5. The closest std for the attributes to it's mean is ‘medianamnt\_loans90=’ - 0.20 and the farthest is ‘sumamnt\_ma\_rech90’ - 16857.793882. Thus more the std, more wide spread the attribute is.

6. There are outliers present as per the statistical summary.

**Finding relationships among the analysis**

The most commonly used techniques for investigating the relationship between two quantitative variables is correlation, which is figured out using heatmap. Thus, The correlation of the numerical data using heatmap determines the relationship and statistics more distinctly among the vriables. It depicts:



1. The strongest positive correlation between - 'medianamnt\_loans30' and 'medianamnt\_loans90',

- 'amnt\_loans30' and 'amnt\_loans90',

- 'amnt\_loans30' and 'cnt\_loans30',

- 'rental30' and 'rental90' &

- 'daily\_decr30' and 'daily\_decr90'

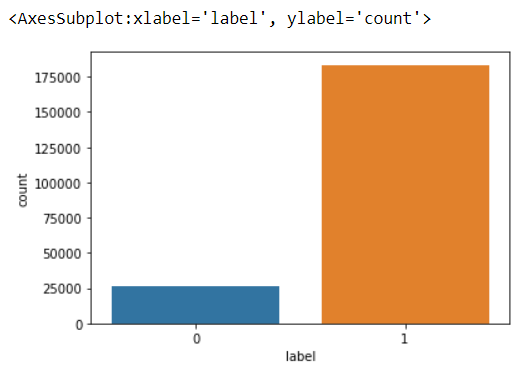
2. The 'cnt\_ma\_rech90' is strongly negatively correlated with the target variable i.e. 'fr\_ma\_rech90'.

3. With the target variable 'label' the attributes are neutral as they are nether highly positively correlated not strongly negatively correlated.

4. Overall the dataset is neither strongly positive nor strongly negatively correlated.

**Analyzing data distinctly through Visualization**

It becomes relatively easy to perceive about the defaulter using the categorical features that generally takes limited number of possible values. Check the target variable which is categorical data from the dataset.

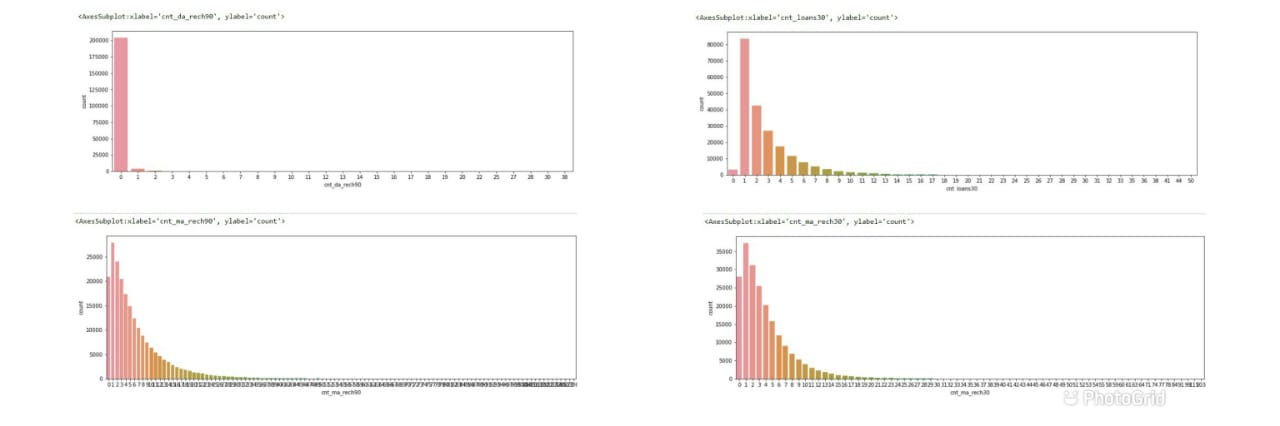


Visualizing the value counts of target variable shows that that the count of customers who have paid back the credit amount within 5 days is more than the customers who haven't paid back the credit amount within the stipulated timeframe. The difference between the count of both the customers is high as the value of not being defaulter being 1 has approximately 87.5% records. While, the defaulters being represented by 0 has approximately 12.5% records. Thus, dataset is imbalanced.

Followed by, Visualized the distribution of columns using Distribution plot. Distribution charts are used to see how quantitative values are distributed along an axis from lowest to highest. The graphs showed that the dataset is not at all normally distributed and totally positively skewed, which needs to be treated.

Plotted the outliers too using boxplot, which showed that Except for pdate\_daya nd pdate\_month the whole dataset contains outliers. 'maxamnt\_loans30', 'payback30' and 'payback90' have the higher number of outliers as compared to the other attributes.

Visualizing counts of column which can be visualized using countplot depicted that:

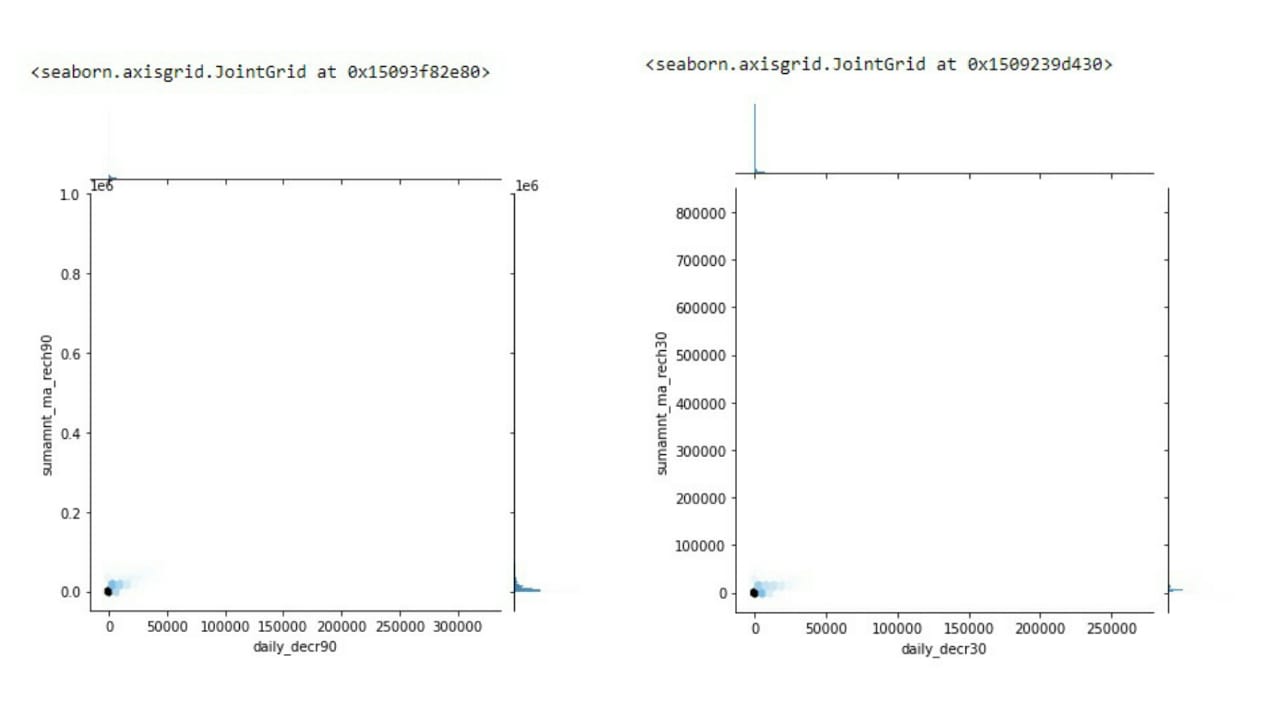


'cnt\_da\_rech90' - The data account in last 90 days majorly doesn't got recharged.

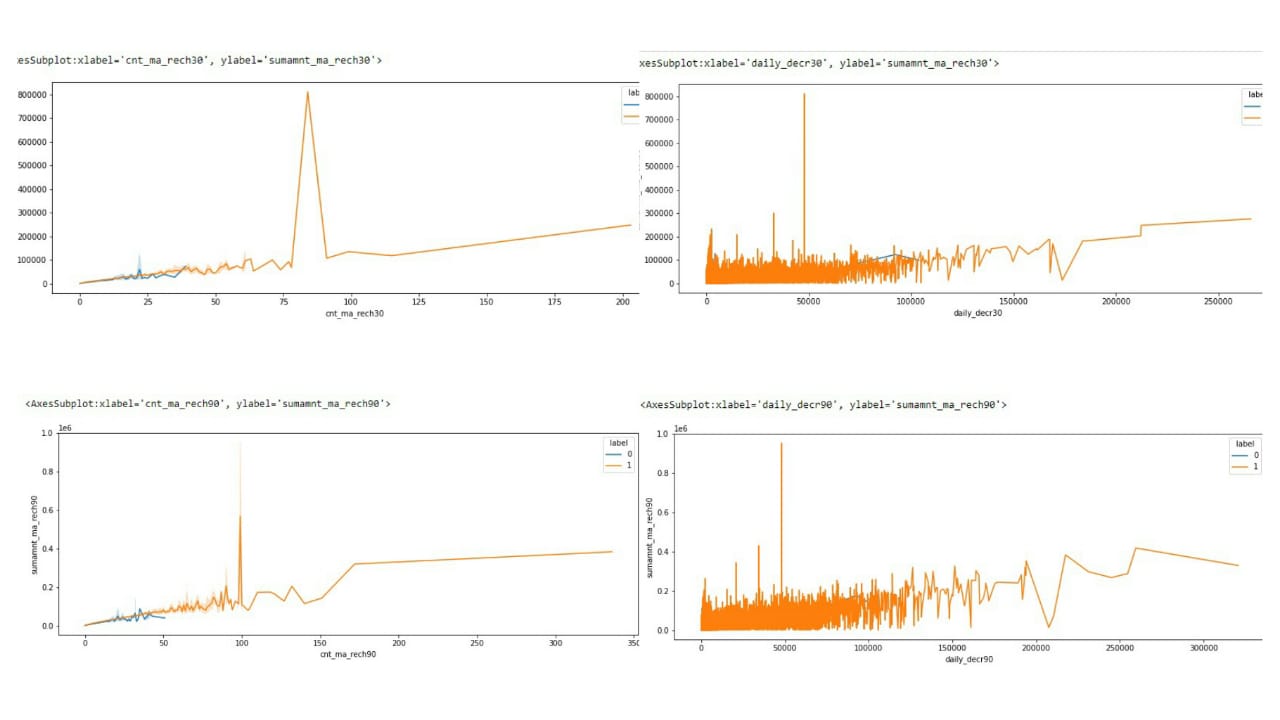
cnt\_loans30 - the users have taken loan majorly 1 time in the last 30 days followed by 2 times. here, a indirect proportionality can be seen as the number increases, the number of loans users taken decreases.

 'cnt\_ma\_rech30' and 'cnt\_ma\_rech90'- In both, maximum only one time the main account got recharged in last 30 days followed by the 2 times. As the number increases the the slope or the account to be recharged no. of times decreases and at a point no recharge is made.

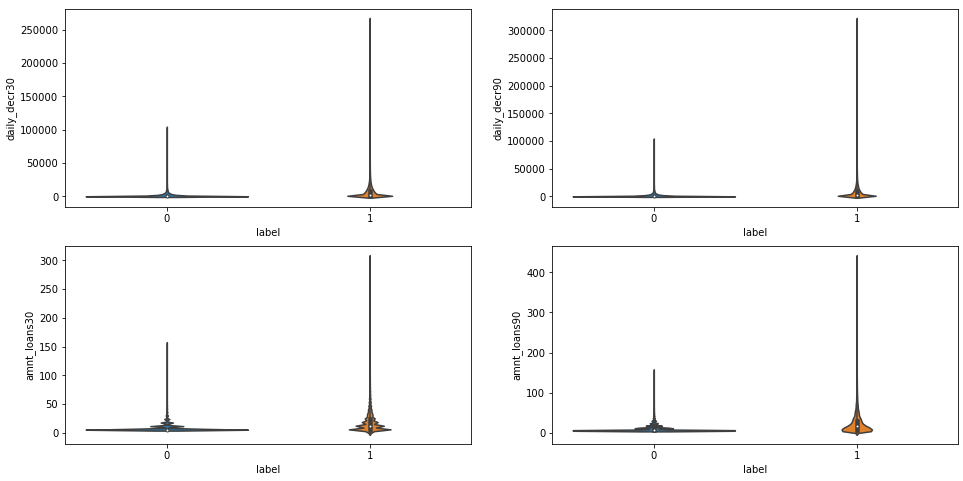
Bivariate analysis looks at two paired data sets, studying whether a relationship exists between them.



The above two jointplot visualize the relation between Total amount of recharge in main account over last 30 days (in Indonesian Rupiah) and over last 90 days (in Indonesian Rupiah) AND Daily amount spent from main account, averaged over last 30 days (in Indonesian Rupiah) and over last 90 days (in Indonesian Rupiah). Both shows that the value being 0.

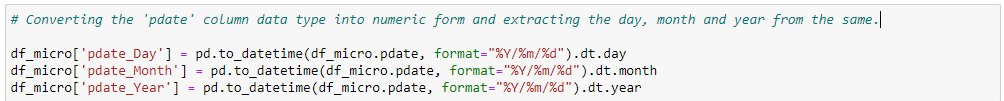


The below Violin plot show how daily\_decr30, daily\_decr90 , amnt\_loans30 and amnt\_loans90 with respect to the customers being defaulter and not.



**DATATYPE TRANSFORMATION**

The attributes -> ‘pdate’ is of object type in the dataset, which needs to be converted into numeric and extracting the day and month from the same using to\_datetime(). Thus, handling the categorical data.



On converting the column data into numeric type and extracting the unnecessary columns, 209593 rows and 35 columns are present in the dataset.

After converting the categorical data into numeric form it is always beneficial to check whether the data is converted into numeric form or not.

Removing skewness gives symmetry or distribute evenly in a set of data. The removed data is referred to as outliers which may cause error if not removed. The highly skewed data affects the standard deviations and counts as an **outliers**. Therefore, after checking the skewness, the highly skewed data was reduced using sqrt function and the outliers are removed using z-score.

After reducing and removing the skewness and outliers respectively present in the dataset. The missing values are found in 'aon', ‘daily\_decr30’, 'daily\_decr90', 'rental30', 'rental90', 'last\_rech\_date\_ma', 'last\_rech\_date\_da', 'medianmarechprebal30' and 'medianmarechprebal90' which were filled by their medians respectively.

The data cleaning is done. So, let's move further and do the testing and training of data by splitting the target and rest variables and selecting the best model by testing various algorithms and then evaluating the same.

**MODEL/S DEVELOPMENT AND EVALUATION**

When separating the input and output variables, in the file of the input variables, each row should correspond to a specific variable. The dataset should be numerical and no null values should be present, therefore preprocessing is done before separating the target and other variables, hence, predicting the results through machine learning.

So, in order to get a more accurate list of variables, the input and output variables are separated.

‘X’ representing the input variables and are separated by dropping the target column “label”. ‘y’ representing the output variable consisting only target variable “label”.

On checking the shape of input and output variables, it is determined that ‘X’ contain 161465 rows and 34 columns whereas ‘y’ contains 161465 rows and 1 column.

**Equalizing the range values (Feature Scaling)**

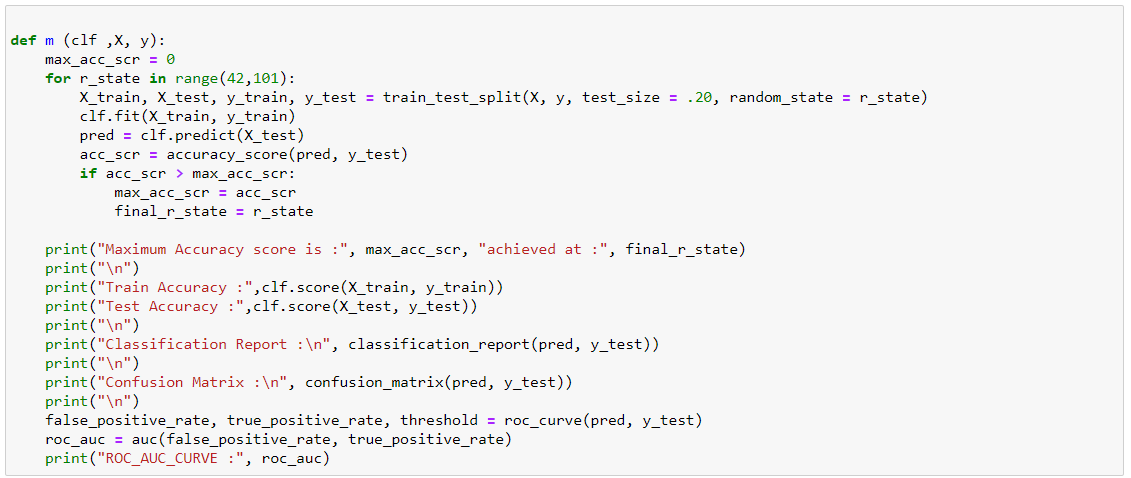
Now, the input variables are scaled because standardization makes the input variables being in a same range of values. Scaled **variable** would be calculated by subtracting mean of the original **variable** from raw value and then divide it by standard deviation of the original **variable**. This can be done by importing ‘StandardScaler’ from sklearn.preprocessing.

**Testing of Identified Approaches**

Training and Testing the models minimizes the effects of data discrepancies and better understand the characteristics of the model.For choosing the best algorithm as a model for this dataset and to get the best accuracy score at the random state, the train and test data is to be split by done by importing ‘train\_test\_split’ from sklearn libraries.

Similarly, cross-validation is used to evaluate machine learning models on a limited data sample like the training data. Thus, this enables the machine to cross check or validate the accuracy scored by training the data. Importing ‘cross\_val\_score’ from sklearn Libraries.

The **training** data is used to make the machine recognizes patterns in the data, the **cross-validation** data is used to ensure better accuracy and efficiency of the algorithm used to train the machine, and the **test** data is used to see how well the machine can predict new answers based on its training.



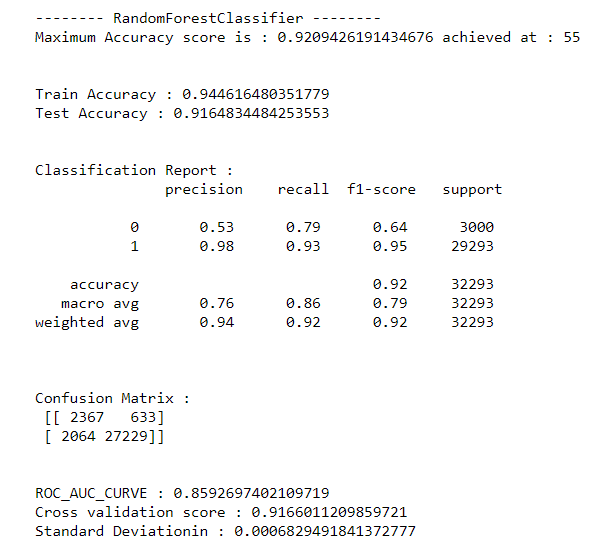
The above code represents a class consisting a ‘for’ loop which will split the train and test data and checking the maximum accuracy score obtained at a random state. Then checking the accuracy score and other scores of algorithms being used in this dataset for the prediction of target value. Thus, each algorithm tested is evaluated too. Other scores includes sklearn classification metrics, which basically helps in evaluating the model that how well the model performance is, it includes:

1. Classification Report, displays the precision, recall, f1-score and support for each class.
2. Confusion Matrix, is a handy presentation of the accuracy of a model with two or more classes.
3. ROC AUC Score, depicts how well our machine learning classifier is performing.

**RESULTS**

Next, algorithms named, Logistic Regression, GaussianNB, DecisionTreeClassifier, and RandomForestClassifier were tested without and with hypertuning parameters using RandomizesSearchCV. On calculating and analyzing all the accuracy score,and on ***Comparing the results of algoritjms with and without hyperparameter tuning,***  **RandomForestClassifier** **is giving the best score among all the algorithms tested i.e. accuracy score being 0.92 at the random state 55.**

Before hyperparameter tuning, the accuracy of RandomForest is mere 0.92 and after parameter tuning, we obtained an accuracy of 0.92 itself. Thus, in this dataset with and without hyperparametr tuning the results are merely same, there is no such big difference. Thus, by comparing all the algorithms tested RANDOMFOREST CLASSIFIER works best as it is giving the higest accuracy among all and hence selected as a model for this dataset.



At last the model was saved as a pickle in a file using joblib from sklearn Externals.

**CONCLUSION**

Predicting Micro Finance Defaulter is really important for the companies or such institutions as it enables them to make strategies and innovations in order to control such scenarios. For any sector, industry etc. customers plays a vital role. Since default is a particularly important issue for client protection, to explore how microfinance institutions (MFIs) treat clients who are unable to repay their loans. It was motivated by a lack of information on the client-facing actions MFIs take when a borrower moves into default.

The dataset used for predicting the defaulters of the Microfinance company concludes that out of the proposed algorithms tested being sample data as 80% for training and 20% for testing, the RandomForestClassifier proved itself efficient as a model giving the maximum and highest accuracy score of 0.92%. The AUC value achieved is 0.85%. Such scores are achieved by performing the effective feature transformation, feature selection, correlation , statistical summary, handling the missing values, skewness, feature engineering, also via pre-processing to remove the outliers, which results the positive effect on the dataset. Hence, the results from this analysis shows that predicting the customer churn with high accuracy is successful.