FLIP ROBO

Micro Credit Defaulter Model

Submitted by:

VINAYAK RATAN

INTRODUCTION

- ## Business Problem Framing

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).

The sample data is provided to us from our client database. It is hereby given to you for this exercise. In order 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.

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.

- ## Conceptual Background of the Domain Problem

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.

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.

Analytical Problem Framing

• Data Sources and their formats

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.

Variable	Definition
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

In [8]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 209593 entries, 1 to 209593
Data columns (total 36 columns):
 #   Column                Non-Null Count   Dtype  
---  ------                --------------   -----  
 0   label                 209593 non-null  int64  
 1   msisdn                209593 non-null  object 
 2   aon                   209593 non-null  float64
 3   daily_decr30          209593 non-null  float64
 4   daily_decr90          209593 non-null  float64
 5   rental30              209593 non-null  float64
 6   rental90              209593 non-null  float64
 7   last_rech_date_ma     209593 non-null  float64
 8   last_rech_date_da     209593 non-null  float64
 9   last_rech_amt_ma      209593 non-null  int64  
 10  cnt_ma_rech30         209593 non-null  int64  
 11  fr_ma_rech30          209593 non-null  float64
 12  sumamnt_ma_rech30     209593 non-null  float64
 13  medianamnt_ma_rech30  209593 non-null  float64
 14  medianmarechprebal30  209593 non-null  float64
 15  cnt_ma_rech90         209593 non-null  int64  
 16  fr_ma_rech90          209593 non-null  int64  
 17  sumamnt_ma_rech90     209593 non-null  int64  
 18  medianamnt_ma_rech90  209593 non-null  float64
 19  medianmarechprebal90  209593 non-null  float64
 20  cnt_da_rech30         209593 non-null  float64
 21  fr_da_rech30          209593 non-null  float64
 22  cnt_da_rech90         209593 non-null  int64  
 23  fr_da_rech90          209593 non-null  int64  
 24  cnt_loans30           209593 non-null  int64  
 25  amnt_loans30          209593 non-null  int64  
 26  maxamnt_loans30       209593 non-null  float64
 27  medianamnt_loans30    209593 non-null  float64
 28  cnt_loans90           209593 non-null  float64
 29  amnt_loans90          209593 non-null  int64  
 30  maxamnt_loans90       209593 non-null  int64  
 31  medianamnt_loans90    209593 non-null  float64
 32  payback30             209593 non-null  float64
 33  payback90             209593 non-null  float64
 34  pcircle               209593 non-null  object 
 35  pdate                 209593 non-null  object 
dtypes: float64(21), int64(12), object(3)
memory usage: 59.2+ MB
```

- One can see that there are only three object data type features - msisdn, pcircle, pdate

- **Data Pre-processing Done**

In [9]:
```
# Checking the presence of null values
df.isnull().sum()
```

Out[9]:
```
label                   0
msisdn                  0
aon                     0
daily_decr30            0
daily_decr90            0
rental30                0
rental90                0
last_rech_date_ma       0
last_rech_date_da       0
last_rech_amt_ma        0
cnt_ma_rech30           0
fr_ma_rech30            0
sumamnt_ma_rech30       0
medianamnt_ma_rech30    0
medianmarechprebal30    0
cnt_ma_rech90           0
fr_ma_rech90            0
sumamnt_ma_rech90       0
medianamnt_ma_rech90    0
medianmarechprebal90    0
cnt_da_rech30           0
fr_da_rech30            0
cnt_da_rech90           0
fr_da_rech90            0
cnt_loans30             0
amnt_loans30            0
maxamnt_loans30         0
medianamnt_loans30      0
cnt_loans90             0
amnt_loans90            0
maxamnt_loans90         0
medianamnt_loans90      0
payback30               0
payback90               0
pcircle                 0
pdate                   0
dtype: int64
```

- There are no null values present in the data

In [12]:
```
# removing duplicates from msisdn
df = df.drop_duplicates(subset = 'msisdn',keep='first')
df.shape
```

Out[12]: (186243, 36)

After removing duplicates, we can see that the data size reduced from 209593 to 186243

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

  Hardware requirements:

  ▪ Laptop or PC to analyse the data

  ▪ Software requirements: 4GB/8GB RAM (8GB or high preferred), minimum 1GB graphics card (any), 512GB HDD.

  ▪ Tools Used: Jupyter Notebook – Entire model

  ▪ Libraries used: NumPy (for numerical computations like imputations), Pandas (used reading the data, data cleaning, data manipulation, correlation, summary statistics, skewness, feature engineering) Seaborn (for plotting), Matplotlib (for plotting), scikit-learn, SciPy (for z score), stats models (for VIF to check multicollinearity), joblib (for pickling i.e. deploying the best model trained)

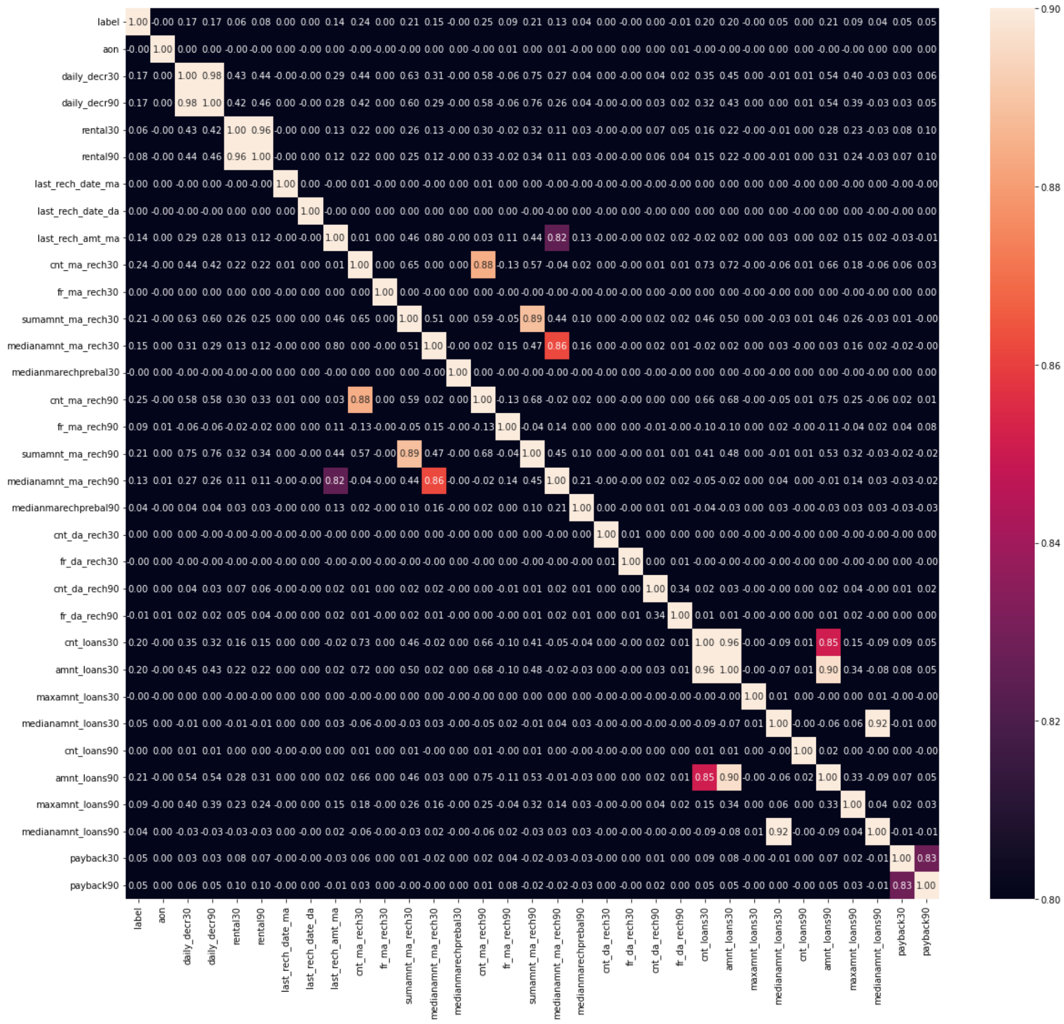
Model/s Development and Evaluation

- **Identification of possible problem-solving approaches (methods)**

In [10]:
```
# For numerical features
df.describe().T
```

Out[10]:

	count	mean	std	min	25%	50%	75%	max
label	209593.0	0.875177	0.330519	0.000000	1.000	1.000000	1.00	1.000000
aon	209593.0	8112.343445	75696.082531	-48.000000	246.000	527.000000	982.00	999860.755168
daily_decr30	209593.0	5381.402289	9220.623400	-93.012667	42.440	1469.175667	7244.00	265926.000000
daily_decr90	209593.0	6082.515068	10918.812767	-93.012667	42.692	1500.000000	7802.79	320630.000000
rental30	209593.0	2692.581910	4308.586781	-23737.140000	280.420	1083.570000	3356.94	198926.110000
rental90	209593.0	3483.406534	5770.461279	-24720.580000	300.260	1334.000000	4201.79	200148.110000
last_rech_date_ma	209593.0	3755.847800	53905.892230	-29.000000	1.000	3.000000	7.00	998650.377733
last_rech_date_da	209593.0	3712.202921	53374.833430	-29.000000	0.000	0.000000	0.00	999171.809410
last_rech_amt_ma	209593.0	2064.452797	2370.786034	0.000000	770.000	1539.000000	2309.00	55000.000000
cnt_ma_rech30	209593.0	3.978057	4.256090	0.000000	1.000	3.000000	5.00	203.000000
fr_ma_rech30	209593.0	3737.355121	53643.625172	0.000000	0.000	2.000000	6.00	999606.368132
sumamnt_ma_rech30	209593.0	7704.501157	10139.621714	0.000000	1540.000	4628.000000	10010.00	810096.000000
medianamnt_ma_rech30	209593.0	1812.817952	2070.864620	0.000000	770.000	1539.000000	1924.00	55000.000000
medianmarechprebal30	209593.0	3851.927942	54006.374433	-200.000000	11.000	33.900000	83.00	999479.419319
cnt_ma_rech90	209593.0	6.315430	7.193470	0.000000	2.000	4.000000	8.00	336.000000



- .

```
In [11]: # for object data type features
         df.describe(include='object').T
```

Out[11]:

	count	unique	top	freq
msisdn	209593	186243	04581I85330	7
pcircle	209593	1	UPW	209593
pdate	209593	82	2016-07-04	3150

- Mean is greater than median for all the columns, hence data is right skewed
- There are outliers present in the dataset when we look at the large difference b/w 75th percentile and maximum, hence the outliers must be treated properly
- since the dataset contains columns with 90days, hence we can see that from pdate column the data is only for three months
- msidn is a mobile number of user and mobile number is unique for each customer. There are 186243 unique number out of 209593, rest are duplicates, hence must be removed

- pcircle column contains only one unique value, hence this can be dropped.

- ## Testing of Identified Approaches (Algorithms)

In [96]:
```python
# Importing the model libraries
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report
```

In [97]:
```python
lg = LogisticRegression()
gnb = GaussianNB()
dtc = DecisionTreeClassifier()
knn = KNeighborsClassifier()

model = [lg, gnb, dtc, knn]
```

- ## Run and evaluate selected models

Maximum accuracy score of DecisionTreeClassifier() is 0.9036225332792966 at Random state 8

In [99]:
```python
x_train, x_test, y_train, y_test= train_test_split(train_x,train_y,test_size= 0.2, random_state=8)
dtc.fit(x_train, y_train)
train_score = dtc.score(x_train, y_train)
test_score = dtc.score(x_train, y_train)
pred = dtc.predict(x_test)
print(f'-------------------- {dtc} --------------------')
acc = accuracy_score(y_test, pred)
print("Accuracy score of ", dtc, 'is:', acc)
print(f'Training score of {dtc} is {train_score}')
print(f'Testing score of {dtc} is {test_score}')
print("Confusion Matrix:\n",confusion_matrix(y_test, pred))
print('Classification Report:\n',classification_report(y_test, pred))
print('*'*120,'\n')
```

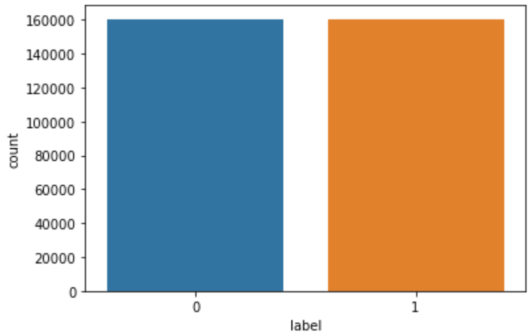
```
-------------------- DecisionTreeClassifier() --------------------
Accuracy score of  DecisionTreeClassifier() is: 0.9032328459643981
Training score of DecisionTreeClassifier() is 0.9999961030661075
Testing score of DecisionTreeClassifier() is 0.9999961030661075
Confusion Matrix:
 [[29117  2840]
 [ 3368 28829]]
Classification Report:
               precision    recall  f1-score   support

           0       0.90      0.91      0.90     31957
           1       0.91      0.90      0.90     32197

    accuracy                           0.90     64154
   macro avg       0.90      0.90      0.90     64154
weighted avg       0.90      0.90      0.90     64154
```

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

In [93]:
```python
# Before proceeding with model building we need to apply SMOTE since the data is imbalanced
from imblearn.over_sampling import SMOTE
smt_loan = SMOTE()
train_x, train_y = smt_loan.fit_resample(X,y)
```

In [94]:
```python
train_y.value_counts()
```

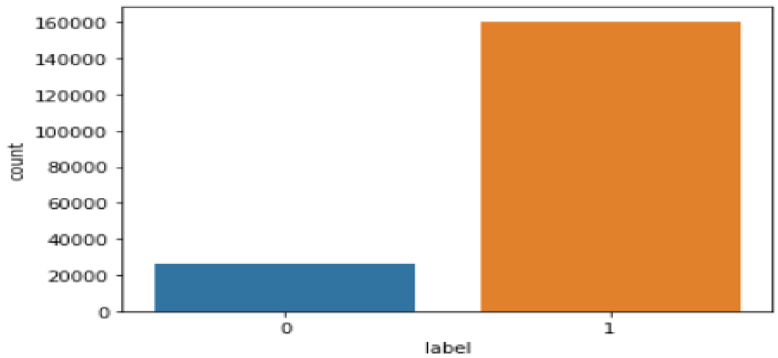
Out[94]:
```
0    160383
1    160383
Name: label, dtype: int64
```



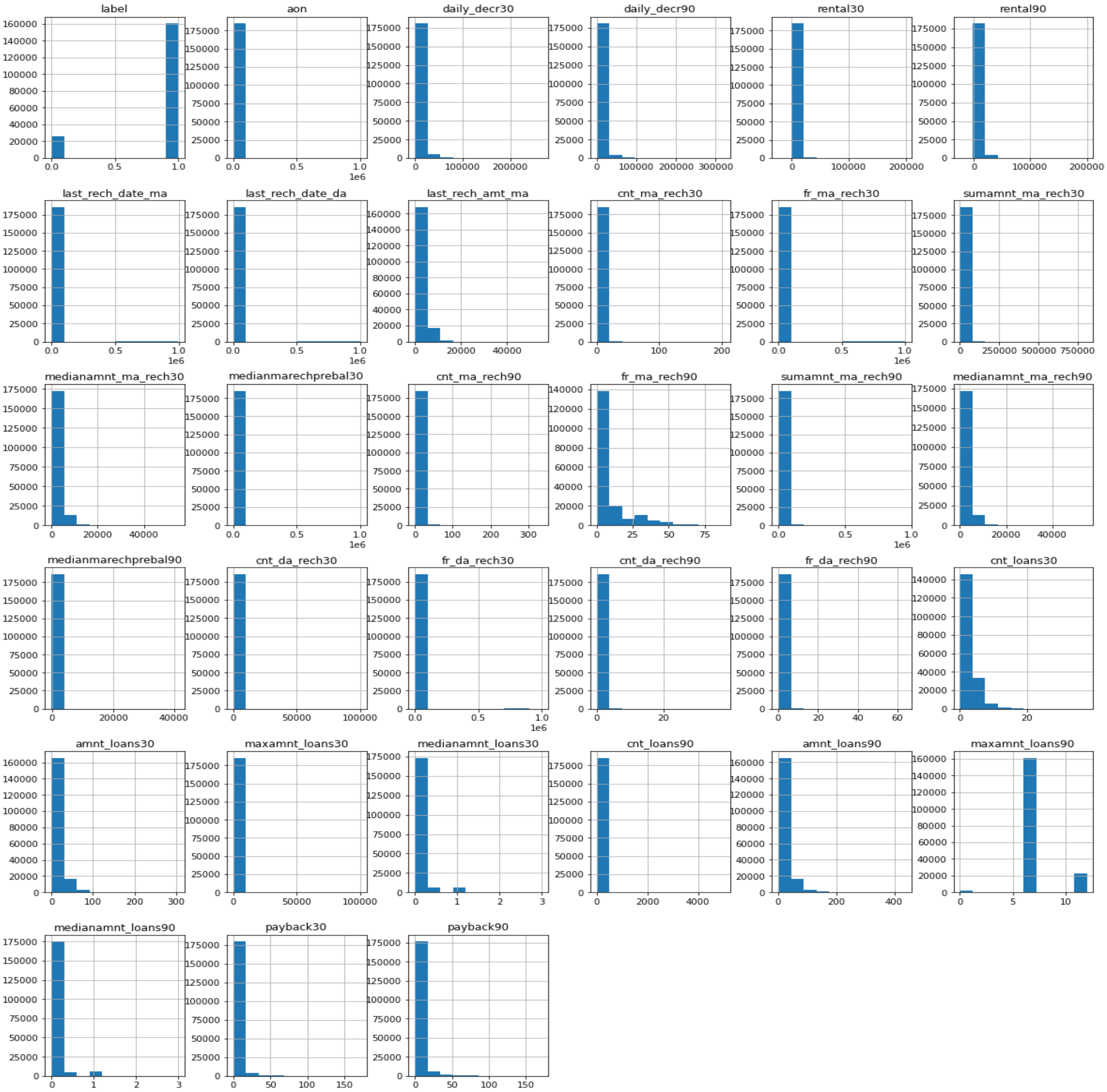

```
Best parameters: {'criterion': 'gini', 'max_depth': 10, 'min_samples_le
af': 5}
Best Estimator: DecisionTreeClassifier(max_depth=10, min_samples_leaf=5
)
Final Accuracy with Decision Tree Classifier: 0.8815506437634442
```

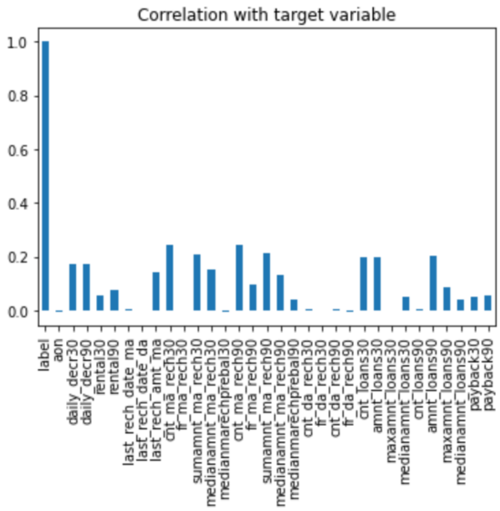
- ## Visualizations

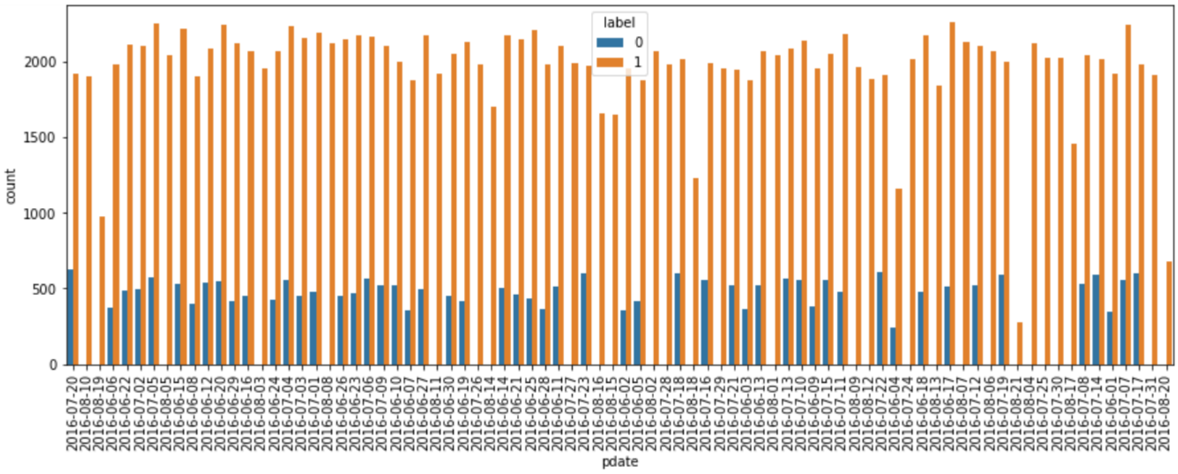
```
In [15]: print(df['label'].value_counts())
         sns.countplot(x = 'label', data = df)

         1    160383
         0     25860
         Name: label, dtype: int64

Out[15]: <AxesSubplot:xlabel='label', ylabel='count'>
```

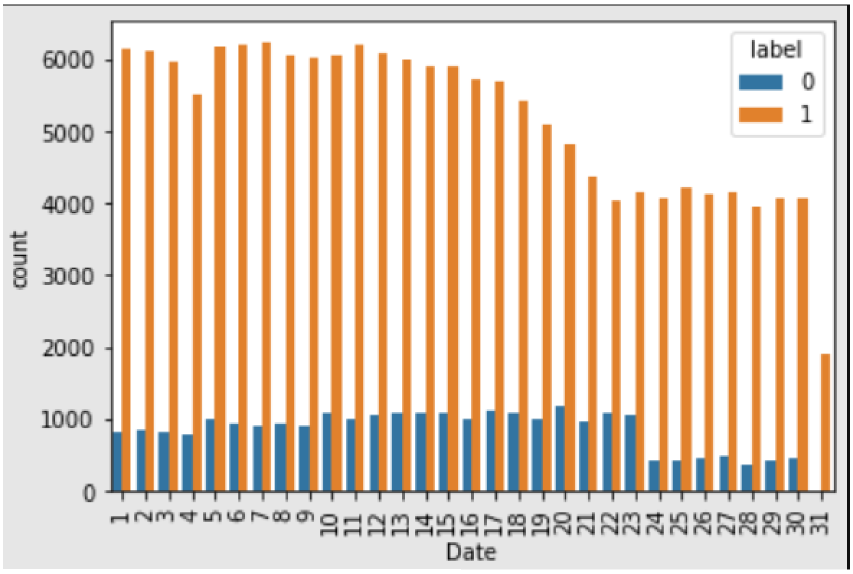


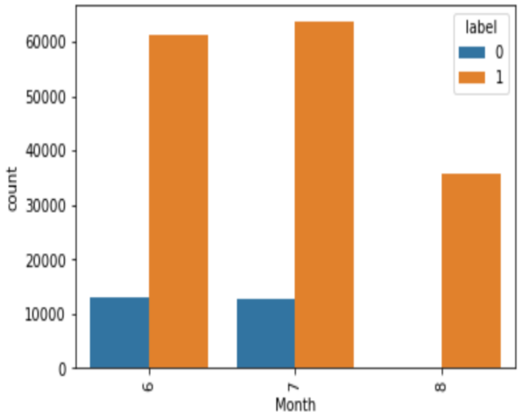
- 'Label' is the target variable wherein '1' indicates loan payed back (Non - defaulter) and '0' indicates loan has not payed back (Defaulter)
- We can see data is not balanced, so we need to balance the data before model building

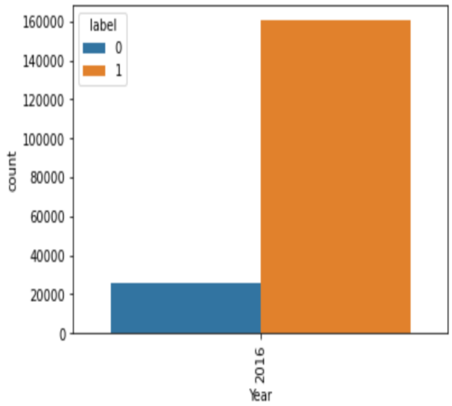

label
aon
daily_decr30
daily_decr90
rental30
rental90
last_rech_date_ma
last_rech_date_da
last_rech_amt_ma
cnt_ma_rech30
fr_ma_rech30
sumamnt_ma_rech30
medianamnt_ma_rech30
medianmarechprebal30
cnt_ma_rech90
fr_ma_rech90
sumamnt_ma_rech90
medianamnt_ma_rech90
medianmarechprebal90
cnt_da_rech30
fr_da_rech30
cnt_da_rech90
fr_da_rech90
cnt_loans30
amnt_loans30
maxamnt_loans30
medianamnt_loans30
cnt_loans90
amnt_loans90
maxamnt_loans90
medianamnt_loans90
payback30
payback90


Correlation with target variable
1.0
0.8
0.6
0.4
0.2
0.0
label
aon
daily_decr30
daily_decr90
rental30
rental90
last_rech_date_ma
last_rech_date_da
last_rech_amt_ma
cnt_ma_rech30
fr_ma_rech30
sumamnt_ma_rech30
medianamnt_ma_rech30
medianmarechprebal30
cnt_ma_rech90
fr_ma_rech90
sumamnt_ma_rech90
medianamnt_ma_rech90
medianmarechprebal90
cnt_da_rech30
fr_da_rech30
cnt_da_rech90
fr_da_rech90
cnt_loans30
amnt_loans30
maxamnt_loans30
medianamnt_loans30
cnt_loans90
amnt_loans90
maxamnt_loans90
medianamnt_loans90
payback30
payback90







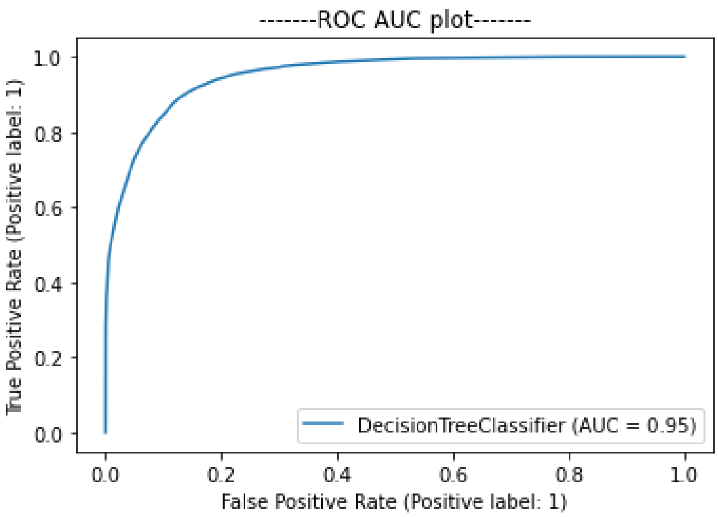


- **Interpretation of the Results**
  - 'Label' is the target variable wherein '1' indicates loan paid back (non-defaulter) and '0' indicates loan has not paid back.

- We can see that the times account got recharged in last 30 days in maximum 1 time. Many recharge less than 10 times every month
- Almost 30% of the data will we lost if we consider the model after removing the outliers, hence data is considered without removing the outliers.

CONCLUSION

- **Key Findings and Conclusions of the Study**
  - Final Accuracy with Decision Tree Classifier: 0.8815506437634442 after hyperparameter training.
  - ROC AUC plot is given below with score of 0.95.



Predicted	Original	Total
1	1	28709
0	0	27846
1	0	4111
0	1	3488

  - We can see that model is predicting well
  - Out of 64154 test data model is predicting accurately on 28709+27846 = 56,555 occasions and wrong on 4111+3488 = 7,599 occasions

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

  We can think of dropping the few features related to 30days like daily_decr30, rental30, amnt_loans30, medianamnt_loans30, cnt_ma_rech30 to improve the accuracy of the trained models.