FLIP ROBO

# Micro Credit Loan Defaulter Prediction

Submitted by:

Moncy Kurien

# ACKNOWLEDGMENT

References:

1. Data description
2. Use case document.

# INTRODUCTION

## Business Problem Framing

A telecom company is collaborating with an MFI to provide micro-credit on mobile balances to be paid back in 5 days for its customers. 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.

## Conceptual Background of the Domain Problem

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 loan amounts provided are only 5 or 10 Indonasian Rupiah.
- The amounts to be paid back are 6 and 12 for the loan amount 5 and 10 respectively.
- 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.

## Motivation for the Problem Undertaken

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 telecom company is collaborating with the MFI to provide a better solution for their consumers in need. It is important to reach the right consumers for this collaboration to win. We need a system that can find the right consumers so that the consumers and the company can win.

# Analytical Problem Framing

## ● Mathematical/ Analytical Modeling of the Problem

Decision Tree is used to solve the problem. Linear models also worked very well in classifying the classes. However, the decision tree did a better job. Also, since there are outliers, and since the decision tree is not impacted by outliers, I have chosen the decision tree as the final model.

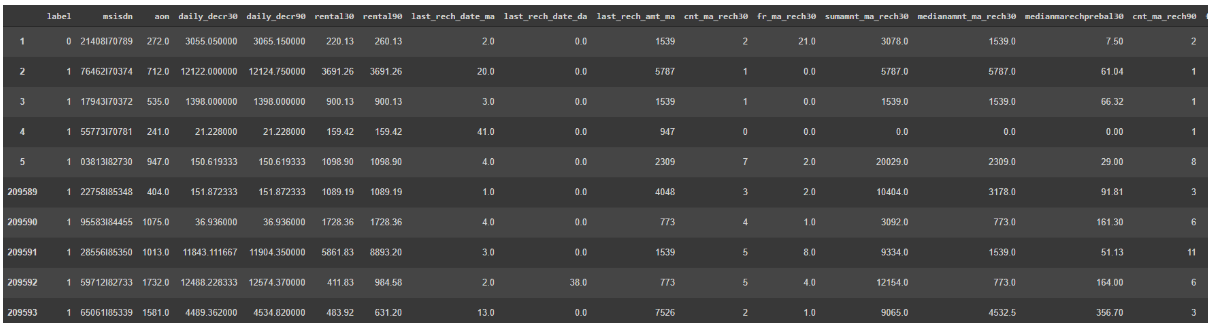
## ● Data Sources and their formats

The data is provided by the client in the .csv format.

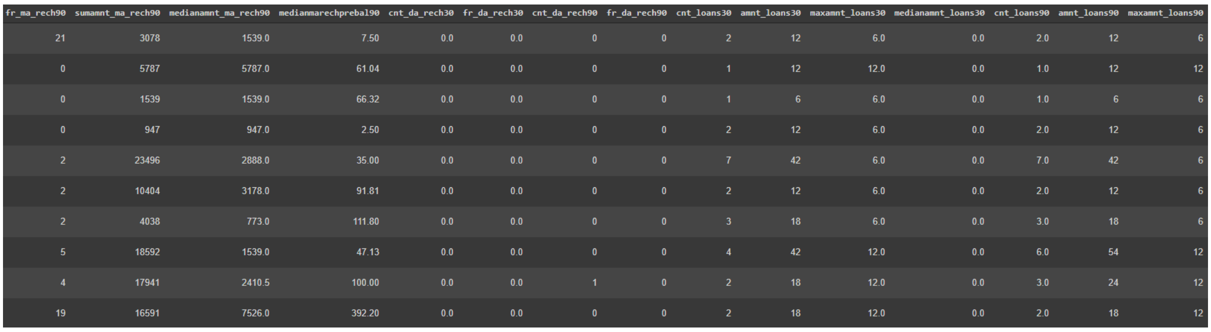
Data description:

| 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) |
| medianmarechpre | Median of main account balance just before recharge in last 90 days at |

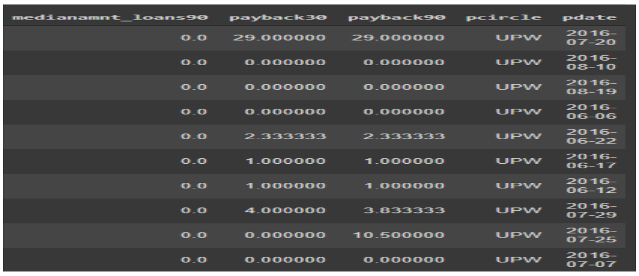
| | |
|---|---|
| bal90 | user level (in Indonasian Rupiah) |
| cnt_da_rech30 | Number of times data account got recharged in last 30 days |
| fr_da_rech30 | Frequency of data account recharged in last 30 days |
| cnt_da_rech90 | Number of times data account got recharged in last 90 days |
| fr_da_rech90 | 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 |



| | label | msisdn | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 21408I70789 | 272.0 | 3055.050000 | 3065.150000 | 220.13 | 260.13 | 2.0 | 0.0 | 1539 | 2 | 21.0 | 3078.0 | 1539.0 | 7.50 | 2 |
| 2 | 1 | 76462I70374 | 712.0 | 12122.000000 | 12124.750000 | 3691.26 | 3691.26 | 20.0 | 0.0 | 5787 | 1 | 0.0 | 5787.0 | 5787.0 | 61.04 | 1 |
| 3 | 1 | 17943I70372 | 535.0 | 1398.000000 | 1398.000000 | 900.13 | 900.13 | 3.0 | 0.0 | 1539 | 1 | 0.0 | 1539.0 | 1539.0 | 66.32 | 1 |
| 4 | 1 | 55773I70781 | 241.0 | 21.228000 | 21.228000 | 159.42 | 159.42 | 41.0 | 0.0 | 947 | 0 | 0.0 | 0.0 | 0.0 | 0.00 | 1 |
| 5 | 1 | 03813I82730 | 947.0 | 150.619333 | 150.619333 | 1098.90 | 1098.90 | 4.0 | 0.0 | 2309 | 7 | 2.0 | 20029.0 | 2309.0 | 29.00 | 8 |
| 209589 | 1 | 22758I85348 | 404.0 | 151.872333 | 151.872333 | 1089.19 | 1089.19 | 1.0 | 0.0 | 4048 | 3 | 2.0 | 10404.0 | 3178.0 | 91.81 | 3 |
| 209590 | 1 | 95583I84455 | 1075.0 | 36.936000 | 36.936000 | 1728.36 | 1728.36 | 4.0 | 0.0 | 773 | 4 | 1.0 | 3092.0 | 773.0 | 161.30 | 6 |
| 209591 | 1 | 28556I85350 | 1013.0 | 11843.111667 | 11904.350000 | 5861.83 | 8893.20 | 3.0 | 0.0 | 1539 | 5 | 8.0 | 9334.0 | 1539.0 | 51.13 | 11 |
| 209592 | 1 | 59712I82733 | 1732.0 | 12488.228333 | 12574.370000 | 411.83 | 984.58 | 2.0 | 38.0 | 773 | 5 | 4.0 | 12154.0 | 773.0 | 164.00 | 6 |
| 209593 | 1 | 65061I85339 | 1581.0 | 4489.362000 | 4534.820000 | 483.92 | 631.20 | 13.0 | 0.0 | 7526 | 2 | 1.0 | 9065.0 | 4532.5 | 356.70 | 3 |



| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 21 | 3078 | 1539.0 | 7.50 | 0.0 | 0.0 | 0 | 0 | 2 | 12 | 6.0 | 0.0 | 2.0 | 12 | 6 |
| 0 | 5787 | 5787.0 | 61.04 | 0.0 | 0.0 | 0 | 0 | 1 | 12 | 12.0 | 0.0 | 1.0 | 12 | 12 |
| 0 | 1539 | 1539.0 | 66.32 | 0.0 | 0.0 | 0 | 0 | 1 | 6 | 6.0 | 0.0 | 1.0 | 6 | 6 |
| 0 | 947 | 947.0 | 2.50 | 0.0 | 0.0 | 0 | 0 | 2 | 12 | 6.0 | 0.0 | 2.0 | 12 | 6 |
| 2 | 23496 | 2888.0 | 35.00 | 0.0 | 0.0 | 0 | 0 | 7 | 42 | 6.0 | 0.0 | 7.0 | 42 | 6 |
| 2 | 10404 | 3178.0 | 91.81 | 0.0 | 0.0 | 0 | 0 | 2 | 12 | 6.0 | 0.0 | 2.0 | 12 | 6 |
| 2 | 4038 | 773.0 | 111.80 | 0.0 | 0.0 | 0 | 0 | 3 | 18 | 6.0 | 0.0 | 3.0 | 18 | 6 |
| 5 | 18592 | 1539.0 | 47.13 | 0.0 | 0.0 | 0 | 0 | 4 | 42 | 12.0 | 0.0 | 6.0 | 54 | 12 |
| 4 | 17941 | 2410.5 | 100.00 | 0.0 | 0.0 | 1 | 0 | 2 | 18 | 12.0 | 0.0 | 3.0 | 24 | 12 |
| 19 | 16591 | 7526.0 | 392.20 | 0.0 | 0.0 | 0 | 0 | 2 | 18 | 12.0 | 0.0 | 2.0 | 18 | 12 |



| medianamnt_loans90 | payback30 | payback90 | pcircle | pdate |
|---|---|---|---|---|
| 0.0 | 29.000000 | 29.000000 | UPW | 2016-07-20 |
| 0.0 | 0.000000 | 0.000000 | UPW | 2016-08-10 |
| 0.0 | 0.000000 | 0.000000 | UPW | 2016-08-19 |
| 0.0 | 0.000000 | 0.000000 | UPW | 2016-06-06 |
| 0.0 | 2.333333 | 2.333333 | UPW | 2016-06-22 |
| 0.0 | 1.000000 | 1.000000 | UPW | 2016-06-17 |
| 0.0 | 1.000000 | 1.000000 | UPW | 2016-06-12 |
| 0.0 | 4.000000 | 3.833333 | UPW | 2016-07-29 |
| 0.0 | 0.000000 | 10.500000 | UPW | 2016-07-25 |
| 0.0 | 0.000000 | 0.000000 | UPW | 2016-07-07 |

## Data Preprocessing Done

| Variable | problem | Assumptions | Cleaning steps |
|---|---|---|---|
| msisdn | Contains the letter ‘I’. | Mobile numbers are integers | Removed the letter ‘I’ |
| aon | 1. Contains negative values.<br>2. Contains very large positive values. Above 10000, the next consecutive value is 500101(1370 years). | 1.Age or number of days is never negative. The negative sign could be a data entry error.<br>2. Extremely large positive values are data entered in minutes instead of days. | 1.Removed negative sign of the negative values.<br>2. Converted very large values from minutes to days. |
| daily_decr30 | 1.Contains negative values. | 1.Amount spent cannot be negative.The negative sign could be a data entry error. | 1.Removed negative sign from the values. |
| daily_decr90 | 1.Contains negative values. | 1.Amount spent cannot be negative.The negative sign could be a data entry error. | 1.Removed negative sign from the values. |
| last_rech_date_ma | 1.Negative values are present.<br>2.Large positive values are present | 1. Number of days cannot be negative.<br>2. Extreme positive values show unrealistic years. For example the next consecutive number after 150 is 500152. Taking into account the values under the 3rd quartile, these large numbers are data in seconds and not in days. | 1.Removed negative sign from the values.<br>2.Considered the large positive values as data entered in seconds. Converted those to days. |
| last_rech_date_da | 1.Negative values are present.<br>2.Large positive values are present | 1. Number of days cannot be negative.<br>2. Extreme positive values show unrealistic years. For example the next consecutive number after 150 is 500032. Taking into | 1.Removed negative sign from the values.<br>2.Considered the large positive values as data entered in seconds. Converted those to days. |

| | | | |
|---|---|---|---|
| | | account the values under the 3rd quartile, these large numbers are data in seconds and not in days. | |
| fr_ma_rech30 | 1. There are some extreme positive values in the variable. These values are very unrealistic. | 1. Since this measure is within 30 days, consider anything greater than 29 as an outlier. | 1. Comparing the 75th quantile, the larger positive outliers will be considered as seconds and will be converted to days. |
| cnt_da_rech30 | Extremely large positive outliers in the data. | 75% of data are 0 and other data except the outliers are very small integers. To bring the outliers to the range, the variable needs a transformation. | Cube root transformation performed on the variable. |
| fr_da_rech30 | 1. There are some extreme positive values in the variable. These values are very unrealistic. | 1. Since this measure is within 30 days, consider anything greater than 29 as an outlier. | 1. Comparing the 75th quantile, the larger positive outliers will be considered as seconds and will be converted to days. |
| maxamnt_loans30 | There are extreme positive values.<br>There are values other than 6 and 12. | There are only two options 6 and 12. And 0 for those that did not take a loan. | Replaced the other values with the mode 6. |
| payback30 | There are outliers. | The customer will be a defaulter if they don't pay back the loan amount within 5 days of issuing the loan,Assuming that the Average payback value to be less than or equal to 5 for records with label = 1 and Average payback value to be greater than 5 for records with label = 0. | Set payback30 as 8 for records that have label = 0 and payback30 within 5. |
| payback90 | There are outliers. | The customer will be a defaulter if they don't pay back the loan amount within 5 days of issuing the loan,Assuming that the | Set payback30 as 8 for records that have label = 0 and payback30 within 5. |

| | | | |
|---|---|---|---|
| | | Average payback value to be less than or equal to 5 for records with label = 1 and Average payback value to be greater than 5 for records with label = 0. | |

- Data Inputs- Logic- Output Relationships
  - All the inputs used are numeric continuous and discrete variables.
  - The output variable is a binary variable with 0s and 1s.
- State the set of assumptions (if any) related to the problem under consideration
  - The customer will be a defaulter if they don't pay back the loan amount within 5 days of issuing the loan,Assuming that the Average payback value to be less than or equal to 5 for records with label = 1 and Average payback value to be greater than 5 for records with label = 0.
  - The days cannot be negative.
  - The amount spent cannot be negative.
- Hardware and Software Requirements and Tools Used

  1. Google Colab
  2. SKLEARN
  3. MATPLOTLIB
  4. SEABORN
  5. PANDAS
  6. NUMPY

## Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)
  - The target variable contains 0s and 1s. So I approached this problem as a binary classification problem.
  - Select features that are useful.

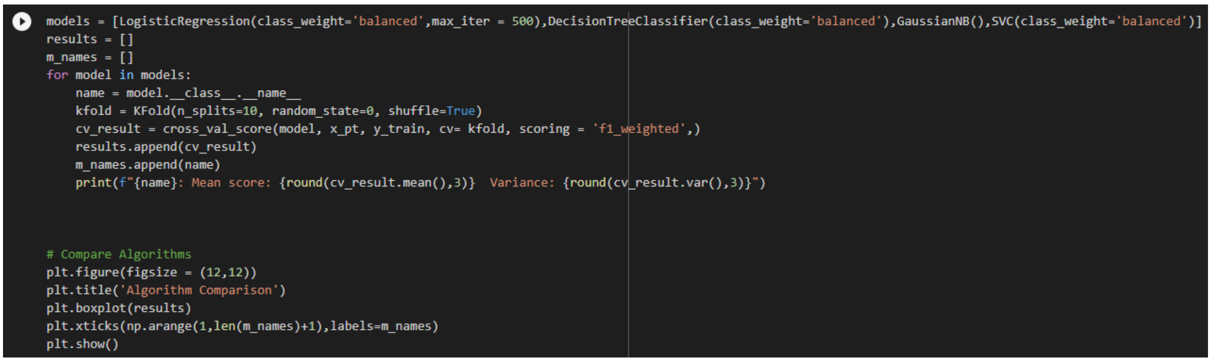
- Scale and Transform Scale the input data.
- Test different Classification models with the data to select an appropriate model.
- Hyper parameters tune the selected models and choose the best model based on the evaluation metrics used.

## Testing of Identified Approaches (Algorithms)

- Simple Classification Models:
  - LogisticRegression
  - DecisionTreeClassifier
  - GaussianNB
  - SVC
- Ensemble Models:
  - AdaBoostClassifier
  - GradientBoostingClassifier
  - RandomForestClassifier
  - XGBClassifier

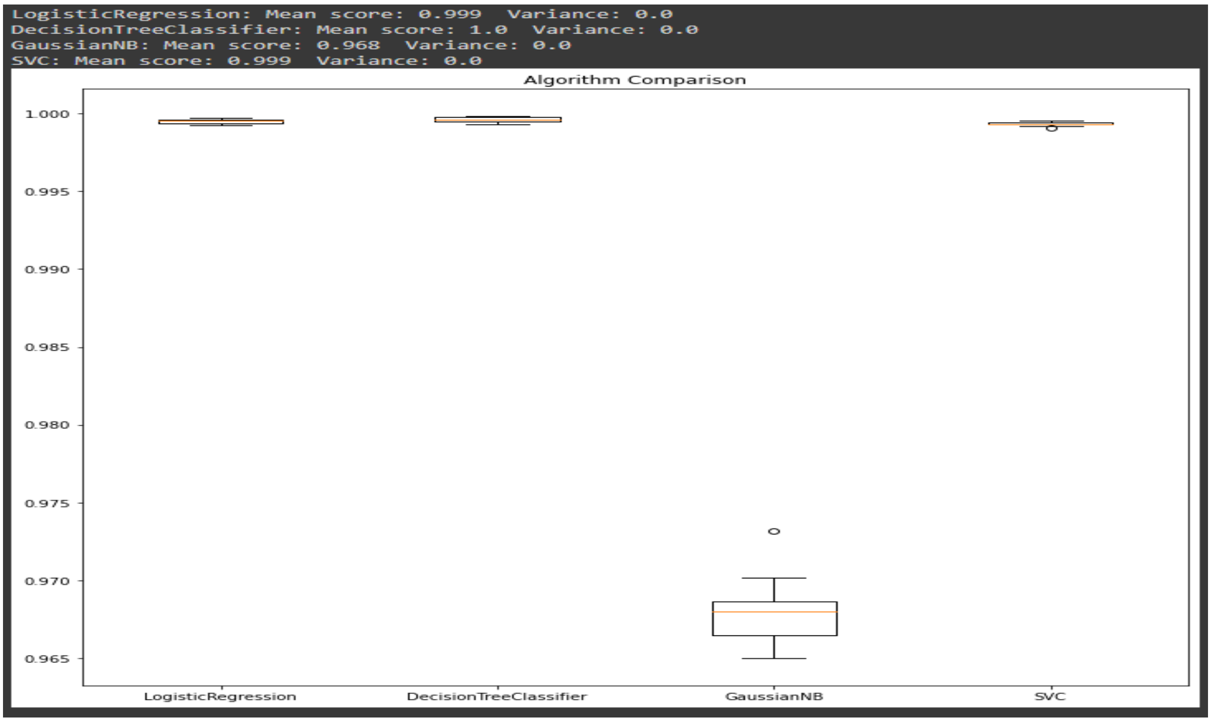
## Run and Evaluate selected models

**Selecting model from simple models:**



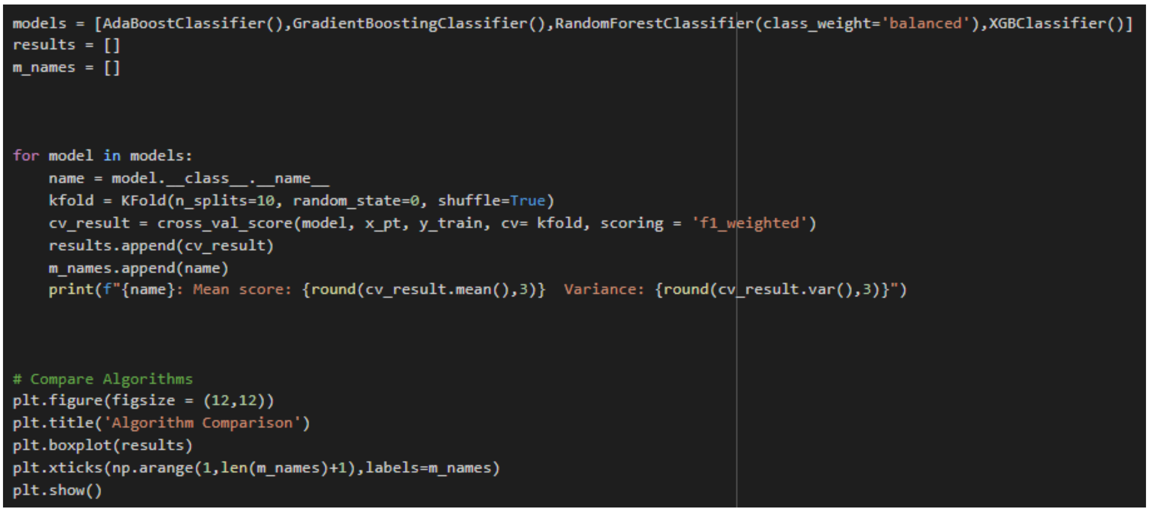
```
models = [LogisticRegression(class_weight='balanced',max_iter = 500),DecisionTreeClassifier(class_weight='balanced'),GaussianNB(),SVC(class_weight='balanced')]
results = []
m_names = []
for model in models:
    name = model.__class__.__name__
    kfold = KFold(n_splits=10, random_state=0, shuffle=True)
    cv_result = cross_val_score(model, x_pt, y_train, cv= kfold, scoring = 'f1_weighted',)
    results.append(cv_result)
    m_names.append(name)
    print(f"{name}: Mean score: {round(cv_result.mean(),3)}  Variance: {round(cv_result.var(),3)}")



# Compare Algorithms
plt.figure(figsize = (12,12))
plt.title('Algorithm Comparison')
plt.boxplot(results)
plt.xticks(np.arange(1,len(m_names)+1),labels=m_names)
plt.show()
```

## Results:



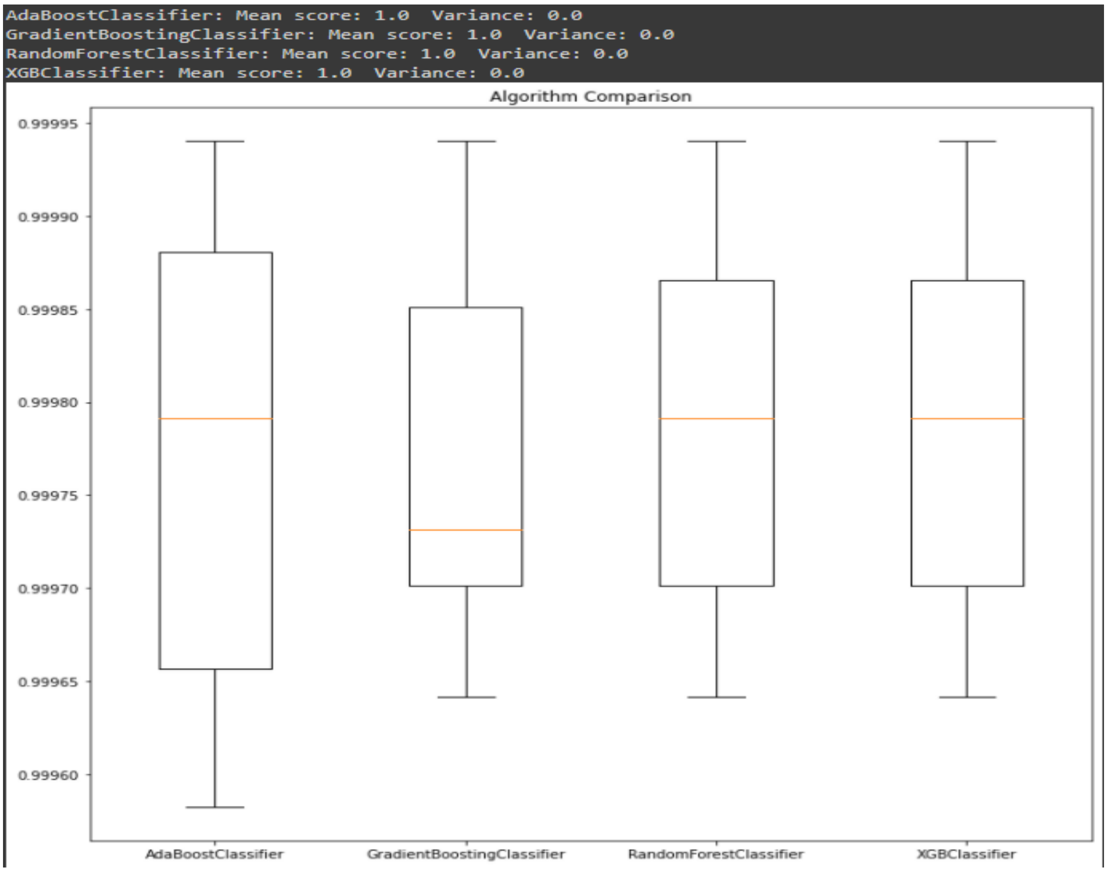
```
LogisticRegression: Mean score: 0.999  Variance: 0.0
DecisionTreeClassifier: Mean score: 1.0  Variance: 0.0
GaussianNB: Mean score: 0.968  Variance: 0.0
SVC: Mean score: 0.999  Variance: 0.0
```

## Ensemble Techniques:



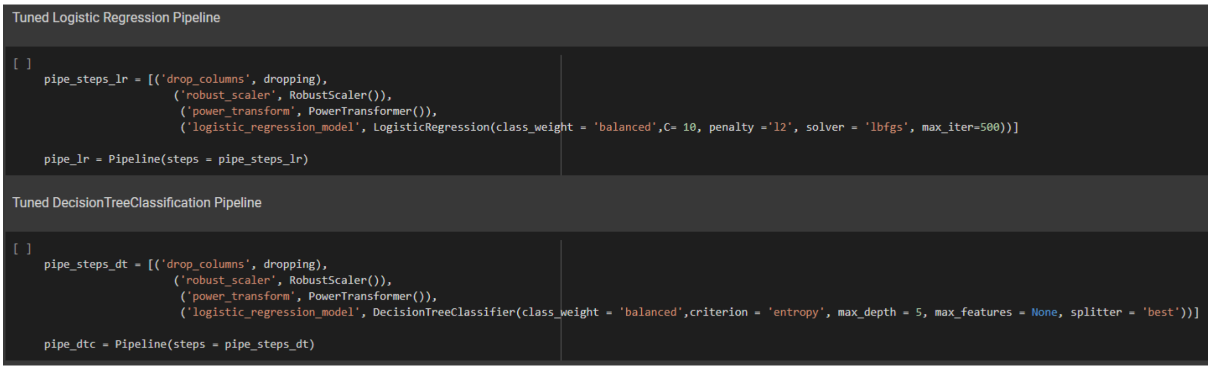
```
models = [AdaBoostClassifier(),GradientBoostingClassifier(),RandomForestClassifier(class_weight='balanced'),XGBClassifier()]
results = []
m_names = []


for model in models:
    name = model.__class__.__name__
    kfold = KFold(n_splits=10, random_state=0, shuffle=True)
    cv_result = cross_val_score(model, x_pt, y_train, cv= kfold, scoring = 'f1_weighted')
    results.append(cv_result)
    m_names.append(name)
    print(f"{name}: Mean score: {round(cv_result.mean(),3)}  Variance: {round(cv_result.var(),3)}")


# Compare Algorithms
plt.figure(figsize = (12,12))
plt.title('Algorithm Comparison')
plt.boxplot(results)
plt.xticks(np.arange(1,len(m_names)+1),labels=m_names)
plt.show()
```

## Results:



```
AdaBoostClassifier: Mean score: 1.0  Variance: 0.0
GradientBoostingClassifier: Mean score: 1.0  Variance: 0.0
RandomForestClassifier: Mean score: 1.0  Variance: 0.0
XGBClassifier: Mean score: 1.0  Variance: 0.0
```

## Evaluation of tuned Final model pipelines:



Tuned Logistic Regression Pipeline

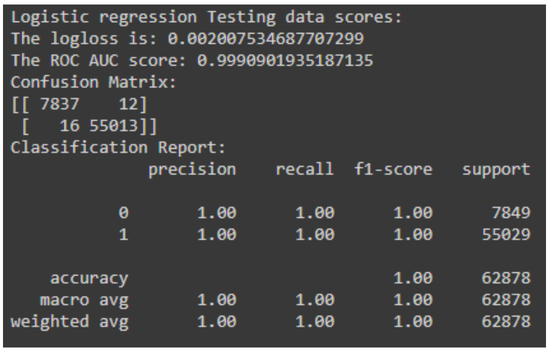
```
pipe_steps_lr = [('drop_columns', dropping),
                 ('robust_scaler', RobustScaler()),
                  ('power_transform', PowerTransformer()),
                  ('logistic_regression_model', LogisticRegression(class_weight = 'balanced',C= 10, penalty ='l2', solver = 'lbfgs', max_iter=500))]

pipe_lr = Pipeline(steps = pipe_steps_lr)
```

Tuned DecisionTreeClassification Pipeline

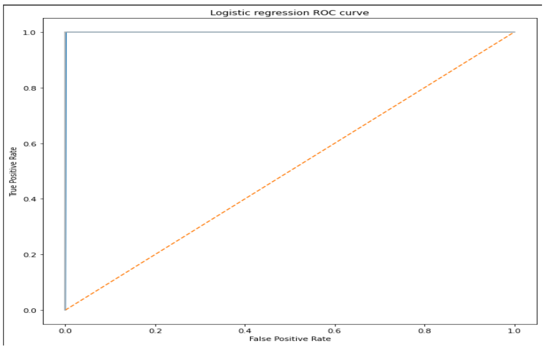
```
pipe_steps_dt = [('drop_columns', dropping),
                 ('robust_scaler', RobustScaler()),
                  ('power_transform', PowerTransformer()),
                  ('logistic_regression_model', DecisionTreeClassifier(class_weight = 'balanced',criterion = 'entropy', max_depth = 5, max_features = None, splitter = 'best'))]

pipe_dtc = Pipeline(steps = pipe_steps_dt)
```

## Evaluation results on Test set:

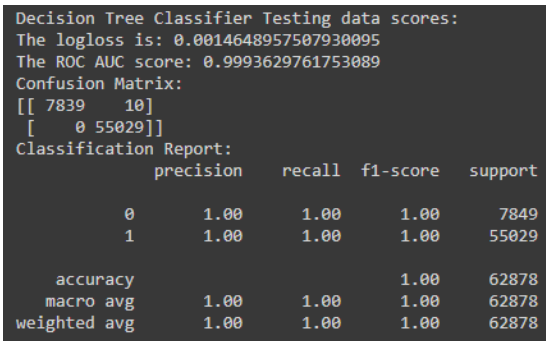
### LogisticRegression Pipeline result



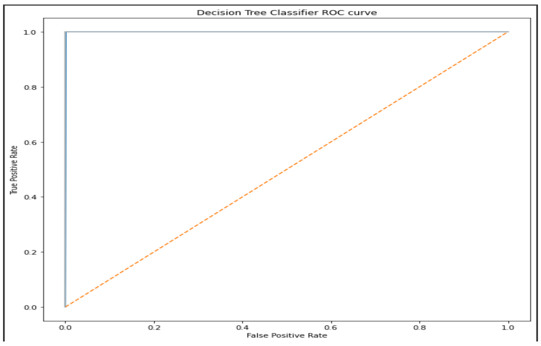
```
Logistic regression Testing data scores:
The logloss is: 0.002007534687707299
The ROC AUC score: 0.9990901935187135
Confusion Matrix:
[[ 7837    12]
 [   16 55013]]
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      7849
           1       1.00      1.00      1.00     55029

    accuracy                           1.00     62878
   macro avg       1.00      1.00      1.00     62878
weighted avg       1.00      1.00      1.00     62878
```



**Final DecisionTreeClassifier Pipeline result**

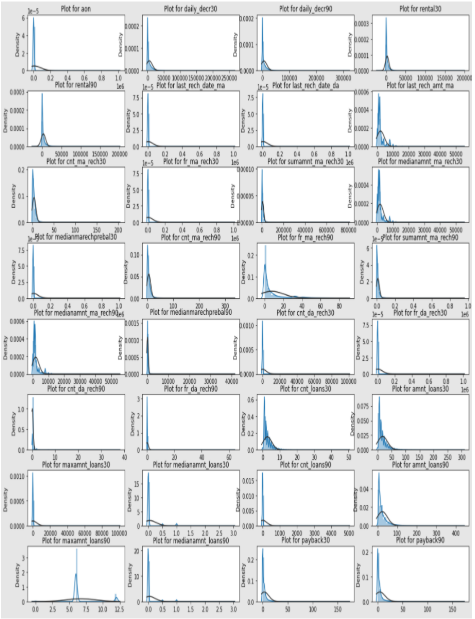


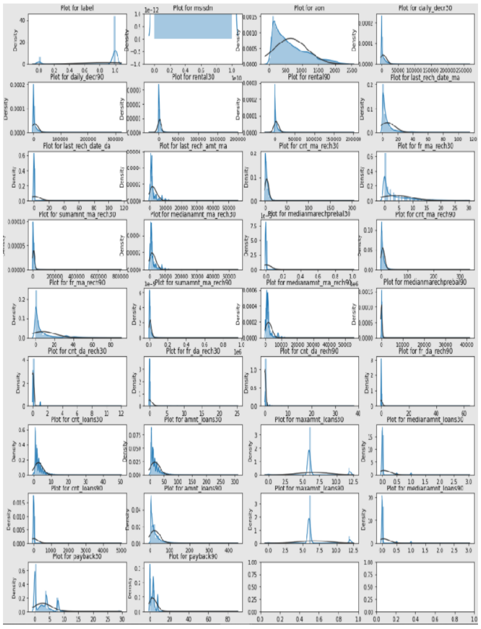
```
Decision Tree Classifier Testing data scores:
The logloss is: 0.0014648957507930095
The ROC AUC score: 0.9993629761753089
Confusion Matrix:
[[ 7839    10]
 [    0 55029]]
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      7849
           1       1.00      1.00      1.00     55029

    accuracy                           1.00     62878
   macro avg       1.00      1.00      1.00     62878
weighted avg       1.00      1.00      1.00     62878
```



- Key Metrics for success in solving problem under consideration
    - ○ The classes in the target variable were imbalance. 12.5% of class 0 and 87.5% of class 1.
    - ○ To manage the imbalance in the problem, Log-loss, weighted Roc Auc score and Confusion Matrix were used as the key Metrics for success.
- **Visualizations**

**Distribution of the Variables before and after cleaning:**

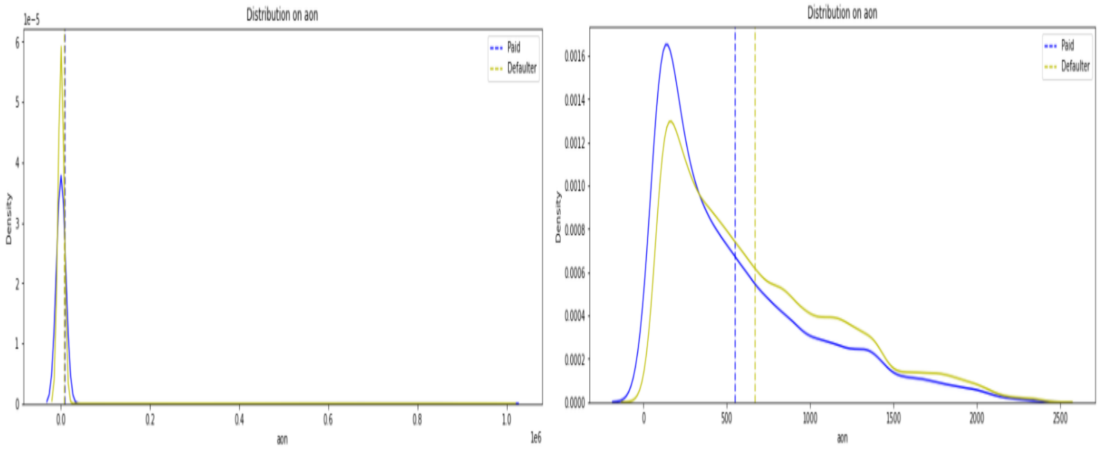




Observations:

1. 1. After cleaning the data we can see some improvements in the data distribution.
2. 2. The distributions still need to be transformed since they are skewed and need to be scaled.

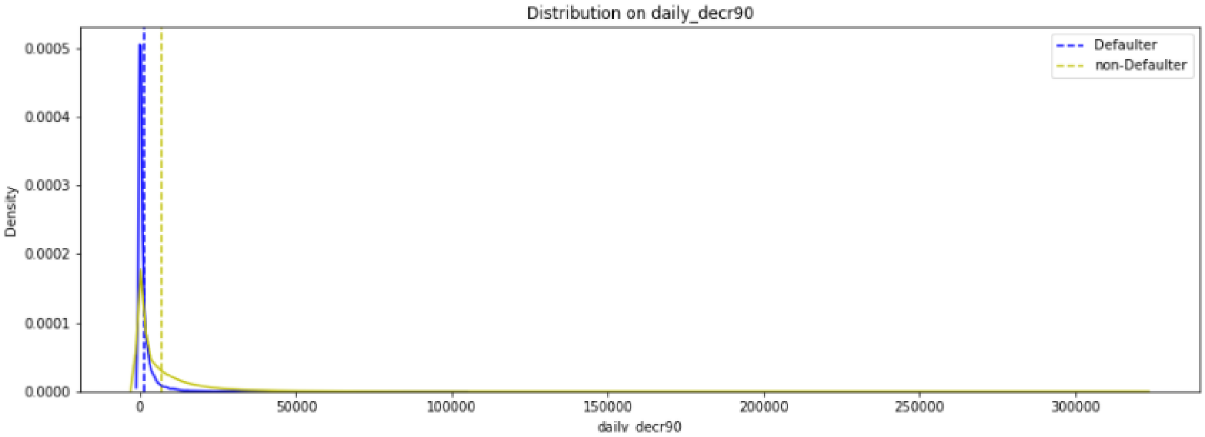
## ‘aon’



Observations:

1. After cleaning the date, the 'aon' variable can distinguish between the labels 1 and 0.
2. The tenure of defaulters on an average is lesser than the tenure of defaulters on the cellular network.

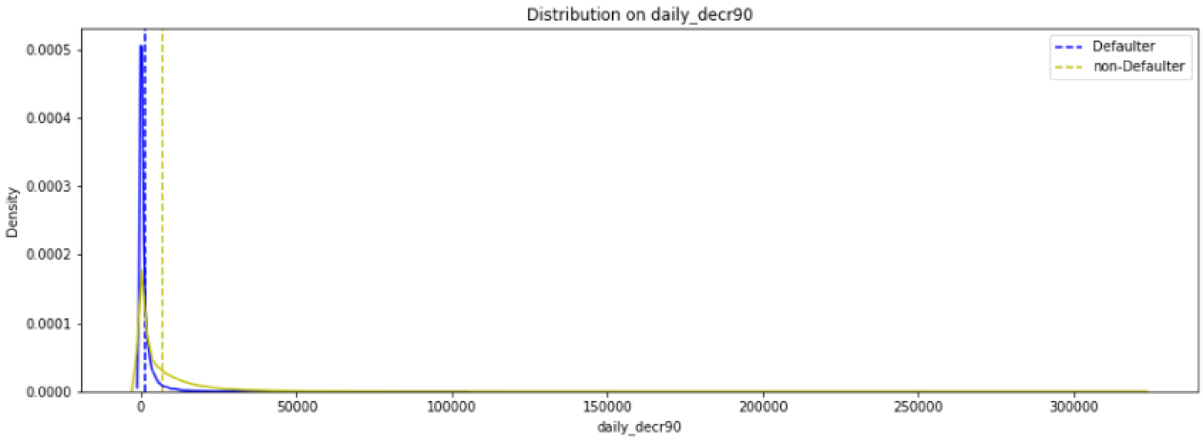
## ‘daily_decr30’



Observations:

1. Most of the values are pretty close.
2. The labels 0 and 1 are somewhat distinguished.
3. The daily spent amount for 30 days for defaulters is lesser than non defaulters on an average.

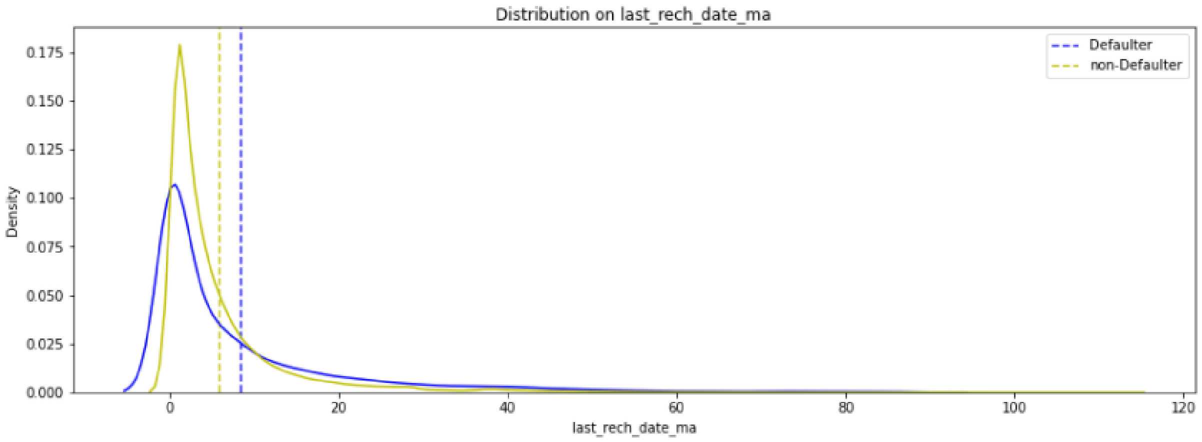
## ‘daily_decr90’



Observations:

1. Most of the values are pretty close. The labels 0 and 1 are somewhat distinguished.
2. The daily spent amount for 90 days for defaulters is slightly lesser than non defaulters on an average.

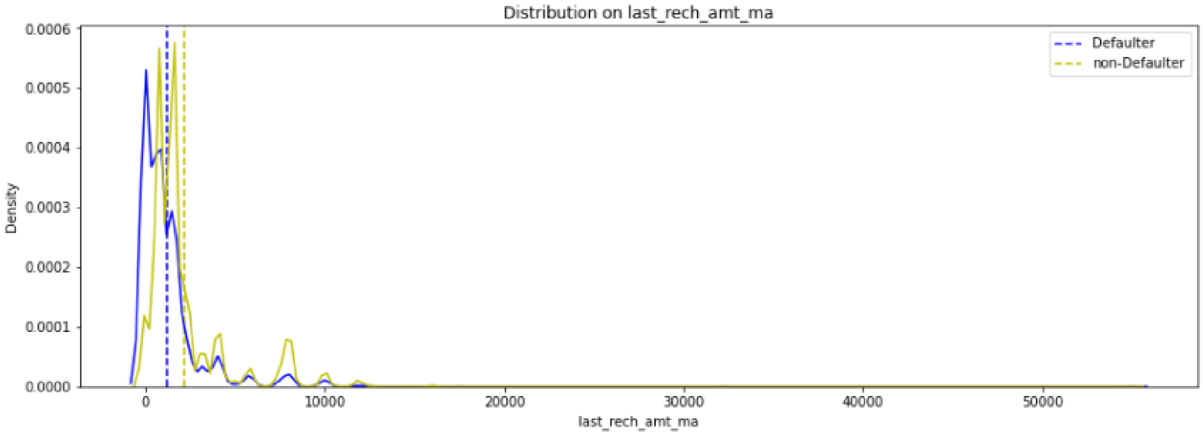
## ‘last_rech_date_ma’



Observations:

1. The classes 0 and 1 are showing some differences.
2. The last recharge date on the main account is slightly lesser for non-defaulters than defaulters.

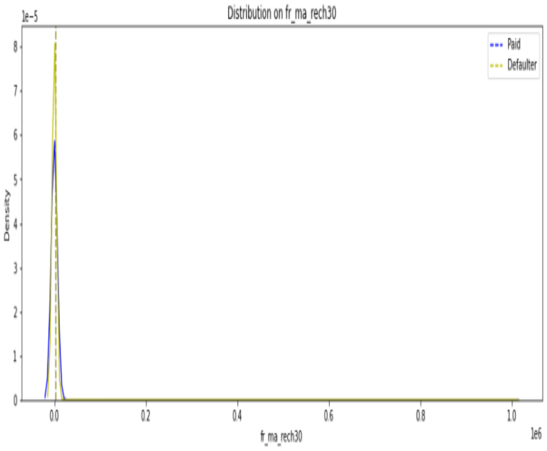
## ‘last_rech_amt_ma’

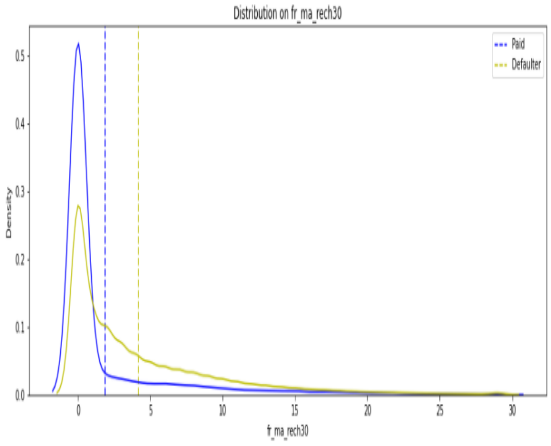


Observations:

1. The classes 0 and 1 in ‘label’ in distributions are showing some difference.
2. The last recharged amount on an average for defaulters is lesser than non-defaulters.

## ‘fr_ma_rech30’

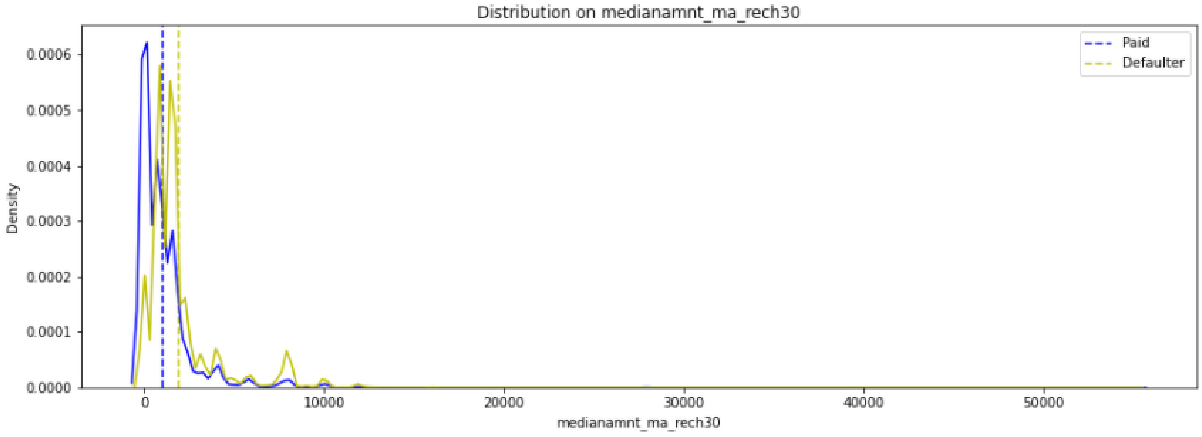




Observations:

1. We can see that the distribution of labels 0 &1 and well distinguished in the fr_ma_rech30.
2. The frequency of recharging the main account done for 30 days by non defaulters is greater than defaulters.

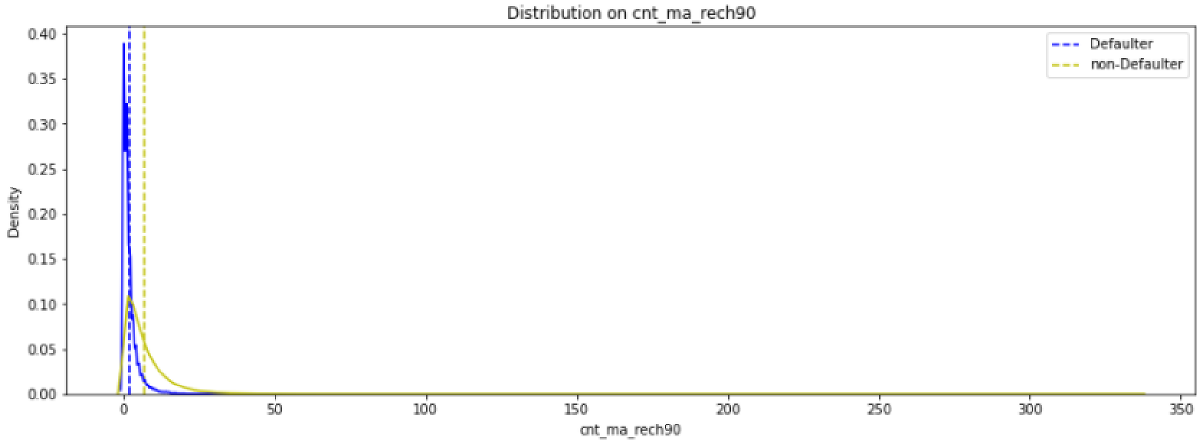
## ‘medianamnt_ma_rech30’



Observations:

1. The graph above shows that the Defaulters have done a slightly lesser amount of recharges to their main account than non-defaulters.

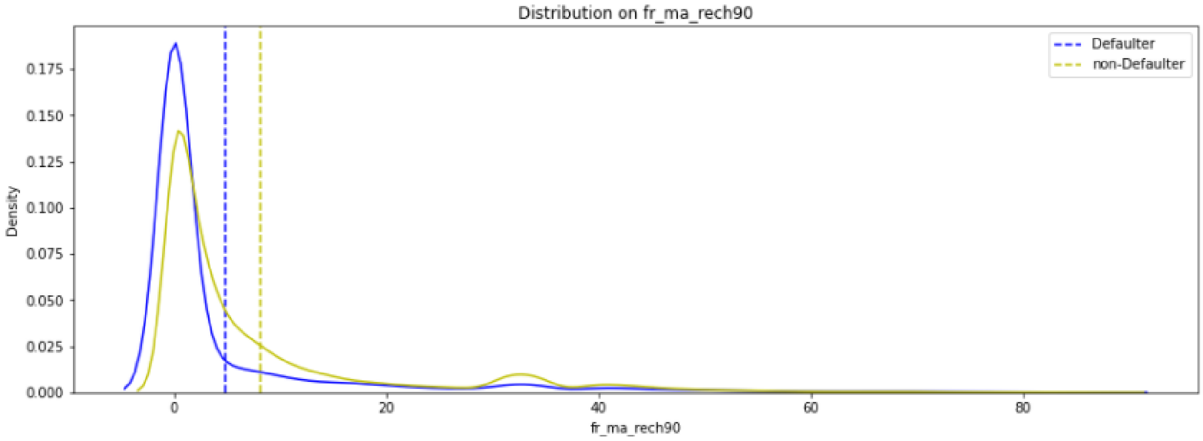
## ‘cnt_ma_rech90’



Observations:

1. On an average, the number of recharges done by the defaulters over 90 days is slightly lesser than the number of recharges done by non-defaulters.

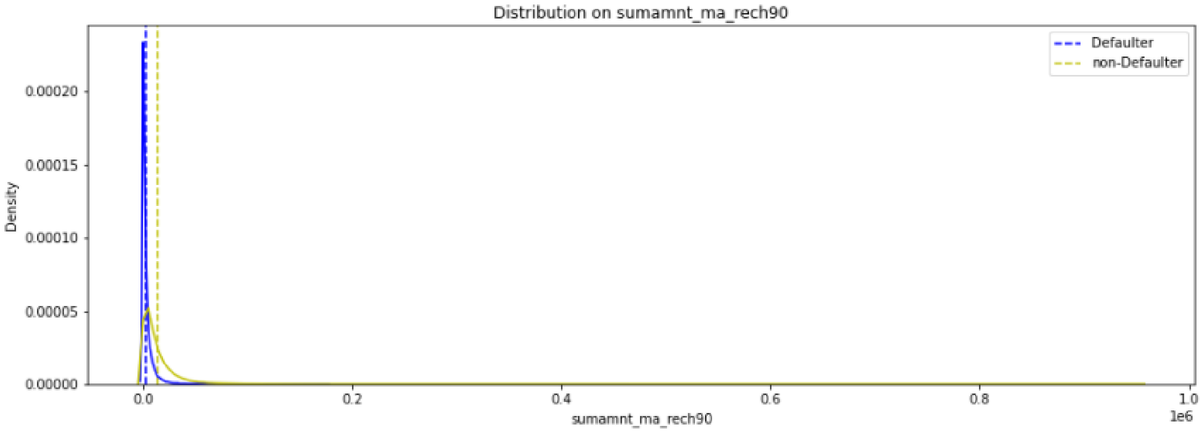
## ‘fr_ma_rech90’



Observations:

1. The classes 0 and 1 show some distinctions.
2. On an average, the frequency of recharges done by defaulters on the main account over a period of 90 days is lesser than that of non-defaulters.

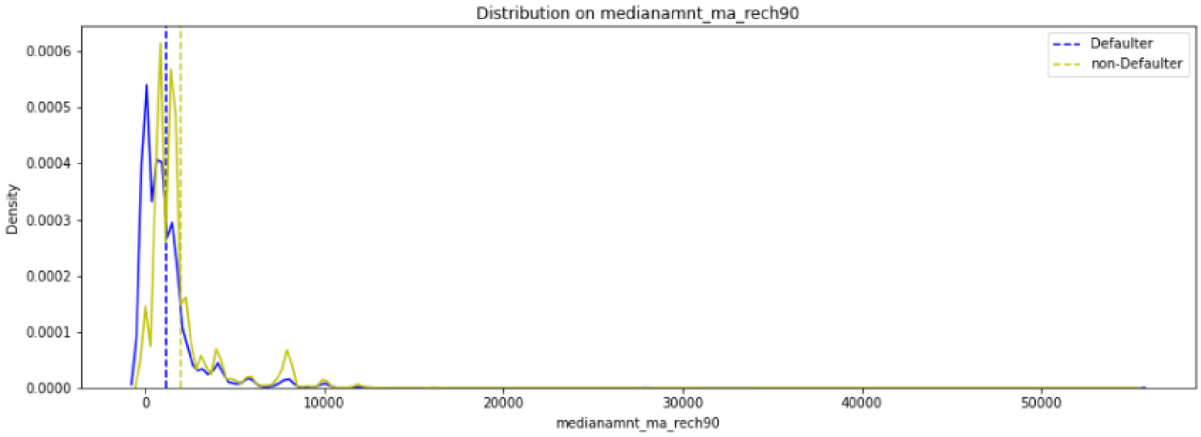
## ‘sumamnt_ma_rech90’



Observations:

1. On an average, the total amount of recharge done by defaulters over 90 days is slightly lesser than done by non-defaulters.

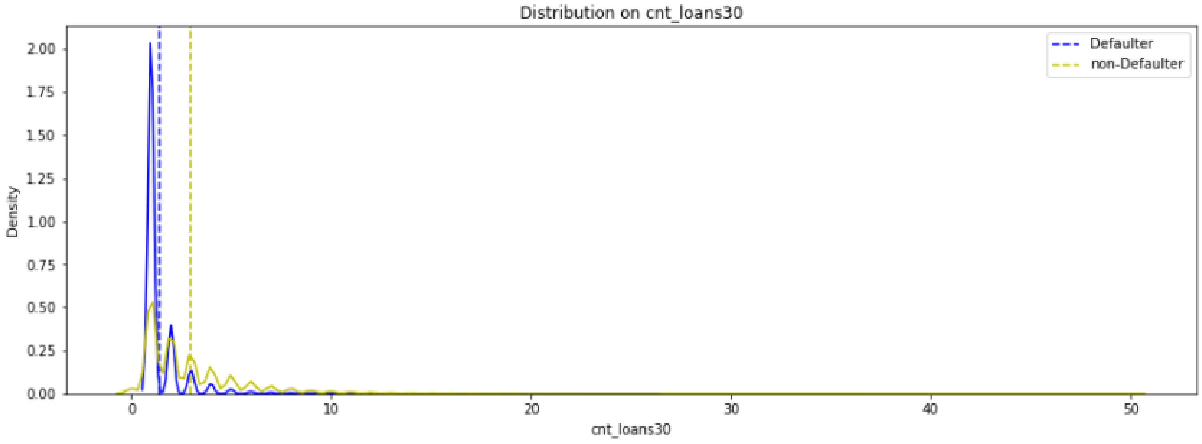
## ‘medianamnt_ma_rech90’



Observations:

1. The label 0 and 1 distribution in medianamnt_ma_rech90 show some difference.
2. On an average, the Median amount of recharge done by defaulters on the main account over 90 days is slightly lesser than that done by non-defaulters.

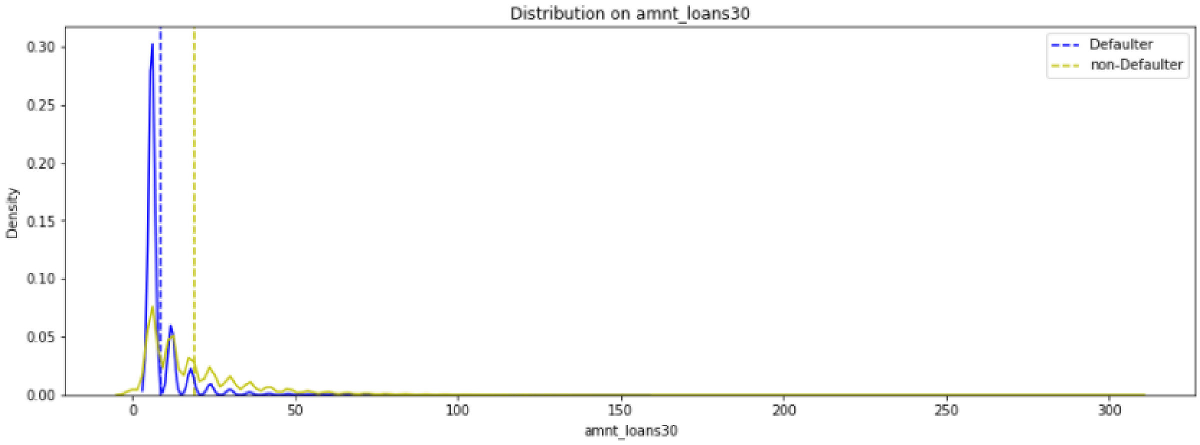
## ‘cnt_loan30’



Observations:

1. The cnt_loans30 distribution shows some difference between labels 0 and 1
2. On an average, the defaulters have taken a lesser number of loans over 30 days than non defaulters.

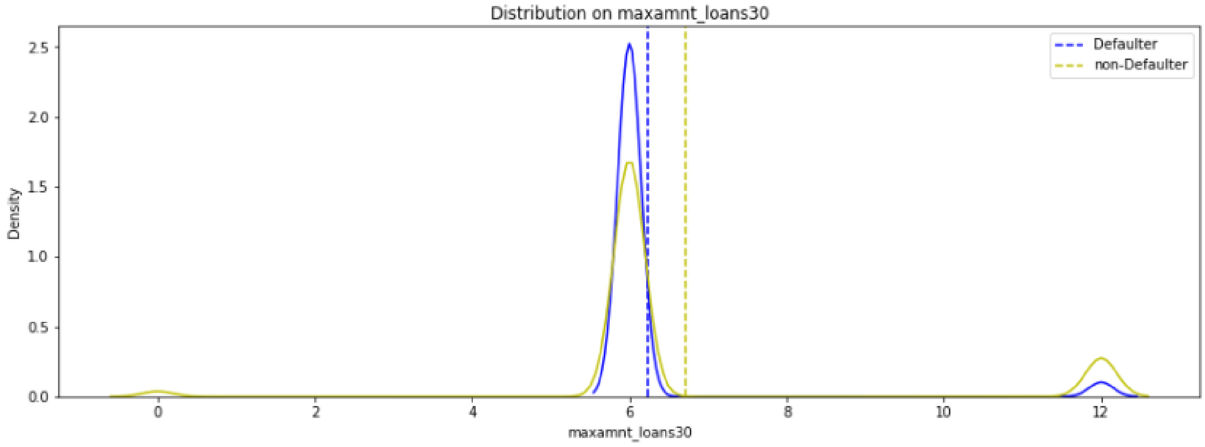
## ‘amnt_loans30’



Observations:

1. The amnt_loans30 distribution shows good difference between labels 0 &1.
2. The total amount of loans taken by defaulters in the last 30 days is lesser than that of non-defaulters on an average.

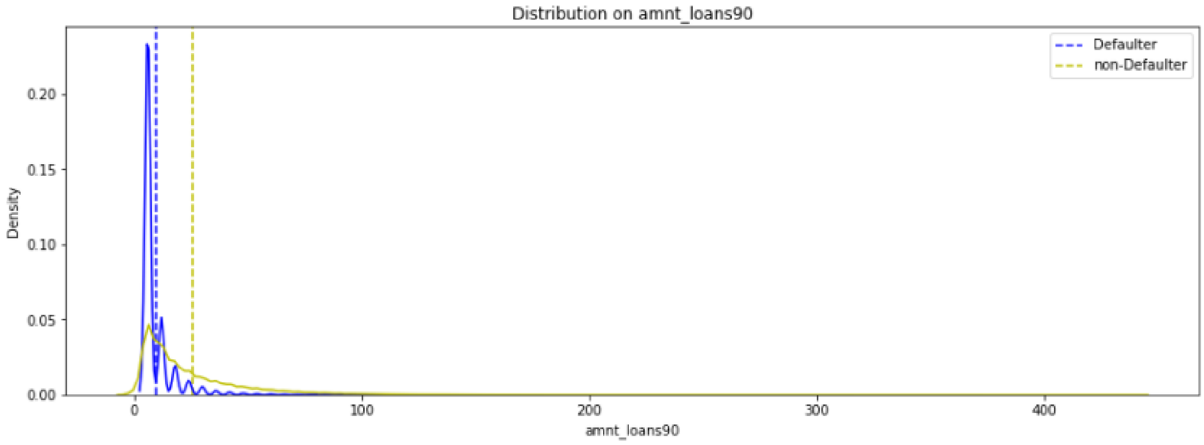
## ‘maxamnt_loans30’



Observations:

1. The distribution means are showing some distinction between labels 0 and 1 for maxamnt_loans30.
2. The maximum amount taken as loan by defaulters is on an average lesser than the non-defaulters.

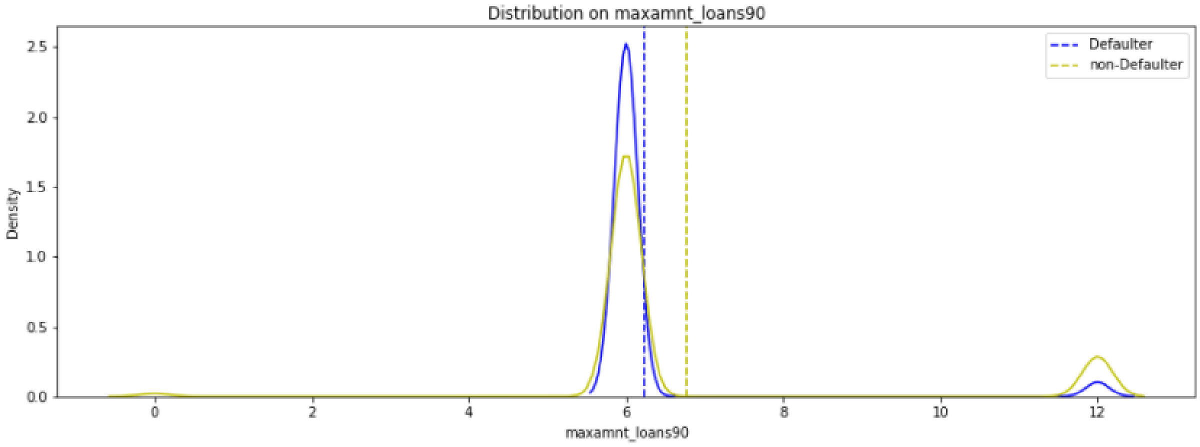
## ‘amnt_loans90’



Observations:

1. The distributions between 0 and 1 labels show good difference in amnt_loans90.
2. On an average, the total amount of loans taken by the defaulters is lesser than non-defaulters.

## ‘maxamnt_loans90’

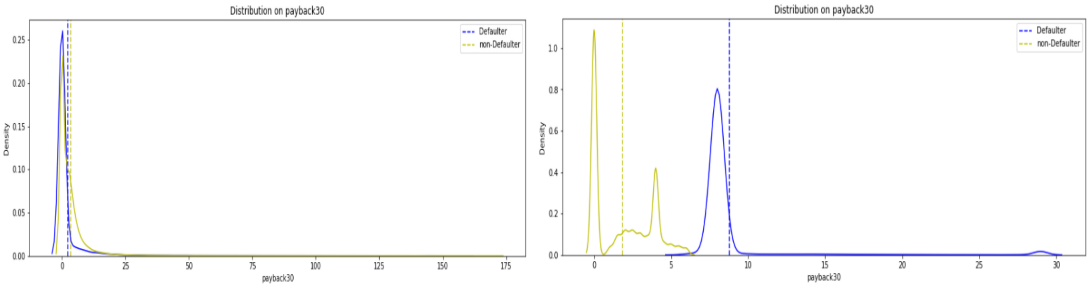


Observations:

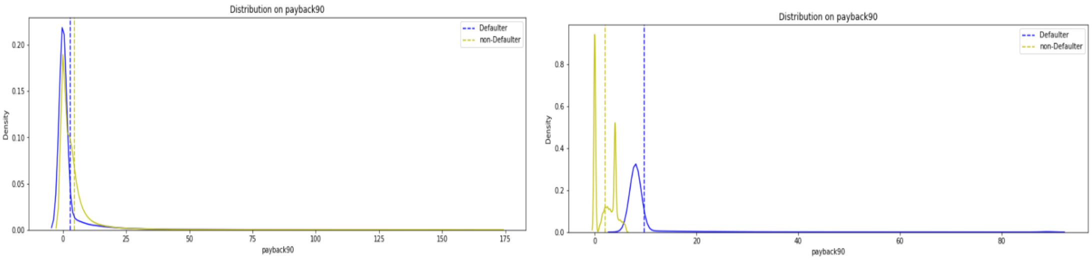
1. On an average, the maximum amount of loans taken by defaulters over a period of 90 days is lesser than the maximum amount of loans taken by non-defaulters.

## Payback

### ‘payback30’ before cleaning left and after cleaning right



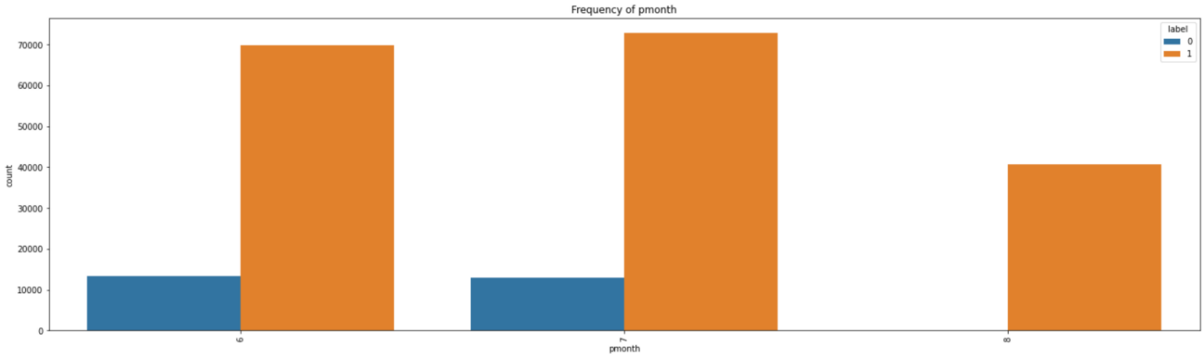
### ‘payback90’ before cleaning(left) and after cleaning(right)



Observations:

1. After cleaning the payback variables show pretty good difference between labels 0 and 1.
2. On an average, the time taken by defaulters to pay back the loan is greater than the time taken by non-defaulter to pay back the loan over a period of 30 or 90.

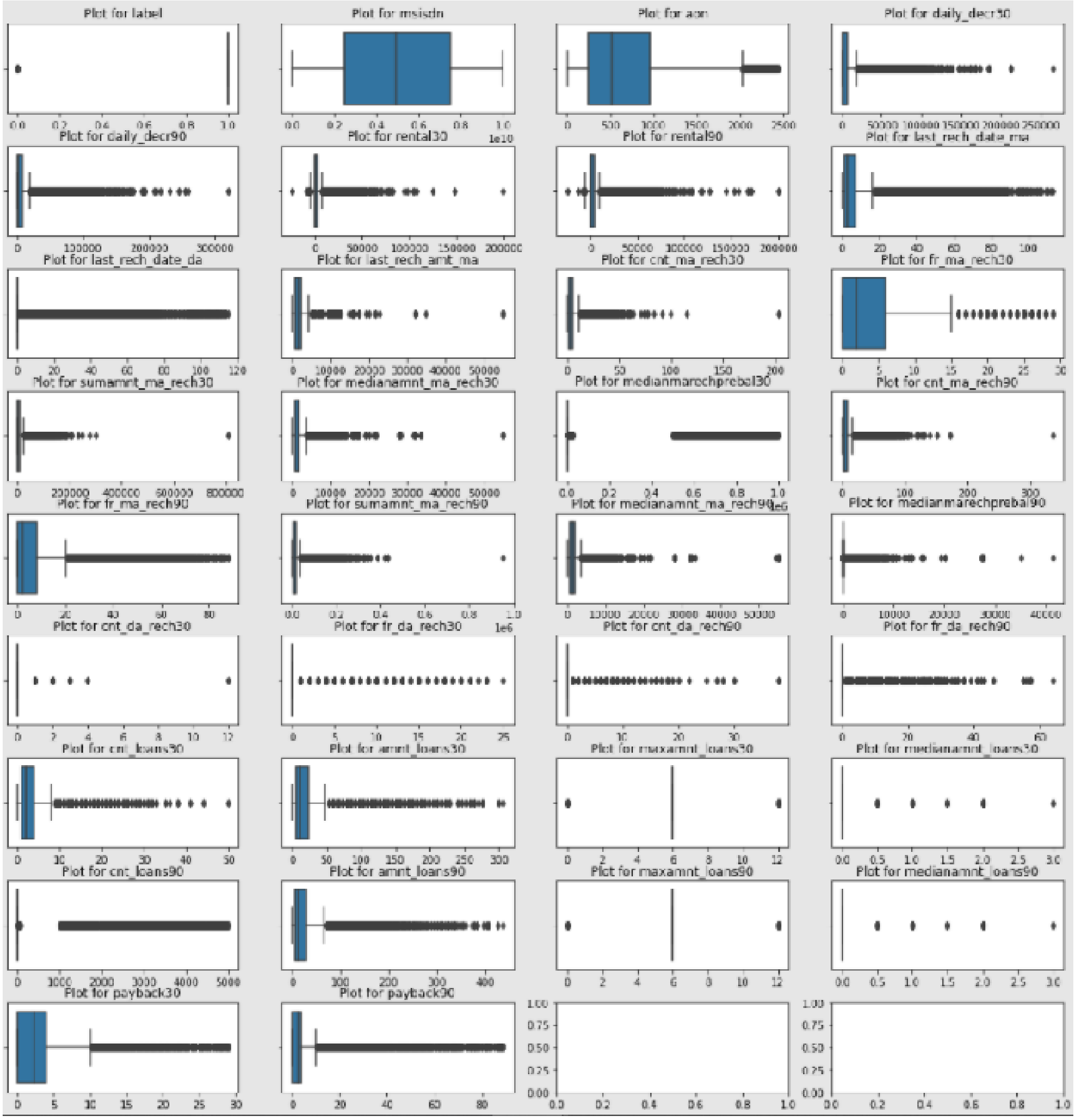
### ‘pdate - Month’



Observations:

1. The month of 6 and 7 have some defaulters. Month 8 doesn't have any defaulters.

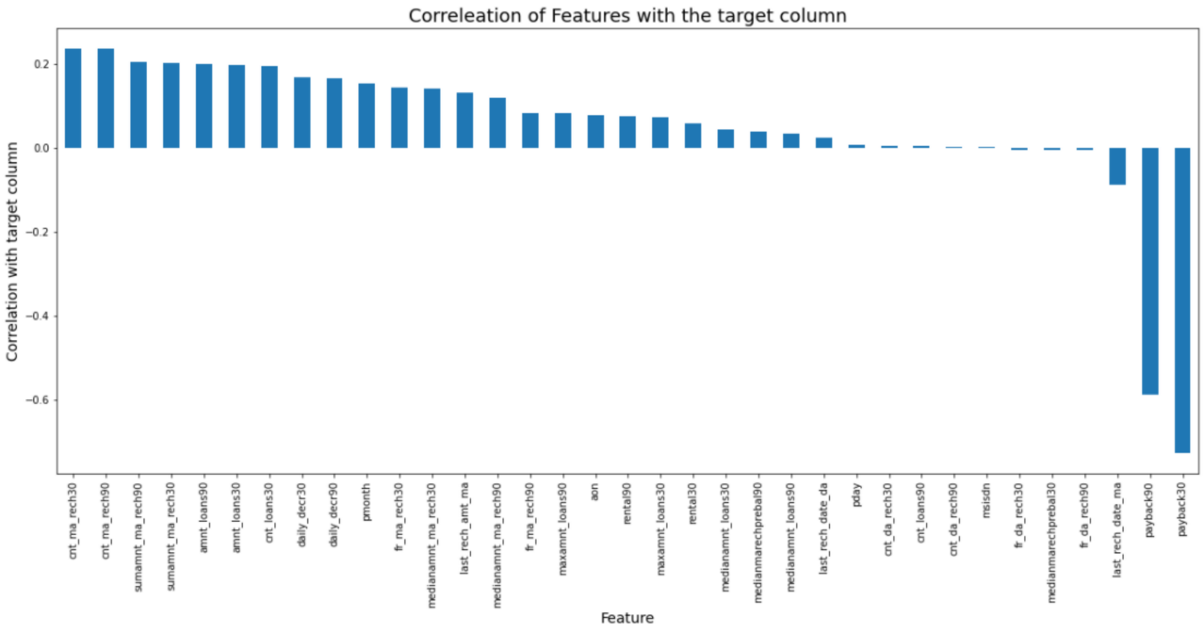
## Outliers:



Observations:

1. There are a lot of outliers in the data. It is not good to remove this much data.
2. Since there is a large percentage of data as outliers, I am going to leave these outliers as they are with a benefit of doubt that these may be natural outliers.
3. A robust method to transform the data needs to be used.

## Correlation:



Observations:

1. The payback30 and payback90 show strong negative correlation with the target variable 'label'.

- ## Interpretation of the Results
  - Most of the visualizations suggest that on an average the defaulters show lesser measurement. For eg, Number of times loans taken, amount of loan taken, number of time recharge done, maximum amount recharge done etc.
  - The Payback time taken is higher for defaulters than non-defaulters.
  - The variables are positively skewed. It requires transformations to make it close to Gaussian distribution.
  - There are outliers in the data. These outliers could be natural outliers. The values are on different scales. Hence, a robust scaling method to scale the data that is not impacted by outliers was required.
  - All of the models used were able to learn the function approximations very well. Even simple models were able to learn from the data. The DecisionTreeClassifier model gave a low bias and low variance result. DecisionTree is robust to outliers. It is best suited for this problem.

## CONCLUSION

- ### Key Findings and Conclusions of the Study

  - Most of the visualizations suggest that on an average the defaulters show lesser measurement. For eg, Number of times loans taken, amount of loan taken, number of time recharge done, maximum amount recharge done etc.
  - The Payback time taken is higher for defaulters than non-defaulters.
  - The Payback variables are the most correlated variables with the target variable negatively.

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

  - The visualization revealed some important behaviours of the customers for example the defaulter were making conservative choices like recharging less number of times, for lesser amounts etc.
  - It is better to use simpler models than complex models if simple models give good performance. This will save a lot in the time complexity and space complexity.
  - Decision tree based models worked really well in datasets which had higher percentage of outliers than linear models that are impacted by the presence of outliers.
  - The most challenging task was the data cleaning. The thinking process to come up with decisions on what needs to be done on the data to clean it based on the available information was the challenge. I had to make informed assumptions based on the information at hand to make several decisions.

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

  - The biggest limitation is the lack of understanding of the business rules of the client. Many decisions could have been taken differently based on what the business is offering to the customers.
  - Also lack of sample data for the label = 0 is also a limitation.As a future scope, more data can be extracted from the client system for the class label =0 and used for training the model. This will also help in testing the model with a more variety of samples. Getting real time data will be more effective than upsampling.