MICRO CREDIT DEFAULTER - PREDICTION

Submitted by

Steffin Varghese

Batch - 26

In partial fulfilment of Data Science – Internship

At

Flip Robo Technologies

AI and Software Development company

Flip Robo Technologies | Indiranagar, Bengaluru - 560 038, Karnataka, India

Month of Submission

August 2022

Acknowledgement

I am highly indebted to FlipRobo Technologies for giving me this opportunity to work on a project and for the guidance and persistent supervision, as well as for providing necessary project information and assistance in completing the project.

I would want to convey my gratitude to the members of FlipRobo Technologies for their kind encouragement and support in completing this project.

I would like to extend my heartfelt gratitude and appreciation to SME, Ms Khushboo Garg for spending such close attention to me and assisting me during the project's completion, as well as towards others who have volunteered to assist me with their skills.

I acknowledge my gratitude towards the authors of papers: “Prediction of Loan Defaulters in Microfinance Using Social Network Data”, “Predicting Credit Default among Micro Borrowers in Ghana and “Strategies for Reducing Microfinance Loan Default in Low-Income Markets” for the insights I could attain through the extensive research which enhanced my knowledge in development of the project.

CONTENTS

INTRODUCTION

Analytical Problem Framing

Model/s Development and Evaluation

CONCLUSION

INTRODUCTION

Business Problem Framing

A Microfinance Institution (MFI) is an organization that offers financial services to low-income populations. Microfinance services (MFS) becomes extremely useful when targeting especially the unbanked poor families living in remote areas with not much sources of income. The 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 extremely useful in such areas, the implementation of MFS has been uneven with both significant challenges and successes.

Conceptual Background of the Domain Problem

Today, microfinance is widely accepted as a poverty-reduction tool, representing 70 billion USD in outstanding loans and a global outreach of 200 million clients.

Many developing nations rely heavily on microfinance institutions to support their economies. However, due to the informal nature of the businesses and people they lend money to, many of these microfinance companies struggle with the issue of default.

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. They are collaborating with an MFI to provide micro-credit on mobile balances to be paid back in 5 days.

Review of Literature

Prediction of Loan Defaulters in Microfinance Using Social Network Data (2018), David Murphy

Three main questions were the focus of this study's objectives. The first step was to introduce a relational learning approach and assess whether it was superior to conventional classification approaches in terms of default prediction. Only one of many relational learner techniques was examined, despite the fact that the results indicate that the predictions from the spreading activation model are typically worse than the classification algorithms and do not provide a major operational benefit to the organisation.

Future work may incorporate different relational learners, such as a network only link based classifier when a denser sample set of loans is available, given that network sparsity may also contribute to the subpar outcomes. Additionally, investigating different transfer functions can result in more encouraging conclusions.

Predicting Credit Default among Micro Borrowers in Ghana (2014),

Kwame Simpe Ofori, Eli Fianu, Kayode Omoregie, Nii Afotey Odai

One of the significant disadvantages of this laudable initiative has been recognised as micro credit default, which depletes these circulating funds and lowers investor confidence. Since effective countermeasures can be established to avoid and reduce the incidences of default, it is crucial to understand the variables that cause a loan recipient to default. Logistic regression was used in this study to determine the variables connected to the occurrence of microcredit default. Age, gender, gross monthly income, length of employment with current employer, loan amount, loan term, number of dependents, other income, and other deductions were found to be significant predictors of default.

Strategies for Reducing Microfinance Loan Default in Low-Income Markets (2017) Patrick Mphaka

The goal of this study was to investigate the methods employed by MFI executives to decrease loan default in the BOP market. According to the study's findings, MFI executives employed a variety of tactics to lower loan default rates, including collaborations, lending to well-established organisations, training in loans and business skills, and routine monitoring. MFI executives also need to place greater emphasis on utilising language that stresses the advantages of loan repayment for prospective borrowers as opposed to language that emphasises the drawbacks of defaulting on loans. The study's conclusions and suggestions provide a list of tactics MFI leaders in low-income economies can employ to decrease loan default and, as a result, increase the profitability and sustainability of their institutions' operations.

Motivation for the Problem Undertaken

In microfinance, default refers to a client's inability to pay back a loan. If loan payback rates are high and reliable, MFIs can continue to deploy loans to help achieve the goal of reducing poverty. If loan payback rates are high, MFIs can lower interest rates and processing costs, which will increase loan demand. An increase in the volume of loans disbursed to various economic sectors is sparked by a high repayment rate. A decrease in credit availability is being felt by the agricultural sector as a result of inadequate loan repayment performance. Poor management practises, loan diversion, loan refusal, and other socioeconomic issues are likely to blame for poor loan payback performance.

Many developing nations depend heavily on microfinance institutions to support their economies. However, due to the informal nature of the businesses and people they lend money to, many of these microfinance companies struggle with the issue of default. So, it would be beneficial for MFIs to identify the possible defaulters based on the historical data.

Data Science and Machine Learning techniques would be apt in this case, as it can identify the complex patterns and analyse the historical data to build models that can predict whether a customer will be a defaulter or not based on various factors and information about the loan and the customer’s loan and repayment history.

With the help of statistics and technology or in short, with the help of data science MFIs can build a predictive model that can identify whether a customer will be a defaulter or not based on his previous credit history, repayment patterns and other crucial information about the loan. Thus, MFIs can reduce the occurrence of defaulters up to an extent. This will be helpful for MFis as well as the economy as a whole as it will help the country to maintain a smooth economic flow and proper control over the economic structure.

Since data science and machine learning allows an option for recalibration, MFIs can always reengineer the machine learning model to perform well with the changing trends in this sector. This will help the MFIs to identify the new changing patterns in credit defaulters better than heuristic approaches.

Analytical Problem Framing

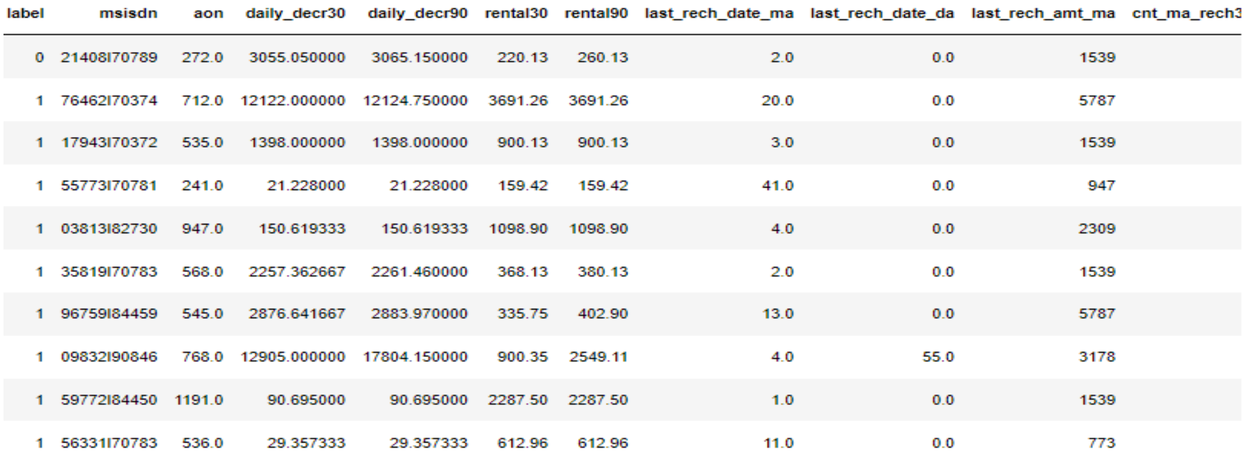
Mathematical/ Analytical Modelling of the Problem

As our client is from Telecom Industry. 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 collaborating with an MFI to provide micro-credit on mobile balances to be paid back in 5 days. The Consumer is believed to be defaulter if he deviates from the path of paying back the loaned amount within the time duration of 5 days. For the loan amount of 5 (in Indonesian Rupiah), payback amount should be 6 (in Indonesian Rupiah), while, for the loan amount of 10 (in Indonesian Rupiah), the payback amount should be 12 (in Indonesian Rupiah).

Data Sources and their formats

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.



label	msisdn	aon	daily_decr30	daily_decr90	rental30	rental90	last_rech_date_ma	last_rech_date_da	last_rech_amt_ma	cnt_ma_rech3
0	21408I70789	272.0	3055.050000	3065.150000	220.13	260.13	2.0	0.0	1539	
1	76462I70374	712.0	12122.000000	12124.750000	3691.26	3691.26	20.0	0.0	5787	
1	17943I70372	535.0	1398.000000	1398.000000	900.13	900.13	3.0	0.0	1539	
1	55773I70781	241.0	21.228000	21.228000	159.42	159.42	41.0	0.0	947	
1	03813I82730	947.0	150.619333	150.619333	1098.90	1098.90	4.0	0.0	2309	
1	35819I70783	568.0	2257.362667	2261.460000	368.13	380.13	2.0	0.0	1539	
1	96759I84459	545.0	2876.641667	2883.970000	335.75	402.90	13.0	0.0	5787	
1	09832I90846	768.0	12905.000000	17804.150000	900.35	2549.11	4.0	55.0	3178	
1	59772I84450	1191.0	90.695000	90.695000	2287.50	2287.50	1.0	0.0	1539	
1	56331I70783	536.0	29.357333	29.357333	612.96	612.96	11.0	0.0	773	

Dataset

- There are no null values in the dataset.
- There may be some customers with no loan history.
- The dataset is imbalanced. Label '1' has approximately 87.5% records, while label '0' has approximately 12.5% records.
- For some features, there may be values which might not be realistic. You may have to observe them and treat them with a suitable explanation.

- You might come across outliers in some features which you need to handle as per your understanding. Keep in mind that data is expensive, and we cannot lose more than 7-8% of the data.
- We have 209593 records in this dataset. We have 209593 rows and 37 columns in the dataset.
- We have string, integer, and float type of data in the dataset.
- We have 209593 non null values in the dataset.

Description of the dataset

Features in Dataset (Independent Variable)

`msisdn` - mobile number of users

`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 number 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

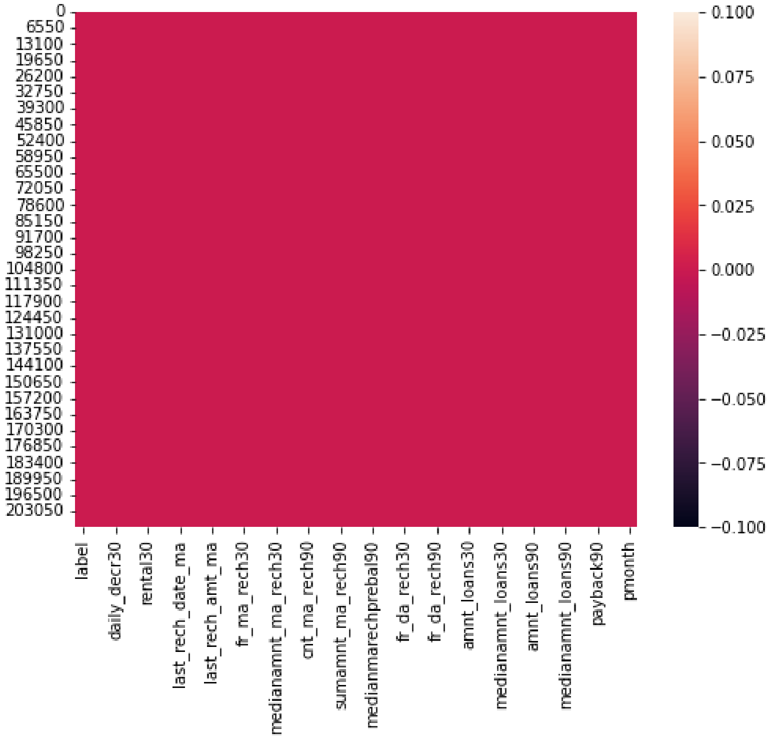
Target in dataset (Dependent Variable)

`label` - Flag indicating whether the user paid back the credit amount within 5 days of issuing the loan. Label ‘1’ indicates that the loan has been paid i.e., non-defaulter, while Label ‘0’ indicates that the loan has not been paid i.e., defaulter.

Data Pre-processing

- The column 'pcirlce' has only one value for all the records. So, we dropped this column from the dataset.
- The first column is just indexed to identify each records. So, we dropped this column as it is not making any impact on the output.
- We have also dropped the column ‘msisdn’ as it contains the user mobile number.
- As a feature engineering, we have extracted the day and month from the column ‘pdate’ and added them to the dataset as new columns.

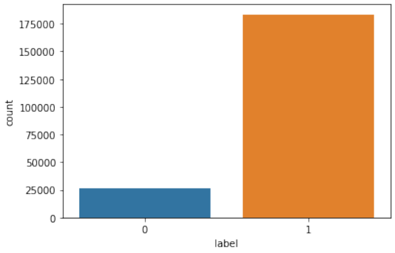
Checking for Missing Values



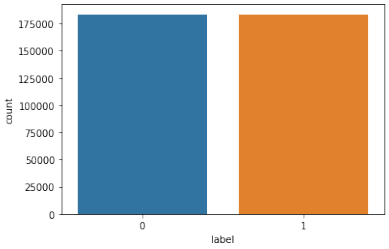
Heatmap of null values

Data Cleansing

Checking whether the data is balanced or not



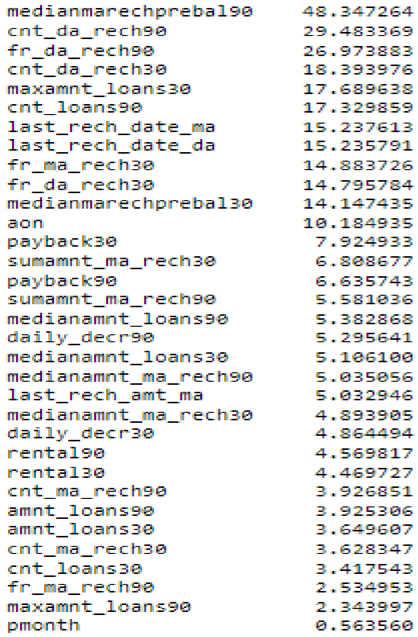
Unbalanced data



Balanced Data

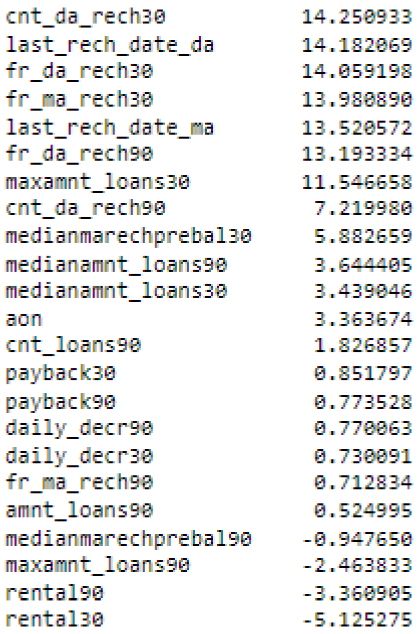
The data is not balanced and the records for non-defaulters is provided more in the datatset. So, we used SMOTE oversampling technique to balance the dataset.

Removing the Skewness



```
medianmarechprebal90     48.347264
cnt_da_rech90            29.483369
fr_da_rech90             26.973883
cnt_da_rech30            18.393976
maxamnt_loans30          17.689638
cnt_loans90              17.329859
last_rech_date_ma        15.237613
last_rech_date_da        15.235791
fr_ma_rech30             14.883726
fr_da_rech30             14.795784
medianmarechprebal30     14.147435
aon                      10.184935
payback30                 7.924933
sumamnt_ma_rech30         6.808677
payback90                 6.635743
sumamnt_ma_rech90         5.581036
medianamnt_loans90        5.382868
daily_decr90              5.295641
medianamnt_loans30        5.106100
medianamnt_ma_rech90      5.035056
last_rech_amt_ma          5.032946
medianamnt_ma_rech30      4.893905
daily_decr30              4.864494
rental90                  4.569817
rental30                  4.469727
cnt_ma_rech90             3.926851
amnt_loans90              3.925306
amnt_loans30              3.649607
cnt_ma_rech30             3.628347
cnt_loans30               3.417543
fr_ma_rech90              2.534953
maxamnt_loans90           2.343997
pmonth                    0.563560
```

Before reducing skewness of data



```
cnt_da_rech30           14.250933
last_rech_date_da       14.182069
fr_da_rech30            14.059198
fr_ma_rech30            13.980890
last_rech_date_ma       13.520572
fr_da_rech90            13.193334
maxamnt_loans30         11.546658
cnt_da_rech90            7.219980
medianmarechprebal30     5.882659
medianamnt_loans90       3.644405
medianamnt_loans30       3.439046
aon                      3.363674
cnt_loans90              1.826857
payback30                0.851797
payback90                0.773528
daily_decr90             0.770063
daily_decr30             0.730091
fr_ma_rech90             0.712834
amnt_loans90             0.524995
medianmarechprebal90    -0.947650
maxamnt_loans90         -2.463833
rental90                -3.360905
rental30                -5.125275
```

After adjusting skewness using power transform

The columns ['cnt_da_rech30', 'fr_ma_rech30','fr_da_rech30', 'last_rech_date_da', 'last_rech_date_ma', 'fr_da_rech90', 'maxamnt_loans30'] are having high skewness. So, we have dropped these columns from the dataset as these columns can have biased impact on the output.

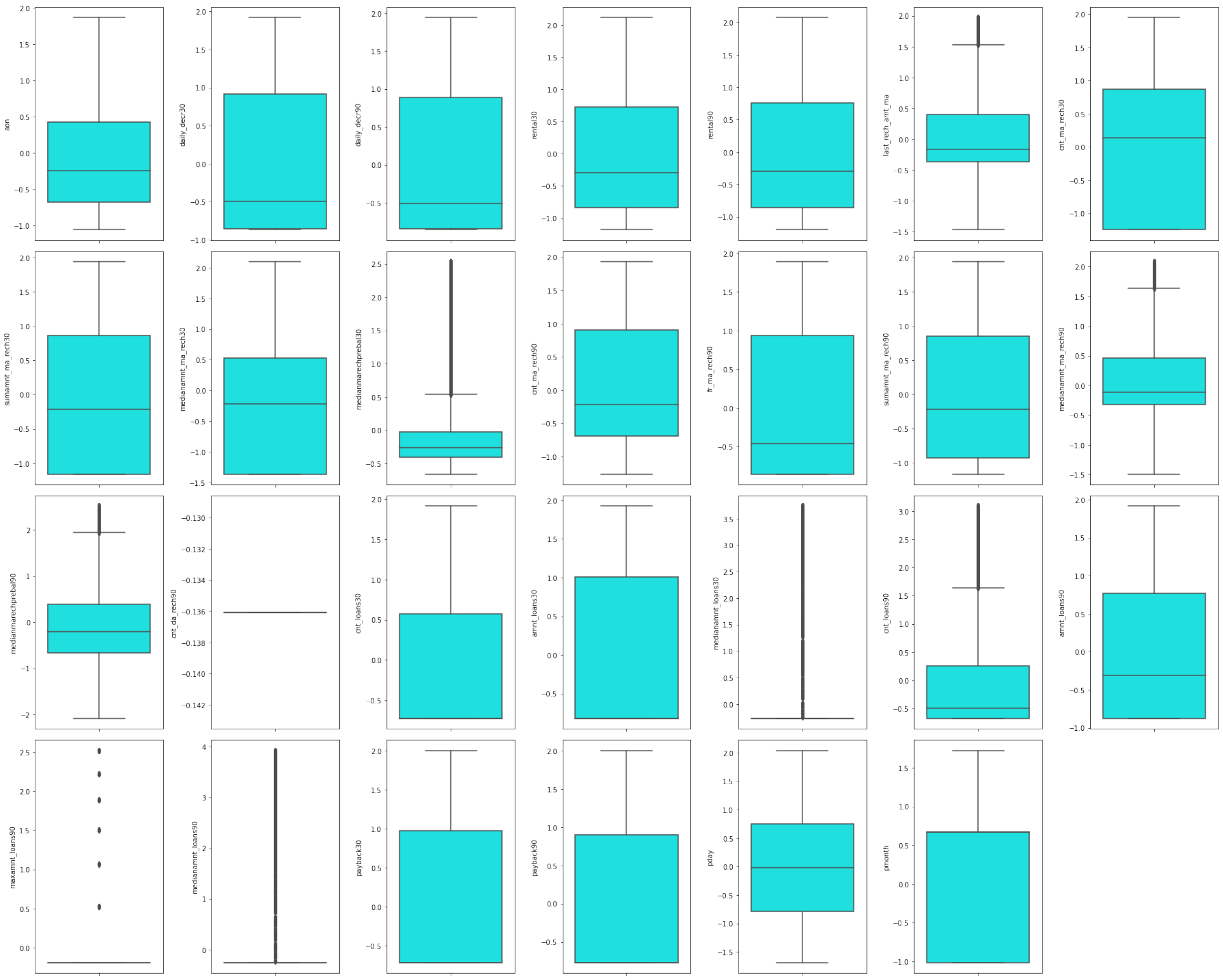
Removing the Outliers

We tried three outlier removal methods on the datasets:

a. Using Zscore method
b. Using Inter-quartile range (IQR)
c. Using percentile method

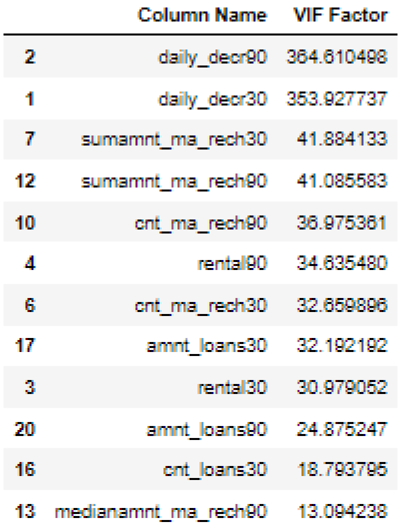
Out of which the percentile method was effective as it was not losing any data from the dataset after removing outliers. Since the data is important and we do not want to lose any crucial information from the dataset, we have proceeded with the percentile outlier removal method.

After removing outliers using percentile method, we were able to reduce the outliers up to an extent.



Outliers in dataset using boxplot after using percentile method

Checked and Removed Multicollinearity from the Datasets.

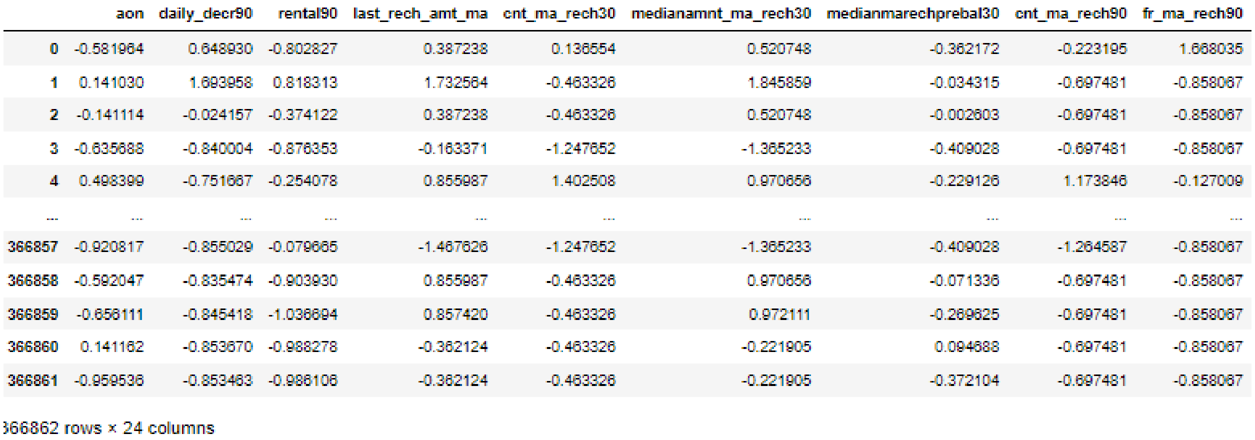


	Column Name	VIF Factor
2	daily_decr90	364.610498
1	daily_decr30	353.927737
7	sumamnt_ma_rech30	41.884133
12	sumamnt_ma_rech90	41.085583
10	cnt_ma_rech90	36.975361
4	rental90	34.635480
6	cnt_ma_rech30	32.659896
17	amnt_loans30	32.192192
3	rental30	30.979052
20	amnt_loans90	24.875247
16	cnt_loans30	18.793795
13	medianamnt_ma_rech90	13.094238

Columns with high VIF

These columns were having high multicollinearity, so we dropped the columns ['daily_decr30', 'sumamnt_ma_rech30','rental30', 'amnt_loans30'] so as to reduce the multicollinearity.

Final Dataset



	aon	daily_decr90	rental90	last_rech_amt_ma	cnt_ma_rech30	medianamnt_ma_rech30	medianmarechprebal30	cnt_ma_rech90	fr_ma_rech90
0	-0.581964	0.648930	-0.802827	0.387238	0.136554	0.520748	-0.362172	-0.223195	1.668035
1	0.141030	1.693958	0.818313	1.732564	-0.463326	1.845859	-0.034315	-0.697481	-0.858067
2	-0.141114	-0.024157	-0.374122	0.387238	-0.463326	0.520748	-0.002603	-0.697481	-0.858067
3	-0.635688	-0.840004	-0.876353	-0.163371	-1.247652	-1.365233	-0.409028	-0.697481	-0.858067
4	0.498399	-0.751667	-0.254078	0.855987	1.402508	0.970656	-0.229126	1.173846	-0.127009
...	...	...	...	...	...	...	...	...	...
366857	-0.920817	-0.855029	-0.079665	-1.467626	-1.247652	-1.365233	-0.409028	-1.264587	-0.858067
366858	-0.592047	-0.835474	-0.903930	0.855987	-0.463326	0.970656	-0.071336	-0.697481	-0.858067
366859	-0.656111	-0.845418	-1.036694	0.857420	-0.463326	0.972111	-0.269625	-0.697481	-0.858067
366860	0.141162	-0.853670	-0.988278	-0.362124	-0.463326	-0.221905	0.094688	-0.697481	-0.858067
366861	-0.959536	-0.853463	-0.986106	-0.362124	-0.463326	-0.221905	-0.372104	-0.697481	-0.858067

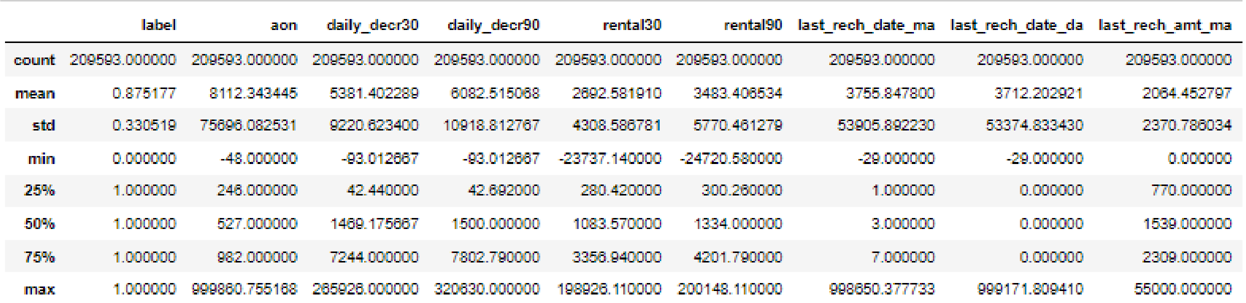
366862 rows × 24 columns

Final Dataset after EDA and pre-processing

Data Inputs- Logic- Output Relationships

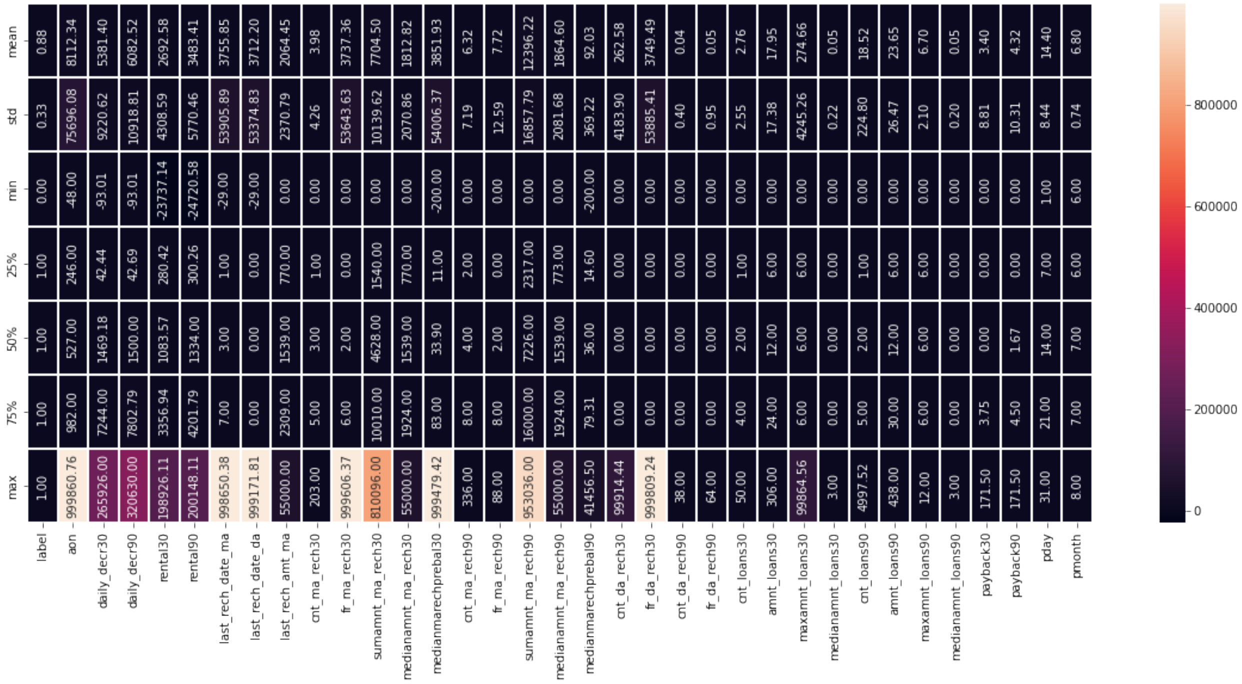
Statistical Summary

Describe of the data



	label	aon	daily_decr30	daily_decr90	rental30	rental90	last_rech_date_ma	last_rech_date_da	last_rech_amt_ma
count	209593.000000	209593.000000	209593.000000	209593.000000	209593.000000	209593.000000	209593.000000	209593.000000	209593.000000
mean	0.875177	8112.343445	5381.402289	6082.515068	2692.581910	3483.406534	3755.847800	3712.202921	2064.452797
std	0.330519	75696.082531	9220.623400	10918.812767	4308.586781	5770.461279	53905.892230	53374.833430	2370.786034
min	0.000000	-48.000000	-93.012667	-93.012667	-23737.140000	-24720.580000	-29.000000	-29.000000	0.000000
25%	1.000000	246.000000	42.440000	42.692000	280.420000	300.260000	1.000000	0.000000	770.000000
50%	1.000000	527.000000	1469.175667	1500.000000	1083.570000	1334.000000	3.000000	0.000000	1539.000000
75%	1.000000	982.000000	7244.000000	7802.790000	3356.940000	4201.790000	7.000000	0.000000	2309.000000
max	1.000000	999860.755168	265926.000000	320630.000000	198926.110000	200148.110000	998650.377733	999171.809410	55000.000000

Describe of the data



	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
mean	0.88	8112.34	5381.40	6082.52	2692.58	3483.41	3755.85	3712.20	2064.45	3.98	3737.36	7704.50	1812.82	3851.93	6.32	7.72	12396.22	1864.60
std	0.33	75696.08	9220.62	10918.81	4308.59	5770.46	53905.89	53374.83	2370.79	4.26	53643.63	10139.62	2070.86	54006.37	7.19	12.59	16857.79	2081.68
min	0.00	-48.00	-93.01	-93.01	-23737.14	-24720.58	-29.00	-29.00	0.00	0.00	0.00	0.00	0.00	-200.00	0.00	0.00	0.00	0.00
25%	1.00	246.00	42.44	42.69	280.42	300.26	1.00	0.00	770.00	1.00	0.00	1540.00	770.00	11.00	2.00	0.00	2317.00	773.00
50%	1.00	527.00	1469.18	1500.00	1083.57	1334.00	3.00	0.00	1539.00	3.00	2.00	4628.00	1539.00	33.90	4.00	2.00	7226.00	1539.00
75%	1.00	982.00	7244.00	7802.79	3356.94	4201.79	7.00	0.00	2309.00	5.00	6.00	10010.00	1924.00	83.00	8.00	8.00	16000.00	1924.00
max	1.00	999860.76	265926.00	320630.00	198926.11	200148.11	998650.38	999171.81	55000.00	203.00	999606.37	810096.00	55000.00	999479.42	336.00	88.00	953036.00	55000.00

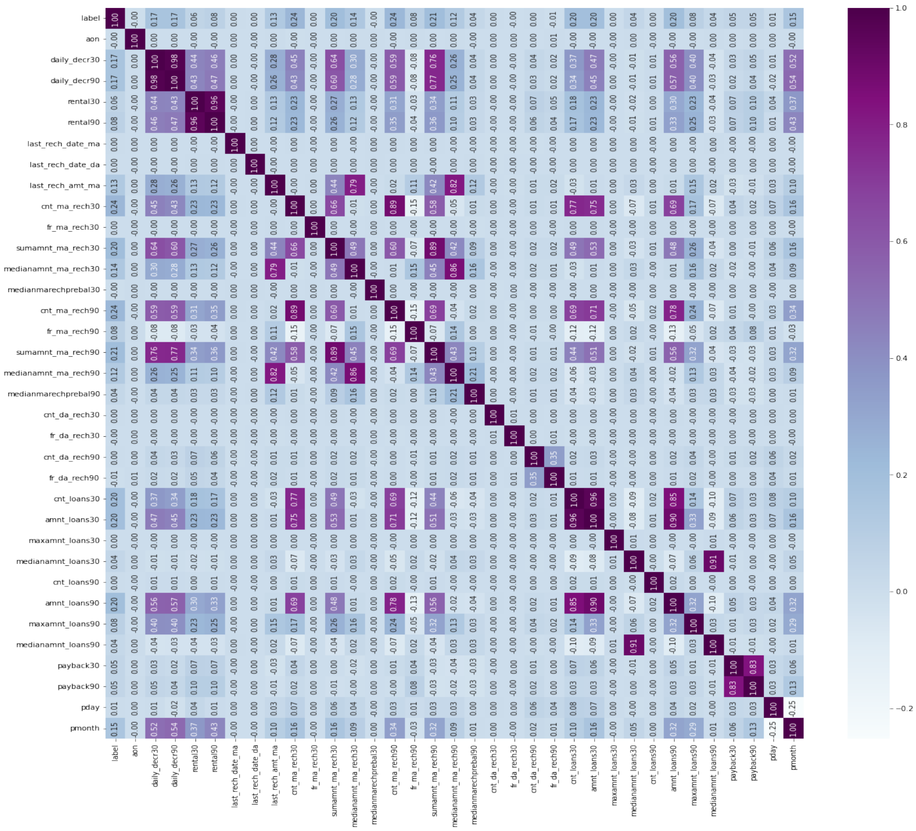
	medianmarechprebal90	cnt_da_rech30	fr_da_rech30	cnt_da_rech90	fr_da_rech90	cnt_loans30	amnt_loans30	maxamnt_loans30	medianamnt_loans30	cnt_loans90	amnt_loans90	maxamnt_loans90	medianamnt_loans90	payback30	payback90	pday	pmonth
mean	92.03	262.58	3749.49	0.04	0.05	2.76	17.95	274.66	0.05	18.52	23.65	6.70	0.05	3.40	4.32	14.40	6.80
std	369.22	4183.90	53885.41	0.40	0.95	2.55	17.38	4245.26	0.22	224.80	26.47	2.10	0.20	8.81	10.31	8.44	0.74
min	-200.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	6.00
25%	14.60	0.00	0.00	0.00	0.00	1.00	6.00	6.00	0.00	1.00	6.00	6.00	0.00	0.00	0.00	7.00	6.00
50%	36.00	0.00	0.00	0.00	0.00	2.00	12.00	6.00	0.00	2.00	12.00	6.00	0.00	0.00	1.67	14.00	7.00
75%	79.31	0.00	0.00	0.00	0.00	4.00	24.00	6.00	0.00	5.00	30.00	6.00	0.00	3.75	4.50	21.00	7.00
max	41456.50	99914.44	999809.24	38.00	64.00	50.00	306.00	99864.56	3.00	4997.52	438.00	12.00	3.00	171.50	171.50	31.00	8.00

Heatmap of describe of data

Observations:

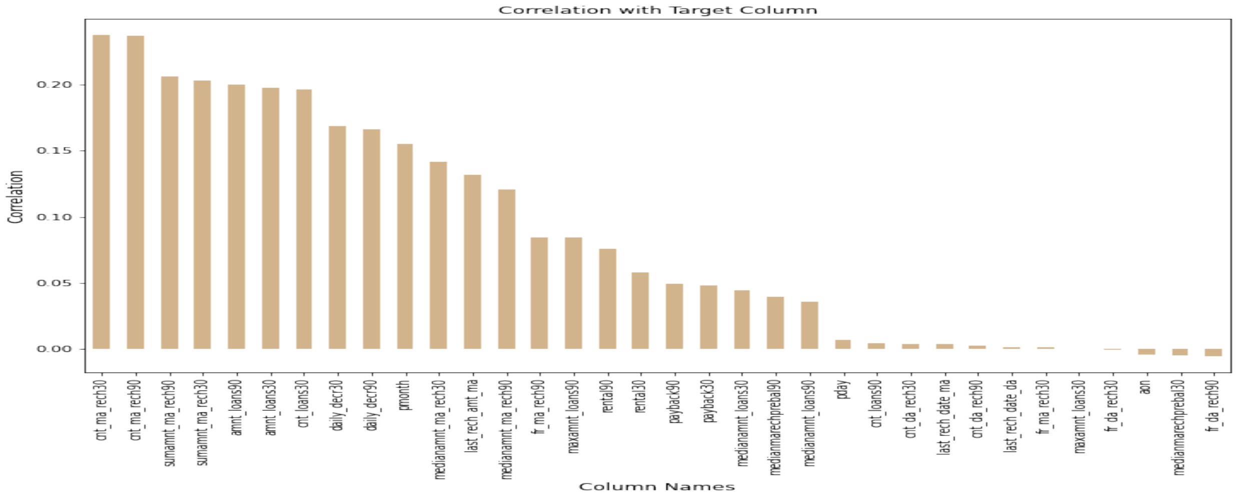
- All the columns except the column 'label' are having higher mean value than the median value. That means the data in these columns are skewed and data is not normally distributed.
- Except the columns ['label', 'medianamnt_loans30', 'maxamnt_loans90', 'medianamnt_loans90', 'pday', 'pmonth'], rest of the columns are having huge difference between the 75% and the maximum value. That means possible outliers are present in the data.

Correlation



Correlation of all variables

Correlation with the Target column



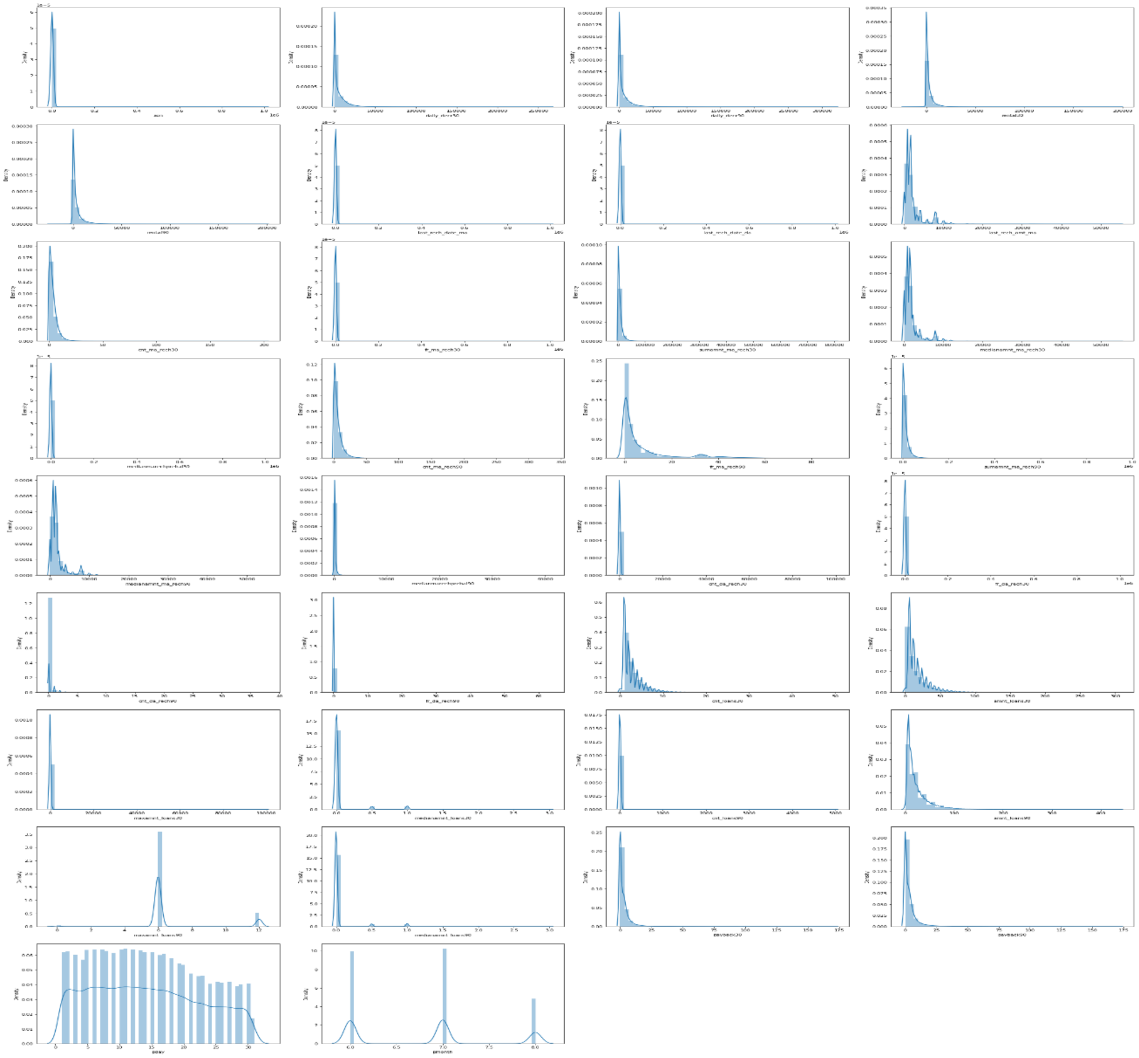
Correlation of variables with target variable

Observations:

- Most of the columns are having positive correlation to the target variable.
- The columns ['fr_da_rech30', 'aon', 'medianmarechprebal30', 'fr_da_rech90'] are having negative correlation to the target variable 'label'. Rest of the variables are having positive correlation to the target variable.
- The column 'cnt_ma_rech30' is having highest positive correlation to the target variable 'label', while the column 'maxamnt_loans30' is having least positive correlation to the target variable 'label'.
- The column 'fr_da_rech30' is having the least negative correlation to the target variable 'label', while the column 'fr_da_rech90' is having the highest negative correlation to the target variable 'label'.

Assumptions Related to the Problem Under Consideration

Distribution of data in columns (Checking skewness of data)

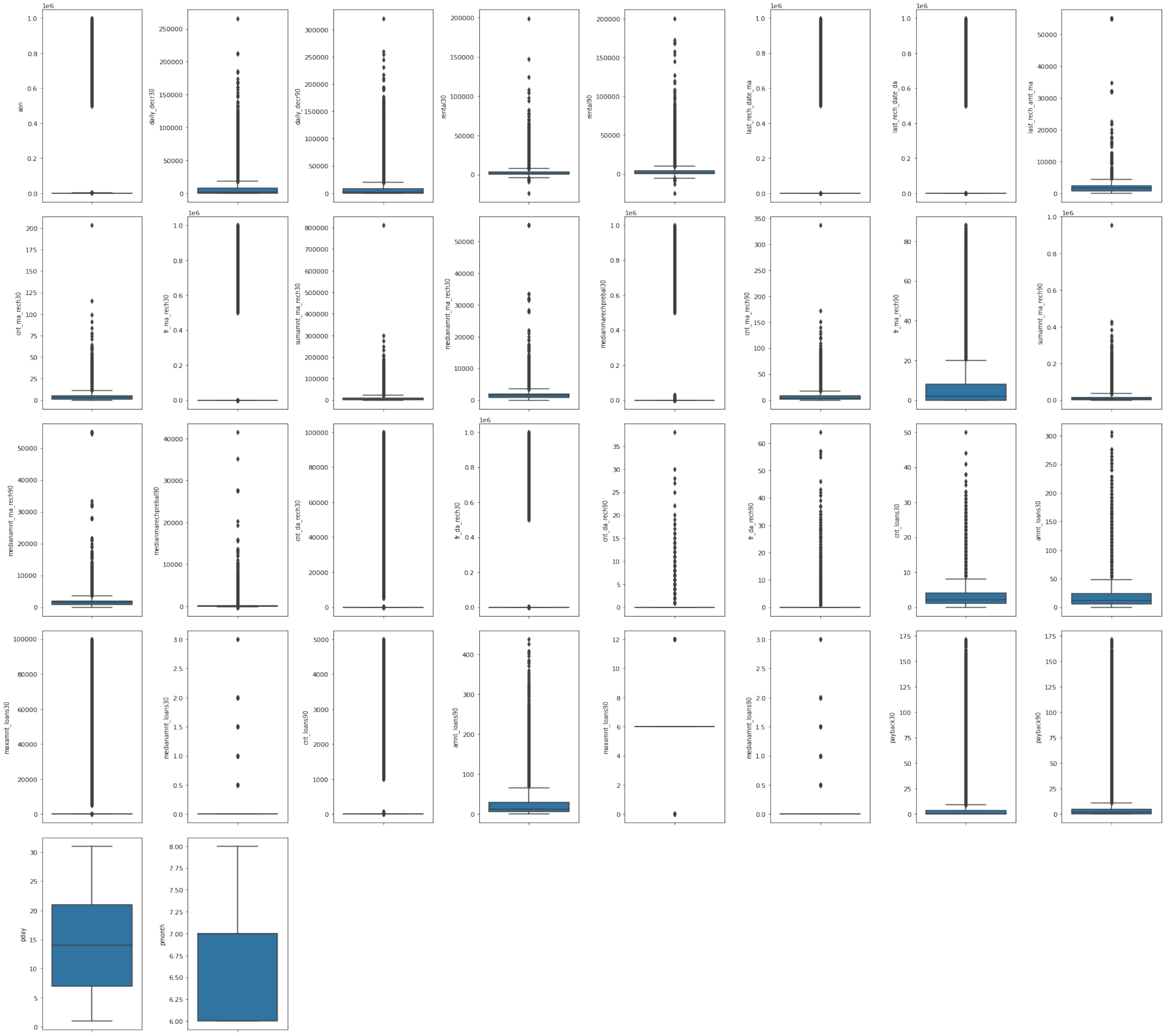


Skewness of Data

Observations:

- None of the columns are having normal distribution of data.
- All the columns except the columns ['pday', 'pmonth'] are having skewness and they are right skewed.

Presence of Outliers in Data



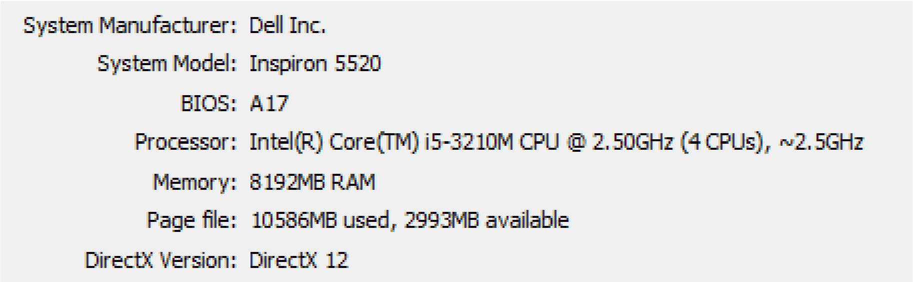
Outliers in dataset using boxplot

Observations:

- All the columns except the columns ['pday', 'pmonth'] are having extreme outliers. We can handle these outliers in later step.

Hardware and Software Requirements and Tools Used

Hardware Requirement:



System Manufacturer:	Dell Inc.
System Model:	Inspiron 5520
BIOS:	A17
Processor:	Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz (4 CPUs), ~2.5GHz
Memory:	8192MB RAM
Page file:	10586MB used, 2993MB available
DirectX Version:	DirectX 12

Hardware Configuration

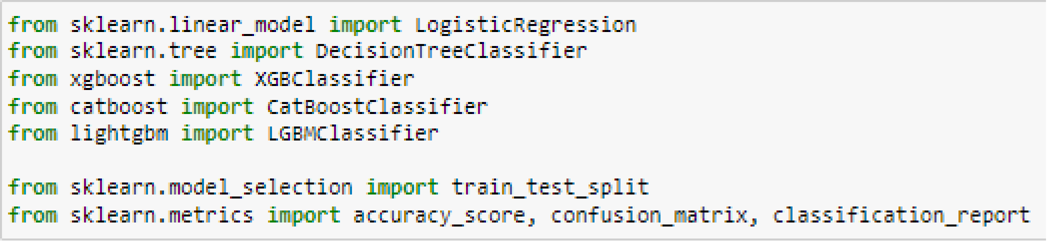
Software Requirements:

- Windows Version: Windows 10 Pro
- Anaconda Navigator: 2.0.3
  Anaconda offers the easiest way to perform Python/R data science and machine learning on a single machine.
- Jupyter Notebook: 6.3.0
  Anaconda offers the easiest way to perform Python/R data science and machine learning on a single machine.
- Python3: Python 3.9.9
  Python3 is used as the base environment for performing the machine learning and data analysis.

  Python Libraries Used:
- Pandas: Data manipulation and analysis
- NumPy: Adding support for large, multi-dimensional arrays and matrices, along with an enormous collection of high-level mathematical functions to operate on these arrays.
- Matplotlib, Seaborn: For visualization of variable relations and data distribution, and analysis.
- Sklearn: Simple and efficient tools for predictive data analysis.
- SciPy: SciPy provides algorithms for optimization, integration, interpolation, eigenvalue problems, algebraic equations, differential equations, statistics, and many other classes of problems.
- Statsmodels: Provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration.
- Xgboost, catboost, lightgbm: Gradient boosting framework that uses tree-based learning algorithms.
- Pickle: Implements binary protocols for serializing and de-serializing

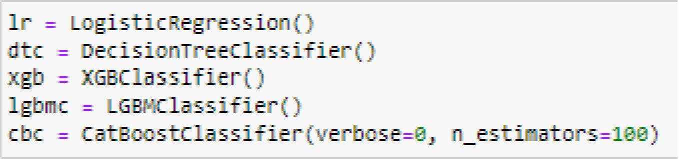
Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods)



```
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from xgboost import XGBClassifier
from catboost import CatBoostClassifier
from lightgbm import LGBMClassifier

from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
```

Importing the required libraries for model building



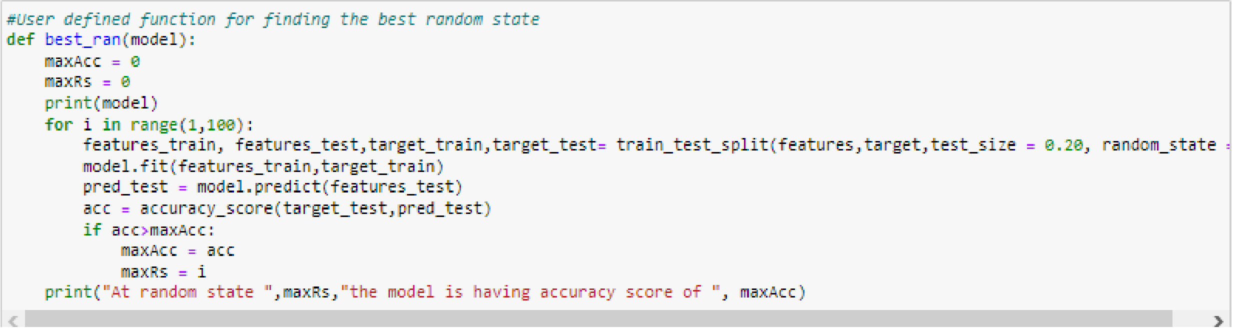
```
lr = LogisticRegression()
dtc = DecisionTreeClassifier()
xgb = XGBClassifier()
lgbmc = LGBMClassifier()
cbc = CatBoostClassifier(verbose=0, n_estimators=100)
```

Creating the instances of algorithms

We created three functions for testing the model and for cross validations:

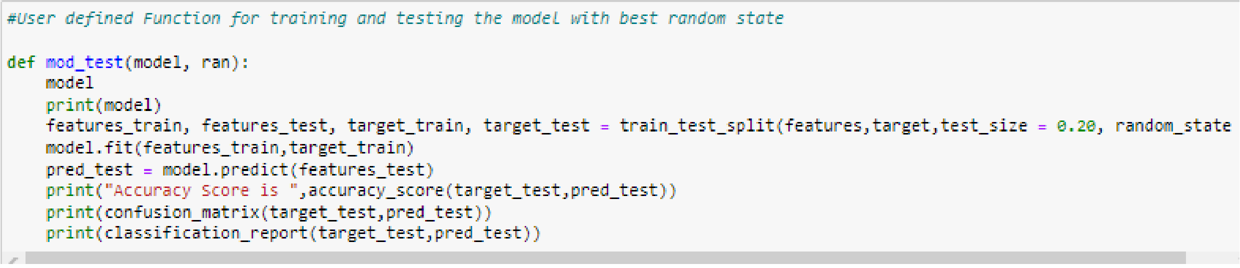
- best_ran: Finding the best random state for the selected model
- mod_test: Training the model with the train data using the best random state.
- cross_val: Finding the best cross validation mean score for each model.

Testing of Identified Approaches (Algorithms) and evaluation of selected models



```
#User defined function for finding the best random state
def best_ran(model):
    maxAcc = 0
    maxRs = 0
    print(model)
    for i in range(1,100):
        features_train, features_test,target_train,target_test= train_test_split(features,target,test_size = 0.20, random_state
        model.fit(features_train,target_train)
        pred_test = model.predict(features_test)
        acc = accuracy_score(target_test,pred_test)
        if acc>maxAcc:
            maxAcc = acc
            maxRs = i
    print("At random state ",maxRs,"the model is having accuracy score of ", maxAcc)
```

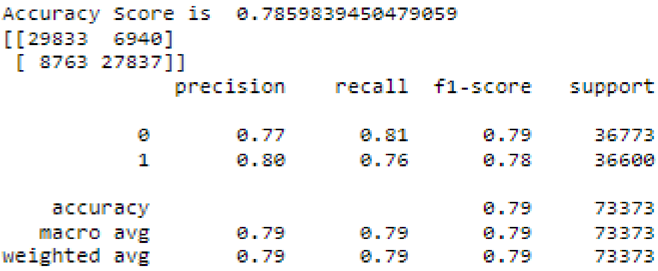
Code Snippet for function to find best random state



```
#User defined Function for training and testing the model with best random state

def mod_test(model, ran):
    model
    print(model)
    features_train, features_test, target_train, target_test = train_test_split(features,target,test_size = 0.20, random_state
    model.fit(features_train,target_train)
    pred_test = model.predict(features_test)
    print("Accuracy Score is ",accuracy_score(target_test,pred_test))
    print(confusion_matrix(target_test,pred_test))
    print(classification_report(target_test,pred_test))
```

Code Snippet for function to test the model

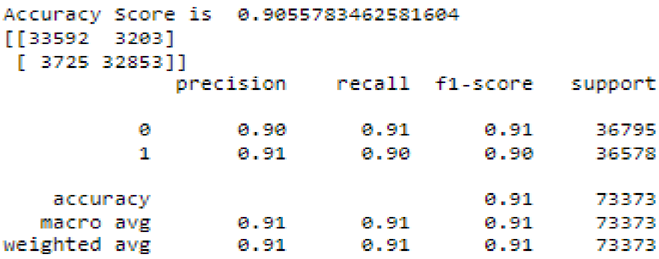
- Logistic Regression



```
Accuracy Score is  0.7859839450479059
[[29833  6940]
 [ 8763 27837]]
              precision    recall  f1-score   support

           0       0.77      0.81      0.79     36773
           1       0.80      0.76      0.78     36600

    accuracy                           0.79     73373
   macro avg       0.79      0.79      0.79     73373
weighted avg       0.79      0.79      0.79     73373
```

Testing the model performance

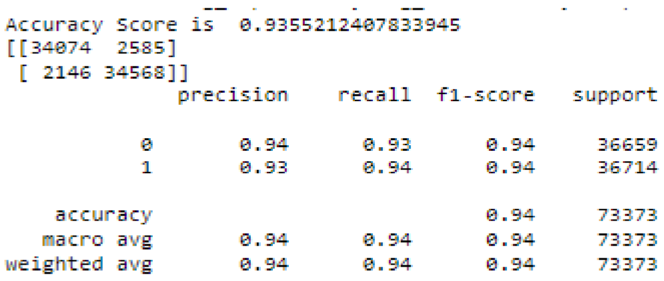
- DecisionTree Classifier



```
Accuracy Score is  0.9055783462581604
[[33592  3203]
 [ 3725 32853]]
              precision    recall  f1-score   support

           0       0.90      0.91      0.91     36795
           1       0.91      0.90      0.90     36578

    accuracy                           0.91     73373
   macro avg       0.91      0.91      0.91     73373
weighted avg       0.91      0.91      0.91     73373
```

Testing the model performance

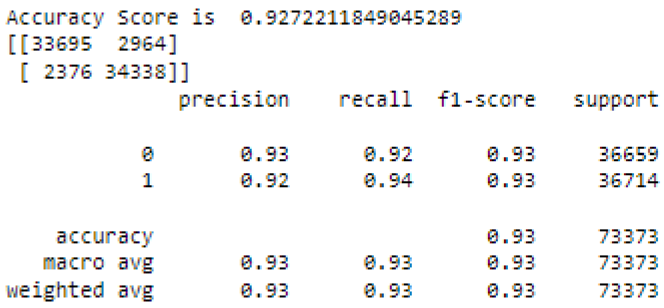
- XGB Classifier



```
Accuracy Score is  0.9355212407833945
[[34074  2585]
 [ 2146 34568]]
              precision    recall  f1-score   support

           0       0.94      0.93      0.94     36659
           1       0.93      0.94      0.94     36714

    accuracy                           0.94     73373
   macro avg       0.94      0.94      0.94     73373
weighted avg       0.94      0.94      0.94     73373
```

Testing the model performance

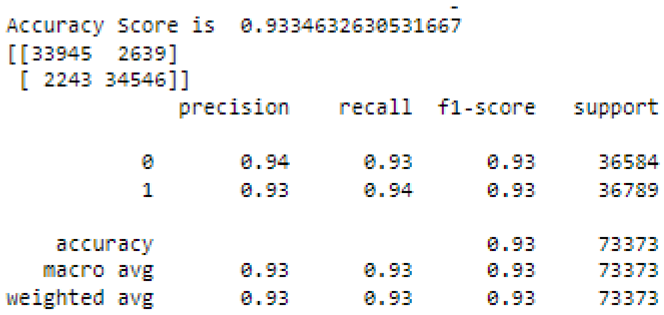
- LGBM Classifier



```
Accuracy Score is  0.9272211849045289
[[33695  2964]
 [ 2376 34338]]
              precision    recall  f1-score   support

           0       0.93      0.92      0.93     36659
           1       0.92      0.94      0.93     36714

    accuracy                           0.93     73373
   macro avg       0.93      0.93      0.93     73373
weighted avg       0.93      0.93      0.93     73373
```

Testing the model performance

- CatBoost Classifier

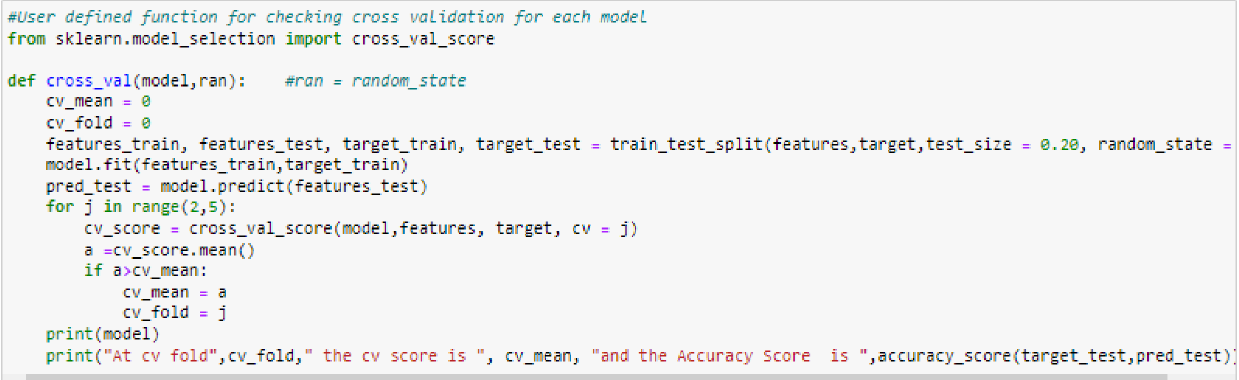


```
Accuracy Score is  0.9334632630531667
[[33945  2639]
 [ 2243 34546]]
              precision    recall  f1-score   support

           0       0.94      0.93      0.93     36584
           1       0.93      0.94      0.93     36789

    accuracy                           0.93     73373
   macro avg       0.93      0.93      0.93     73373
weighted avg       0.93      0.93      0.93     73373
```

Testing the model performance

After testing the models, the xgb classifier(xgb) is performing well by providing the maximum accuracy score of 93.55%. After testing the models, the xgb classifier(xgb) is performing well by providing the maximum accuracy score of 93.55%.

Cross Validation



```
#User defined function for checking cross validation for each model
from sklearn.model_selection import cross_val_score

def cross_val(model,ran):    #ran = random_state
    cv_mean = 0
    cv_fold = 0
    features_train, features_test, target_train, target_test = train_test_split(features,target,test_size = 0.20, random_state =
    model.fit(features_train,target_train)
    pred_test = model.predict(features_test)
    for j in range(2,5):
        cv_score = cross_val_score(model,features, target, cv = j)
        a =cv_score.mean()
        if a>cv_mean:
            cv_mean = a
            cv_fold = j
    print(model)
    print("At cv fold",cv_fold," the cv score is ", cv_mean, "and the Accuracy Score  is ",accuracy_score(target_test,pred_test)
```

Code Snippet for function to find the cross validation mean score

- LogisticRegression

At cv fold 4 the cv score is 0.7816590794547977 and the Accuracy Score is 0.7859839450479059

- DecisionTree Classifier

At cv fold 4 the cv score is 0.8985968400822768 and the Accuracy Score is 0.9066005206274788

- XGB Classifier

At cv fold 4 the cv score is 0.9258223768870628 and the Accuracy Score is 0.9355212407833945

- LGBM Classifier

At cv fold 3 the cv score is 0.9197056224400632 and the Accuracy Score is 0.9272211849045289

- CatBoost Classifier

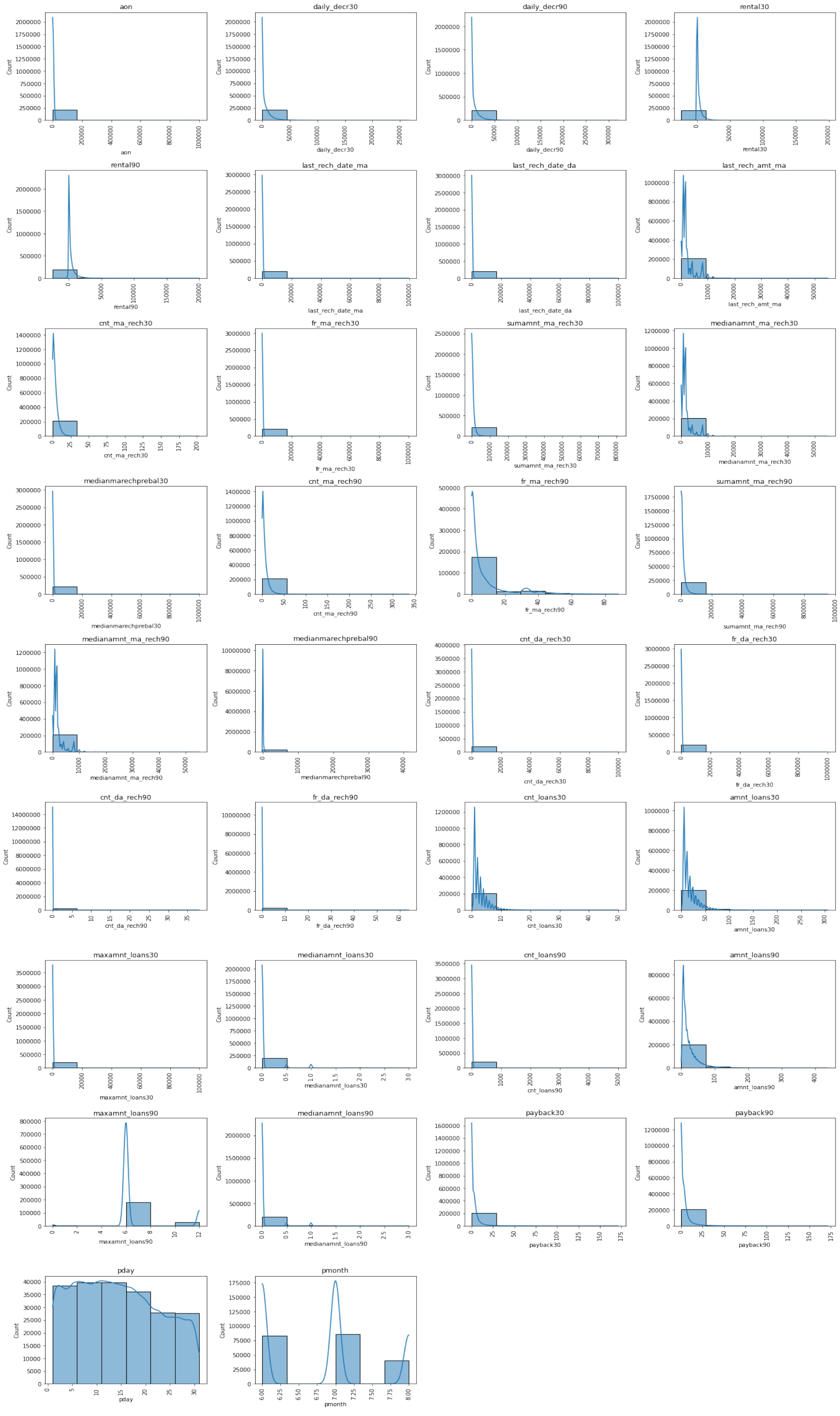
At cv fold 4 the cv score is 0.9243395300600212 and the Accuracy Score is 0.9334632630531667

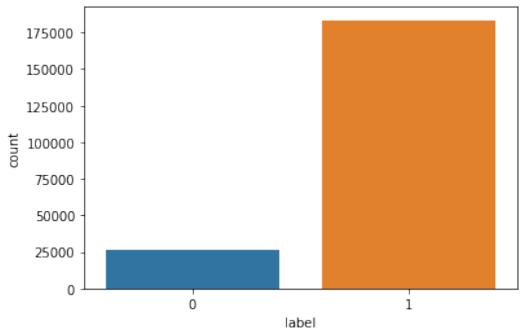
Key Metrics for success in solving problem under consideration

- Accuracy Score: Metric that summarizes the performance of a classification model as the number of correct predictions divided by the total number of predictions.
- Confusion Matrix: Performance measurement for machine learning classification problem where output can be two or more classes. It is a table with 4 different combinations of predicted and actual values.
- Classification Report: Displays your model's precision, recall, F1 score and support. It provides a better understanding of the overall performance of our trained model.

Visualization

Univariate Analysis



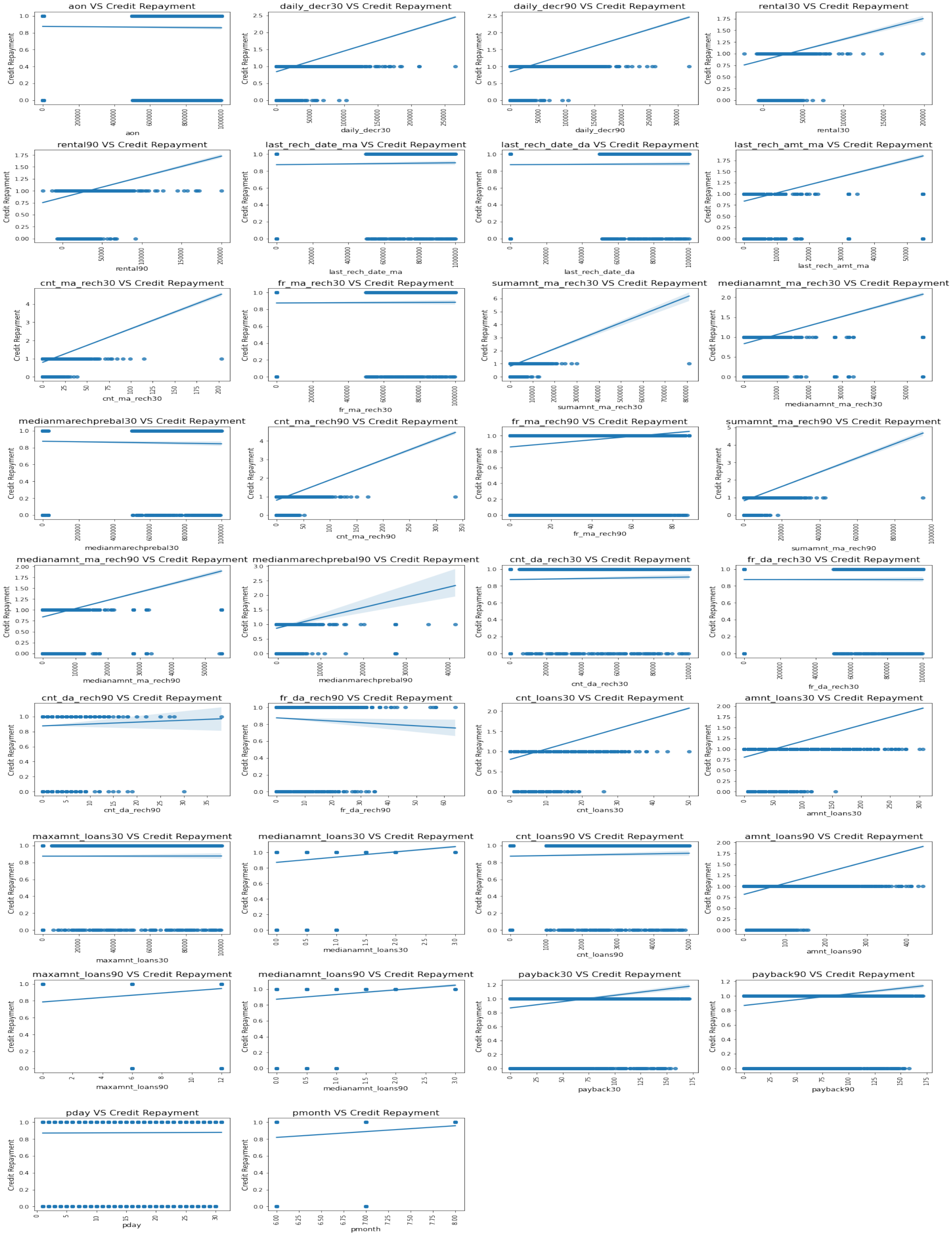


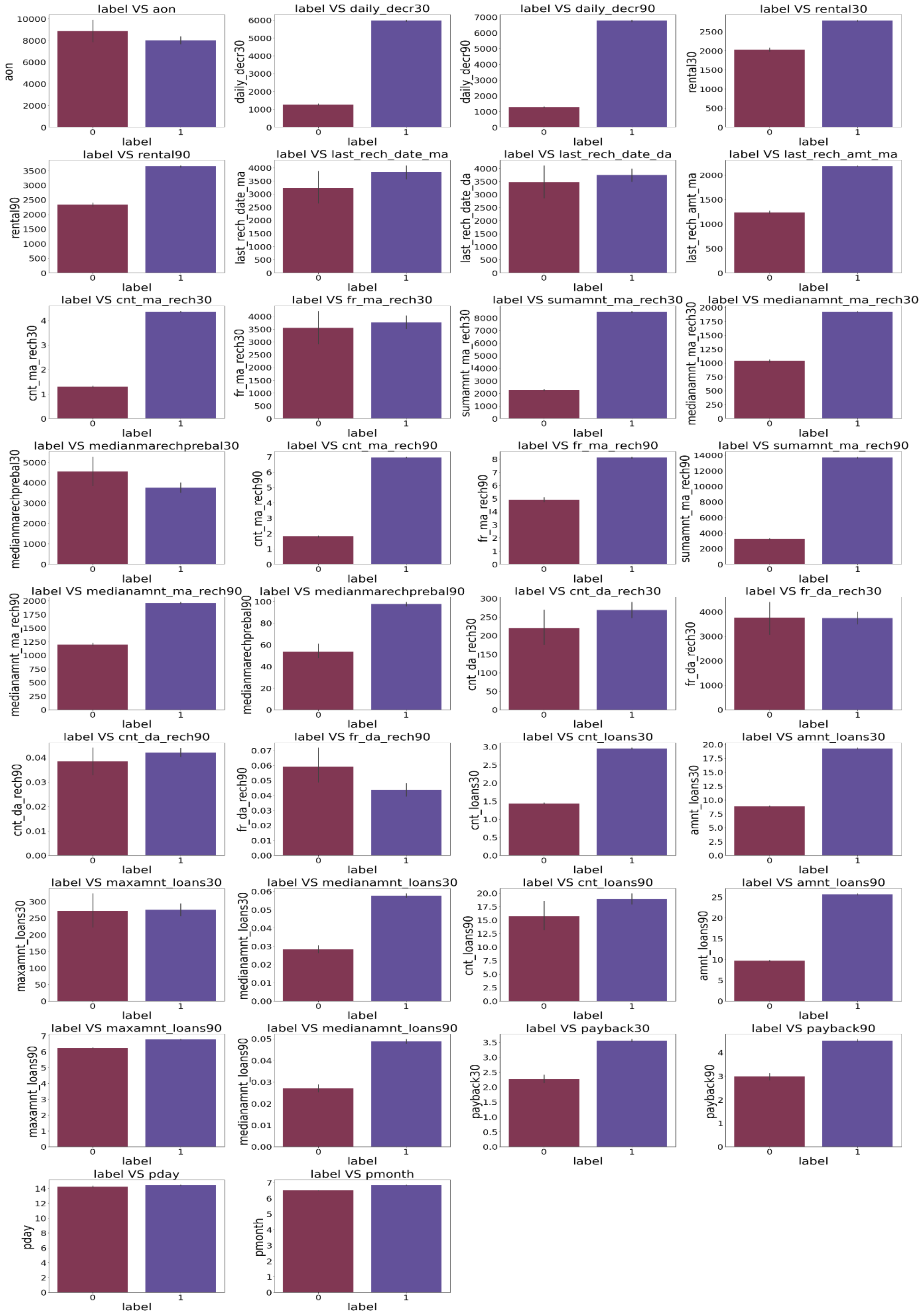
Observations:

- The average age on cellular network in days for customers ranging between -48 to 1.7 lakh.
- Most customers have spent less than Rs. 44,000 from main account averaged over last 30 and spent more than Rs. 53,000 over last 90 days.
- The average main account balance of most of customers are ranging between Rs. -23000 to Rs. 14000 over last 30 days and Rs. -25000 to Rs. 13000 over last 90 days.
- The average number of days from recharge of main account and data account for most of customers ranging between -29 to 1.67 lakh.
- Most customers have recharged their main account for an amount ranging between Rs. 0 to Rs. 9000.
- The number of times customers recharged their main account in last 30 days is ranging between 0 to 34 times and in last 90 days is ranging between 0 to 56 times
- Frequency of main account recharge for most of the customers in last 30 days is ranging between 0 to 1.67 lakh and in last 90 days is ranging between 0 to 14.67.
- Total amount of recharge for most of the customers for their main account in last 30 days is between the range Rs. 0 to Rs. 1.35 lakh and the median amount is between the range Rs. 0 to Rs. 9000. While the total amount of recharge for their main account in last 90 days is ranging between Rs. 0 to Rs. 1.59 lakh and the median amount of recharge is between Rs. 0 to Rs. 9000.
- The median main balance of most of customers just before recharge in last 30 days is between Rs-200 to Rs. 1.67 Lakh and in last 90 days is between Rs. 200 to Rs. 6700.
- Most of the customers have recharged their data account between 0 to 16000 times in last 30 days and 0 to 6.33 times in last 90 days.
- The frequency of recharge for most of customers for their data account is ranging from 0 to 1.67 lakh times in last 30 days and 0 to 11 times in last 90 days.
- Most customers have recharged their data account between 0 to 1.67 lakh times in last 30 days and 0 to 6 times in last 90 days.
- Number of loans taken by most customers is between 0 to 8 times in last 30 days and 0 to 832 times in last 90 days

- Total amount of loan taken by most customers is ranging between Rs. 0 to Rs. 51 in last 30 days and Rs. 73 in last 90 days.
- Maximum amount of loan taken by most customers is between Rs. 0 to Rs. 17000 in last 30 days and Rs. 2 in last 90 days.
- Median amount of loan taken by most customers is between Rs. 0 to Rs. 0.5 in last 30 days and Rs. 0.5 in last 90 days
- Average payback time for most customers is between the range 0 to 28 days over last 30 and 90 days.
- The data was collected for three months: - June, July, and August. The data is uniform for most of the days, but a greater number of data is collected from the beginning of the month, and it is reducing towards the end of month.
- Most of the customers have paid back the credit amount within 5 days of issuing loan amount.

Bivariate Analysis




label VS aon
aon
0
2000
4000
6000
8000
10000
label
0
1
label VS daily_decr30
daily_decr30
0
1000
2000
3000
4000
5000
6000
label VS daily_decr90
daily_decr90
0
1000
2000
3000
4000
5000
6000
7000
label VS rental30
rental30
0
500
1000
1500
2000
2500
label VS rental90
rental90
0
500
1000
1500
2000
2500
3000
3500
label VS last_rech_date_ma
last_rech_date_ma
0
500
1000
1500
2000
2500
3000
3500
4000
label VS last_rech_date_da
last_rech_date_da
0
500
1000
1500
2000
2500
3000
3500
4000
label VS last_rech_amt_ma
last_rech_amt_ma
0
500
1000
1500
2000
label VS cnt_ma_rech30
cnt_ma_rech30
0
1
2
3
4
label VS fr_ma_rech30
fr_ma_rech30
0
500
1000
1500
2000
2500
3000
3500
4000
label VS sumamnt_ma_rech30
sumamnt_ma_rech30
0
1000
2000
3000
4000
5000
6000
7000
8000
label VS medianamnt_ma_rech30
medianamnt_ma_rech30
0
250
500
750
1000
1250
1500
1750
2000
label VS medianmarechprebal30
medianmarechprebal30
0
1000
2000
3000
4000
5000
label VS cnt_ma_rech90
cnt_ma_rech90
0
1
2
3
4
5
6
7
label VS fr_ma_rech90
fr_ma_rech90
0
1
2
3
4
5
6
7
8
label VS sumamnt_ma_rech90
sumamnt_ma_rech90
0
2000
4000
6000
8000
10000
12000
14000
label VS medianamnt_ma_rech90
medianamnt_ma_rech90
0
250
500
750
1000
1250
1500
1750
2000
label VS medianmarechprebal90
medianmarechprebal90
0
20
40
60
80
100
label VS cnt_da_rech30
cnt_da_rech30
0
50
100
150
200
250
300
label VS fr_da_rech30
fr_da_rech30
0
1000
2000
3000
4000
label VS cnt_da_rech90
cnt_da_rech90
0.00
0.01
0.02
0.03
0.04
label VS fr_da_rech90
fr_da_rech90
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
label VS cnt_loans30
cnt_loans30
0.0
0.5
1.0
1.5
2.0
2.5
3.0
label VS amnt_loans30
amnt_loans30
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
label VS maxamnt_loans30
maxamnt_loans30
0
50
100
150
200
250
300
label VS medianamnt_loans30
medianamnt_loans30
0.00
0.01
0.02
0.03
0.04
0.05
0.06
label VS cnt_loans90
cnt_loans90
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
label VS amnt_loans90
amnt_loans90
0
5
10
15
20
25
label VS maxamnt_loans90
maxamnt_loans90
0
1
2
3
4
5
6
7
label VS medianamnt_loans90
medianamnt_loans90
0.00
0.01
0.02
0.03
0.04
0.05
label VS payback30
payback30
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
label VS payback90
payback90
0
1
2
3
4
label VS pday
pday
0
2
4
6
8
10
12
14
label VS pmonth
pmonth
0
1
2
3
4
5
6
7

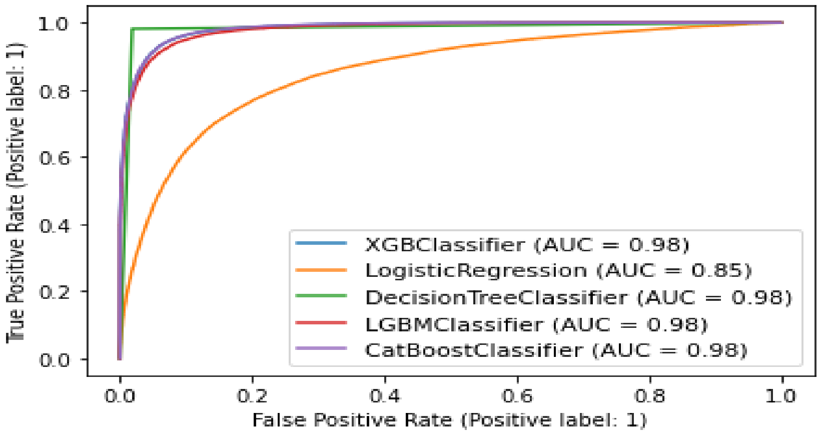
Observations:

- Age of cellular network in days is not having much impact on the loan defaulters. The number of credit defaulters are slightly higher if the age of cellular network is higher.
- Customers who have least amount of spent from main account over last 30 and 90 days are more likely to be a defaulter.
- Customers who maintain low average main account balance over last 30 and 90 days are more likely to be a defaulter.
- Customers who had a greater number of recharges of main account and data account in past 30 and 90 days are not likely to be a defaulter.
- Customers who have recharged main account for higher amount have made the repayment of credit on time.
- Customers who are having frequent recharges in last 30 and 90 days are less likely to be a defaulter.
- Customers whose total amount and median amount of recharge in last 30 and 90 days is lower are more likely to be a defaulter.
- Customers who have higher median main balance just before last recharge in 30 days are more likely to be a defaulter, while the customers who have higher median main balance just before last recharge in 90 days are less likely to be a defaulter.
- Customers who have higher number of data account recharges in last 30 and 90 days are less likely to be a defaulter.
- Customers who have higher frequency in recharge of data account in last 90 days are more likely to be a defaulter, while the frequency of recharges of data account in last 30 days is not having much impact on identifying whether the customer will be a defaulter or not.
- Most number of customers who have taken most number of loans and higher amount of loan over the last 30 days and 90 days are less likely to be a defaulter.
- Maximum amount taken by customers as loan over the last 30 and 90 days are not providing much information on whether the customer would be a defaulter or not.
- Customers who have higher median loan amount over the last 30 and 90 days are less likely to be defaulter.
- Customers who payback the amount in more days over the last 30 and 90 days are less likely to be a defaulter.
- The day and month of which the data was collected is not having much impact on the identifying whether the customer would be a defaulter or not.

Interpretation of the Results

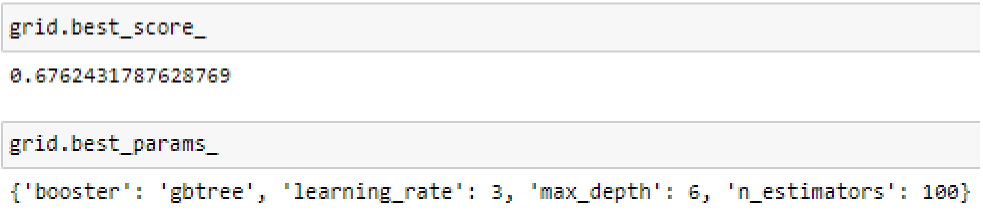
- Most of the columns are having positive correlation to the target variable.
- The columns ['fr_da_rech30', 'aon', 'medianmarechprebal30', 'fr_da_rech90'] are having negative correlation to the target variable 'label'. Rest of the variables are having positive correlation to the target variable.
- The column 'cnt_ma_rech30' is having highest positive correlation to the target variable 'label', while the column 'maxamnt_loans30' is having least positive correlation to the target variable 'label'.
- The column 'fr_da_rech30' is having the least negative correlation to the target variable 'label', while the column 'fr_da_rech90' is having the highest negative correlation to the target variable 'label'.

AUC ROC Curve



All the models except logistic regression model is providing the maximum AUC ROC score. Since the XGB model is performing well with all the training, testing, cross validations and auc score, we can consider this model as the best performing model.

Hyperparameter Tuning



```
grid.best_score_
```

```
0.6762431787628769
```

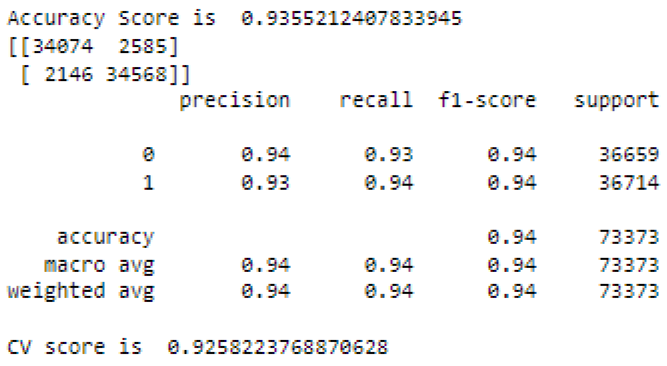
```
grid.best_params_
```

```
{'booster': 'gbtree', 'learning_rate': 3, 'max_depth': 6, 'n_estimators': 100}
```

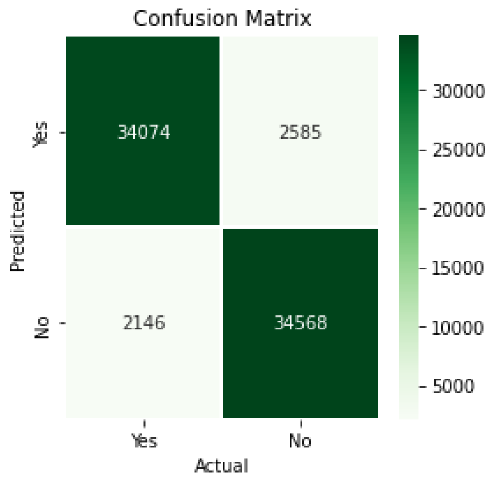
Best tuning score and best tuning parameters

After all the tests, cross validations and tuning the XGB classifier(xgb) model is performing well and providing maximum accuracy score of 93.55%.

Finalized Model Performance

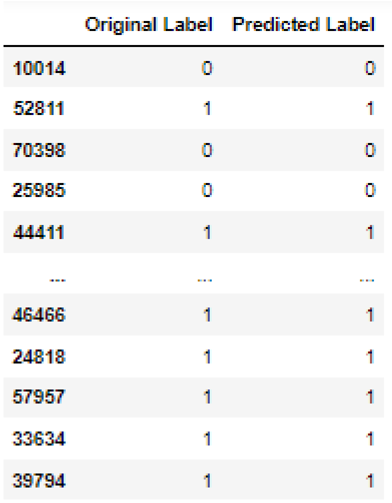


```
Accuracy Score is  0.9355212407833945
[[34074  2585]
 [ 2146 34568]]
              precision    recall  f1-score   support

           0       0.94      0.93      0.94     36659
           1       0.93      0.94      0.94     36714

    accuracy                           0.94     73373
   macro avg       0.94      0.94      0.94     73373
weighted avg       0.94      0.94      0.94     73373

CV score is  0.9258223768870628
```

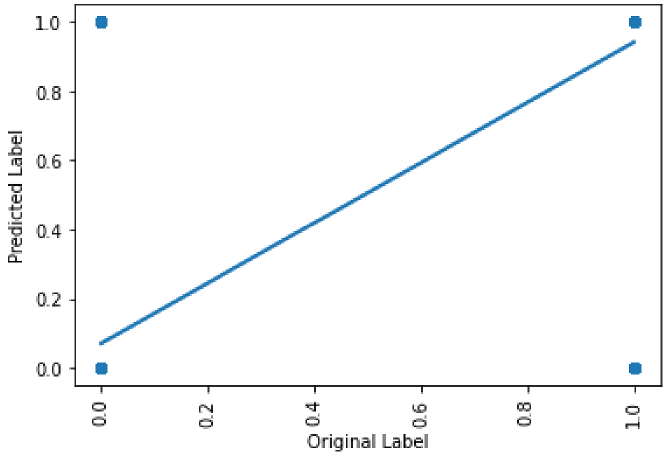


Final Model Performance

Now we have trained our model and it is ready to test with the actual data to cross verify the performance.



	Original Label	Predicted Label
10014	0	0
52811	1	1
70398	0	0
25985	0	0
44411	1	1
...	...	...
46466	1	1
24818	1	1
57957	1	1
33634	1	1
39794	1	1

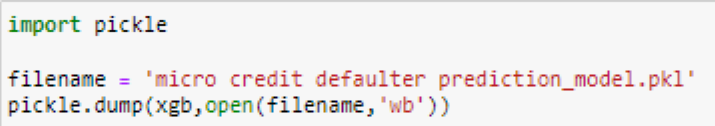


Model Predictions and Actual Label

Model performance with predictions vs actual label

Our model is performing well with predictions and provided accurate results.

Saving the best model



```
import pickle

filename = 'micro credit defaulter prediction_model.pkl'
pickle.dump(xgb,open(filename,'wb'))
```

We have saved the machine learning model for future predictions. We have serialized and saved the binary file as “micro credit defaulter prediction_model.pkl” using the pickle library.

CONCLUSION

Key Findings and Conclusions of the Study

With the help of data science and machine learning, we were able to create a machine learning model using XGBoost algorithm, which can predict whether the customer will be a defaulter or not.

Now this model can be used to predict whether a customer will be a defaulter or non-defaulter of credit repayment based on the following features:

`cnt_ma_rech90`

`sumamnt_ma_rech90`

`last_rech_amt_ma`

`medianamnt_loans30`

`daily_decr90`

`medianmarechprebal30`

`pday`

`amnt_loans90`

`medianamnt_ma_rech90`

`cnt_loans90`

`payback30` `payback90`

`pmonth`

`rental90` `maxamnt_loans90`

`cnt_da_rech90`

`cnt_ma_rech30`

`cnt_loans30`

`medianamnt_loans90`

`medianamnt_ma_rech30`

`medianmarechprebal90`

`fr_ma_rech90`

`aon`

Impact of Variables on Target Variable (Correlation)

- Most of the columns are having positive correlation to the target variable.
- The columns ['fr_da_rech30', 'aon', 'medianmarechprebal30', 'fr_da_rech90'] are having negative correlation to the target variable 'label'. Rest of the variables are having positive correlation to the target variable.
- The column 'cnt_ma_rech30' is having highest positive correlation to the target variable 'label', while the column 'maxamnt_loans30' is having least positive correlation to the target variable 'label'.
- The column 'fr_da_rech30' is having the least negative correlation to the target variable 'label', while the column 'fr_da_rech90' is having the highest negative correlation to the target variable 'label'.

Learning Outcomes of the Study in respect of Data Science

Artificial intelligence (AI) and machine learning are not new, but microcredit organisations are reluctant to incorporate these techniques into their process for determining credit risk. Instead, they continue to use traditional credit scoring techniques that are based on the linear calculation of a limited number of indicators. Results from this scoring model are inconsistent and unreliable. On the other hand, machine learning provides a far broader perspective of a client and can be used to control not only credit risk but also other company hazards.

In order to improve predictions for other risks as well as credit risks, such as the potential for early repayment leading to interest income losses, the potential for money laundering, etc., machine learning algorithms enable the creation of new models using historical anonymized data that has already been collected. Financial institutions might use a decent model to forecast the likelihood that a client will return the loan before the maturity date and then set up procedures with pre-set preventive actions to be taken, prior to this occurring.

Limitations of this work and Scope for Future Work

Limitations

- The data was vast, and the client tried to include all the important information that would benefit the model building. But the data was noisy, and we had to left out some part of data during the pre-processing.
- Most of the features were having skewed data and huge outliers and high multicollinearity between the variables. This has resulted in complex model.
- Even though the data included many crucial features that had impact on the output, there are still many other factors that influence the output such as the financial freedom of the customer, their spending behaviour, economic condition, inflation etc.

Scope

- Future work may incorporate different relational learners, such as a network only link based classifier when a denser sample set of loans is available, given that network sparsity may also contribute to the substandard outcomes.
- We can also consider including less noisy data as it will be helpful for the model to easily understand the patterns impacting the outcomes and also it will not lose much data during pre-processing, thus we can preserve the true nature of the data throughout the model building.
- This study has only focused on whether the customer would be a defaulter or not, but there are other important inferences which can be beneficial to the company such as predicting the time of repayment of loan for each customer, chances of early settlement of loan etc.

Thank You