FLIP ROBO

MICRO CREDIT DEFAULTER MODEL

Submitted By -

Ganesh Kumbhar

Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. All that I have done is only due to such supervision and assistance and I would not forget to thank them.

I respect and thank Mr. Harsh Ayush, for providing me an opportunity to do the project work in FlipRobo Technologies and giving us all support and guidance which made me complete the project duly. I am extremely thankful to him for providing such a nice support and guidance.

I owe my deep gratitude to everyone ho guided me.

- ➢ *FlipRobo Technologies*
- ➢ *DataTrained Academy*
- ➢ *Krush Naik*
- ➢ *Code With Harry*
- ➢ *Dr. Abhinanda Sarkar*

Some mentors helped me with their research work mentioned as follow -

https://towardsdatascience.com/style-pandas-dataframe-like-a-master-6b02bf6468b0

https://towardsdatascience.com/feature-selection-correlation-and-p-value-da8921bfb3cf

https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/

I am thankful to and fortunate enough to get constant encouragement, support and guidance, which helped me in successfully completing this project work.

INTRODUCTION

Business Problem Framing

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,

Our client that is in Telecom Industry is 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.

This problem is very much related to real world as we see in india also companies like Idea, Airtel, Vodafone give emergency loan within one second and after digital revolution in india we see that many companies suffering losses and for them to survive and thrive they can use this approach to see detect the defaulters and increase the revenue of company.

Conceptual Background of the Domain Problem

The telecom companies work on credit basis, you give them value in form of money and they provide you their service but as this data is from 2016 and the economic condition of Indonesia is towards downward spiral and so there doing a recharge is not easy as it now in 2020, people has to go to a certain location and to a specific person to do a recharge, and giving them loan is easy option to increase the revenue as well as to help the person in need when he/she may not have money at hand or person to recharge.

People can take a loan of 5 and 10 rupees and for that they have to pay 6 and 12 rupees respectively ,which means 1 rupees interest for 5 rupees and that they have to pay in 5 days.

Once a person becomes a defaulter then the company will not give him/her loan.

A person can have more than 1 and 2 connections from one single documentation thus duplicate numbers with different data can be seen.

Review of Literature

Z Score - Plays an important role when is comes to know the distribution of the of data with this we can detect the various things like outliers,
It is a numerical measurement that describes a value's relationship to the mean of a group of values. Z-score is measured in terms of standard deviation from the mean. If a Z-score is 0, it indicates that the data point's score is identical to the mean score. A Z-score of 1.0 would indicate a value that is one standard deviation from the mean. Z-scores may be positive or negative, with a positive value indicating the score is above the mean and a negative score indicating it is below the mean.
The Z-score is limited to get the values within range but not the genuine values which are outside the range; it detects them as outliers.

Feature Selection - To avoid the curse of dimensionality, and also to avoid overfitting and under filling we should select features which are very important to the data. All of the features we find in the dataset might not be useful in building a machine learning model to make the necessary prediction. Using some of the features might even make the predictions worse. So, feature selection plays a huge role in building a machine learning model.
I learned various methods to select the appropriate features.

- Variance
- P-Value
- Correlation
- Chi-Square Test
- Anova Test

- Co-Independence
- Visualization

Conclusion and Need for Additional Research - Removal of outliers plays very important role in as it manipulate a fine percentage of data and currently the known methods are zero, mean, median , mode , Z-score but i need to do more research in order to get the data which is outside the standard deviation.

Motivation for the Problem Undertaken

Use of Telecom is very essential when it comes to development of a country and providing these micro loans to a poor living in underdeveloped areas or maybe a person who doesn't have money or resources to recharge immediately this loan can really help them to be connected which can decrease many problems.

Giving the loan process should be monitored very carefully as there are more customers in this category and if loan is provided anonymously then the company can suffer heavy loss which will decrease the expansion of a company. Hence, it will affect both Nation people and the company with the implementation of data science with developing country as well as company help the company by not giving the loan to a person which is going to be defaulter therefor it'll help the company to thrive more and a person in need will get a loan every time.

Analytical Problem Framing

Mathematical/ Analytical Modeling of the Problem

Outliers - When you get the description of a data set we see that data that has extreme outliers and these outliers are not 1 or 2 points above 3rd quartile meanwhile they are the distribution of data which is disturbing the mean,variance and standard deviation of data.

We chose to remove the outliers by the distribution of data and for that we used Z score and as our data is important to us so expanded our standard threshold value of Z score that is +-3 to +-5 so that we will lose less data.

Negative Values - In this data set many features have negative value which is not even possible so with the understanding of description of feature we change data from negative to positive.

Grouping - The data is given to us in continuous format and it will be very hard for us to classify people into 4 categories i.e in 1st , 2nd ,3rd and 4th quartile, and we made a group of people based on their age on the network, amount they spent, loan they take.

Dropping Unnecessary columns

1. *By Uniqueness :-* First we separated the mobile number by I , to see the relationship between first and second number if any, but we see no relationship and as the number are unique even though some numbers are repeating but still their data is different so we can consider them as unique and they provide no information in data analysis and as well as machine learning whatsoever, so we dropped that column.

   We drop the column of serial number at it has all unique value

2. *By Zero Variance:-* We also dropped P-circle and Year column that is extracted from date column because they have zero variance.*By*
3. *Correlation:-* We dropped all the column in feature selection who has more than 80% correlation between themselves

Data Sources and their formats

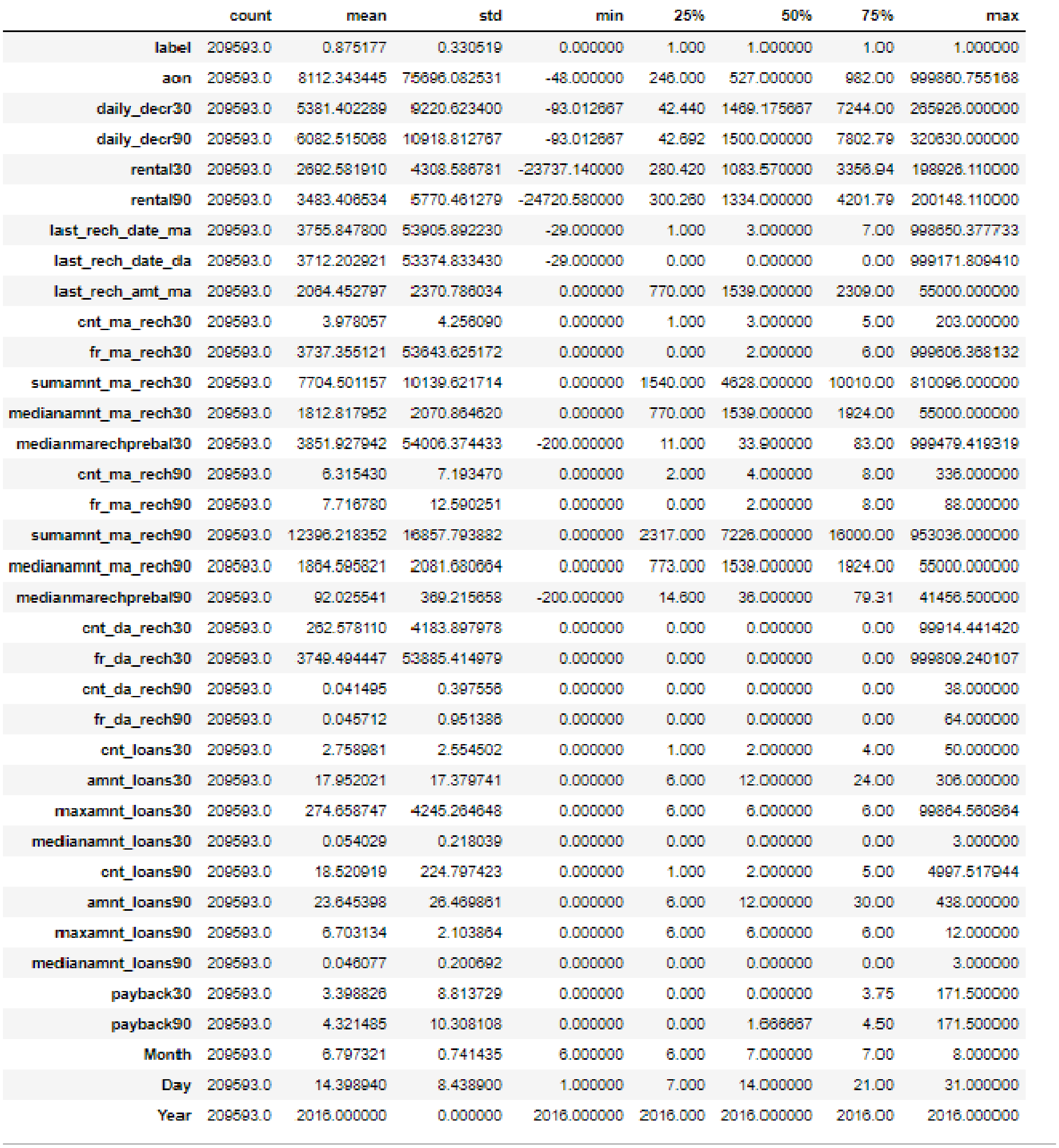
The data has 209593 rows × 37 columns, there are 36 independent feature and 1 dependent feature which is the output.

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

The statical summary of the data are follow



	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
fr_ma_rech90	209593.0	7.716780	12.590251	0.000000	0.000	2.000000	8.00	88.000000
sumamnt_ma_rech90	209593.0	12396.218352	16857.793882	0.000000	2317.000	7226.000000	16000.00	953036.000000
medianamnt_ma_rech90	209593.0	1864.595821	2081.680664	0.000000	773.000	1539.000000	1924.00	55000.000000
medianmarechprebal90	209593.0	92.025541	369.215658	-200.000000	14.600	36.000000	79.31	41456.500000
cnt_da_rech30	209593.0	262.578110	4183.897978	0.000000	0.000	0.000000	0.00	99914.441420
fr_da_rech30	209593.0	3749.494447	53885.414979	0.000000	0.000	0.000000	0.00	999809.240107
cnt_da_rech90	209593.0	0.041495	0.397556	0.000000	0.000	0.000000	0.00	38.000000
fr_da_rech90	209593.0	0.045712	0.951386	0.000000	0.000	0.000000	0.00	64.000000
cnt_loans30	209593.0	2.758981	2.554502	0.000000	1.000	2.000000	4.00	50.000000
amnt_loans30	209593.0	17.952021	17.379741	0.000000	6.000	12.000000	24.00	306.000000
maxamnt_loans30	209593.0	274.658747	4245.264648	0.000000	6.000	6.000000	6.00	99864.560864
medianamnt_loans30	209593.0	0.054029	0.218039	0.000000	0.000	0.000000	0.00	3.000000
cnt_loans90	209593.0	18.520919	224.797423	0.000000	1.000	2.000000	5.00	4997.517944
amnt_loans90	209593.0	23.645398	26.469861	0.000000	6.000	12.000000	30.00	438.000000
maxamnt_loans90	209593.0	6.703134	2.103864	0.000000	6.000	6.000000	6.00	12.000000
medianamnt_loans90	209593.0	0.046077	0.200692	0.000000	0.000	0.000000	0.00	3.000000
payback30	209593.0	3.398826	8.813729	0.000000	0.000	0.000000	3.75	171.500000
payback90	209593.0	4.321485	10.308108	0.000000	0.000	1.666667	4.50	171.500000
Month	209593.0	6.797321	0.741435	6.000000	6.000	7.000000	7.00	8.000000
Day	209593.0	14.398940	8.438900	1.000000	7.000	14.000000	21.00	31.000000
Year	209593.0	2016.000000	0.000000	2016.000000	2016.000	2016.000000	2016.00	2016.000000

Data Preprocessing

1. We Checked if the mobile number can give us any information and as a result it didn't so we dropped that column.
2. We remove outliers by the main distribution method that is by Z-Score, keeping the threshold value + -5.
3. We made a group of various columns into 1st ,2nd, 3rd and 4th quartile to do the data analysis.
4. The data set has very high skewness and we removed it by using a cube root method.
5. As data set is imbalance by 87% so we did the oversampling by SMTOTE
6. To make data in standard scale we used the Standard Scaling method.
7. We removed all the data which has 0 variance as they provide no value to machine learning.
8. Then we removed all the features which have high correlation between themselves.
9. We used train_test_split to split data for machine learning.

Data Inputs- Logic- Output Relationships

1. Average Balance has a direct relationship with output as when people have more balance they tend to return the loan.
2. People who do recharge more frequently have more probability after returning the loan.
3. People who do data recharge are more likely that they will not take the loan.
4. If people want to return the long then they pay back within 5 days otherwise the probability of these returning their loan decreases

The set of assumptions related to the problem under consideration

<u>Mobile no</u> - We see that there are many approximately 23000 similar numbers in the data set but when we see their features they all are different so we assumed two things here.

1. The mobile number is a Unique Identification number that is provided to a document and bi that document multiple number is issued to friends and family.
2. When the user changed and the mobile number didn't, so the company took data of both the instances.

<u>Negative Values-</u>

Many features of negative values which are not possible like age , amount spent , days, date for that we assumed that. It's just human error and that's why the number is written as negative.

Hardware and Software Requirements and Tools Used

Software

- ➔ Jupyter Notebook (Python 3.8)
- ➔ Microsoft Excel
- ➔ Microsoft Word

Hardware

- ➔ Processor - Intel i5 9th Gen
- ➔ RAM - 8 GB
- ➔ Graphic Memory - 4Gb , Nvidia 1060

Libraries

- ➔ Pandas
- ➔ Numpy
- ➔ Matplotlib
- ➔ Seaborn

- ➔ Scipy
- ➔ Sklearn

Model/s Development and Evaluation

Problem-solving approaches

Feature Selection - We used two approaches to select the feature first selection of feature by 0 variance and second selection of feature by the internal correlation.
We splitted the data by using train_test_split method and analysed the features on x_train and removed them from x_train and then simply removed the feature from x_test to reduce the chance of overfitting.

Imbalance Dataset - The data is highly imbalanced about 87% and to make a model more precise we used over sampling method to do the balancing of a data

Standard Scaling - The scale of data has high variance and to put data in in one scale which will increase the efficiency of our model for that we use standard scaler library.

Skewness - The data is very much skewed and to remove the skewness we used cube root method this method is capable of dealing with high skewness as well s features with 0 value but still skewness is not completely gone but we cannot remove the data any further as data is precious to us.

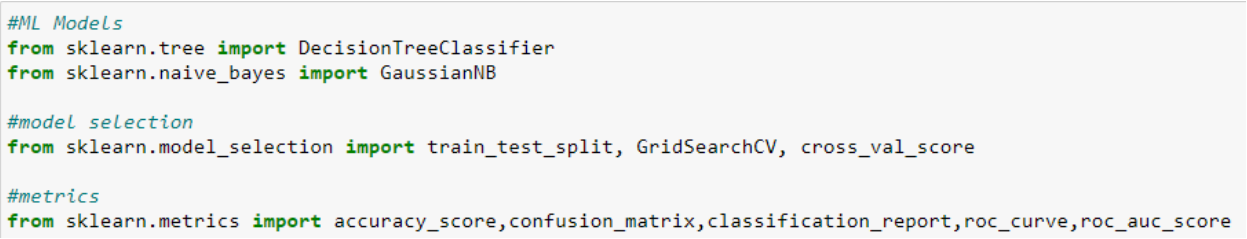
Testing of Identified Approaches.

- DecisionTreeClassifier
- GaussianNB
- RandomForestClassifier
- AdaBoostClassifier

- GradientBoostingClassifier
- BaggingClassifier
- ExtraTreesClassifier

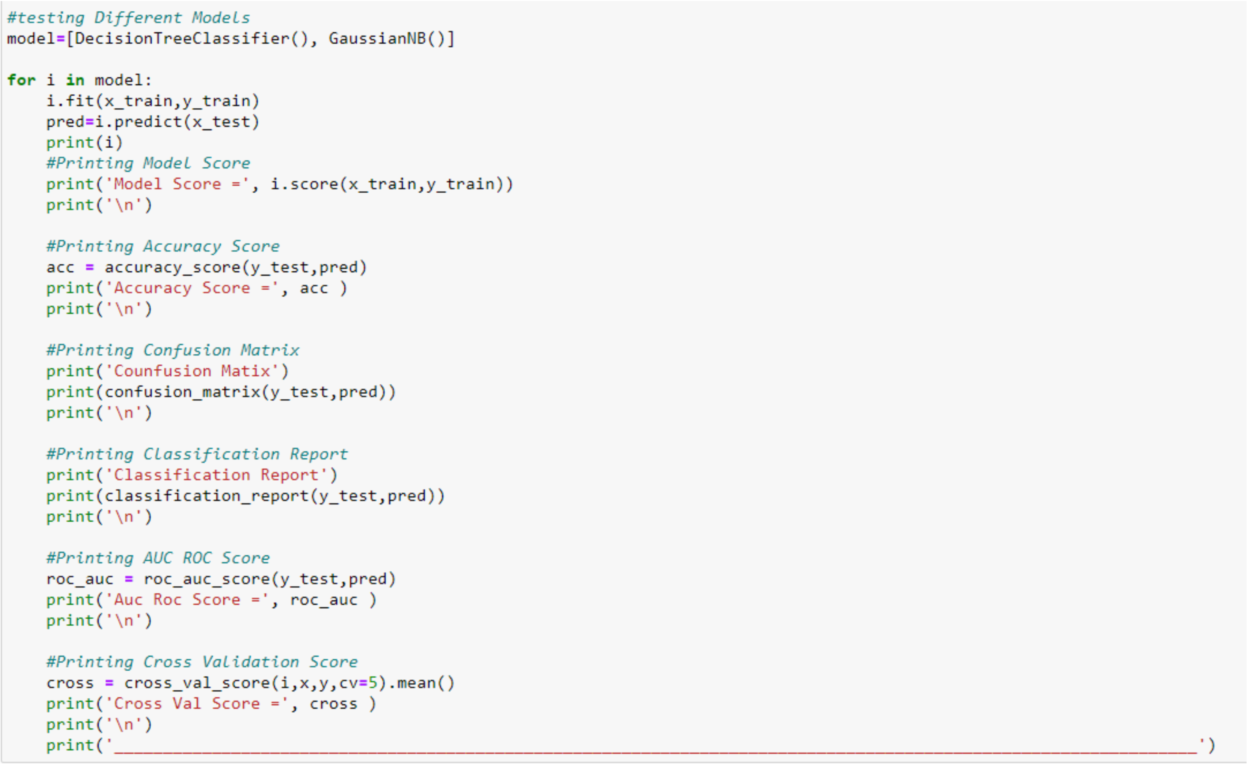
Run and Evaluate selected models

First we used two models



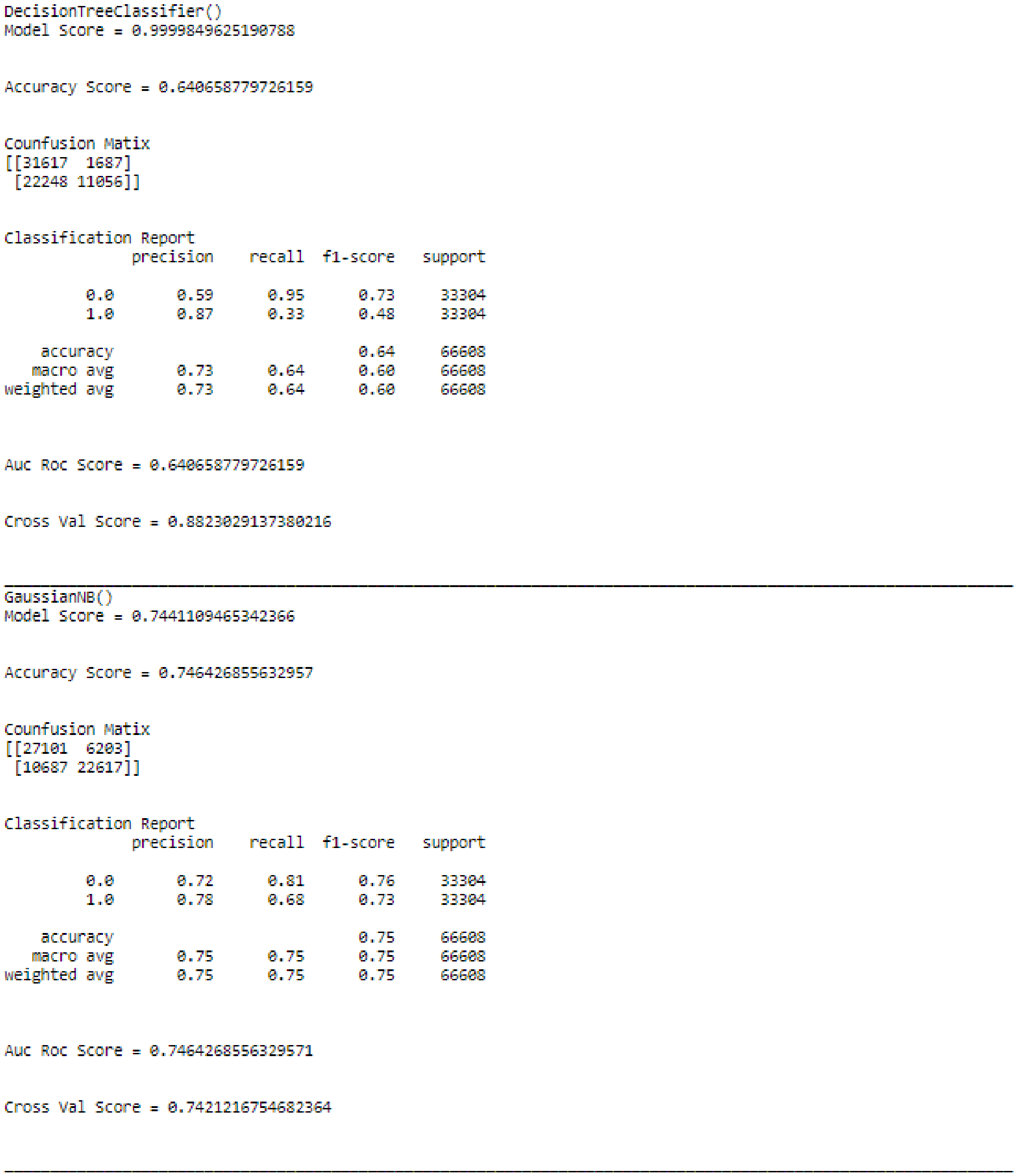
```
#ML Models
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import GaussianNB

#model selection
from sklearn.model_selection import train_test_split, GridSearchCV, cross_val_score

#metrics
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report,roc_curve,roc_auc_score
```

Then we made a for loop to test and print the result of both the two models



```
#testing Different Models
model=[DecisionTreeClassifier(), GaussianNB()]

for i in model:
    i.fit(x_train,y_train)
    pred=i.predict(x_test)
    print(i)
    #Printing Model Score
    print('Model Score =', i.score(x_train,y_train))
    print('\n')

    #Printing Accuracy Score
    acc = accuracy_score(y_test,pred)
    print('Accuracy Score =', acc )
    print('\n')

    #Printing Confusion Matrix
    print('Counfusion Matix')
    print(confusion_matrix(y_test,pred))
    print('\n')

    #Printing Classification Report
    print('Classification Report')
    print(classification_report(y_test,pred))
    print('\n')

    #Printing AUC ROC Score
    roc_auc = roc_auc_score(y_test,pred)
    print('Auc Roc Score =', roc_auc )
    print('\n')

    #Printing Cross Validation Score
    cross = cross_val_score(i,x,y,cv=5).mean()
    print('Cross Val Score =', cross )
    print('\n')
    print('________________________________________________________________________________________________________________')
```

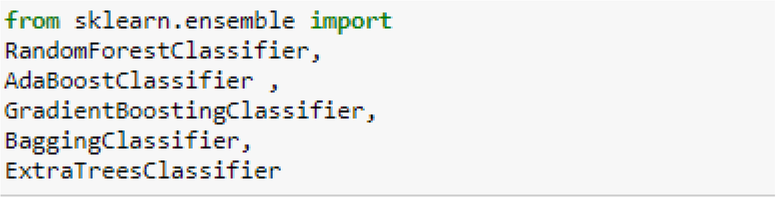
In result we got all the values of matrices then with our choice of matrix, we selected the model



```
DecisionTreeClassifier()
Model Score = 0.9999849625190788


Accuracy Score = 0.640658779726159


Counfusion Matix
[[31617  1687]
 [22248 11056]]


Classification Report
              precision    recall  f1-score   support

         0.0       0.59      0.95      0.73     33304
         1.0       0.87      0.33      0.48     33304

    accuracy                           0.64     66608
   macro avg       0.73      0.64      0.60     66608
weighted avg       0.73      0.64      0.60     66608



Auc Roc Score = 0.640658779726159


Cross Val Score = 0.8823029137380216


________________________________________________________________________________________________________________
GaussianNB()
Model Score = 0.7441109465342366


Accuracy Score = 0.746426855632957


Counfusion Matix
[[27101  6203]
 [10687 22617]]


Classification Report
              precision    recall  f1-score   support

         0.0       0.72      0.81      0.76     33304
         1.0       0.78      0.68      0.73     33304

    accuracy                           0.75     66608
   macro avg       0.75      0.75      0.75     66608
weighted avg       0.75      0.75      0.75     66608



Auc Roc Score = 0.7464268556329571


Cross Val Score = 0.7421216754682364


________________________________________________________________________________________________________________
```

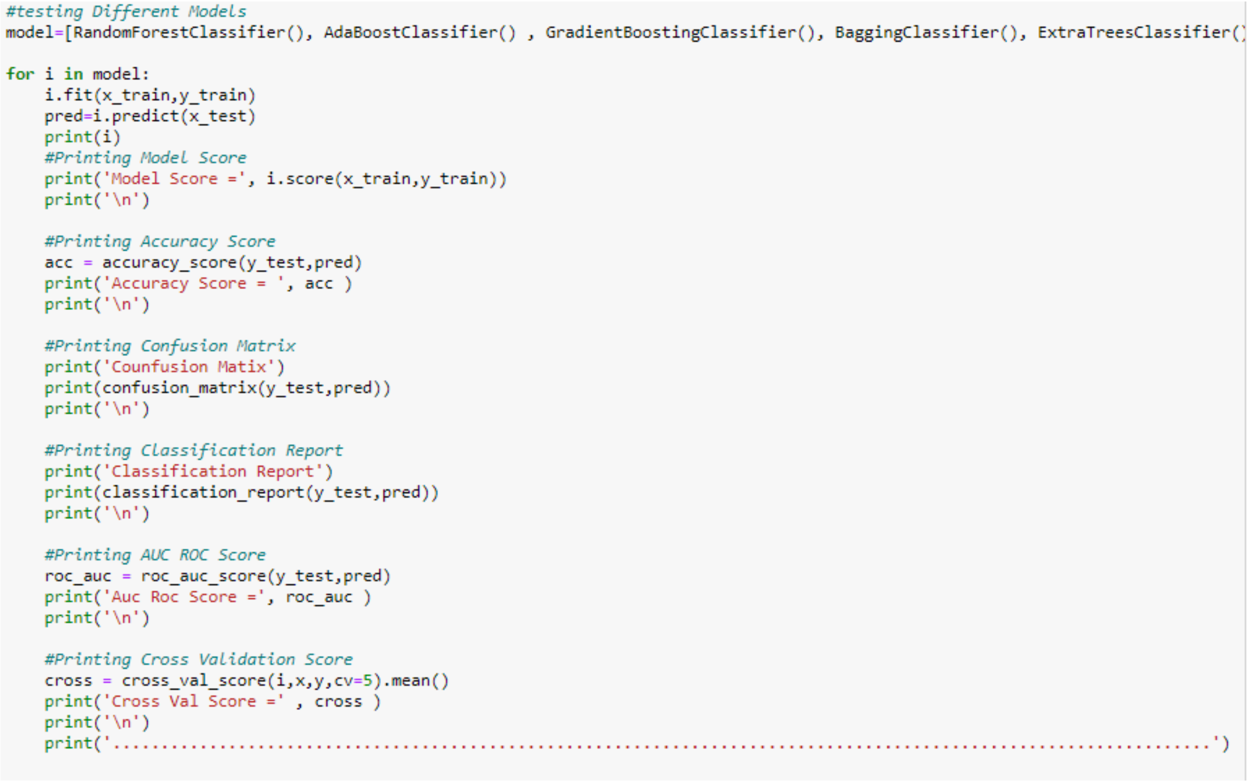
we see that GaussianNB has accuracy score = 74%, f1 score = 75% , auc roc score = 74% and average cross validation score = 74 but this can further be increased to we will first see all ensemble technique

Then we used different ensemble bagging and boosting method



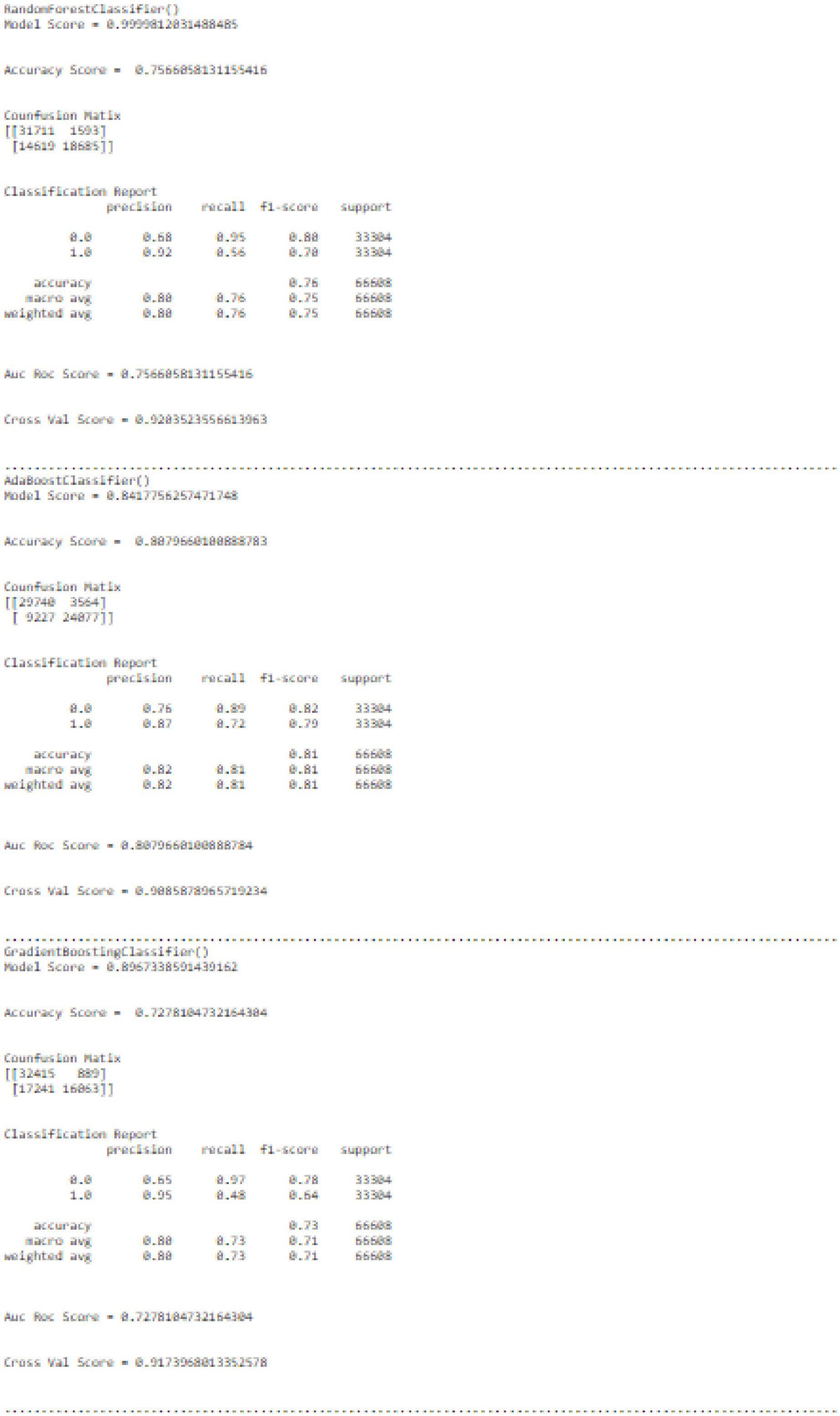
```
from sklearn.ensemble import
RandomForestClassifier,
AdaBoostClassifier ,
GradientBoostingClassifier,
BaggingClassifier,
ExtraTreesClassifier
```

Again we used for loop to test different models and get the result

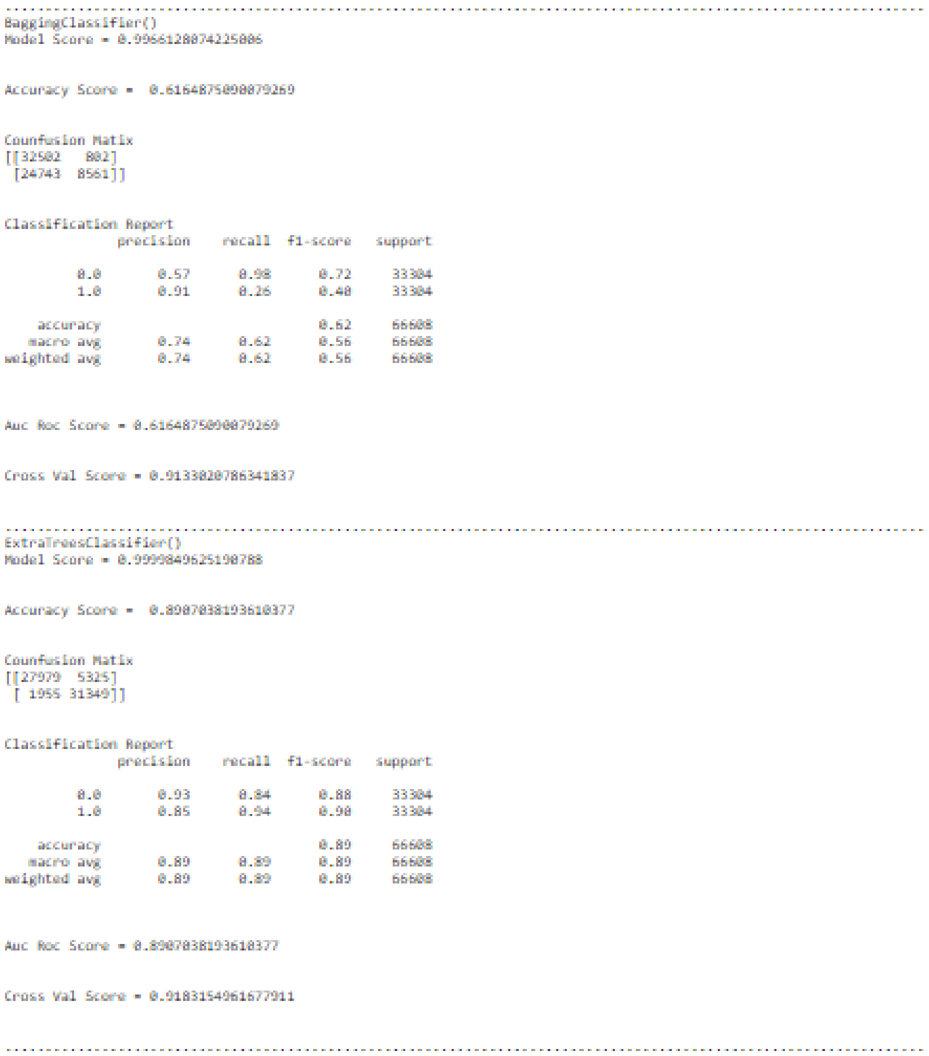


```
#testing Different Models
model=[RandomForestClassifier(), AdaBoostClassifier() , GradientBoostingClassifier(), BaggingClassifier(), ExtraTreesClassifier(

for i in model:
    i.fit(x_train,y_train)
    pred=i.predict(x_test)
    print(i)
    #Printing Model Score
    print('Model Score =', i.score(x_train,y_train))
    print('\n')

    #Printing Accuracy Score
    acc = accuracy_score(y_test,pred)
    print('Accuracy Score = ', acc )
    print('\n')

    #Printing Confusion Matrix
    print('Counfusion Matix')
    print(confusion_matrix(y_test,pred))
    print('\n')

    #Printing Classification Report
    print('Classification Report')
    print(classification_report(y_test,pred))
    print('\n')

    #Printing AUC ROC Score
    roc_auc = roc_auc_score(y_test,pred)
    print('Auc Roc Score =', roc_auc )
    print('\n')

    #Printing Cross Validation Score
    cross = cross_val_score(i,x,y,cv=5).mean()
    print('Cross Val Score =' , cross )
    print('\n')
    print('..................................................................................................................')
```

We got list of result with all the ensemble technique



```
RandomForestClassifier()
Model Score = 0.9999812031488485


Accuracy Score =  0.7566058131155416


Counfusion Matix
[[31711  1593]
 [14619 18685]]


Classification Report
              precision    recall  f1-score   support

         0.0       0.68      0.95      0.80     33304
         1.0       0.92      0.56      0.70     33304

    accuracy                           0.76     66608
   macro avg       0.80      0.76      0.75     66608
weighted avg       0.80      0.76      0.75     66608



Auc Roc Score = 0.7566058131155416


Cross Val Score = 0.9283523556613963


------------------------------------------------------------------------------------------------------------------
AdaBoostClassifier()
Model Score = 0.8417756257471748


Accuracy Score =  0.8079660100888783


Counfusion Matix
[[29740  3564]
 [ 9227 24077]]


Classification Report
              precision    recall  f1-score   support

         0.0       0.76      0.89      0.82     33304
         1.0       0.87      0.72      0.79     33304

    accuracy                           0.81     66608
   macro avg       0.82      0.81      0.81     66608
weighted avg       0.82      0.81      0.81     66608



Auc Roc Score = 0.8079660100888784


Cross Val Score = 0.9085878965719234


------------------------------------------------------------------------------------------------------------------
GradientBoostingClassifier()
Model Score = 0.8967338591439162


Accuracy Score =  0.7278104732164304


Counfusion Matix
[[32415   889]
 [17241 16063]]


Classification Report
              precision    recall  f1-score   support

         0.0       0.65      0.97      0.78     33304
         1.0       0.95      0.48      0.64     33304

    accuracy                           0.73     66608
   macro avg       0.80      0.73      0.71     66608
weighted avg       0.80      0.73      0.71     66608



Auc Roc Score = 0.7278104732164304


Cross Val Score = 0.9173968013352578


------------------------------------------------------------------------------------------------------------------
```



```
--------------------------------------------------------------------------------------------------------------------------
BaggingClassifier()
Model Score = 0.9966128874225806


Accuracy Score =  0.6164875090079269


Counfusion Matix
[[32502   802]
 [24743  8561]]


Classification Report
              precision    recall  f1-score   support

         0.0       0.57      0.98      0.72     33304
         1.0       0.91      0.26      0.40     33304

    accuracy                           0.62     66608
   macro avg       0.74      0.62      0.56     66608
weighted avg       0.74      0.62      0.56     66608



Auc Roc Score = 0.6164875090079269


Cross Val Score = 0.9133020786341837


--------------------------------------------------------------------------------------------------------------------------
ExtraTreesClassifier()
Model Score = 0.9999849625190788


Accuracy Score =  0.8907038193610377


Counfusion Matix
[[27979  5325]
 [ 1955 31349]]


Classification Report
              precision    recall  f1-score   support

         0.0       0.93      0.84      0.88     33304
         1.0       0.85      0.94      0.90     33304

    accuracy                           0.89     66608
   macro avg       0.89      0.89      0.89     66608
weighted avg       0.89      0.89      0.89     66608



Auc Roc Score = 0.8907038193610377


Cross Val Score = 0.9183154961677911


--------------------------------------------------------------------------------------------------------------------------
```

we see that extra tree classifier has best accuracy score, f1 score, auc_roc score ann cross val score

so we will process with **Extra Tree Classifier**

Key Metrics for success in solving problem under consideration

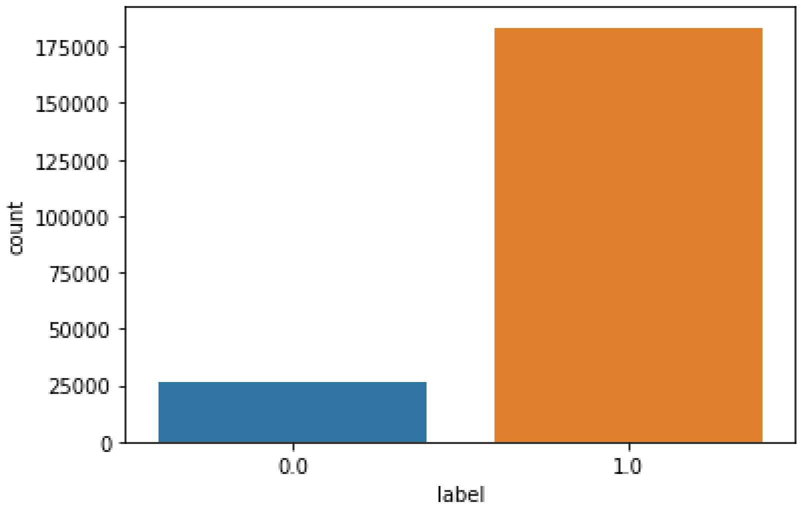
1. **Accuracy Score** - at first our dataset was imbalanced but later we balanced it so first we will look at accuracy score shows best result when it comes to balance the dataset.
2. **F1 - Score** - in this data set the target will decide who is defaulter or not, Hence 0 and 1, and as both zero and one is important to us

therefore recall and precision what will be our preferred metric and as we all know that , it combines precision and recall into one metric by calculating the harmonic mean between those two.and preferred metric is F1 score.

3. **AUC ROC** - We can see a healthy ROC curve, pushed towards the top-left side both for positive and negative classes. It is not clear which one performs better across the board as with FPR = 0.15 positive class is higher and starting from FPR= 0.15 the negative class is above.In order to get one number that tells us how good our curve is, we can calculate the Area Under the ROC Curve, or ROC AUC score. The more top-left your curve is the higher the area and hence higher ROC AUC score.
4. **Cross Validation Score** - to check if our model is overfitting or not we use cross validation score, higher the cross validation score higher the cross validation score means the model is not overfitting.

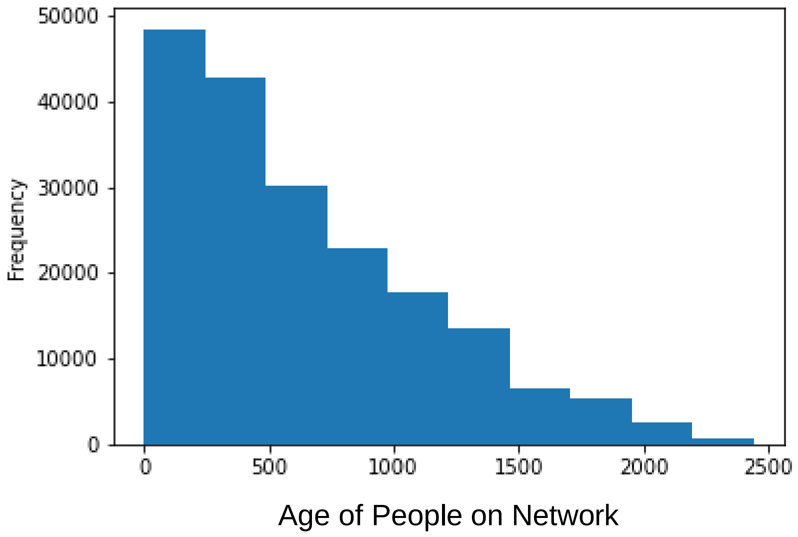
Visualizations

Label -



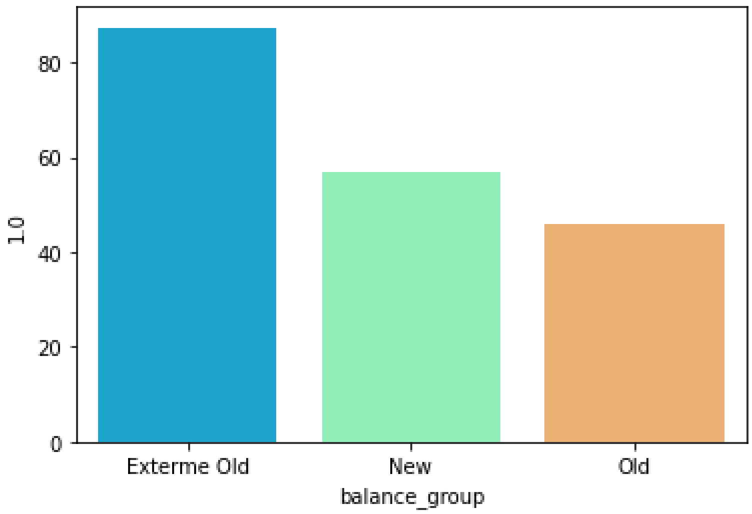
We see that the data is highly imbalanced and approx 87%, so after seeing this graph we decided to balance the data by oversampling.

Effect of Age on Network on Loan Payment -



The almost uniform distribution shows us that there are more new people to the network and less old people we can draw two consultation

1. More people are joining the network
2. With the time people are leaving the network

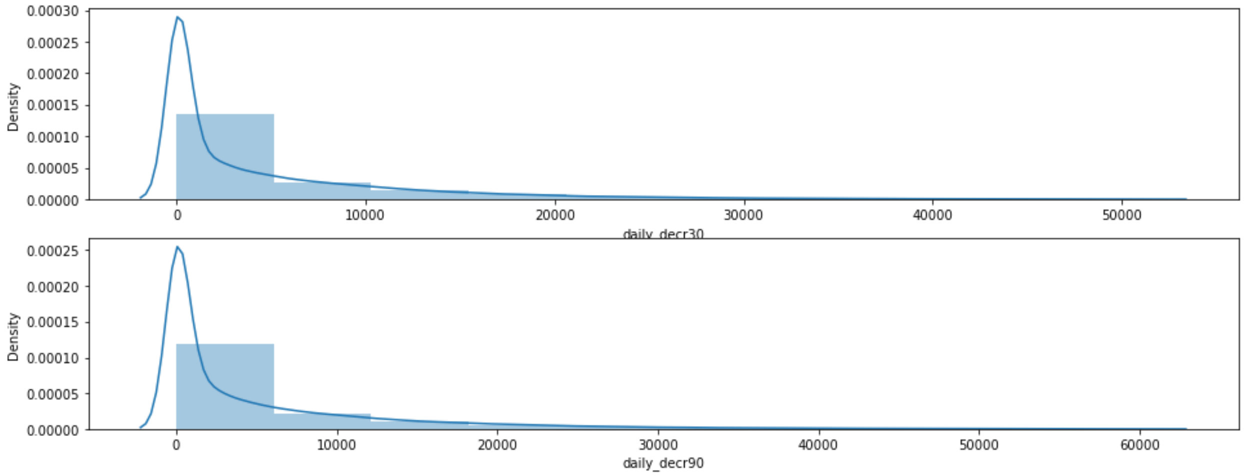


Age of people vs Loan Repayment

87% extreme old and 56% new people are most likely to return loan as they develop loyalty towards telecom

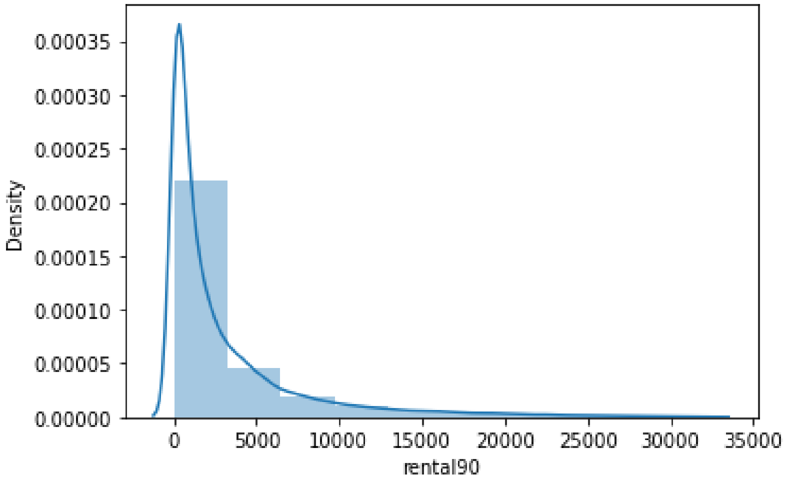
but as person become little old then they are more likely to be faulty only 45% returned the loan, as they are not sure whether they want to continue with the teleco or not and we have already seen that with time people are leaving the network

Daily Amount Spent



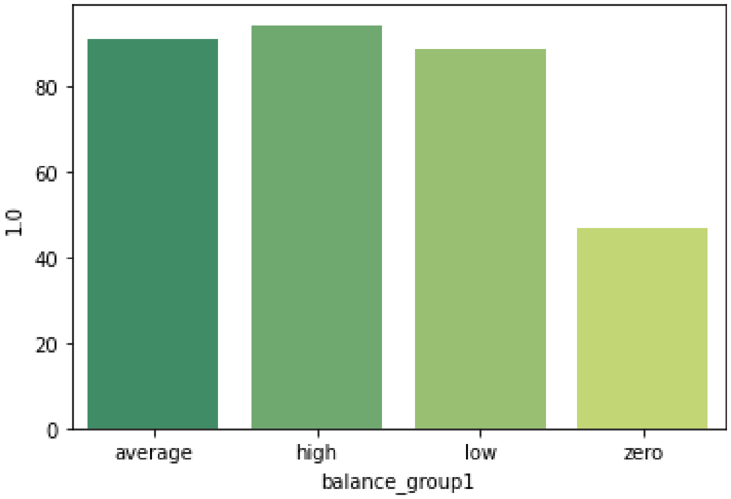
we see that most of the people spent amount between 0-5000 and when we go beyond that we see sudden drop in the customer who spent more than 5000 as their daily amount

Average Main Account Balance



Average account Balance on 90 days

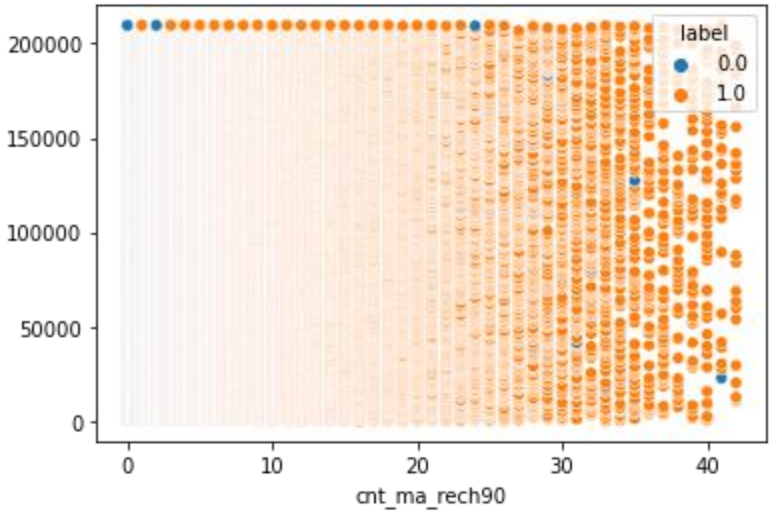
If we analyse the above graph and this graph side by side, we see people have the main balance just little more than their amount spent.



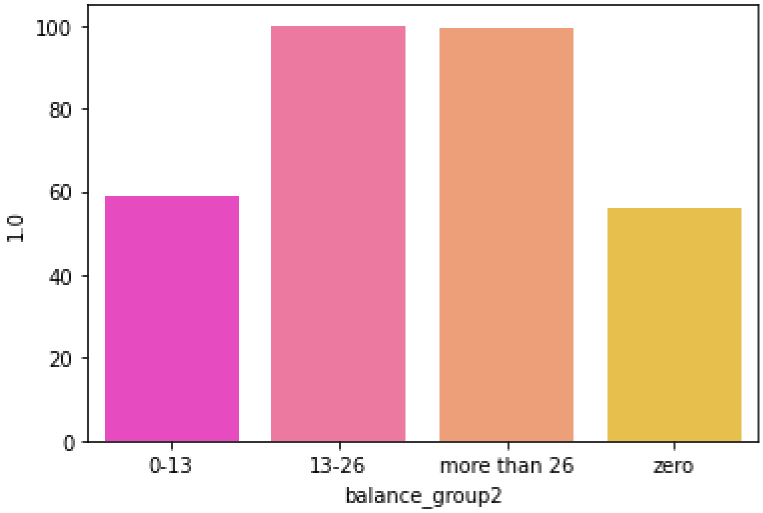
Average Balance vs Loan Repayment

People with High balance have 94% probability of returning the loan, Average people have 90% probability of returning the loan, this becomes more low when it reaches 88% for low balance and 46% for zero balance.
we see direct relationship, when people keep account balance high are more likely to return loan

Number of times main account got recharged in last 90 days



We see that the majority of people do recharge almost more than 20 times a month but they are also in the non defaulter category.

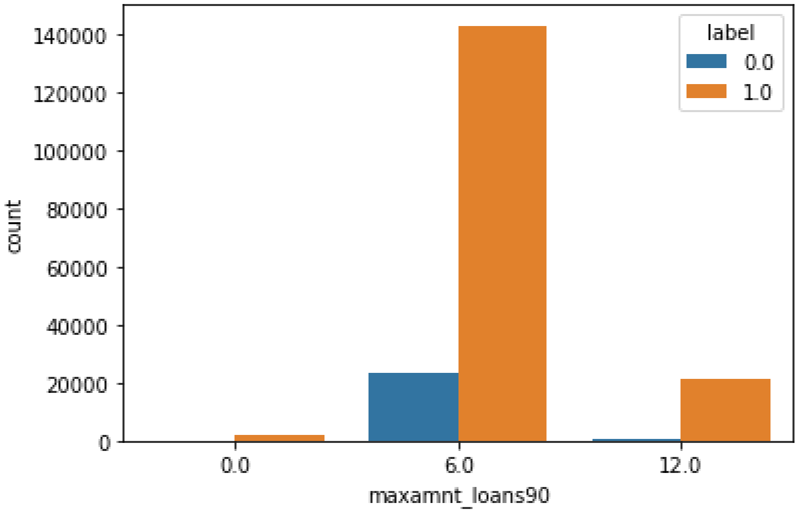


Number of times account rechange vs Loan Repayment

People who are doing recharge more than 26 times have 99% probability of returning the loan, People who are doing recharge between 13-26 times have 100% probability of returning the loan, this becomes more low when it reaches 58% for recharges between 0-13 and 56% for times.

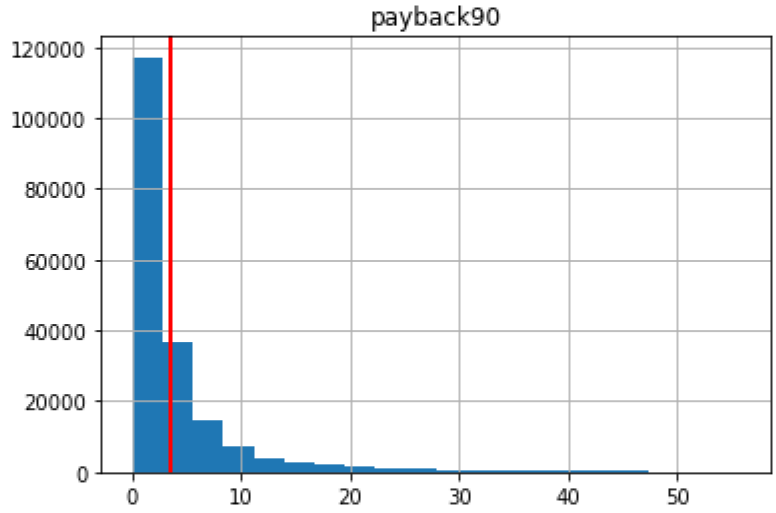
People who are doing recharge more than 13 times in 90 days are surely going to return the loan amount.

Maximum amount of loan taken by the user in last 90 days



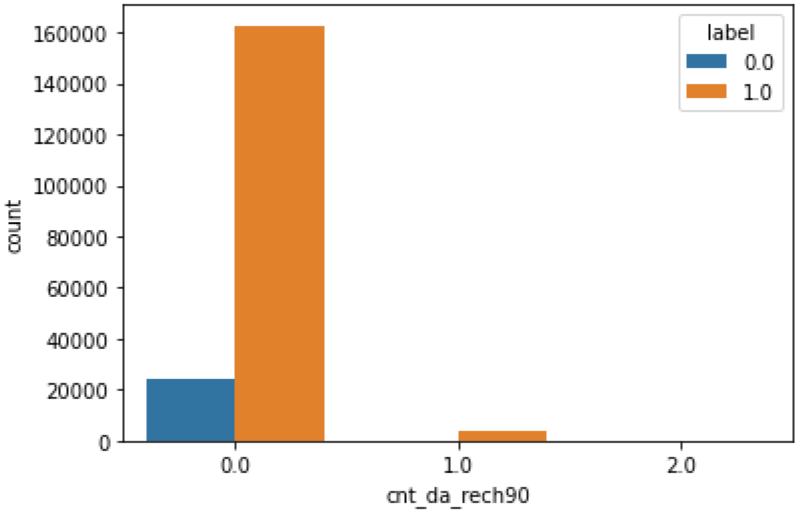
Most of the people took the loan 6 times in the last 30 days, but they also returned the loan, alos most of the people who didn't return the loan falls in this category. so, it is advised that max loan amount reaches 6, red flags should be raised.

Payback time in 90 days



Most of the people paid back around in 3 days, so people who didn't return the money till 10 days are most likely to be defaulters.

Data account got recharged in last 90 days



people who didn't do data recharge are most likely to take a loan and the factor of taking loan and not taking loan is another thing, here we see if a person is doing data recharge are more likely that they will not take any loan.

Interpretation of the Results.

When we see the result of matrices we see that the we see that extra tree classifier has best accuracy score, f1 score, auc_roc score ann cross val score

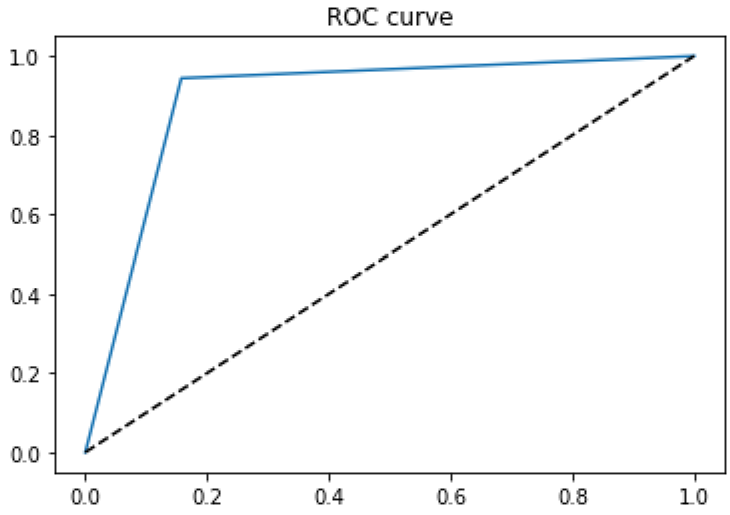
Accuracy = 89%

F1-Score = 89%

Auc Roc Score = 89%

Cross Val Score = 91%

and AUC_ROC_Curve touching corner



CONCLUSION

Key Findings and Conclusions of the Study

1. With the introduction of micro loans many people join the network but because of various factors they start leaving the network.
2. We see that people you have low balance or jio recharge and have probability of not giving a loan more than 50% still got the loan
3. People with high probability returning the loan should have provided a loan instantly and bigger amount is possible
4. All the people who are using the internet can be considered as literate and literacy rate has a huge effect on returning the loan.

Learning Outcomes of the Study in respect of Data Science

1. We learnt how to deal with negative values and extreme outliers.
2. To do a visualisation when data has high standard deviation and no classification
3. Ways to select features and to do hyperparameter tuning efficiently
4. Ways of removing skewness and what are the best methods still not versatile when it comes to data with 0 value

Limitations of this work and Scope for Future Work

The data is collected not efficiently and because of that we are getting values which are impossible for the features the data has extreme outliers that cannot be removed even when we put I should equal to 5 in Z score which decrease the efficiency of model the data is highly skewed and even using the best method the effectiveness of some features are so high that they are hindering with the efficiency of a model

The SVC model and Ken neighbours model is taking too much time and when we do hyperparameter tuning of and simple method the grid search CV is also taking too much time

In future we can increase the efficiency of a model by selecting a better method to remove outliers and skewness also how to make the search of perfect model in a way that if we want to change some parameters in model then we don't have to run all the model again

The ways to collect data should have be more efficient so that the anomaly of a data could be less