



## MICRO-CREDIT DEFAULTER MODEL

Submitted by:  
ABDUL WASHIF AHMED

## **ACKNOWLEDGMENT**

This includes mentioning of all the references, research papers, data sources, professionals and other resources that helped you and guided you in completion of the project.

# INTRODUCTION

A Microfinance Institution (MFI) is an organization that offers financial services to low income populations. Today, microfinance is widely accepted as a poverty-reduction tool, representing \$70 billion in outstanding loans and a global outreach of 200 million clients. 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.

## Conceptual Background of the Domain Problem

Many donors, experts, and microfinance institutions (MFI) have become convinced that using mobile financial services (MFS) is more convenient and efficient, and less costly, than the traditional high-touch model for delivering microfinance services. MFS becomes especially useful when targeting the unbanked poor living in remote areas. The implementation of MFS, though, has been uneven with both significant challenges and successes.

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

One of our Client in Telecom collaborates with an MFI to provide micro-credit on mobile balances to be paid back in 5 days. The Consumer is believed to be delinquent if he deviates from the path of paying back the loaned amount within 5 days.

## Review of Literature

A machine learning model is created which can predict in terms of a probability for each loan transaction, whether the customer will be paying back the loaned amount within 5 days of insurance of loan and after the research it is shown that out of different models used, RANDOM FOREST will have minimal errors with maximum accuracy.

## Motivation for the Problem Undertaken

To Build a model which can be used to predict in terms of a probability for each loan transaction, whether the customer will be paying back the loaned amount within 5 days of insurance of loan..

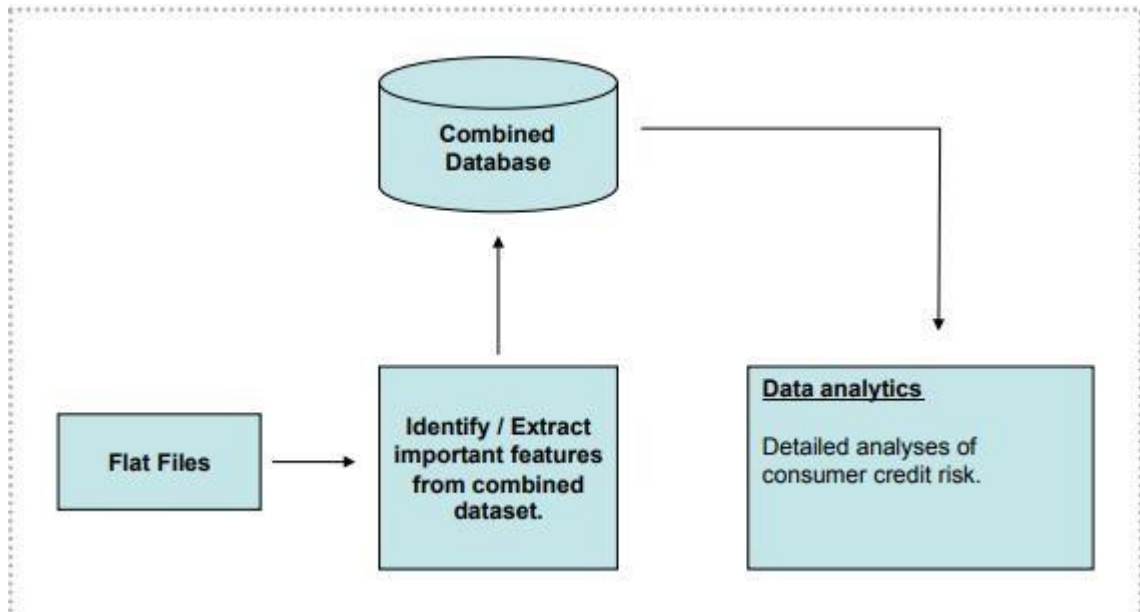
## Analytical Problem Framing

### Mathematical/ Analytical Modelling of the Problem

Different type of classification modelling is performed to find the best predictive model for the micro-credit defaulter data like Random forest ,Logistic Regression ,GAUSSIAN NAIVEBAYES ,Decision Tree, Support Vector Machine

### Data Sources and their formats

The raw data was provided to us in flat-file format. A set of programs were then used to pre-process the data, producing higher-level variables or “feature vectors” and time-aggregated to yield monthly observations that were stored in a database see Figure below. The following is a description of each of these components.



## Data Preprocessing

- STEPS-

1. Dropped unnecessary columns as "pcircle,msisdn,pdate".
2. As the data was Imbalanced so scaling over data was done.
3. Imported normalizer and normalise the data.
4. Imported stats from "scipy" and refined the outliers have z score more than 3.

## Data Inputs- Logic- Output Relationships

It is shown that Via box plots, the data was completely co related with the "label" (target column), as it is completely showing how much defaulters carry average balance and data in last 30 - 90 days, another way to find relationship is the syntax "df.corr()" will show the relationships between all.

## Hardware and Software Requirements and Tools Used

Listing down the hardware and software requirements along with the tools, libraries and packages used. Describe all the software tools used along with a detailed description of tasks done with those tools.

Lists of packages and libraries used in modelling --

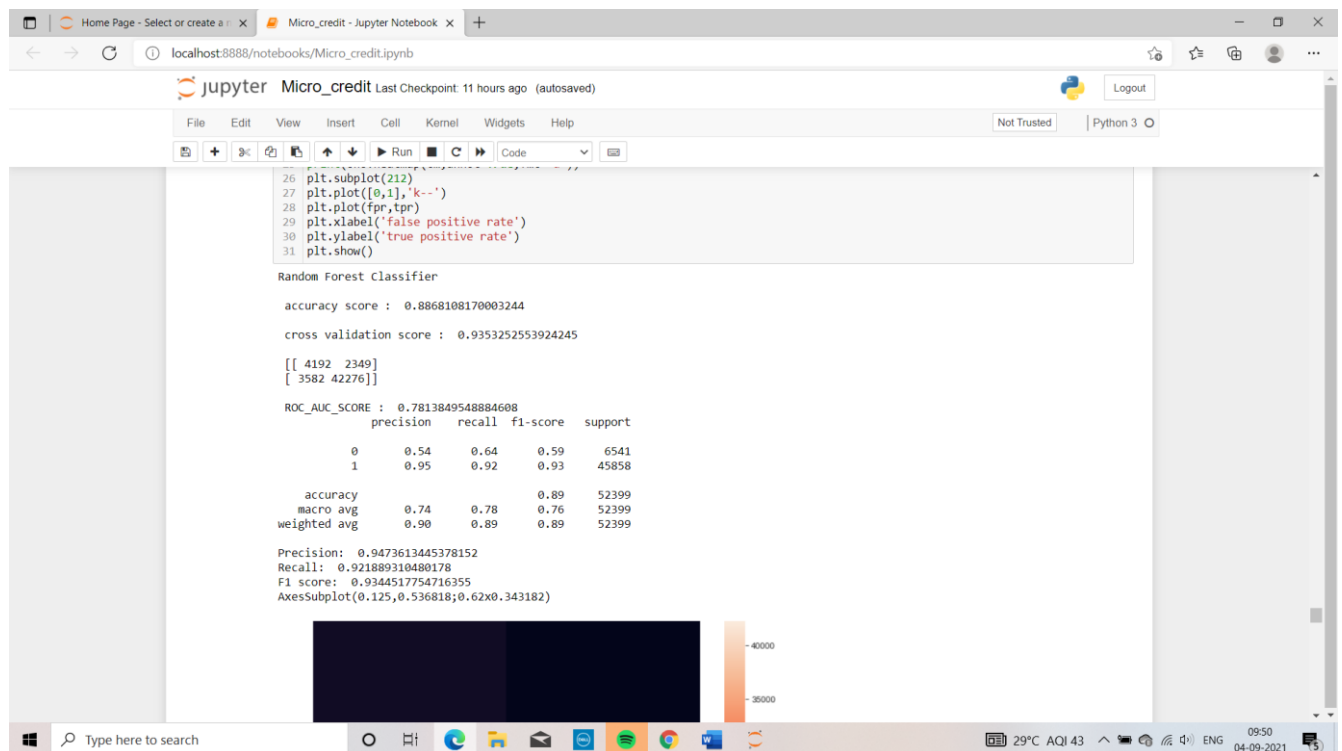
1. from sklearn.metrics import f1\_score, roc\_curve, classification\_report, confusion\_matrix
2. from sklearn.linear\_model import LogisticRegression
3. from sklearn import metrics
4. from sklearn.ensemble import RandomForestClassifier
5. from sklearn import model\_selection
6. import matplotlib.pyplot as plt
7. from sklearn.preprocessing import LabelEncoder
8. from sklearn.naive\_bayes import GaussianNB
9. from sklearn.tree import DecisionTreeClassifier
10. from sklearn.linear\_model import LogisticRegression
11. from sklearn.svm import SVC
12. from sklearn.metrics import accuracy\_score, recall\_score, precision\_score, precision\_recall\_curve
13. from sklearn.model\_selection import train\_test\_split
14. import pandas as pd
15. import numpy as np
16. import seaborn as sns

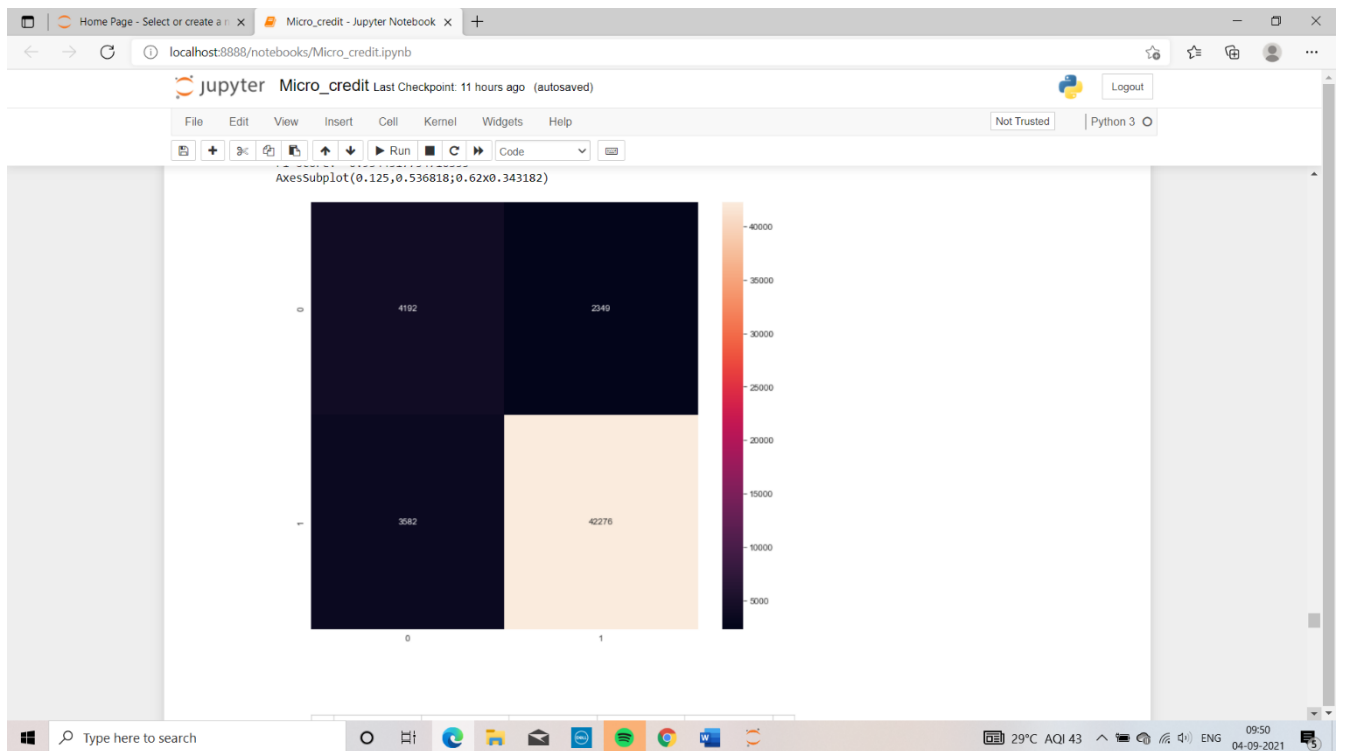
# Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods)

After train test splitting the data, the basic approach followed was checking the difference in confusion matrix and comparing f1score between models with comparing AUC-ROC curve

EXAMPLE-







# Testing of Identified Approaches (Algorithms)

Listing down all the algorithms used for the training and testing.

all the algorithms used for the training and testing-

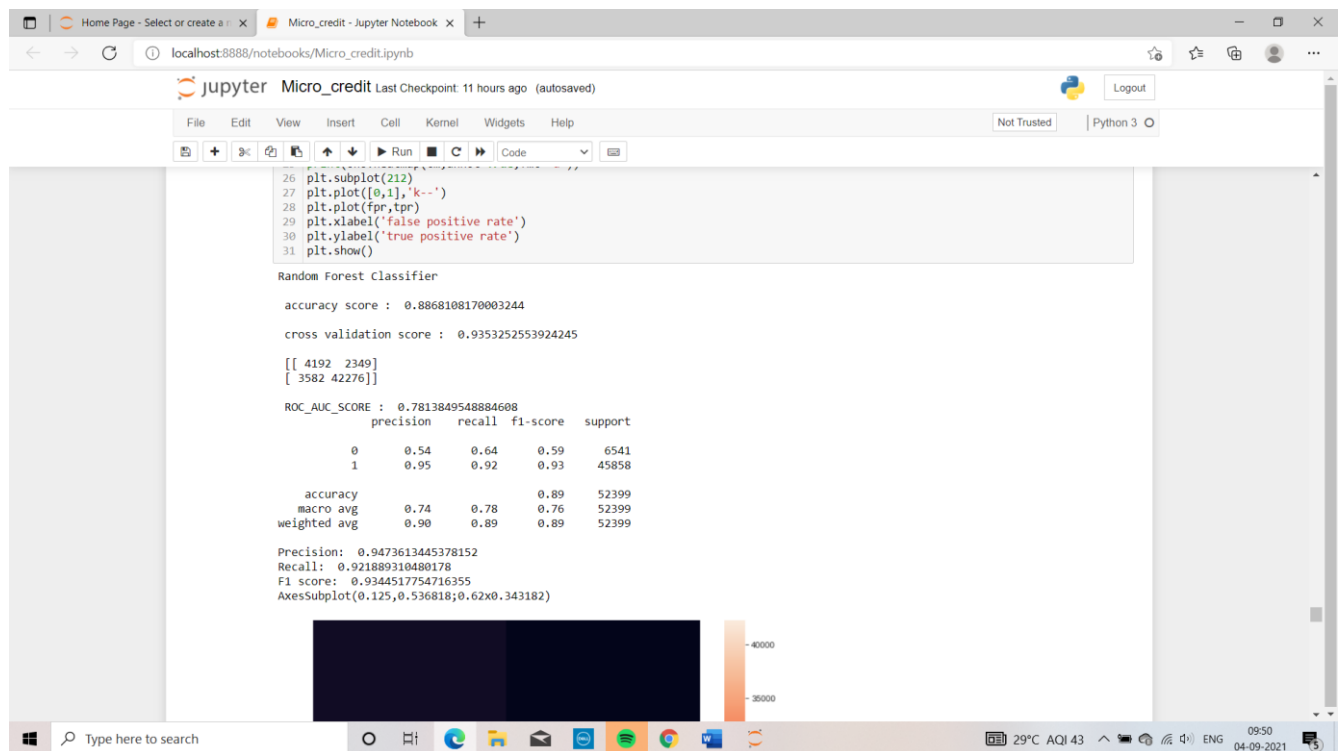
Random forest

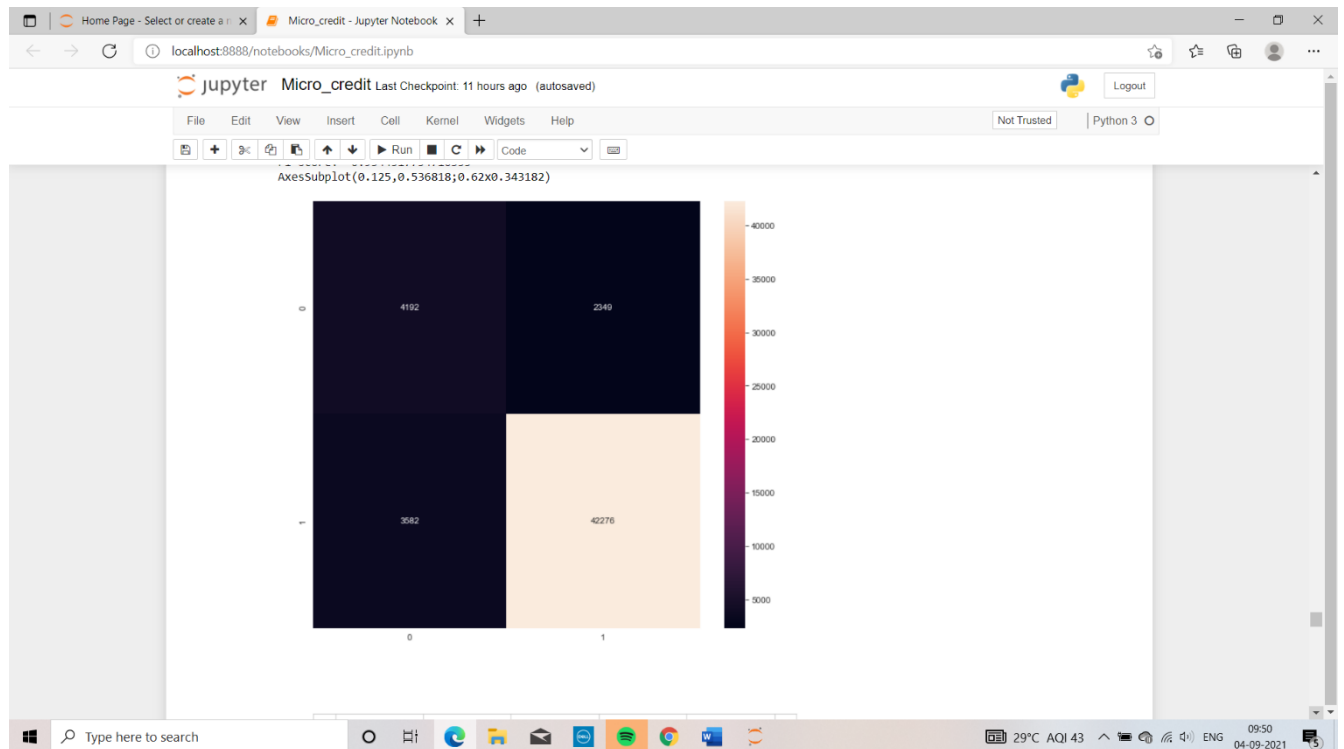
Logistic Regression

Decision Tree

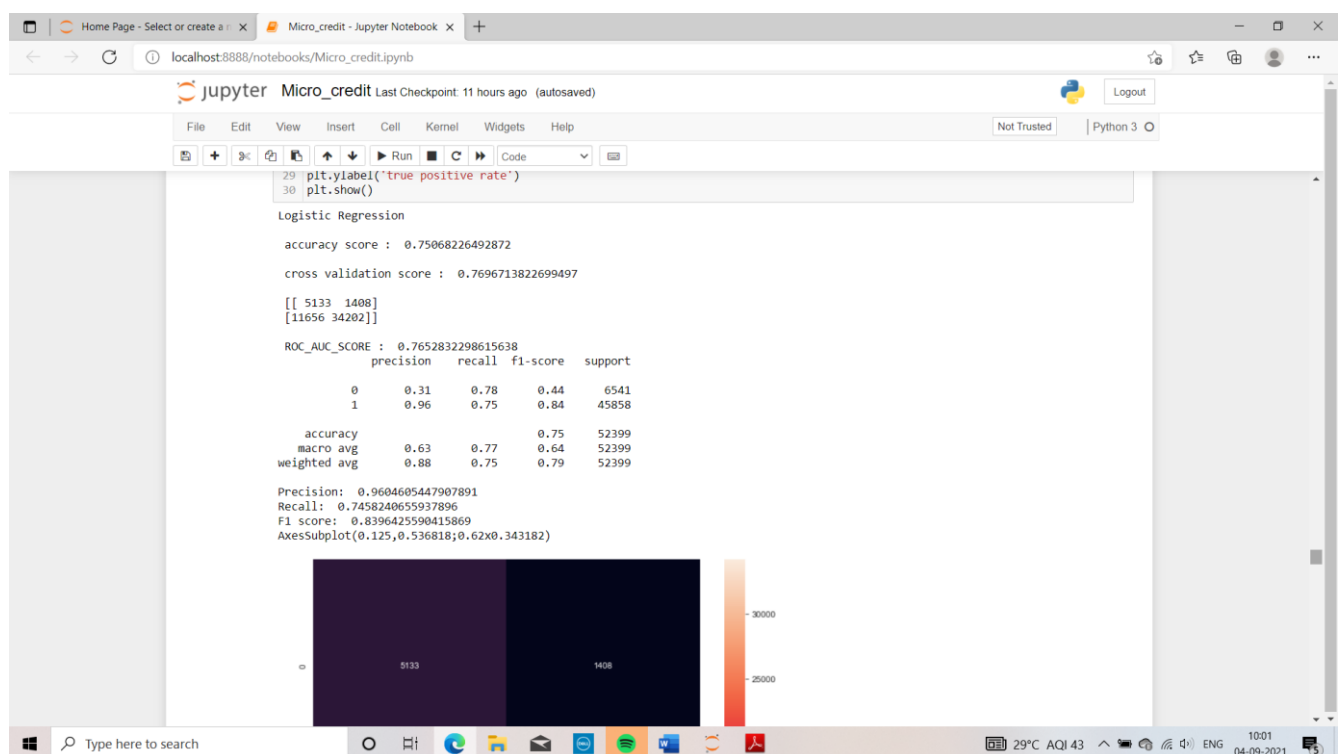
## Run and Evaluate selected models

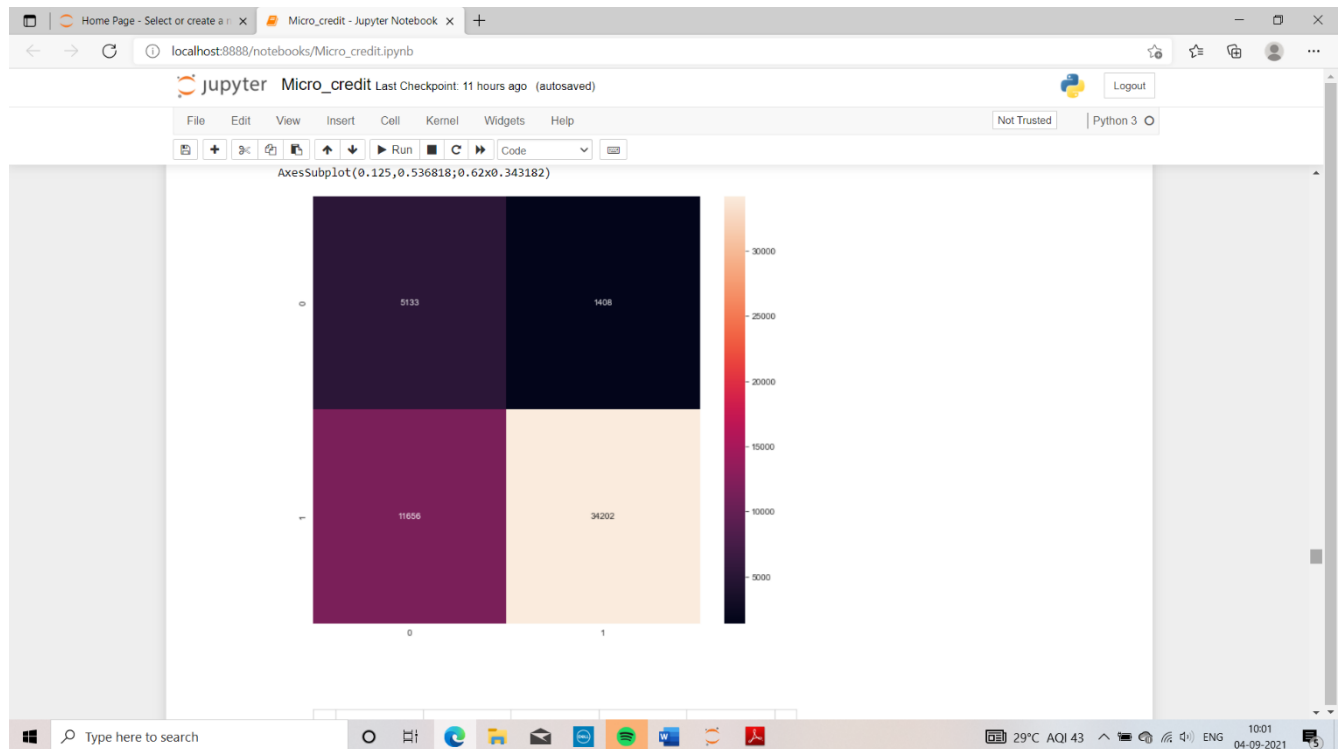
Random forest



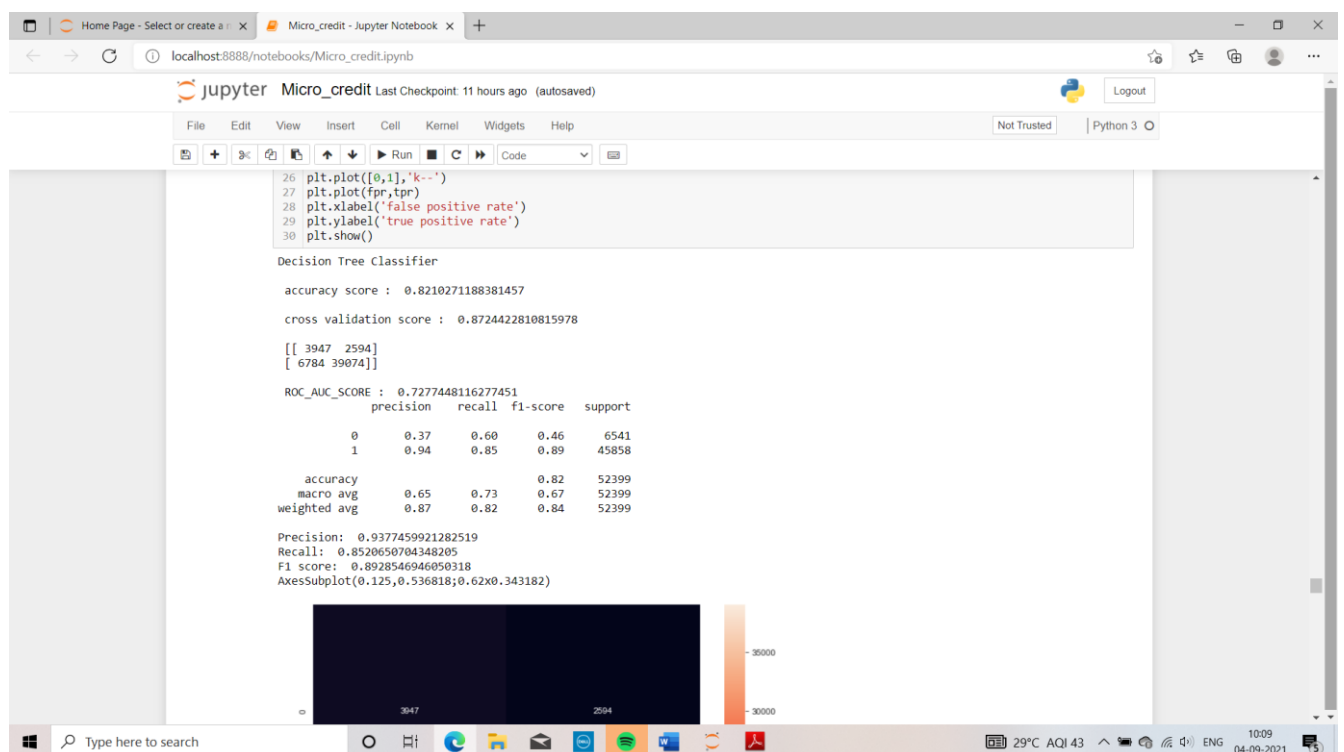


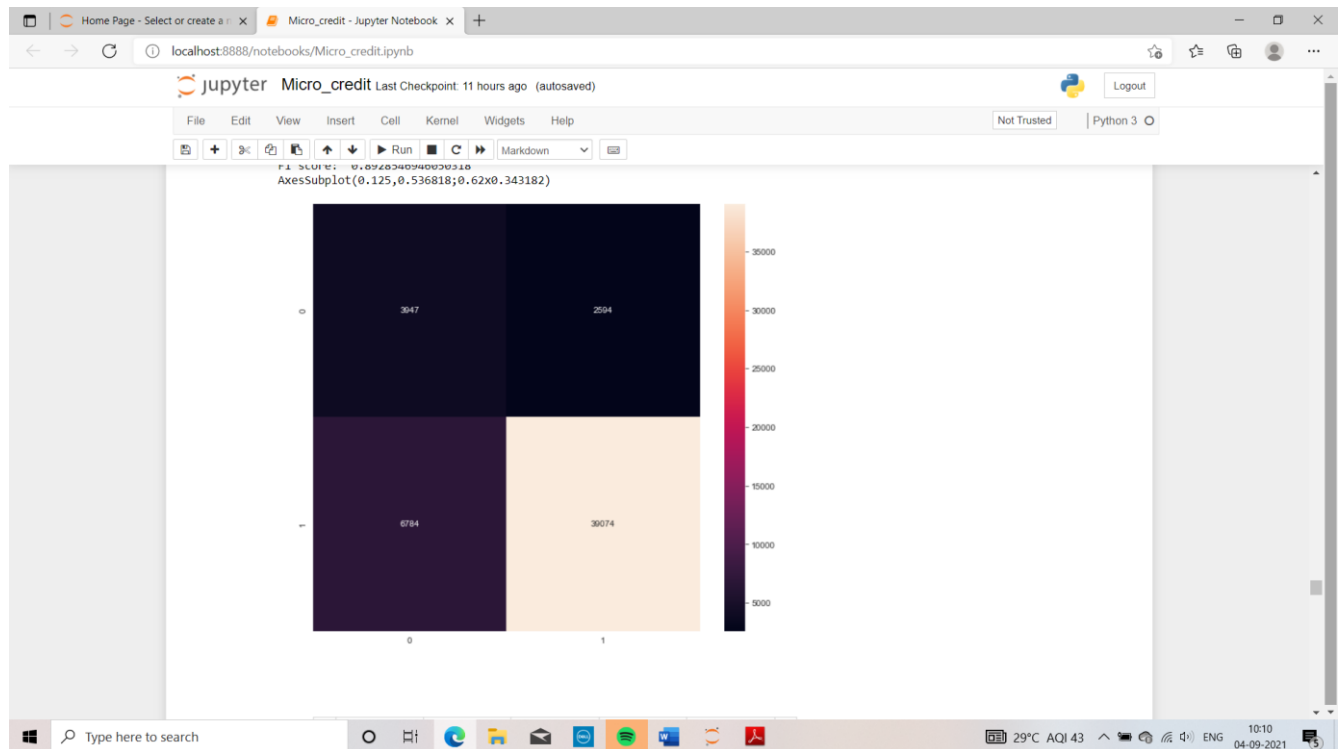
## Logistic Regression





## Decision Tree





(Out of all, Random forest model has an Accuracy of 0.88 and AUC\_ROC score of 0.78. we also have....

True Negative = 4192 False Positive = 3582

False Negative = 2349 True Positive = 42276 Also, we had low Type-1 Error as FP is less.)

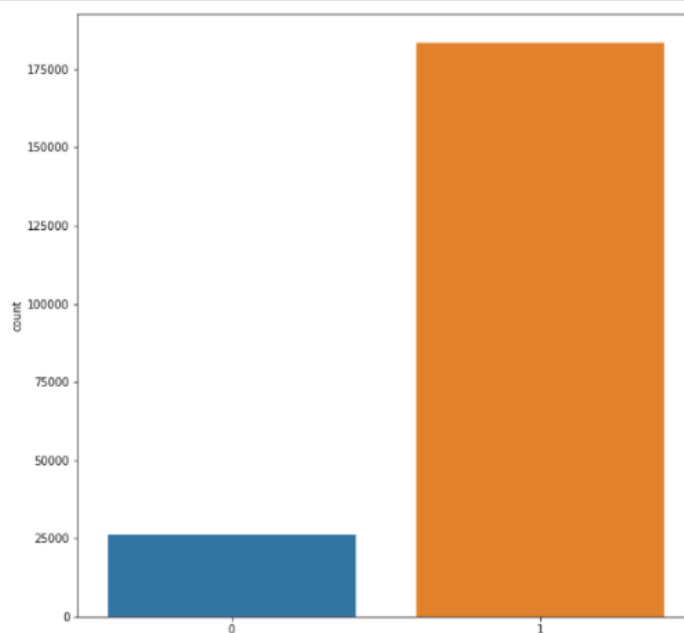
## Key Metrics for success in solving problem under consideration

Accuracy score, recall score, precision score, confusion matrix and classification reports were generated compared for solving model.

- Visualizations

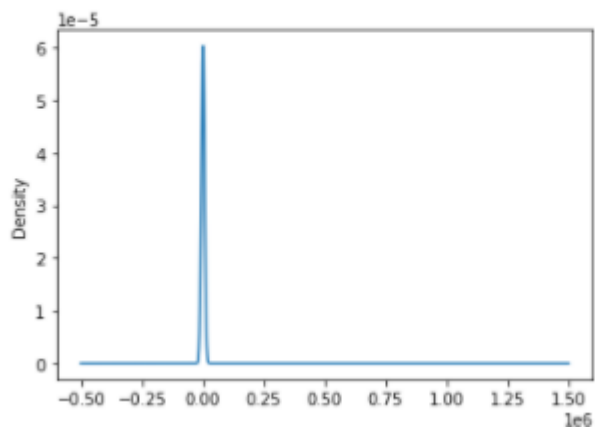
Target column (labelcount plot)

```
In [100]: plt.figure(figsize=(10,10))  
sns.countplot(x="label", data=df)  
plt.show()
```



Column “aon” plot to check the density of values which shows that it lies near 0 only

```
n [105]: df.aon.plot(kind='density')  
ut[105]: <AxesSubplot:ylabel='Density'>
```

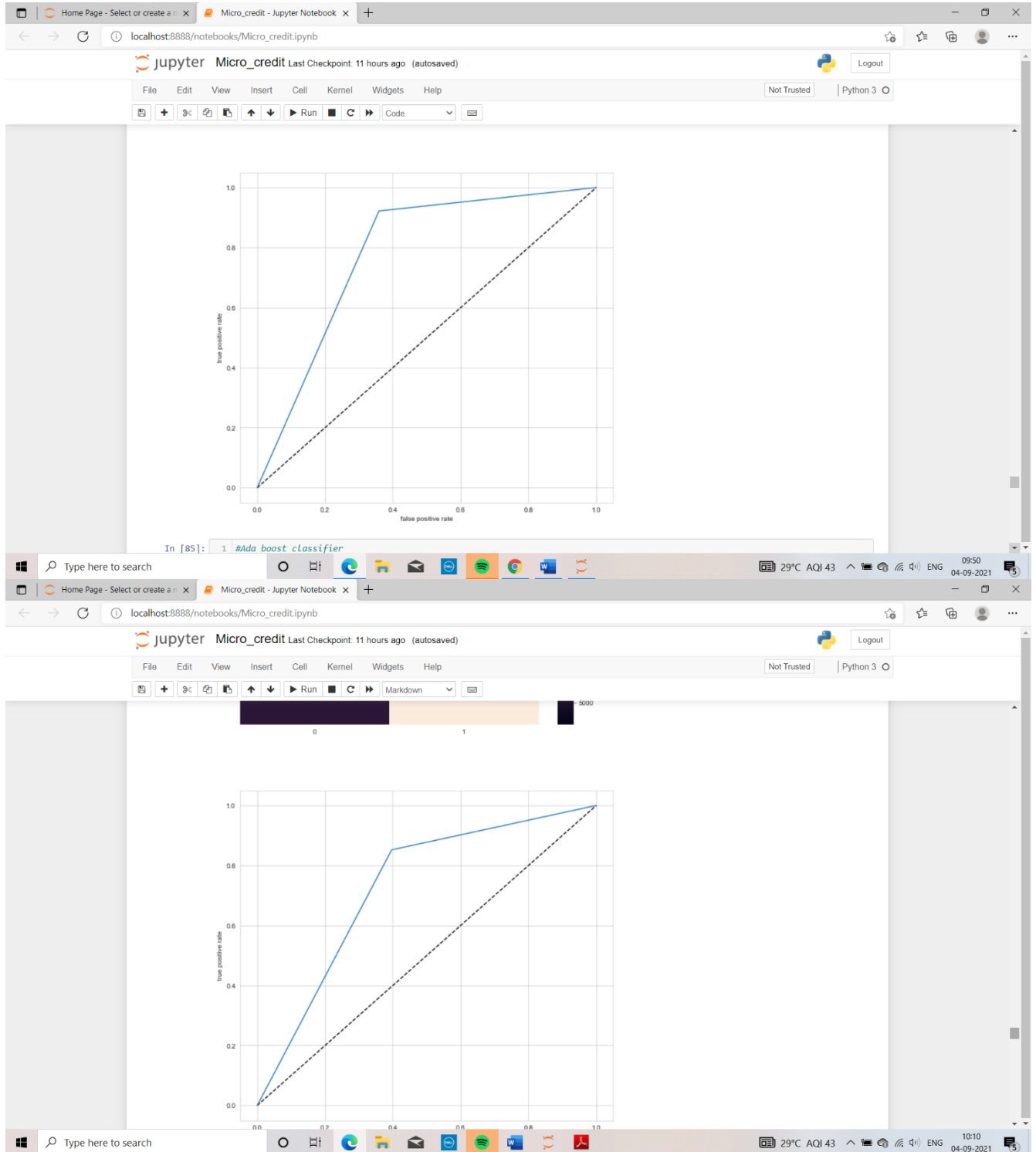


Interpretation of the Results

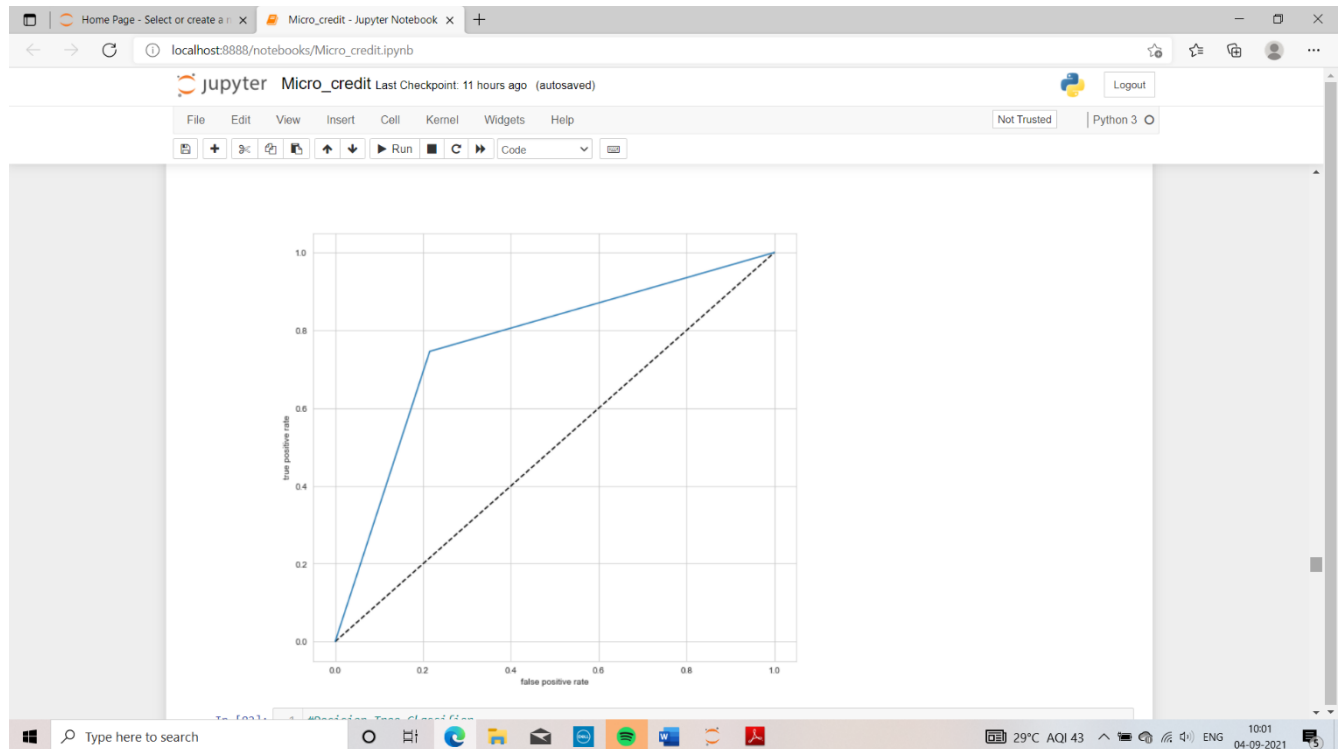
After all data cleaning and processing and scaling the data, I summarize that we can go with the most effective classification modelling technique for better result showing high accuracy and that is RANDOM FOREST.

# CONCLUSION

- Key Findings and Conclusions of the Study







Out of all of the roc curves above, we can conclude that random forest have better area under curve value

- Learning Outcomes of the Study in respect of Data Science

What I learnt during my whole project is that I should be aware of my data and how to handle my imbalanced data so that I can get perfect accuracy.

I should be aware of outliers and null values and how to over come these natural situations

I faced step wise problem, over how to write a particular code but after research I was able to get all confusion matrix along.

