

PROJECT REPORT ON

**“Micro-Credit Defaulter Model”**

**SUBMITTED BY**

**DISHANT DOSHI**

# ACKNOWLEDGMENT

Firstly, I would like to thank FlipRobo Technologies for giving me the opportunity to work on this project. Also, I would like to thank the DataTrained team, for providing me the knowledge and guidance which helped me a lot to work on this project.

References: <https://stackoverflow.com/> <https://seaborn.pydata.org/>

# INTRODUCTION

## Business Problem Framing

The main objective of this project is 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.

## Conceptual Background of the Domain Problem

A Microfinance Institution (MFI) is an organization that offers financial services to low income populations. MFS becomes very useful when targeting especially the unbanked poor families living in remote areas with not much sources of income. The Microfinance services (MFS) provided by MFI are Group Loans, Agricultural Loans, Individual Business Loans and so on.

Many microfinance institutions (MFI), experts and donors are supporting the idea of using mobile financial services (MFS) which they feel are more convenient and efficient, and cost saving, than the traditional high-touch model used since long for the purpose of delivering microfinance services. Though, the MFI industry is primarily focusing on low income families and are very useful in such areas, the implementation of MFS has been uneven with both significant challenges and successes.

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

FlipRobo is 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.

## Review of Literature

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).

## Motivation for the Problem Undertaken

* 1. The objective behind to take this project is to harness the required data science skills.
  2. Improve the analytical thinking.
  3. Get into the real world problem solving mechanics.

# Analytical Problem Framing

## Data Sources and their formats

The sample data is provided to us from FlipRobo client database. 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. The summary of the dataset are as follows:



**Features Information:**

1. label : Flag indicating whether the user paid back the credit amount within 5 days of issuing the loan{1:success, 0:failure}

2. msisdn : mobile number of user

3. aon : age on cellular network in days

4. daily\_decr30 : Daily amount spent from main account, averaged over last 30 days (in Indonesian Rupiah)

5. daily\_decr90 : Daily amount spent from main account, averaged over last 90 days (in Indonesian Rupiah)

6. rental30 : Average main account balance over last 30 days

7. rental90 : Average main account balance over last 90 days

8. last\_rech\_date\_ma : Number of days till last recharge of main account

9. last\_rech\_date\_da: Number of days till last recharge of data account

10. last\_rech\_amt\_ma : Amount of last recharge of main account (in Indonesian Rupiah)

11. cnt\_ma\_rech30 : Number of times main account got recharged in last 30 days

12. fr\_ma\_rech30 : Frequency of main account recharged in last 30 days

13. sumamnt\_ma\_rech30 : Total amount of recharge in main account over last 30 days (in Indonesian Rupiah)

14. medianamnt\_ma\_rech30 : Median of amount of recharges done in main account over last 30 days at user level (in Indonesian Rupiah)

15. medianmarechprebal30 : Median of main account balance just before recharge in last 30 days at user level (in Indonesian Rupiah)

16. cnt\_ma\_rech90 : Number of times main account got recharged in last 90 days

17. fr\_ma\_rech90 : Frequency of main account recharged in last 90 days

18. sumamnt\_ma\_rech90 : Total amount of recharge in main account over last 90 days (in Indonasian Rupiah)

19. medianamnt\_ma\_rech90 : Median of amount of recharges done in main account over last 90 days at user level (in Indonasian Rupiah)

20. medianmarechprebal90 : Median of main account balance just before recharge in last 90 days at user level (in Indonasian Rupiah)

21. cnt\_da\_rech30 : Number of times data account got recharged in last 30 days

22. fr\_da\_rech30: Frequency of data account recharged in last 30 days

23. cnt\_da\_rech90 : Number of times data account got recharged in last 90 days

24. fr\_da\_rech90 : Frequency of data account recharged in last 90 days

25. cnt\_loans30 : Number of loans taken by user in last 30 days

26. amnt\_loans30: Total amount of loans taken by user in last 30 days

27. maxamnt\_loans30 : maximum amount of loan taken by the user in last 30 days

28. medianamnt\_loans30 : Median of amounts of loan taken by the user in last 30 days

29. cnt\_loans90 : Number of loans taken by user in last 90 days

30. amnt\_loans90 : Total amount of loans taken by user in last 90 days

31. maxamnt\_loans90 : maximum amount of loan taken by the user in last 90 days

32. medianamnt\_loans90 : Median of amounts of loan taken by the user in last 90 da ys

33. payback30 : Average payback time in days over last 30 days

34. payback90 : Average payback time in days over last 90 days

35. pcircle : telecom circle

36. pdate : date

## Data Preprocessing Done

Below are the steps which we have taken in data pre - processing:

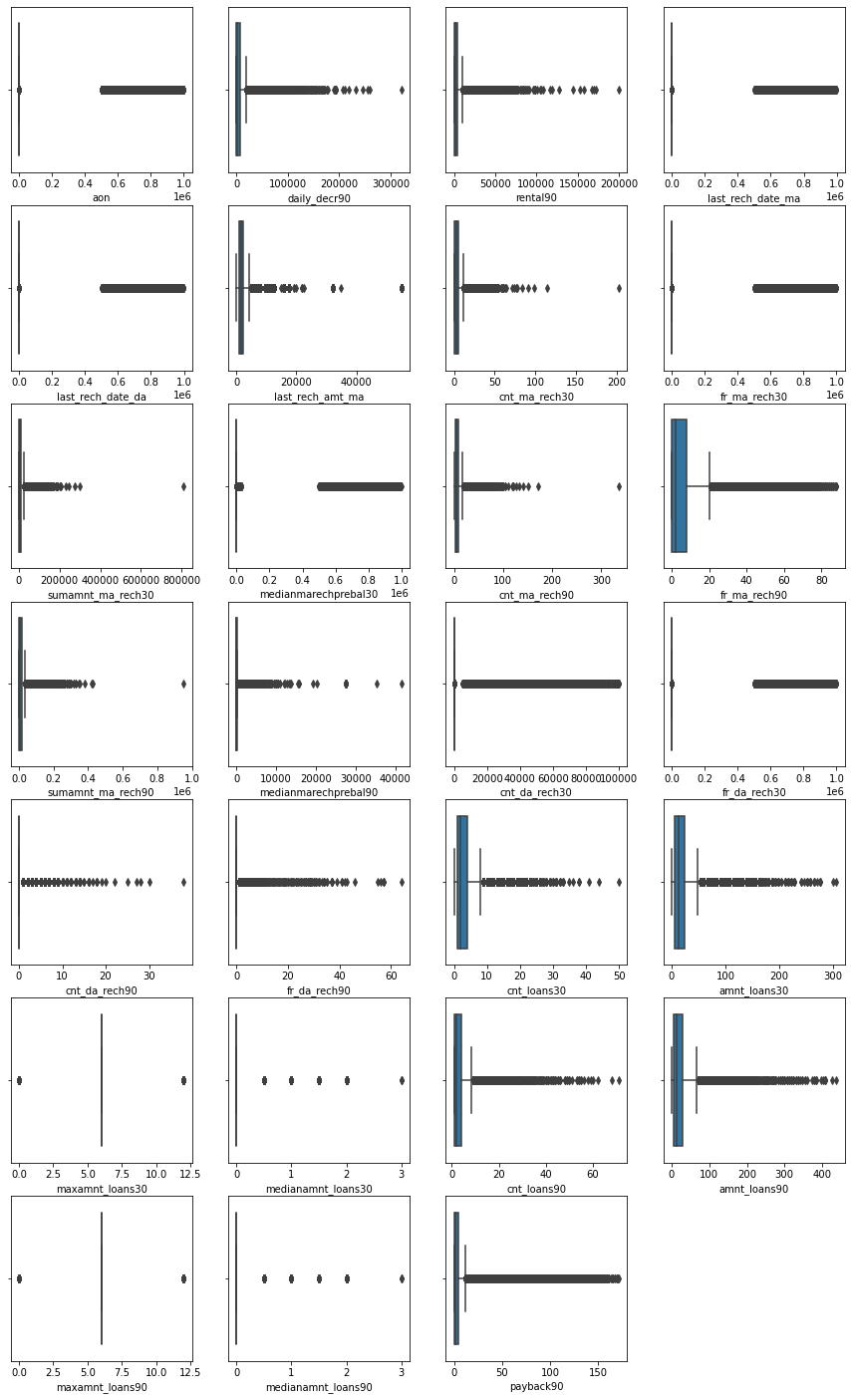
* Null Values:

We checked for the null values (missing values) and found that there is no null values in the given dataset.

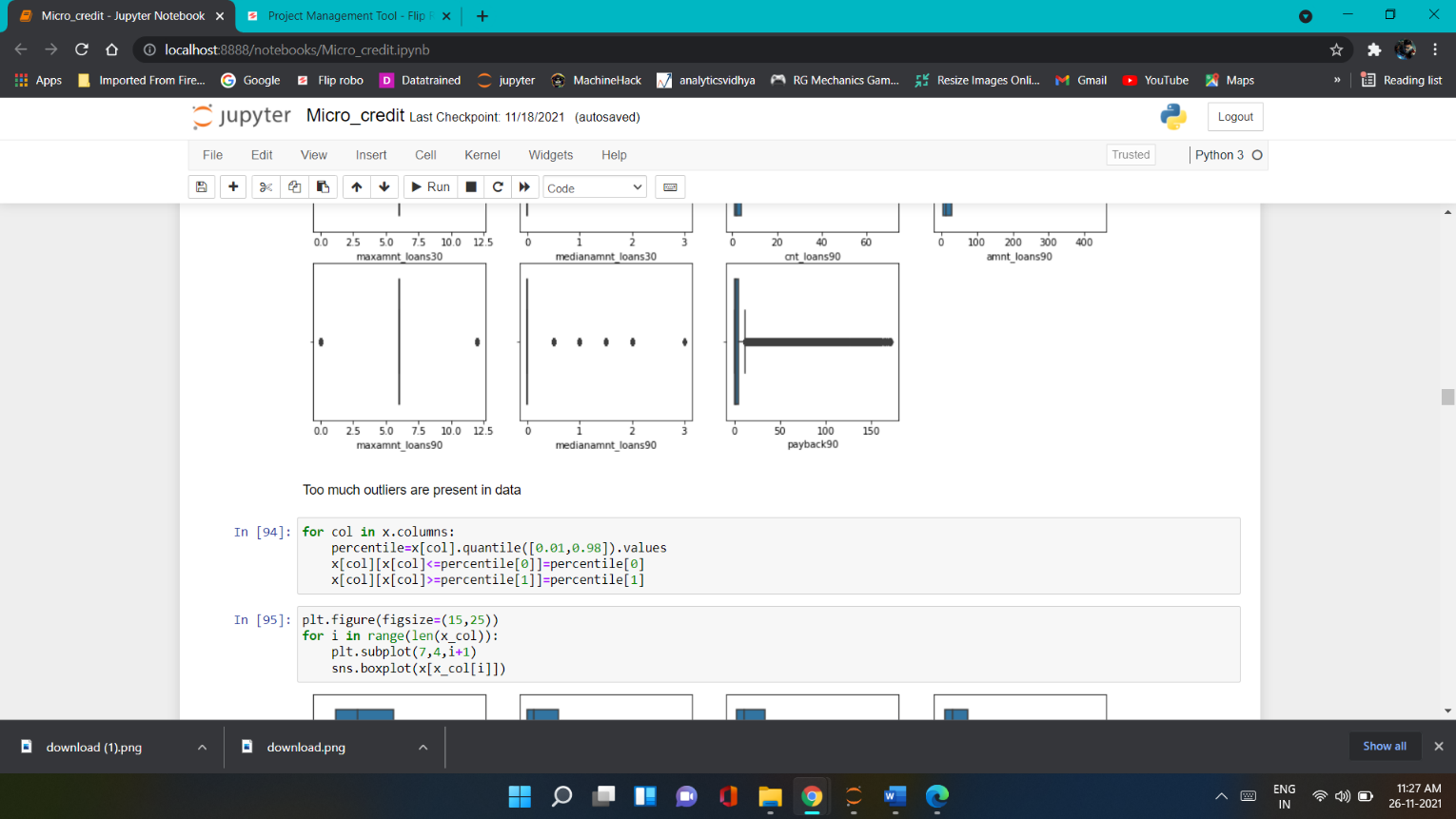
* Data Cleaning:
  + 1. Dropped ‘Unnamed:0’ column as it was not contributing to

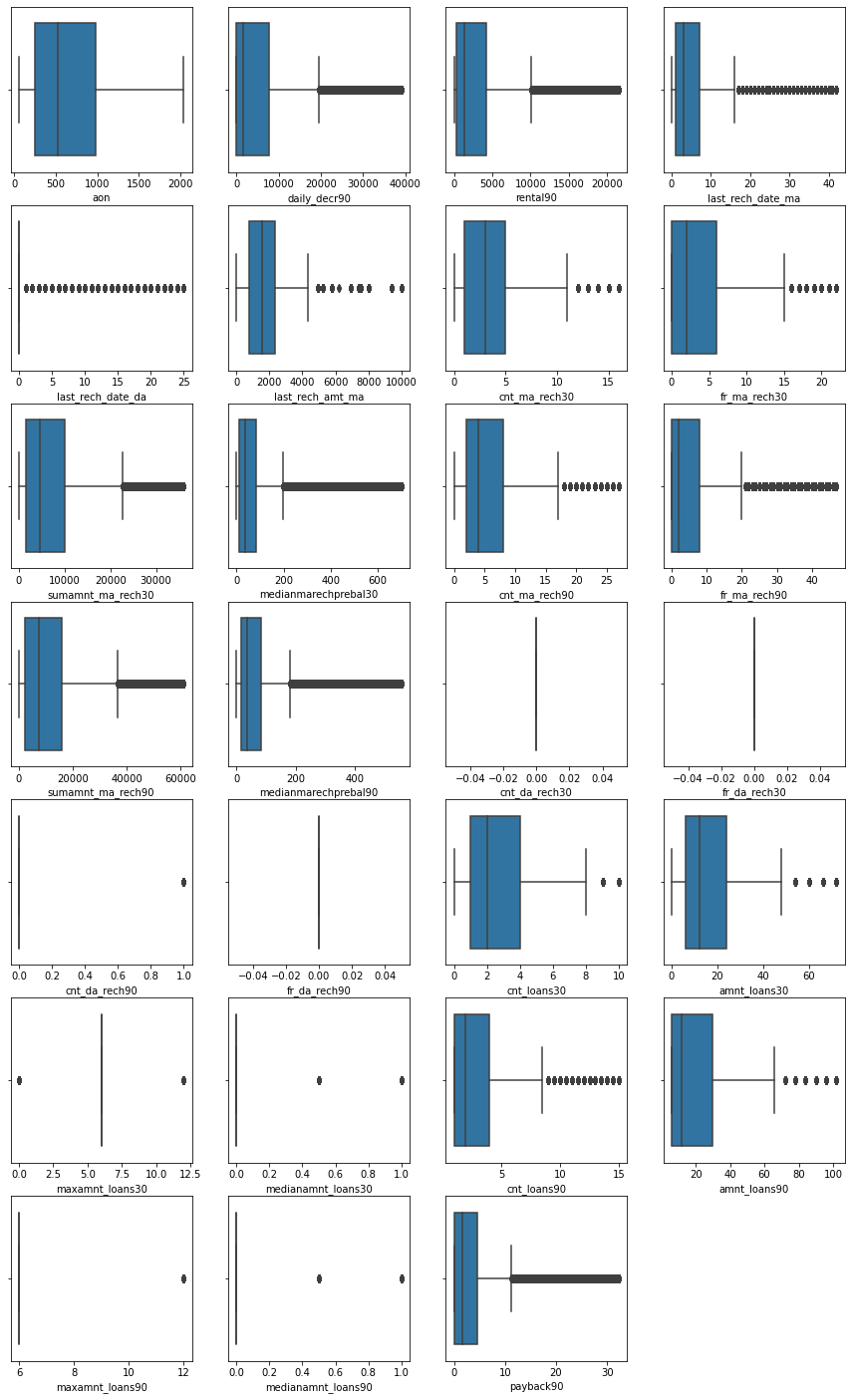
the dataset.

* + 1. Dropped 'msisdn' as it’ll not help in the model building.
    2. Split the ‘pdate’ column into day, month, and year and dropped the ‘pdate’ column.
    3. Dropped ‘year’ column as it only contains 2016 as value.
    4. Dropped ‘pcircle’ column as it contains single value (UPW).
    5. Observed negative values in dataset so convert them by abs() command.
    6. In problem statement maximum loan amount should be 0,6,12 so in loan\_30 amount in floating values so convert treat them.
    7. From statistical analysis skewness and outliers observed in data so treat them with suitable methods



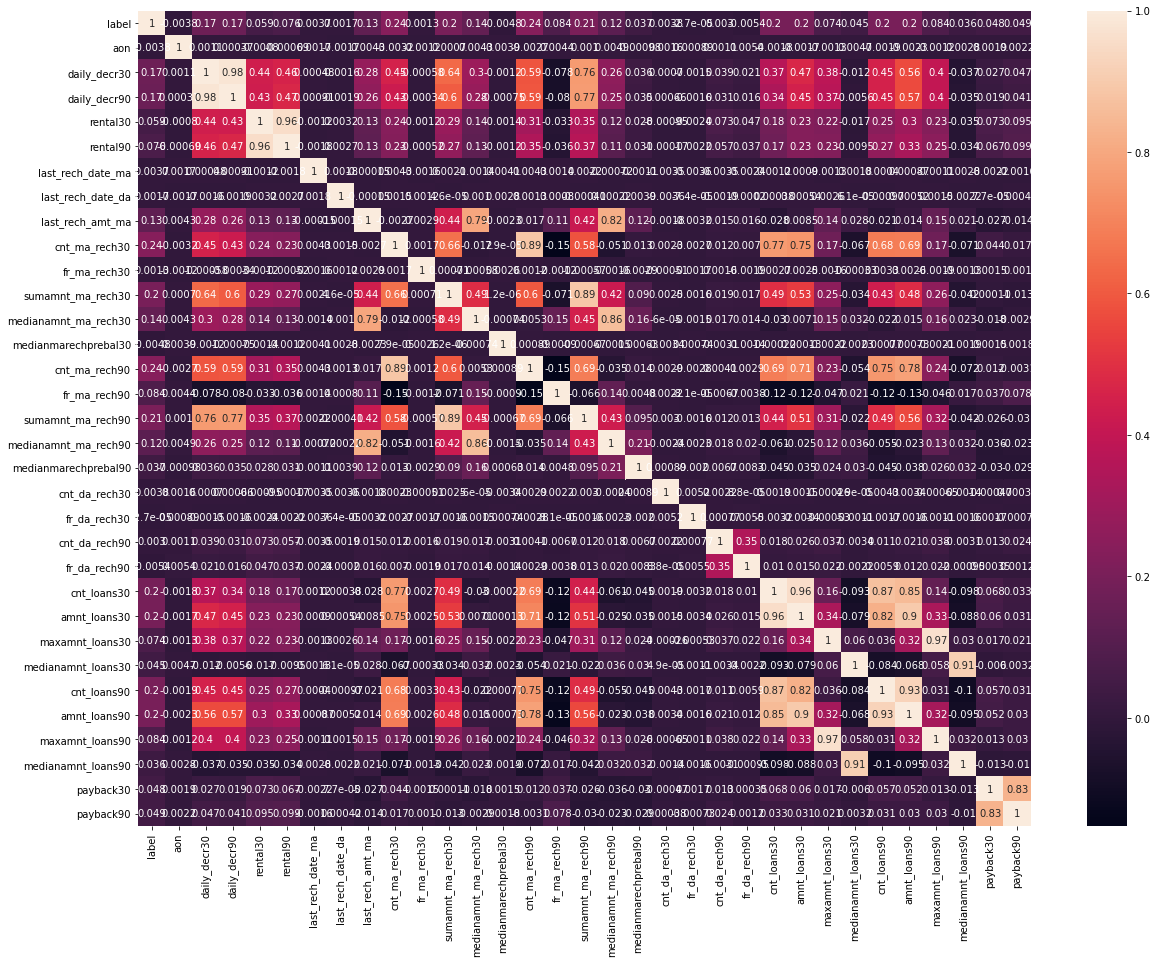
* Remove Outlier by percentile methods





## Data Inputs- Logic- Output Relationships

EDA was performed by creating valuable insights using various visualization libraries.



The main relationship between the input variable and the output variable is their correlation and covariance value. The value must lie between -1 to 1 for correlation and 0 to 1 for covariance for a strong relationship between input and the output.

For example ‘cnt\_loans90’ (number of loans taken in last 90 days)

By examining this column we can establish a relation between input and output, whether the user had taken the loan or not if he had taken whether he was able to pay it or not.

* **Libraries** **required :-**
* Import pandas as pd
* Import numpy as np
* Import matplotlib.pyplot as plt
* Import seaborn as sns
* Import warings
* From sklearn.model\_selection import train\_test\_split,GridSearchCV
* from sklearn.preprocessing import MinMaxScaler
* from sklearn.ensemble import RandomForestClassifier
* from sklearn.ensemble import AdaBoostClassifier
* from sklearn.ensemble import GradientBoostingClassifier
* from sklearn.ensemble import BaggingClassifier
* from sklearn.tree import DecisionTreeClassifier
* from linear\_model import LogisticRegression,SGDClassifier
* from xgboost import XGBClassifier
* from sklearn.metrics import classification\_report
* from sklearn.metrics import accuracy\_score
* from sklearn.model\_selection import cross\_val\_score

## Hardware and Software Requirements and Tools Used

### Hardware Configuration:

**Operating System:** Windows 10

**System Type:** 64-bit operating system, x64-based processor **Processor:** Intel® Core™ i3-5005U @ 2.00 GHz 2.00 GHz **RAM:** 4GB

### Software & Tools:

1. Jupyter Notebook (used as a notebook to code)
2. Python (used for scientific computation)
3. Pandas (used for scientific computation)
4. Numpy (used for scientific computation)
5. Matplotlib (used for visualization)
6. Seaborn (used for visualization)
7. Scikit-learn (used as algorithmic libraries)

# Models Development and Evaluation

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

* Performed EDA (Exploratory Data Analysis).
* Data Cleaning and dropping the columns which were not contributing to the dataset.
* Checked for the outliers and tried to remove the outliers of the dataset.
* Checked for the skewness in the dataset and removed the skewness for better model building.
* Train- Test the dataset into independent and dependent variables.
* Balancing dataset by SMOTE technique
* Model Building.
* Cross validation score to check if the model is over-fitted.

## Testing of Identified Approaches (Algorithms)

Below are the algorithms used for the training and testing:

* 1. Logistic Regression.
  2. XGBoost Classifier
  3. Random Forest Classifier.
  4. Decision Tree Classifier.
  5. AdaBoost Classifier
  6. Gradient Boosting Classifier
  7. SGD Classifier

## Run and Evaluate selected models

### Logistic Regression:

### 

From Logistic Regression we got 75% accuracy score.

### Decision Tree Classifier:

### 

From Decision Tree Classifier we got 84.4% accuracy score.

### Random Forest Classifier:

### 

From Random Forest Classifier we got 89.76% accuracy score.

### Ada Boost Classifier:

### 

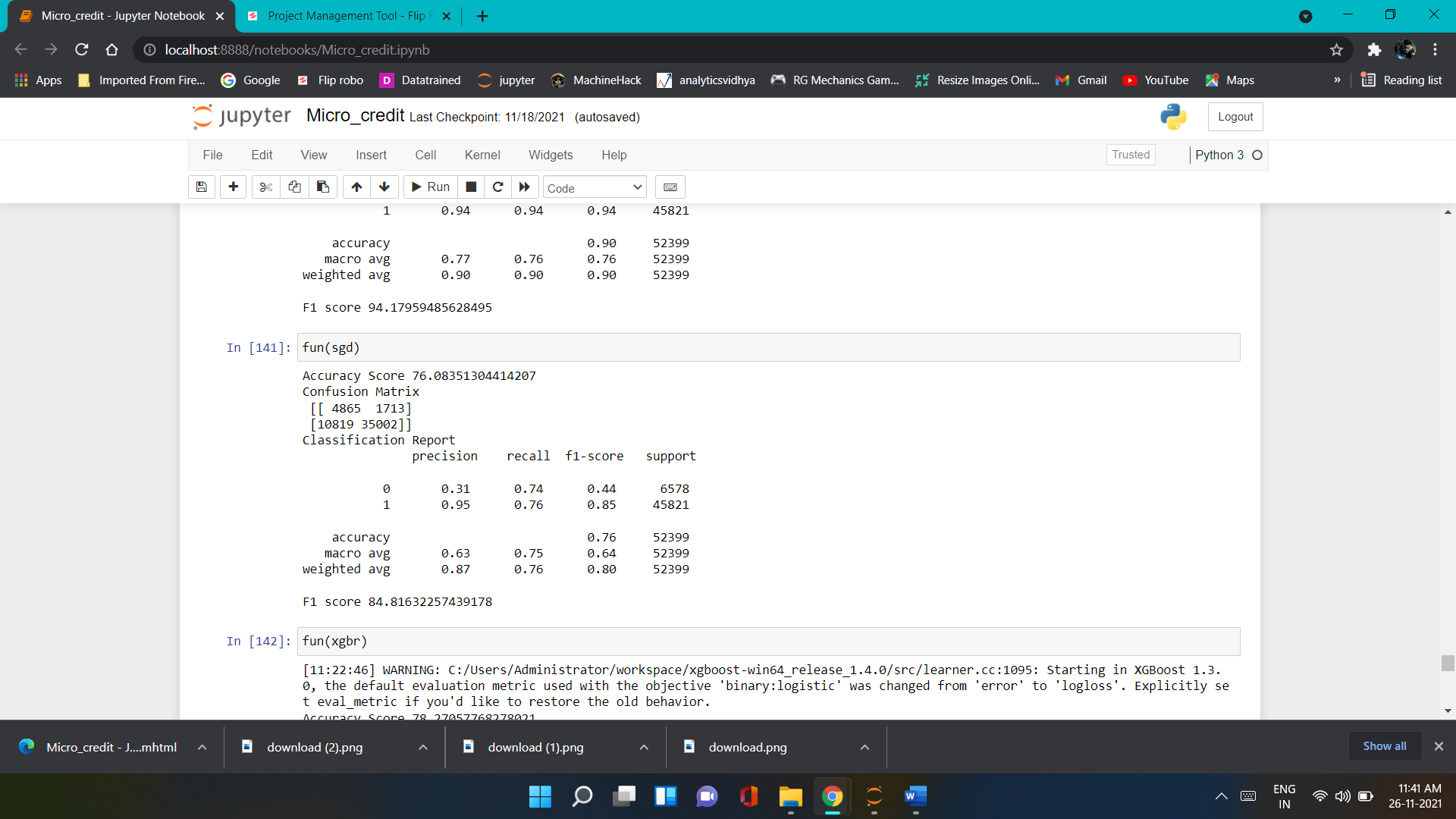
From Ada Boost Classifier we got 79.64% accuracy score.

### Gradient Boosting Classifier:

### 

From Gradient boosting classifier we got 83.18% accuracy score.

### SGD Classifier:



From SGD classifier we got 76.08% accuracy score.

### XGBoost Classifier:

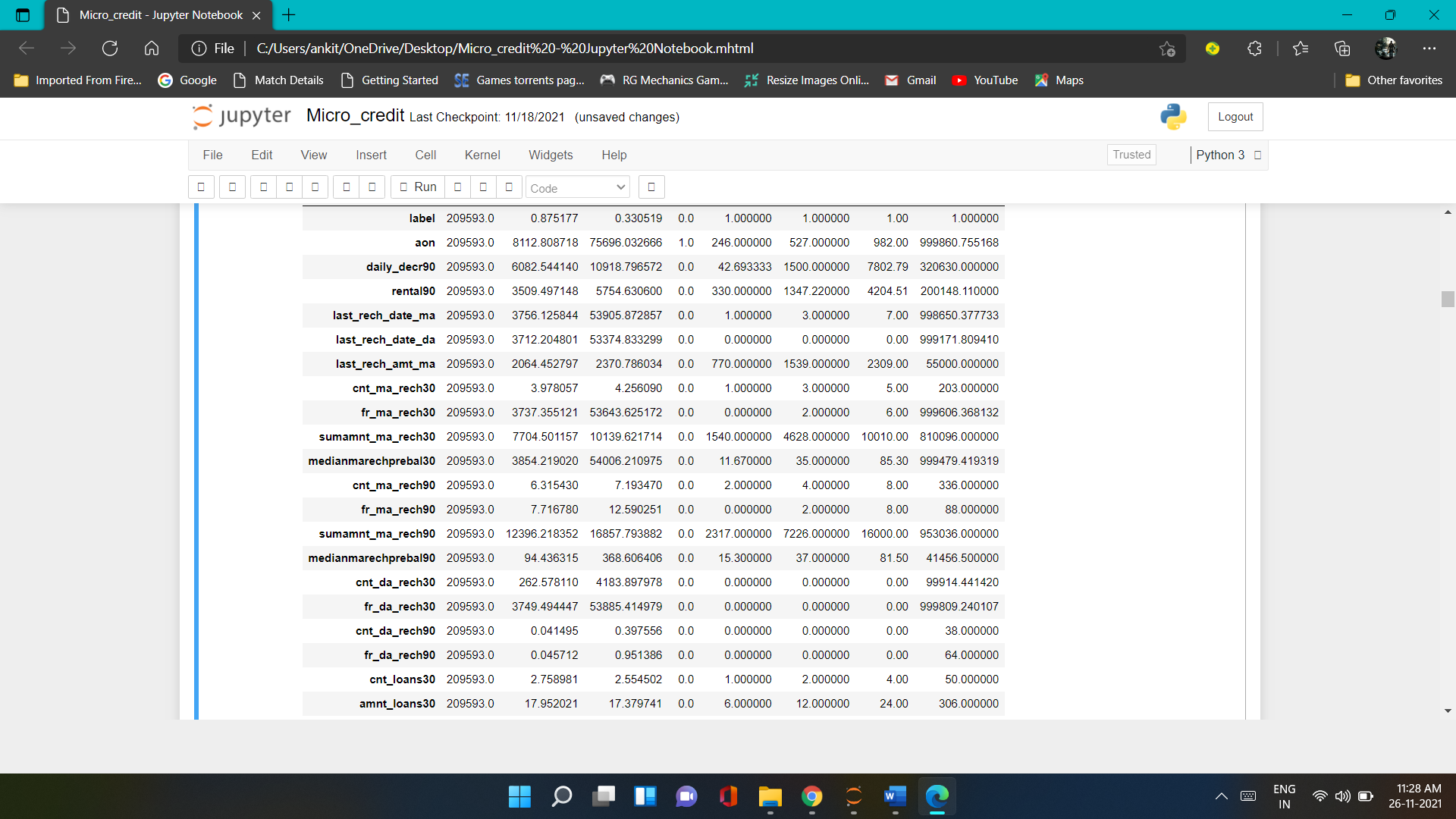
### 

From XGBoost classifier we got 89.78% accuracy score.

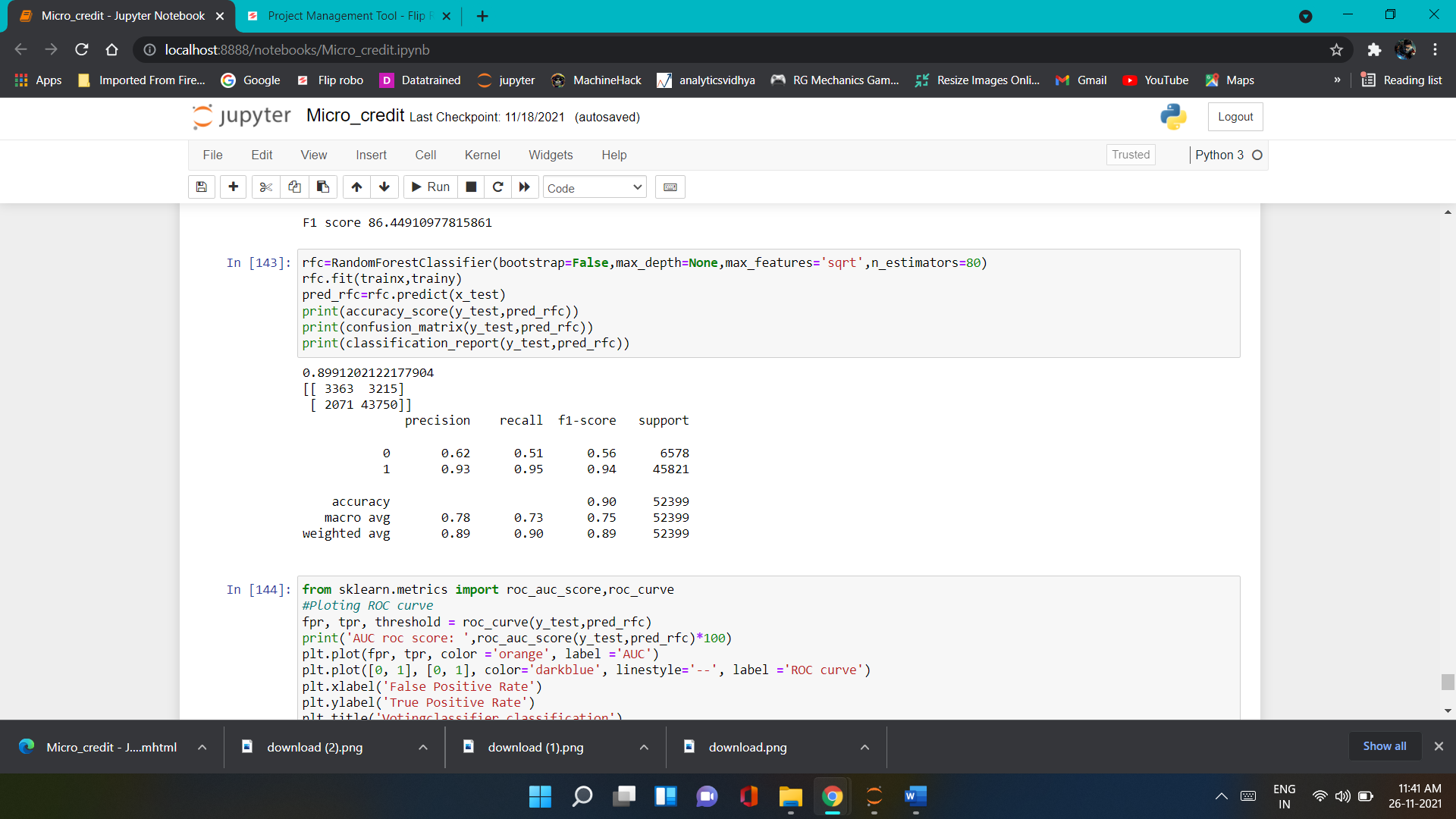
## Key Metrics for success in solving problem under consideration

The key metrics used are as follows:

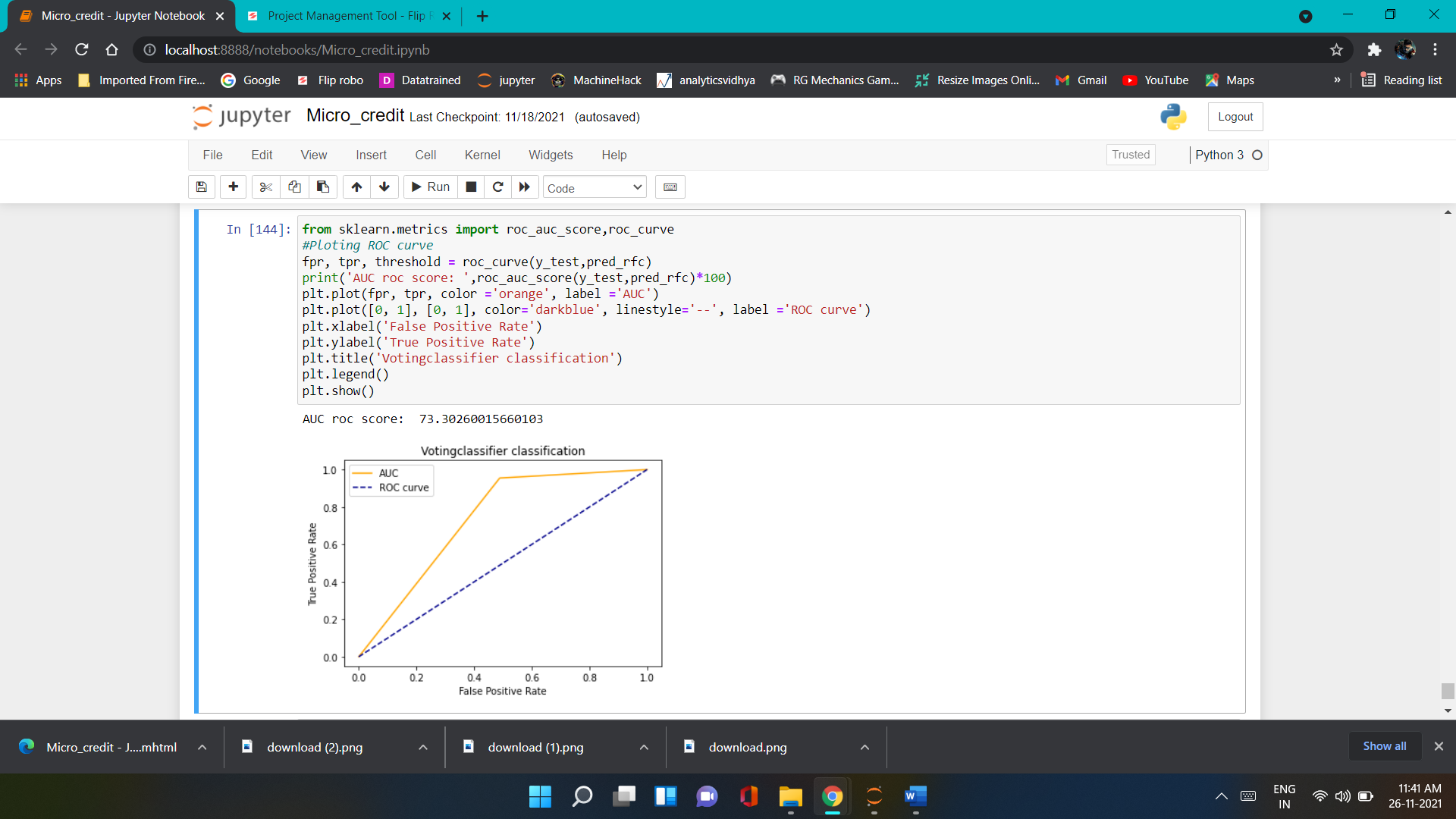
* 1. Accuracy Score
  2. Confusion Matrix
  3. Classification Report
  4. F1 Score
  5. Precision & Recall
  6. Cross validation score
* Statistical Summary
* Statistical Summary



* Final Model Selection and Hyper parameter using

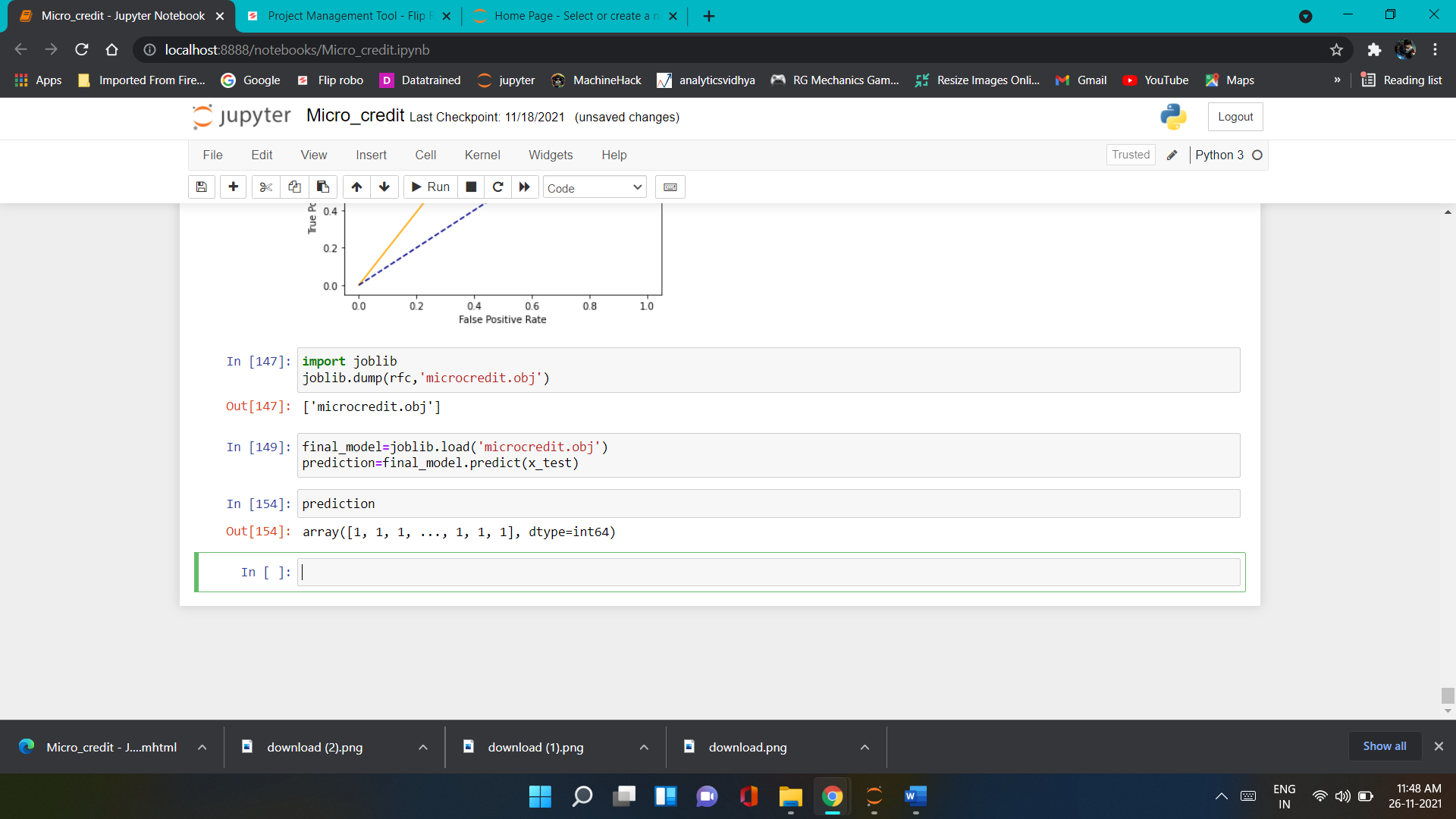


* ROC AUC Curve:



Area for the AUC ROC SCORE: 73.30 %

* SAVING BEST MODEL FOR PREDICTION



# CONCLUSION

## Key Findings and Conclusions of the Study

* If the number of days of payback is increasing the chance of defaulters is also increasing. So, we should look for the payback duration.
* If the loan amount is below 100 and the number of loans taken by users is 90 days, the number of defaulters is increasing.

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

This project helped me to work on the real time industrial data, which helped me to gain the real time experience. In the project I got to work on the different type of algorithms and fitting the best model based on the accuracy score and cross validation score. We achieved accuracy score of 89.99% using the Random Forest Classifier Tree Classifier.