An Efficient Novel Approach for Prediction of Start-Up Company Success Rates through ML Paradigms

In the past many peoples were investing millions of money in new start-ups blindly without knowing about its success and this is one kind of gambling which leads person to success or failure in his start-up. To overcome from this issue we are employing machine learning algorithms to predict start-up success by using parameters like funding, participants, milestones etc.

Propose algorithm will predict start-up success or failure and then give suggestion to make start-up successful. In propose work for accurate prediction we have employed and evaluate performance of many ML algorithms like Random Forest, KNN, SVM, Naïve Bayes and Logistic Regression. Each algorithm performance is evaluated in terms of accuracy, precision, recall, FSCORE, confusion matrix and ROC graph. Among all algorithms Random Forest achieved highest accuracy.

To train and test above algorithm performance we have utilize start-up dataset from KAGGLE repository which can be download from below URL

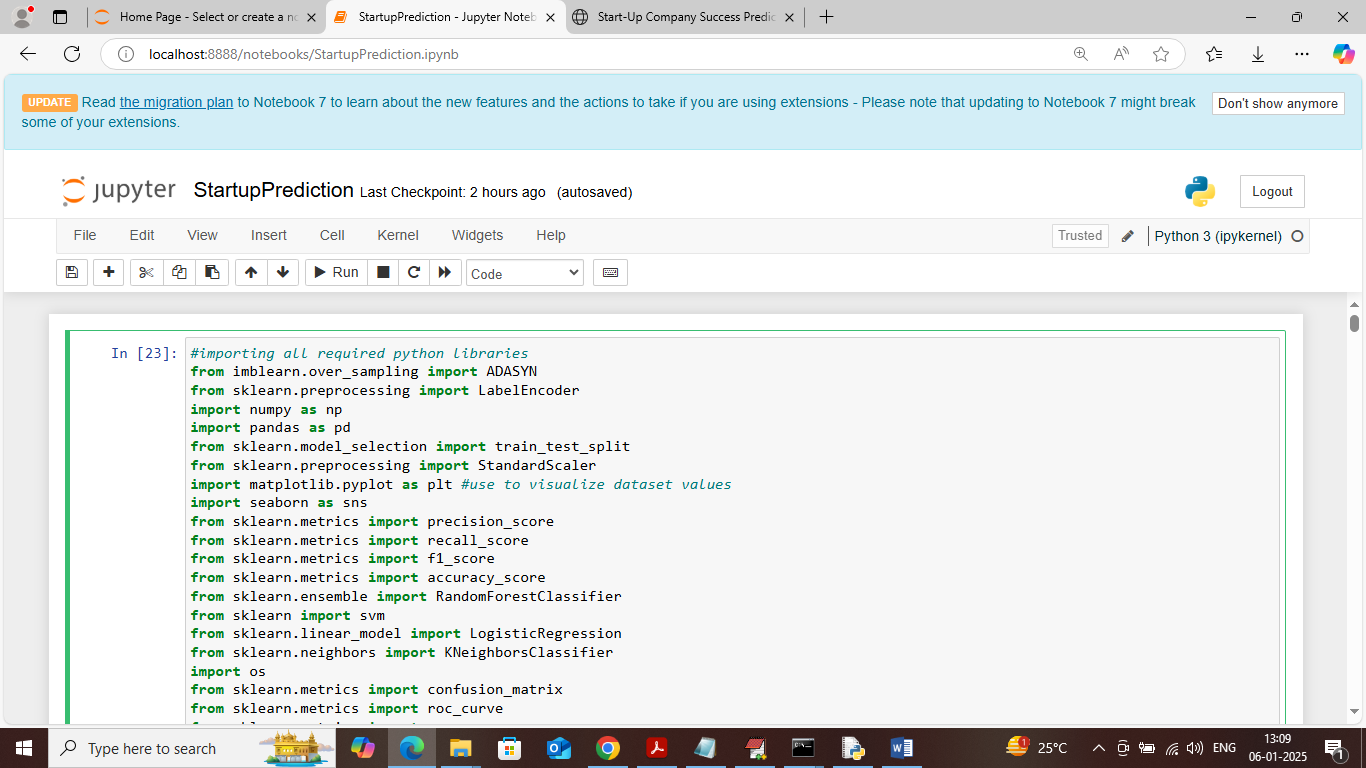
<https://www.kaggle.com/datasets/manishkc06/startup-success-prediction>

while training ML algorithms we have explore dataset with various visualization and then handle missing values, handle data imbalance issue using ADASYN algorithm and then normalize all training features. Each algorithm run with different and default tuning parameters and then choose default or tuned parameters based on algorithm performance.

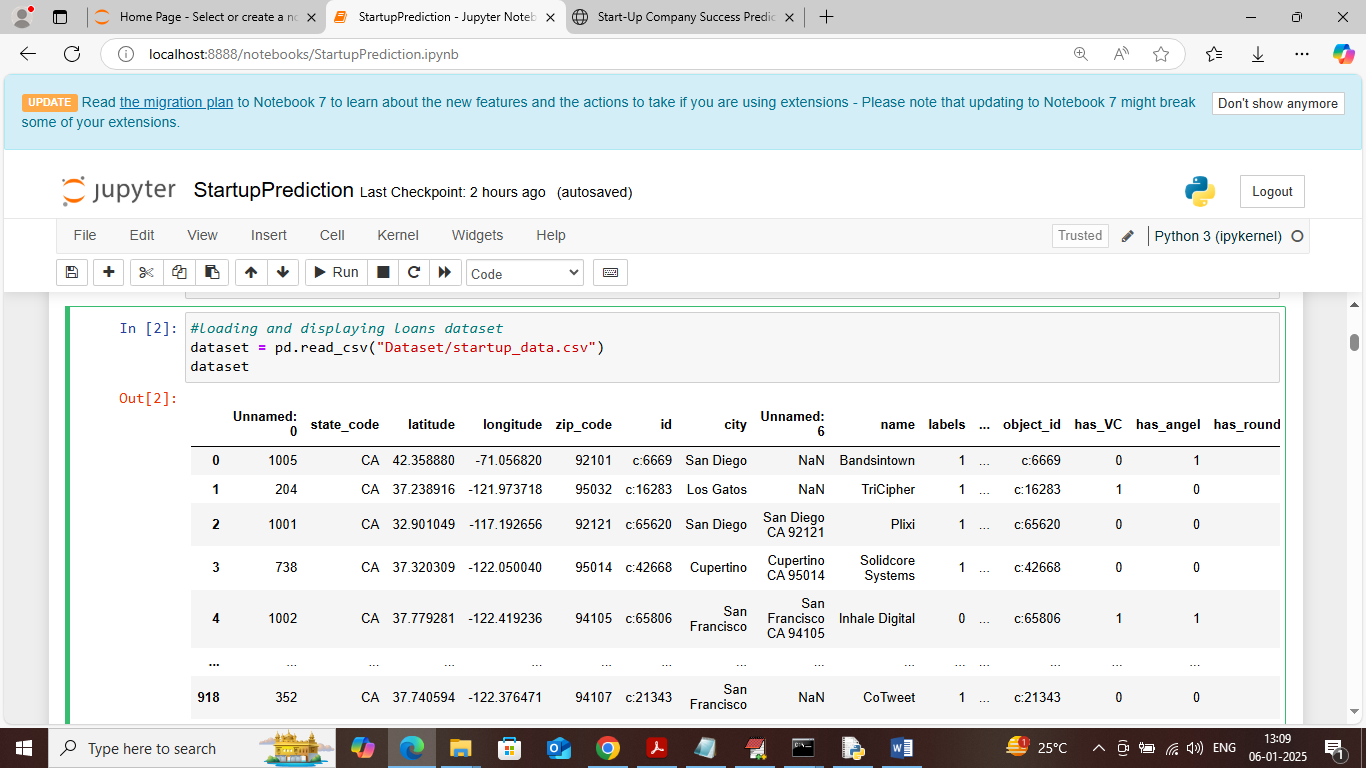
For dataset analysis, visualization, training and testing we have utilized JUPYTER notebook and then utilize flask web framework for start-up success prediction.

JUPYTER SCREEN SHOTS

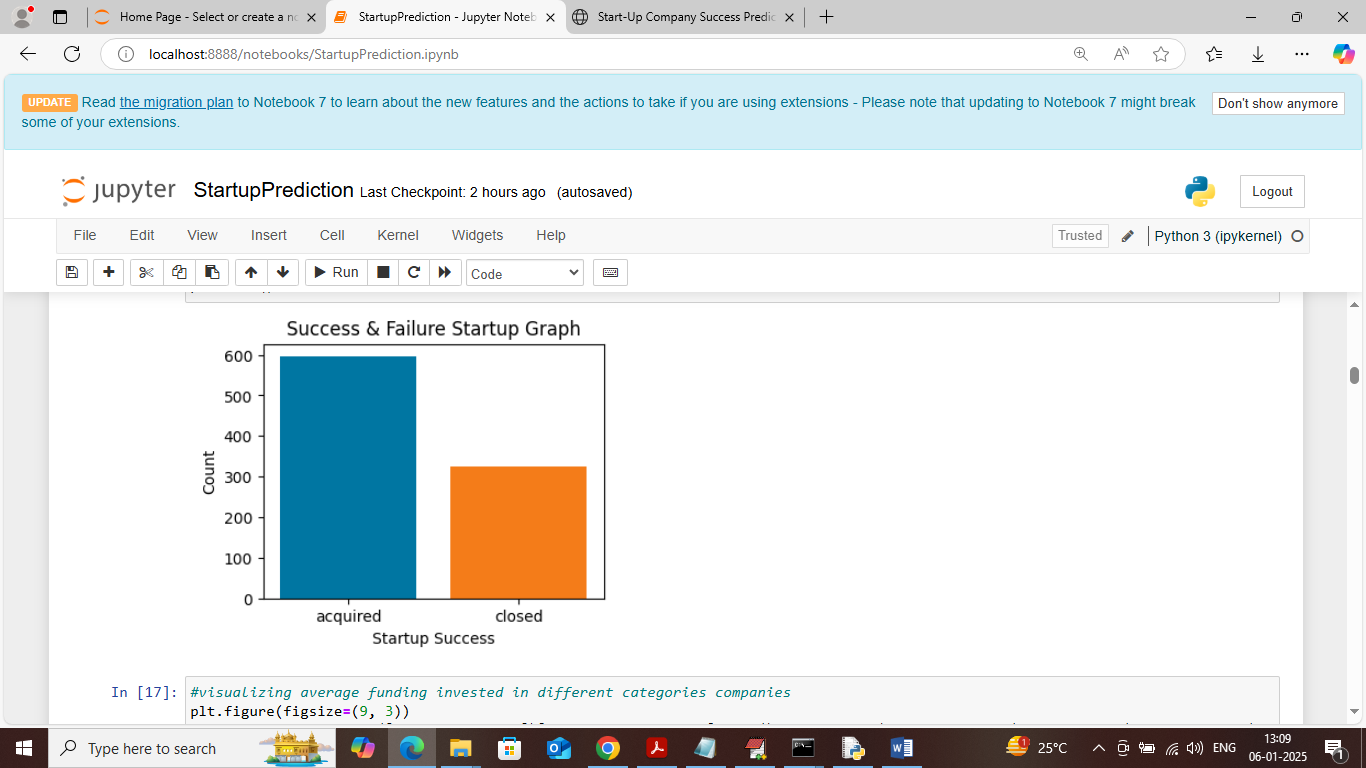
To run JUPYTER notebook double click on ‘runJUPYTER.bat’ file to start notebook and get below page



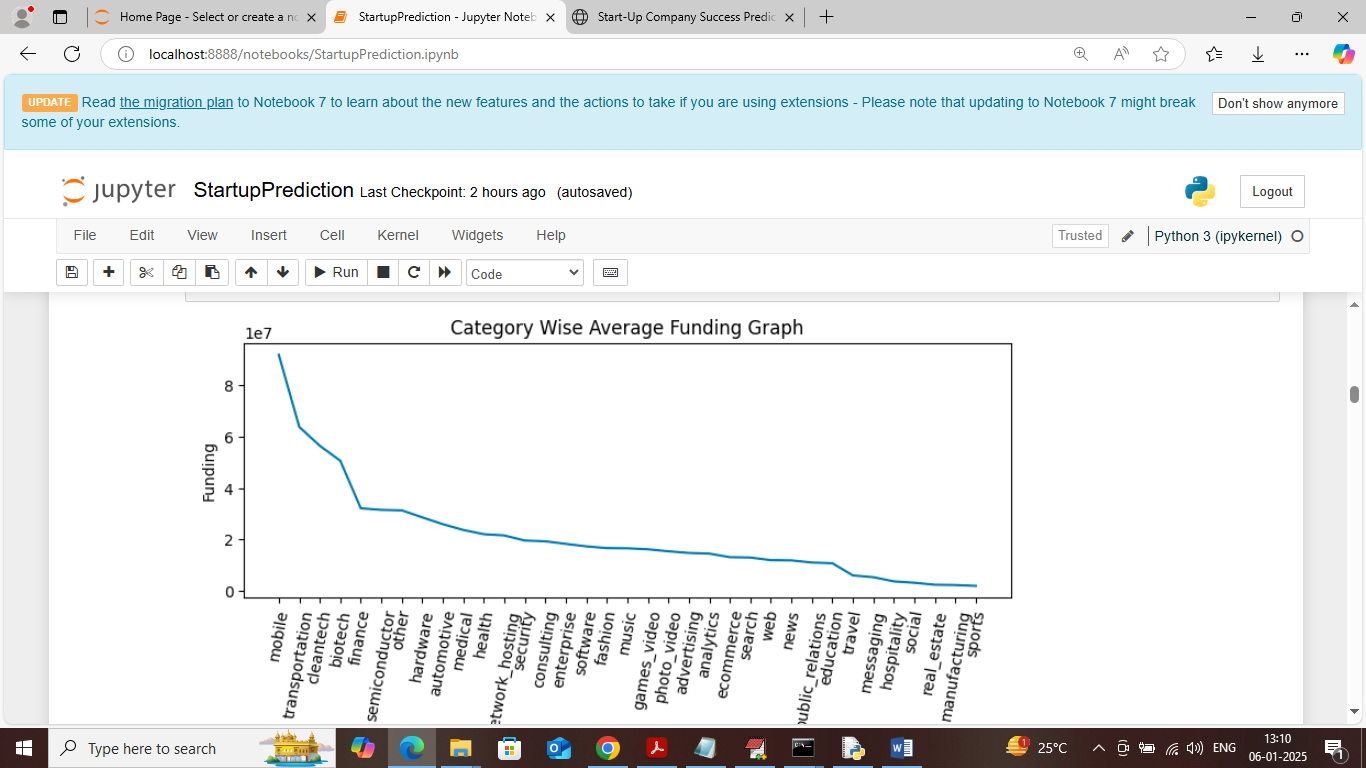
In above screen importing required python classes and packages



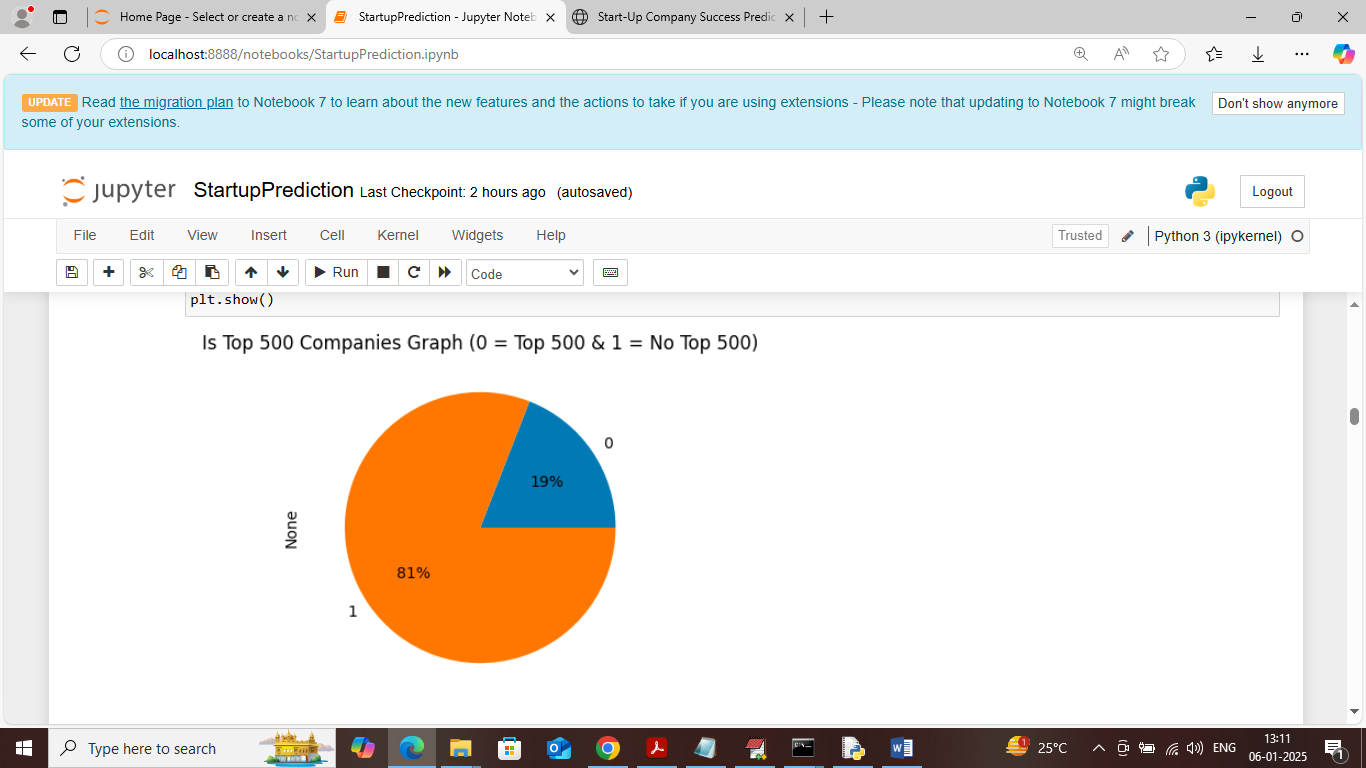
In above screen loading and displaying dataset values



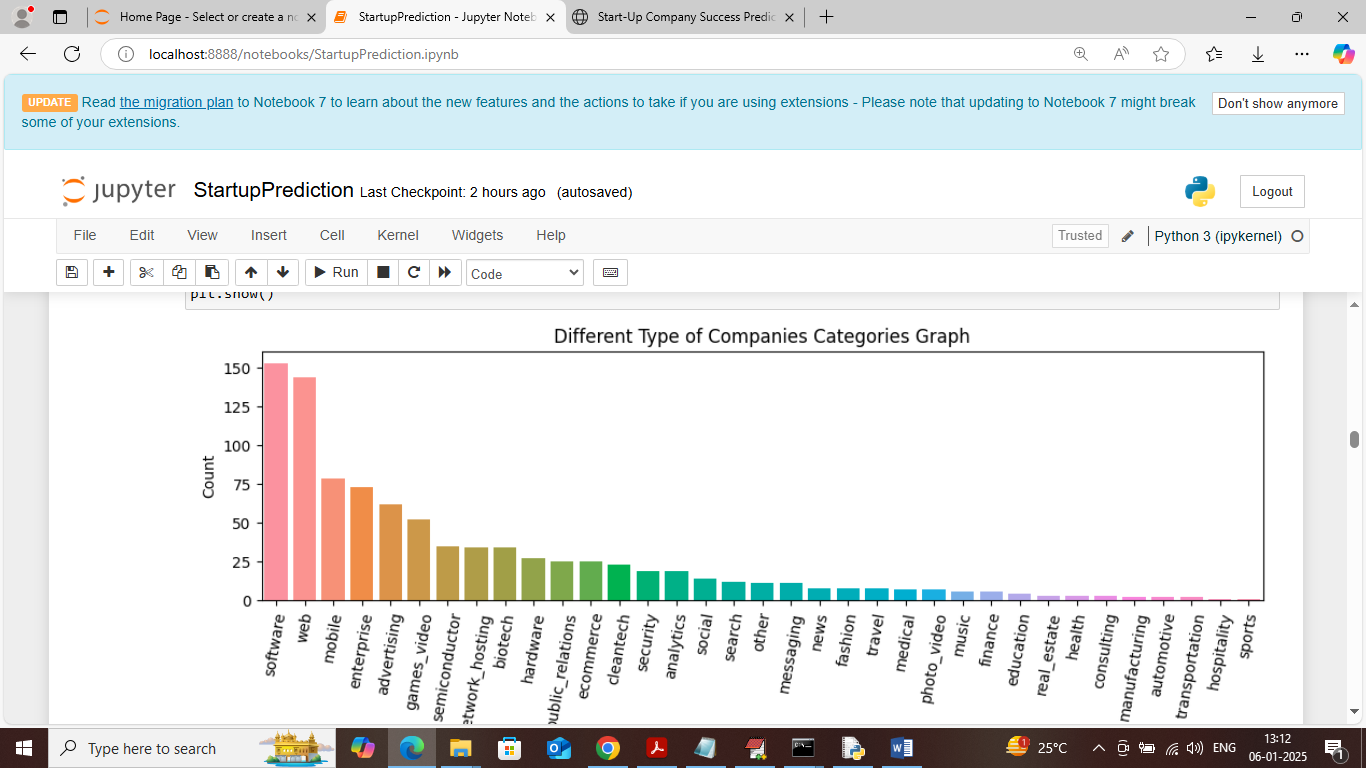
In above screen visualizing graph of start-up success where x-axis represents start-up acquired or closed status and y-axis represents counts



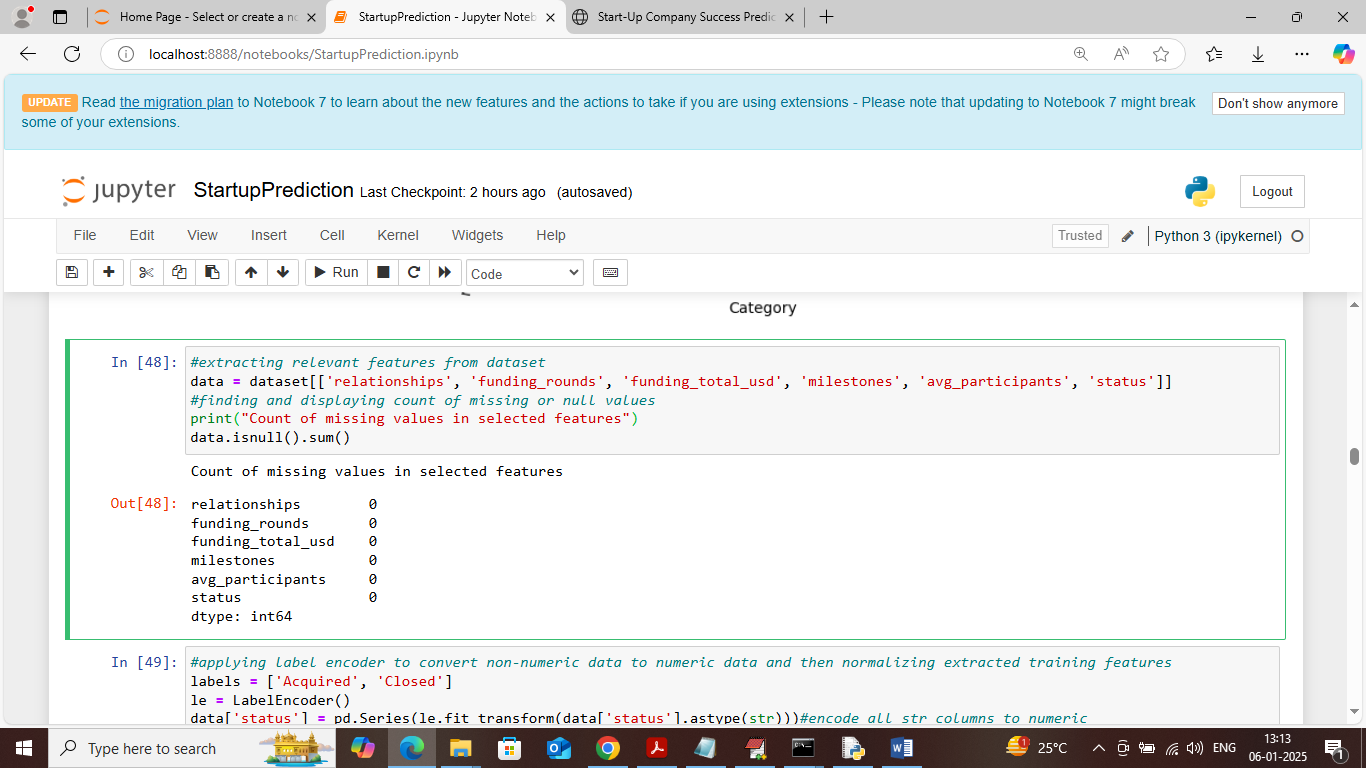
In above screen visualizing graph of total amount invested in start-ups of different categories where x-axis represents start-up category and y-axis represents average invested amount



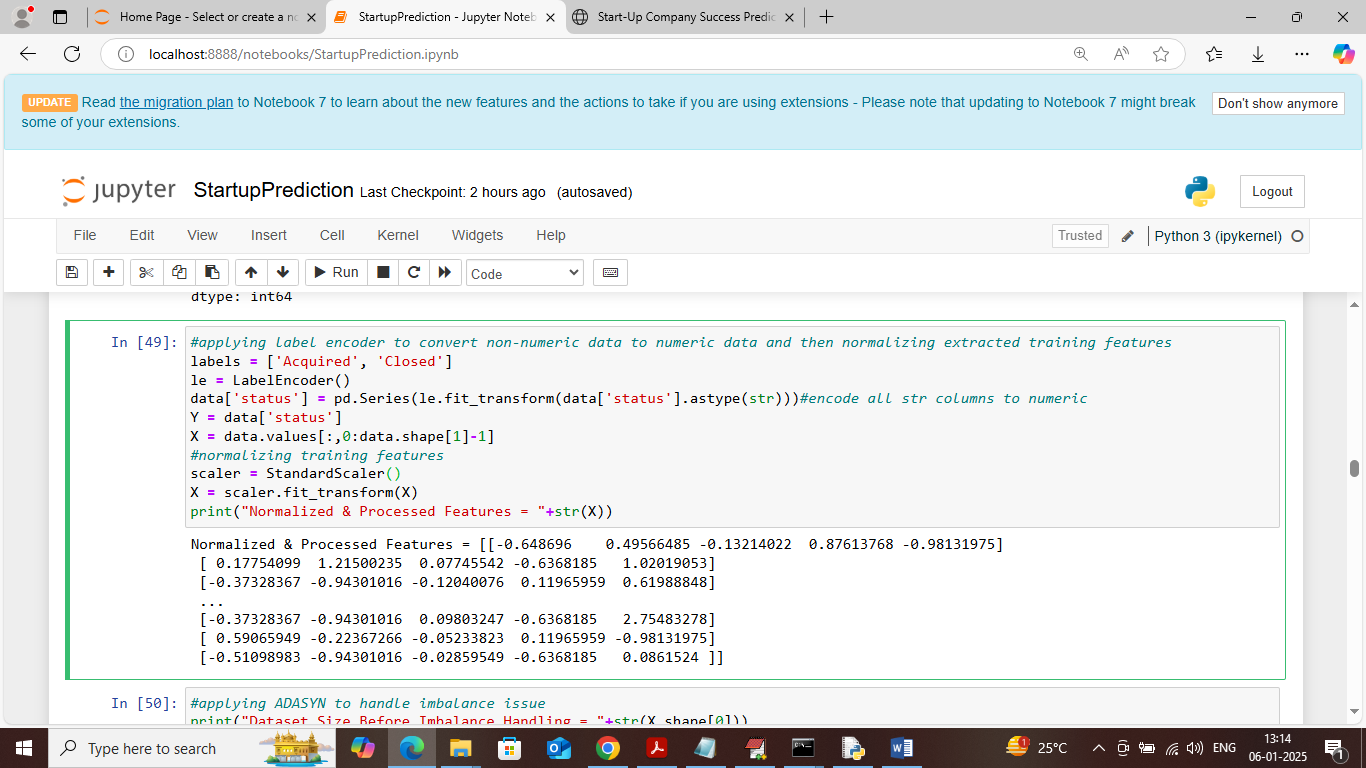
In above screen visualizing graph of percentage of companies who are in top 500 where 0 represents top 500 and 1 represents not in top 500



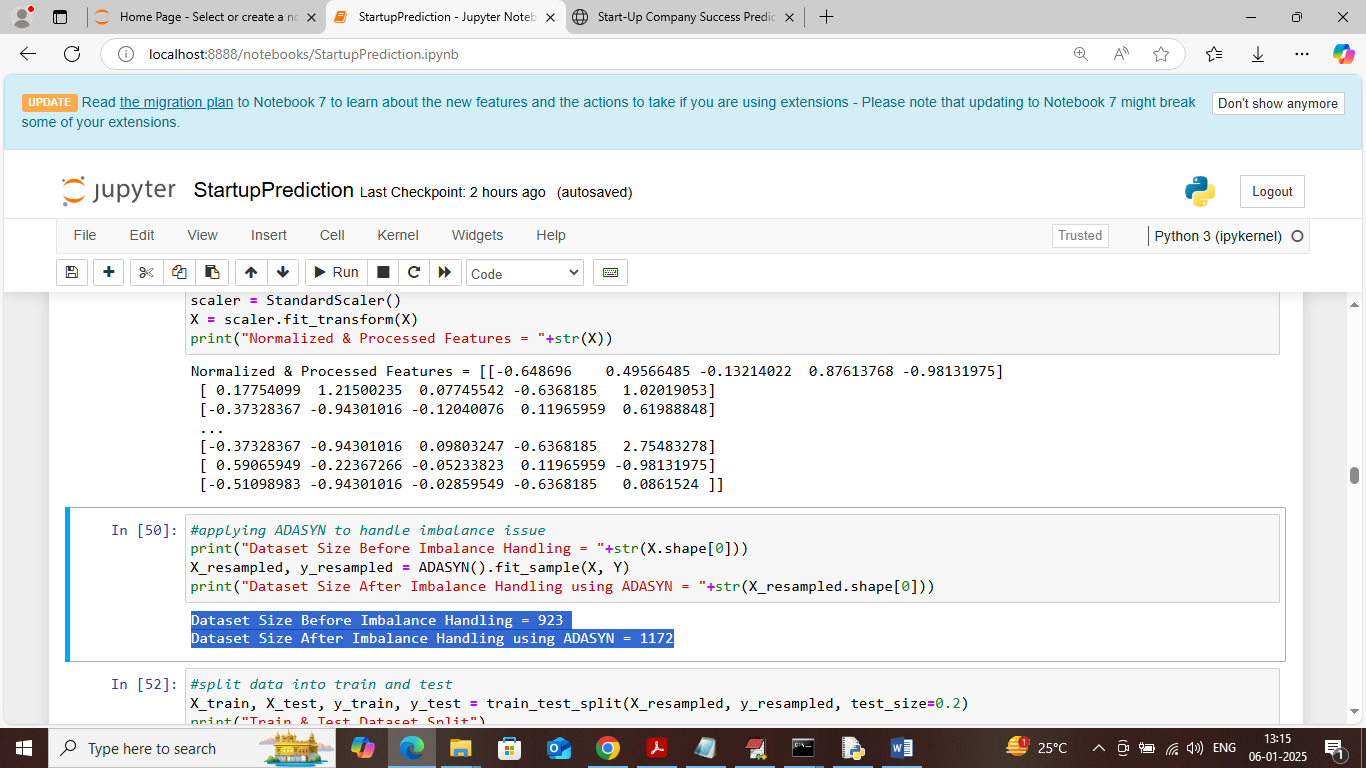
In above screen visualizing graph of number of different categories start-up found in dataset where x-axis represents category name and y-axis represents count



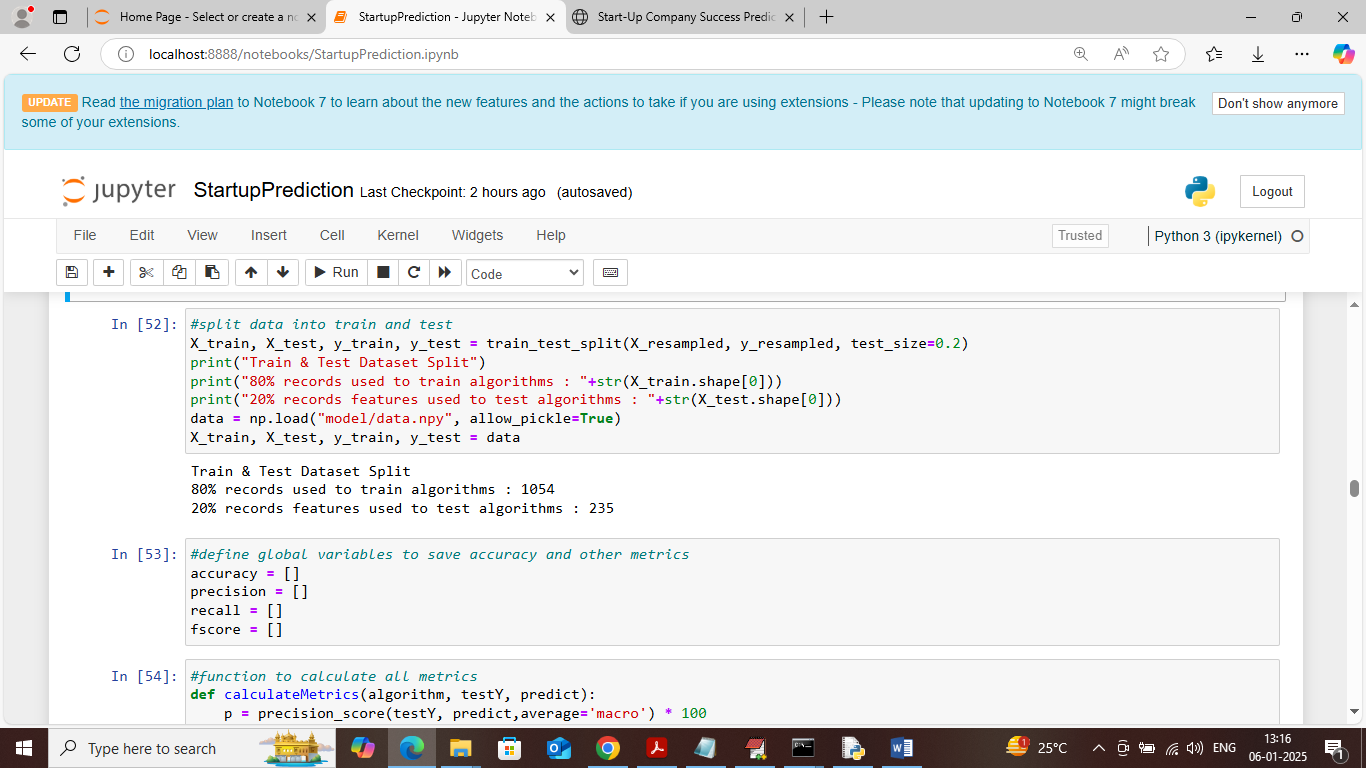
In above screen applying processing techniques such as extracting training features and then counting number of missing values and in above dataset no missing data exists



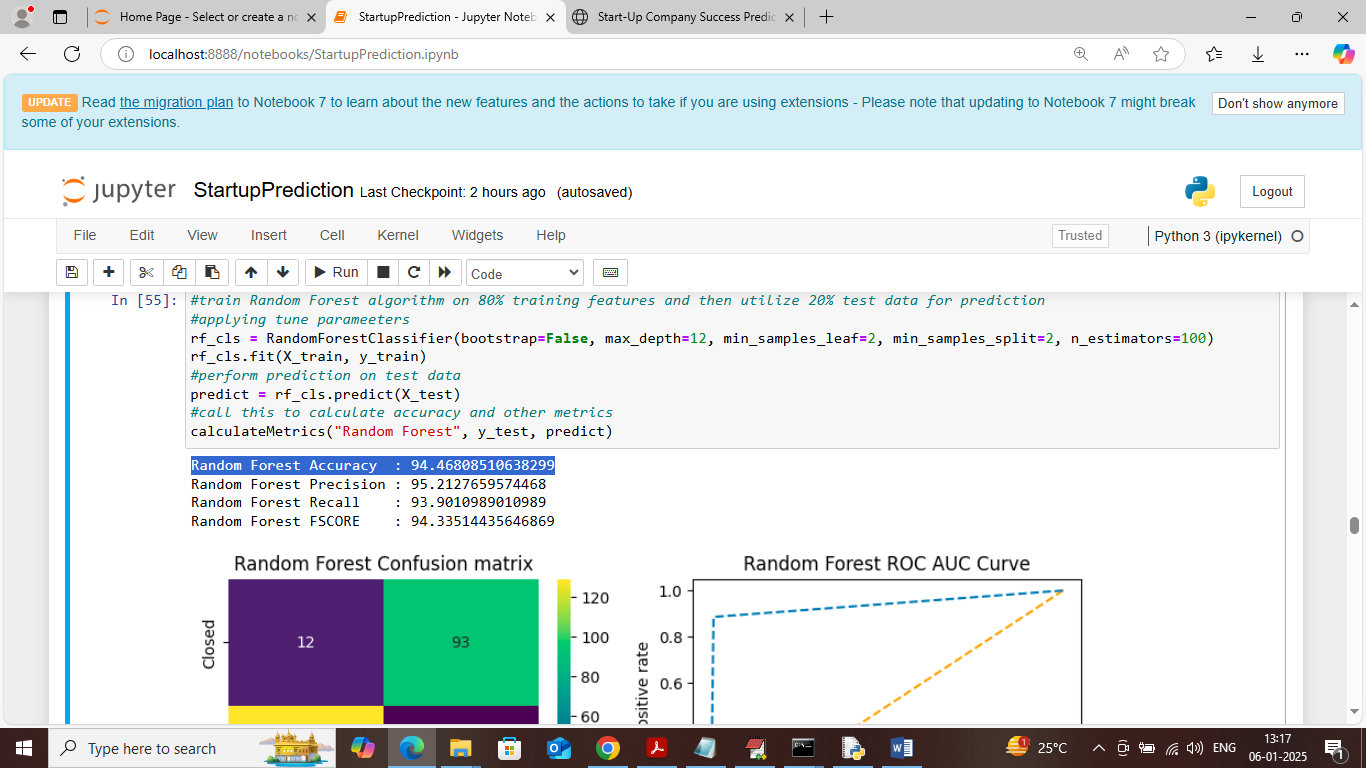
In above screen extracting X and Y training features from dataset and then normalizing training features and then displaying all normalized features



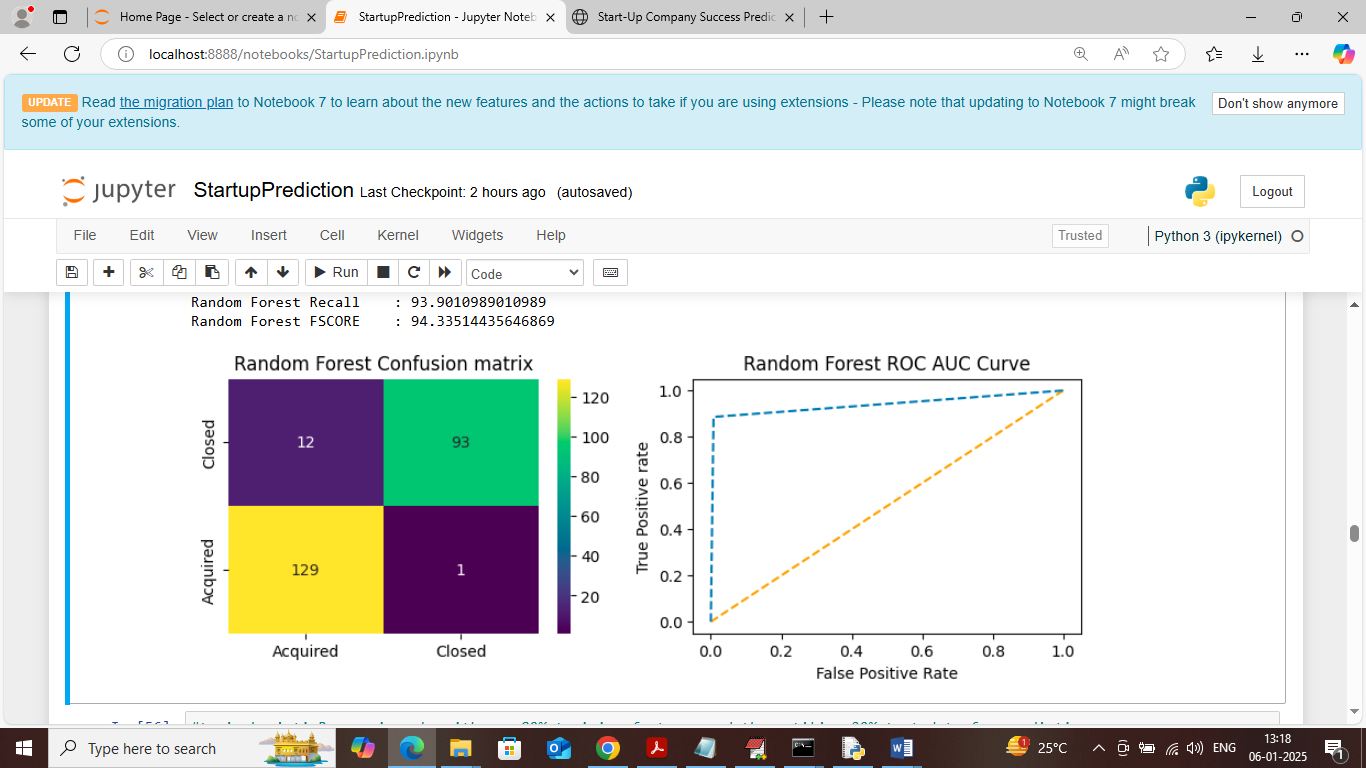
In above screen applying ADASYN algorithm to generate synthetic instances for imbalance classes and in above blue text before applying ADASYN dataset were having 923 instances and after applying ADASYN dataset sze increased to 1172



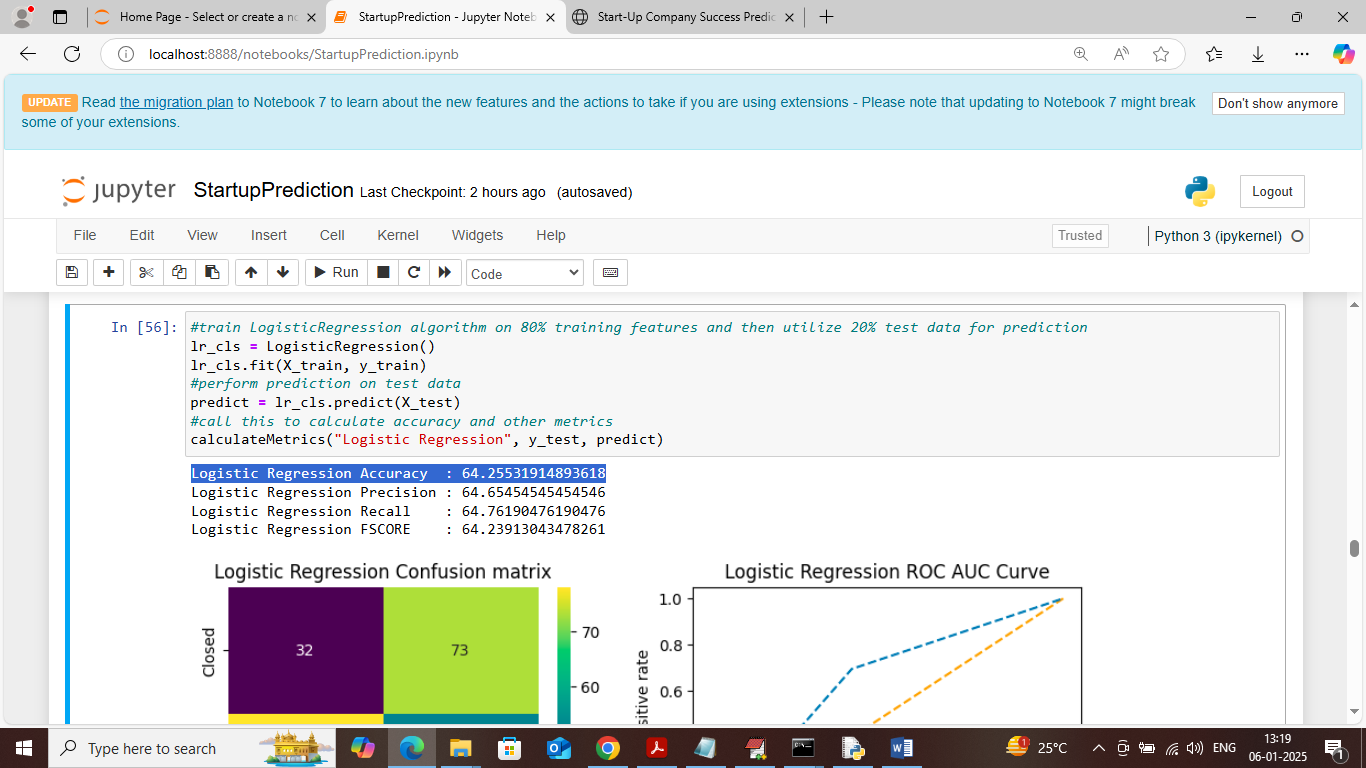
In above screen splitting processed features into train and test where application using 80% dataset features for training and 20% for testing and then defining function to calculate accuracy and other metrics



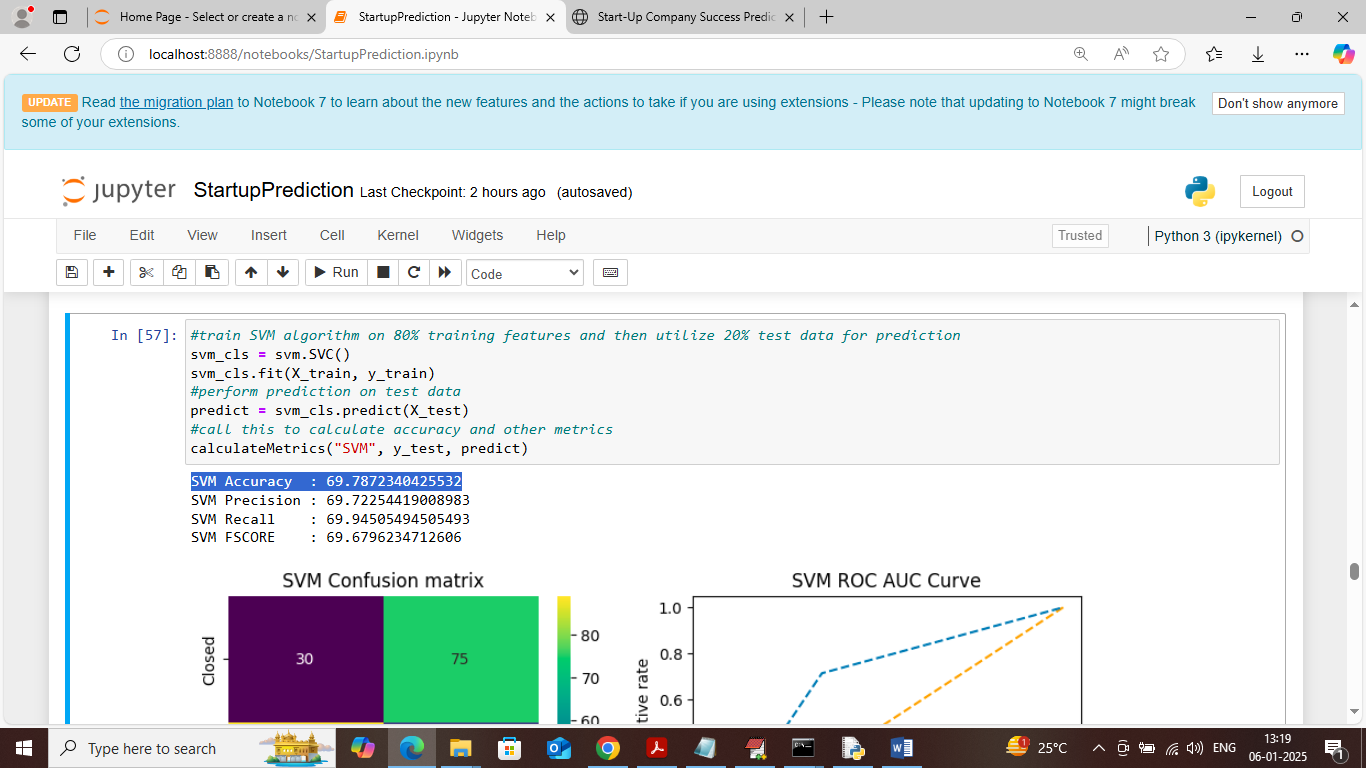
In above screen training Random Forest algorithm on training data and then performing prediction on test data and then Random Forest got more than 94% accuracy and can see other metrics like precision, recall and FCSORE. Below is the confusion matrix graph



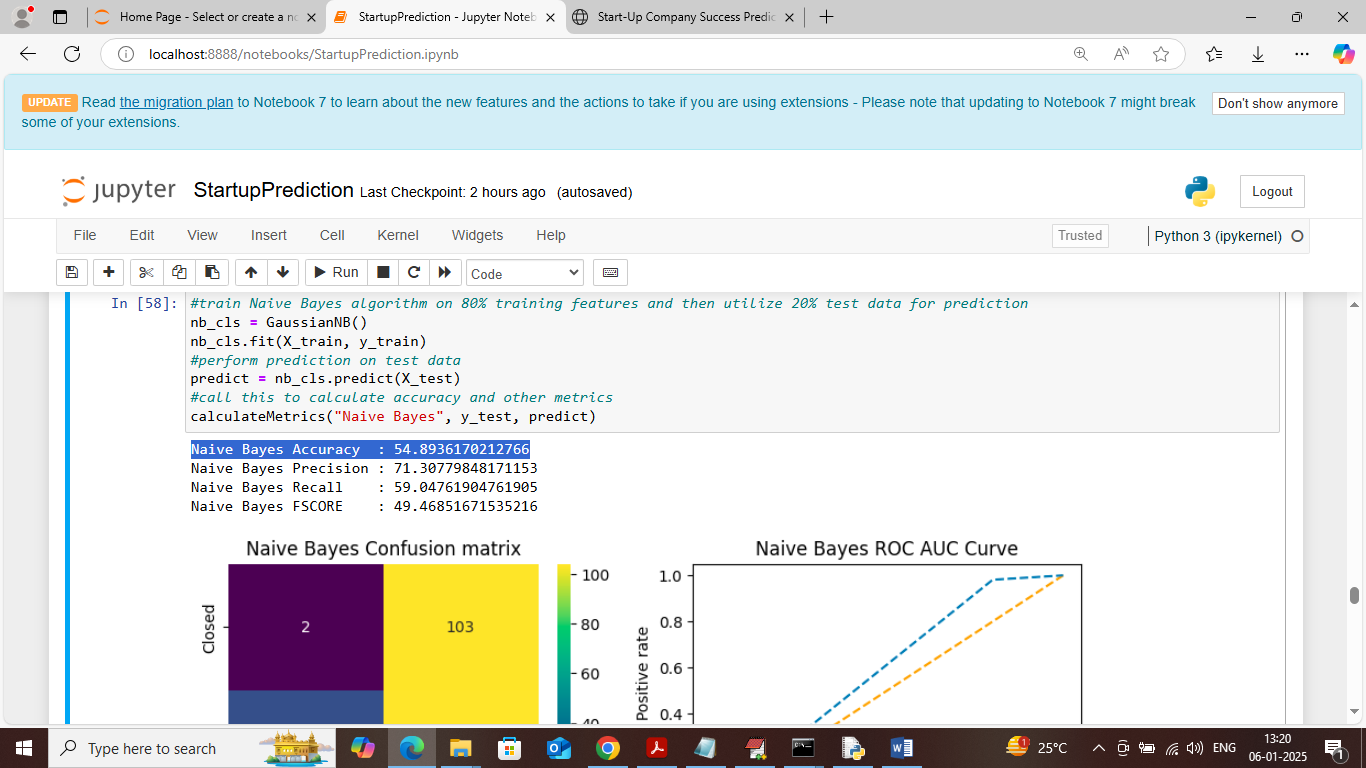
In above Random Forest confusion matrix graph x-axis represents Predicted Labels and y-axis represents True labels and then different color boxes like yellow and green represents correct prediction count and blue boxes represents incorrect prediction count which are very few. In ROC graph x-axis represents False Positive Rate and y-axis represents True Positive Rate and if blue lines comes below orange line then all predictions are incorrect or false and if goes above orange line then all predictions are correct or true.



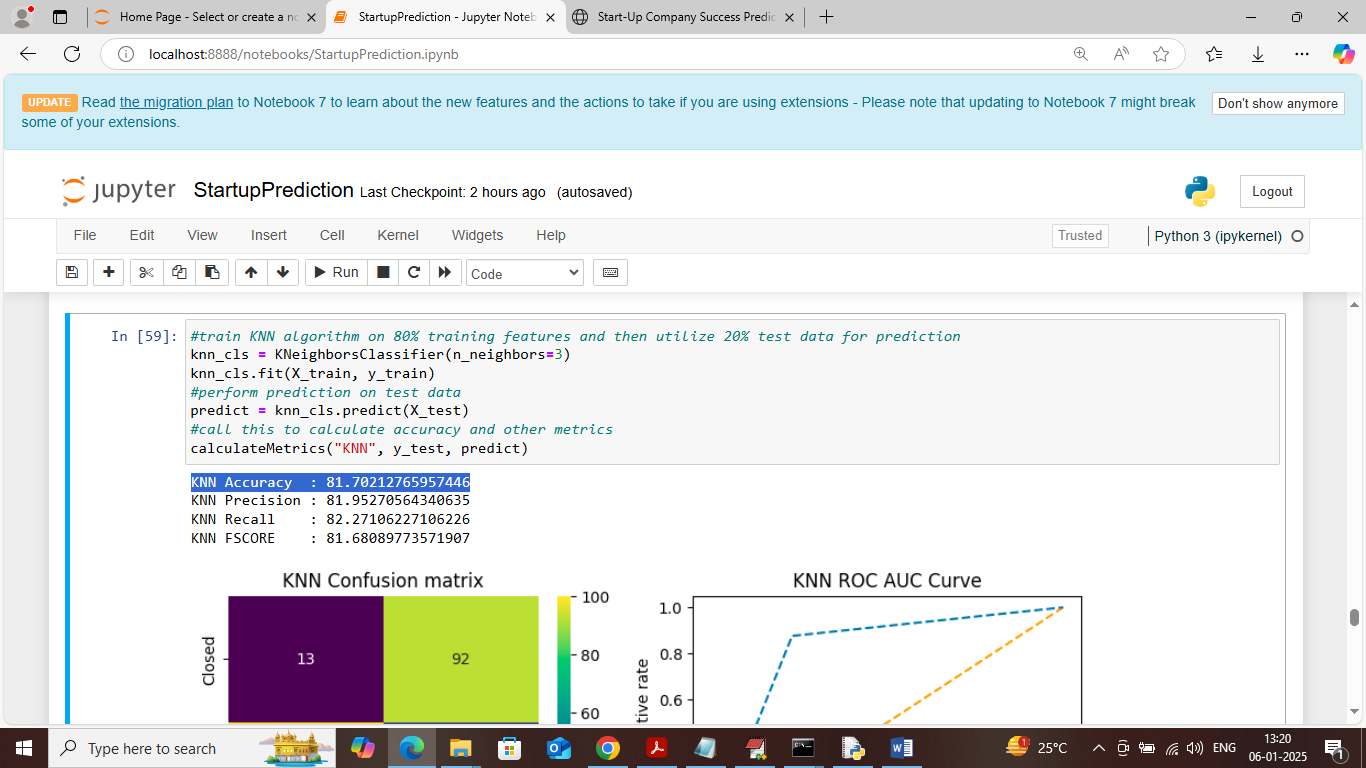
In above screen Logistic Regression got 64% accuracy and can see other metrics also



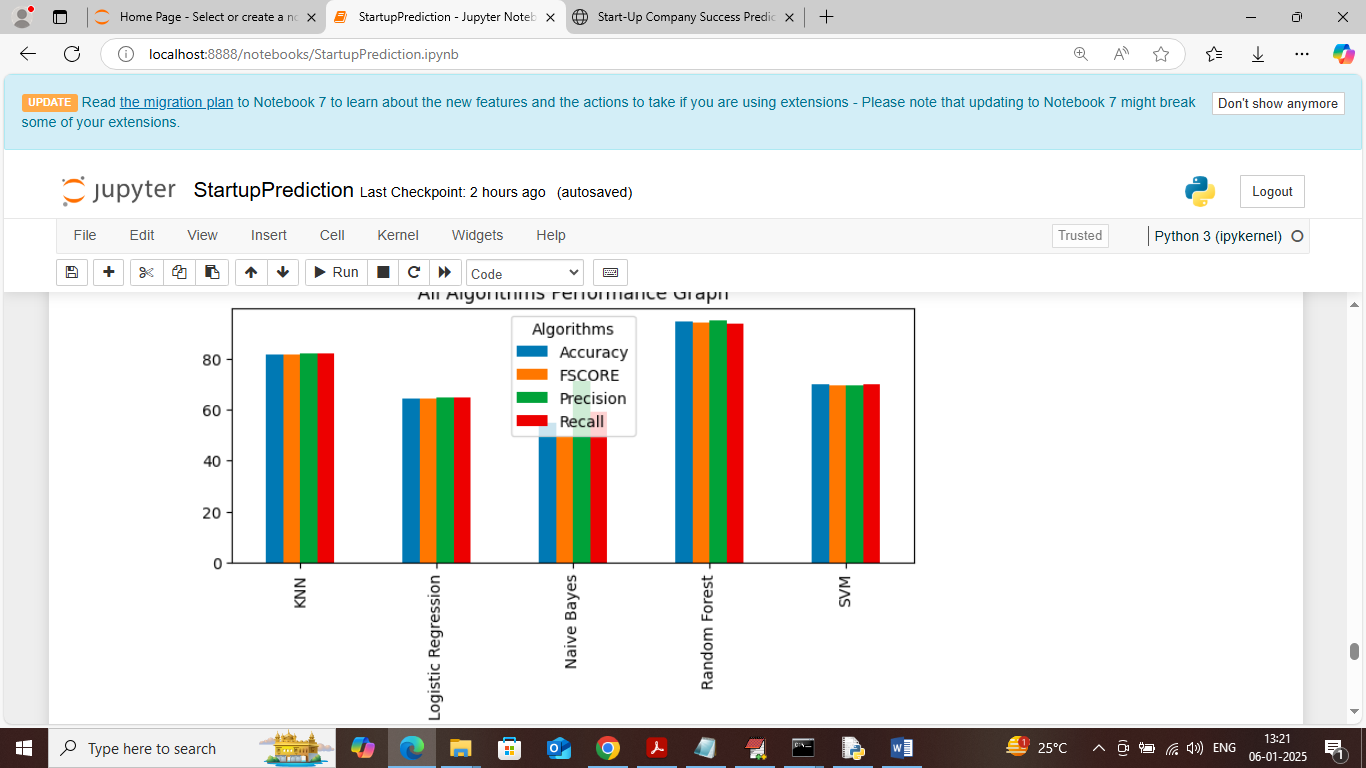
In above screen SVM got 69% accuracy and can see other metrics also



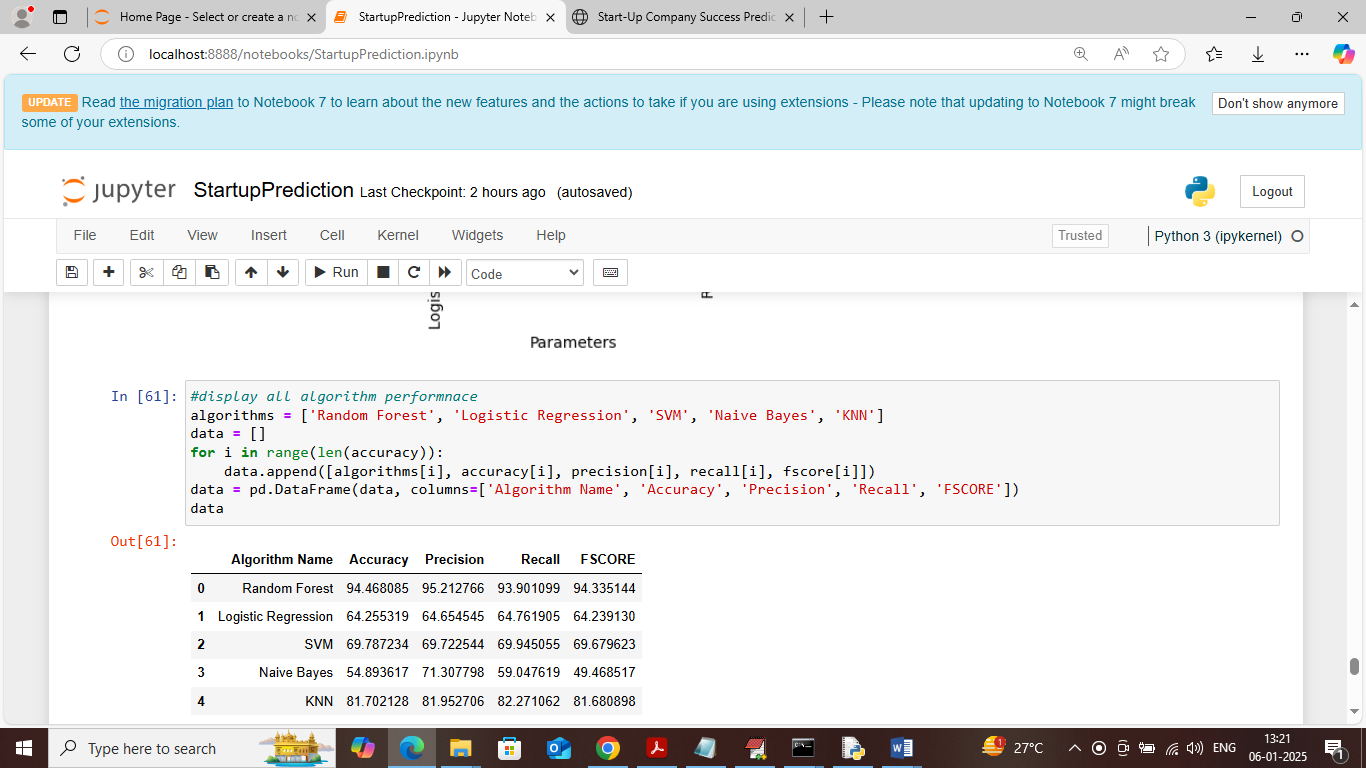
In above screen Naïve Bayes got 54% accuracy



In above screen KNN got 81% accuracy and can see other metrics output also

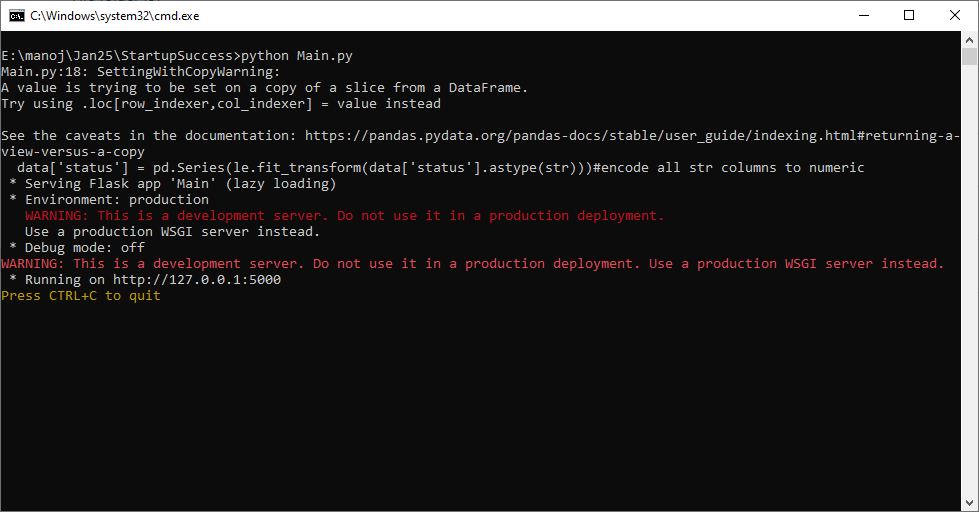


In above screen displaying all algorithms performance in graph format where x-axis represents algorithm names and y-axis represents accuracy and other metrics in different color bars and in all algorithms Random Forest got high performance

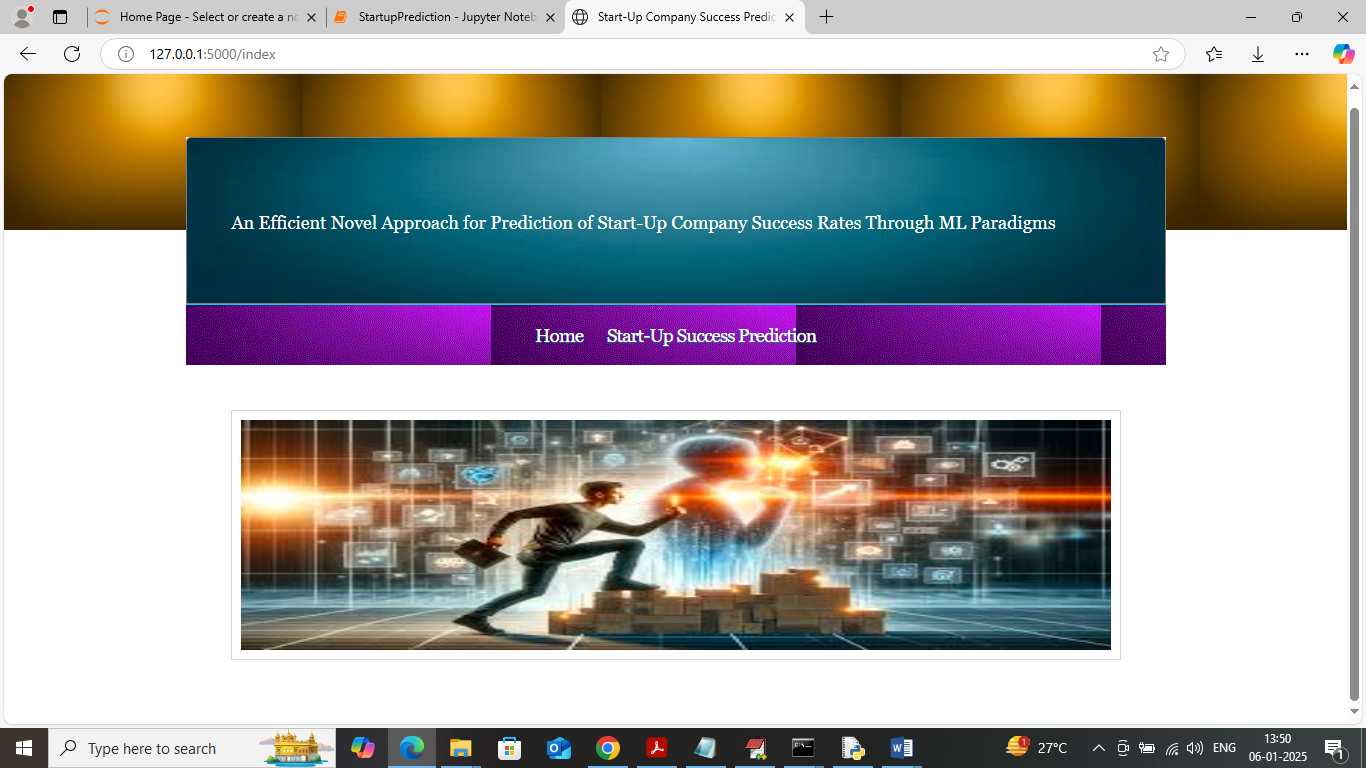


In above screen displaying all algorithm performance in tabular format

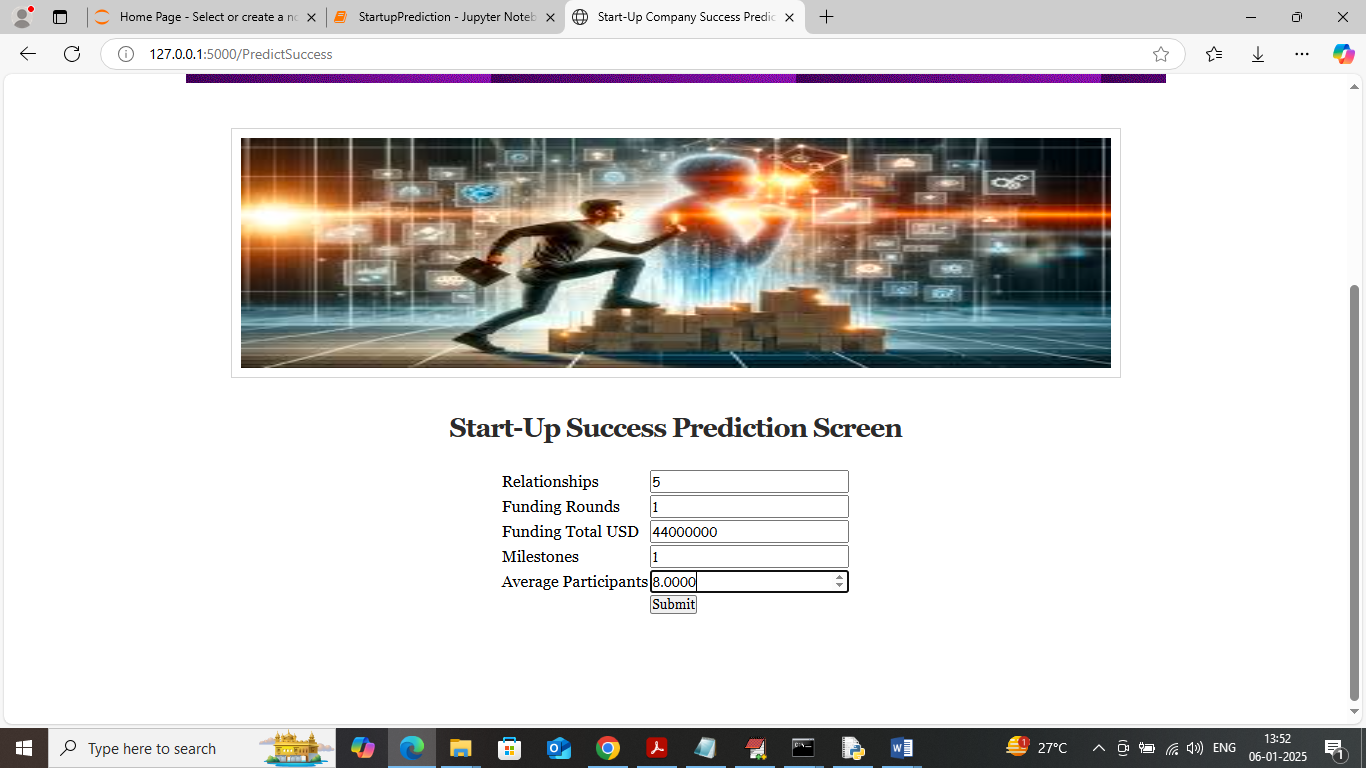
So in above screens we have seen training, testing and visualization output and now double click on ‘runFlaskServer.bat’ to start flask server and get below page



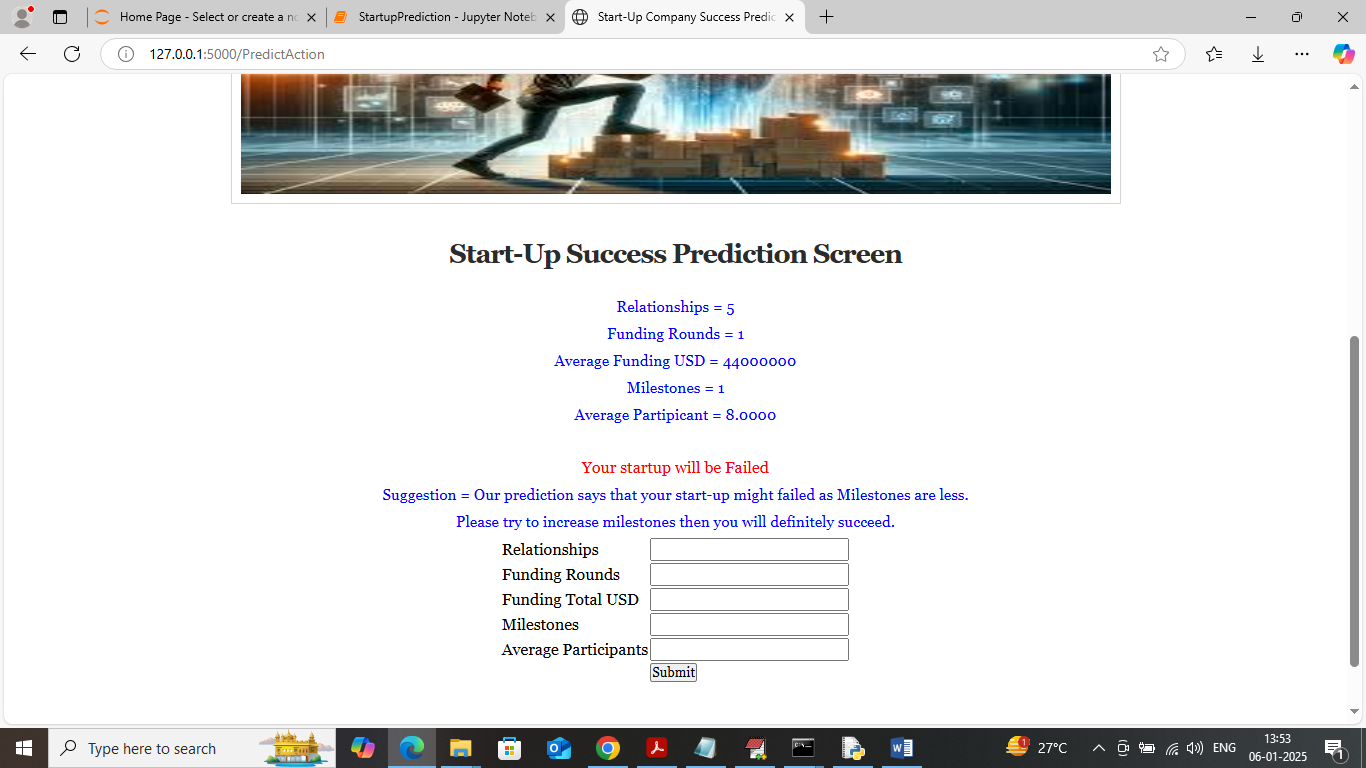
In above screen FLASK server started and now open browser and enter URL as <http://127.0.0.1:5000/index> and then press enter key to get below page



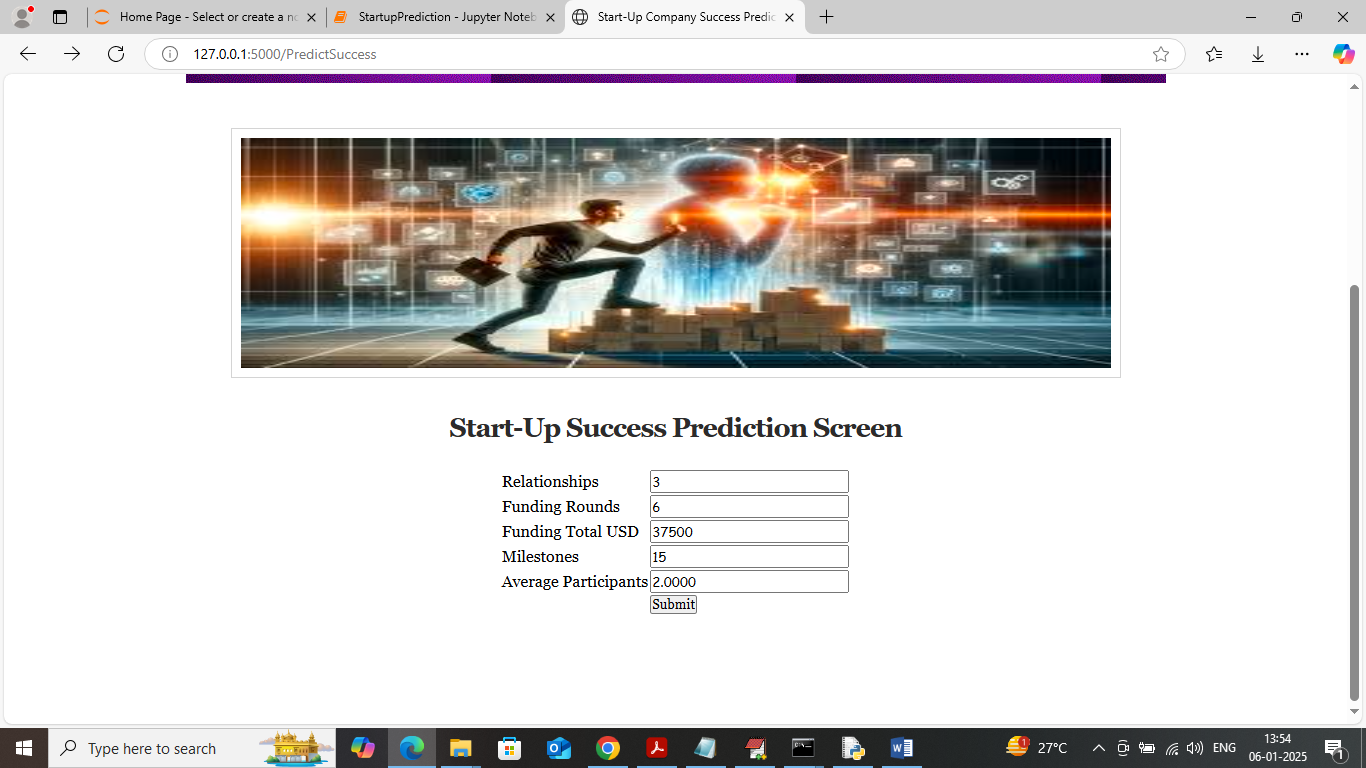
In above screen click on ‘Start-up Success Prediction’ link to get below page



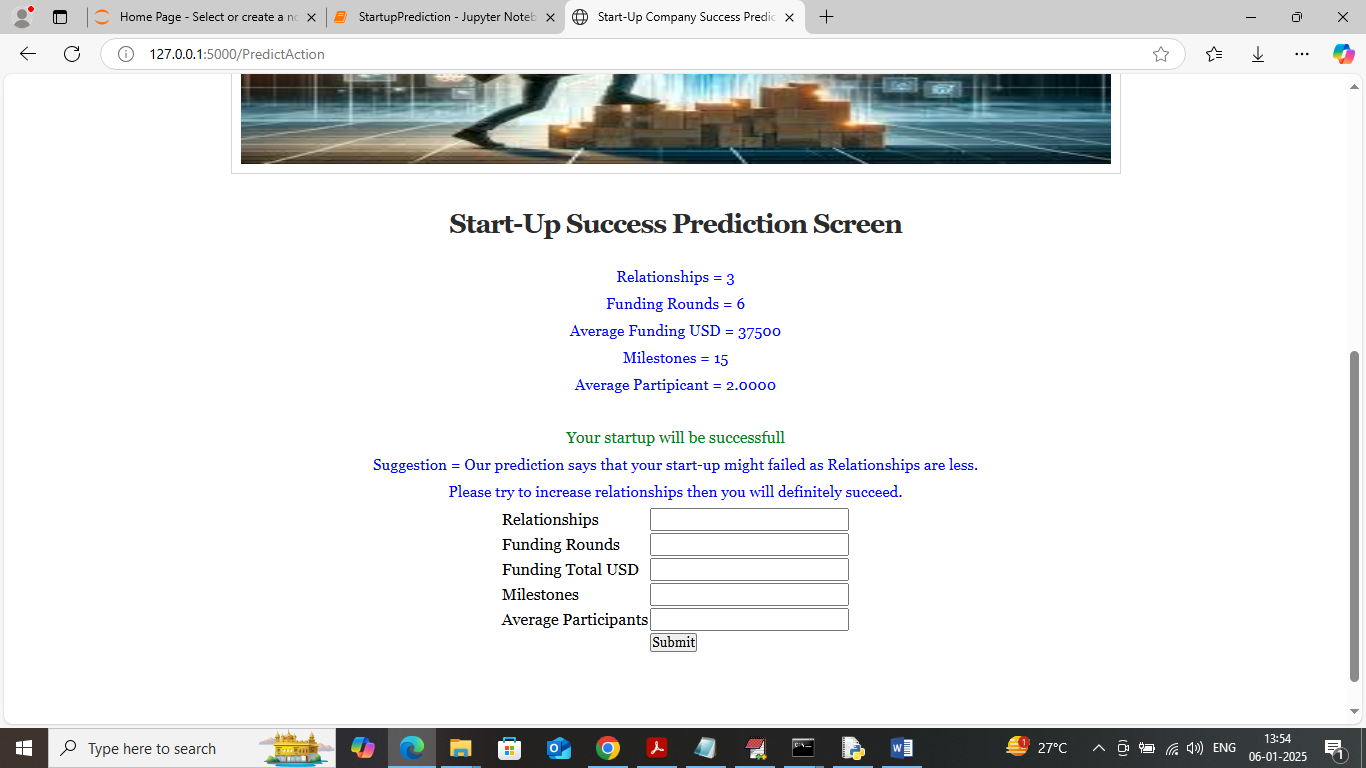
In above screen enter some input values and then click on button to get below prediction



In above screen in first 5 blue lines can see input values and then red text can see predicted output and then can see suggestion from the application and similarly you can enter input and get output. Below is another example



In above screen giving another example and below is the output



In above screen in green text can see ‘Start-up is successful’ and then can see some suggestion for further improvements.