

PROJECT REPORT ON

**“RATING PREDICTION”**

**SUBMITTED BY**

**Ankit Dadarwala**

**ACKNOWLEDGMENT**

I would like to express my special gratitude to “Flip Robo” team, who has given me this opportunity to deal with a beautiful dataset and it has helped me to improve my analyzation skills. And I want to express my huge gratitude to **Mr. Shubham Yadav** (SME Flip Robo), he is the person who has helped me in tickets to get out of all the difficulties I faced while doing the project. And also inspired me in so many aspects and also encouraged me a lot with his valuable words and with his unconditional support I have ended up with a beautiful Project.

A huge thanks to my academic team “Data trained” who are the reason behind what I am today. Last but not least my parents who have been my backbone in every step of my life. And also thank you for many other persons who has helped me directly or indirectly to complete the project.

**INTRODUCTION**

1.1 Business Problem Framing:

We have a client who has a website where people write different reviews for technical products. Now they are adding a new feature to their website i.e. The reviewer will have to add stars(rating) as well with the review. The rating is out 5 stars and it only has 5 options available 1 star, 2 stars, 3 stars, 4 stars, 5 stars. Now they want to predict ratings for the reviews which were written in the past and they don’t have a rating. So, we have to build an application which can predict the rating by seeing the review.

* 1. Conceptual Background of the Domain Problem

Multi-class classification originated from the investigation of text categorisation problem, where each document may belong to several predefined topics simultaneously.

For example, *multi-label classification* makes the assumption that each sample is assigned to one than more class: a fruit can be either an apple or a pear but not both at the same time. Whereas, an instance of *multi-class classification* can be that a text might be about any of religion, politics, finance or education at the same time or none of these

Our goal is to build a prototype for rating the of online electronic product based on review comments text given by customers.

1.3 Review of Literature

Now today people like do shopping from online by showing demo video of product, reviews from other customers and other financial benefits and easy payment options,

So product popularity on sale base on their performance and review given by other users for that review is an important role for product popularity base on review rating given by customers. In that we have to do rating 1,2,3,4,5 out of 5.

**Analytical Problem Framing**

2.1 Mathematical/ Analytical modelling of the Problem

In this particular problem I have multiclass categorical variables as my target column and it was having of the different class of rating like 1,2,3,4,5. So clearly it is a Multiclass Classification base problem and I have to use all classification algorithms while building the model. There were some null values in the dataset. To get better insight on the features I have used plotting like distribution plot, bar plot, Pie plot and Word cloud plot. Using the **TFIDF** vectorizer extract the 18,763 features from dataset. I have used all the linear regression and Tree based algorithms while building model then tuned the best model and saved the best model.

2.2 Data Sources and their formats

The data was collected from different e-commerce websites like amazon, flip kart by scraping using selenium tool of different products like phones, laptops, professional cameras, printers, routers, smart watches, monitors.It is hereby given to us for this exercise. In order to build model for online products review rating classification between 1 to 5 stars.

Scraping data stored in csv and excel file and that use for out data pre-processing, cleaning, and model building part.

Also, my dataset was having 35470 rows and 3 columns including target. In this target have multiclass and review comments have text data that mean NLP sentiment analysis problem. We have to classify base on word’s Verb, Noun, Adjective, Adverb etc...

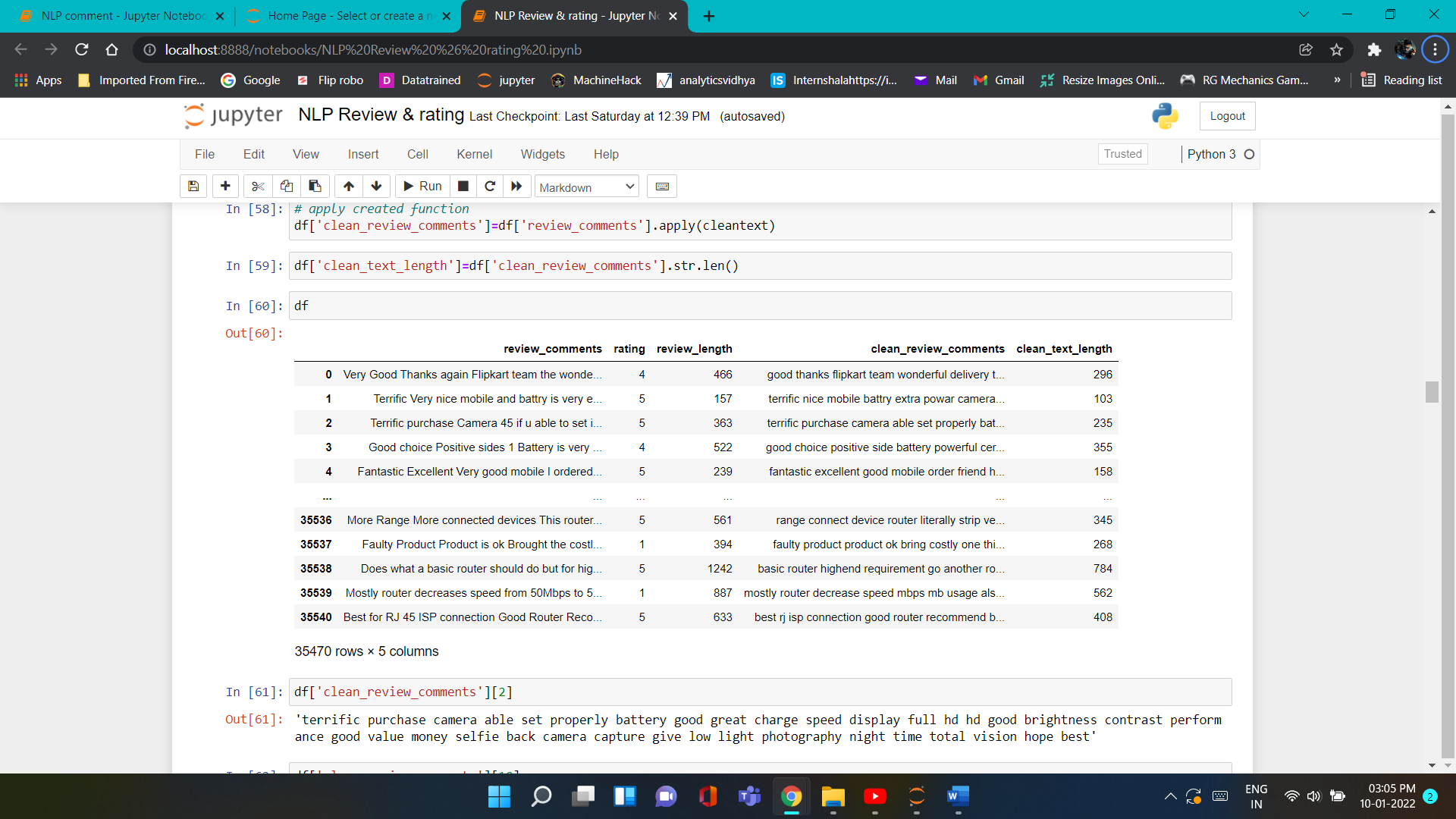
**Features Information:**

The data set includes

* review titel --> title of review
* review comments --> text of review written by users
* rating --> No.of rating out of 5

2.3 Data Pre-processing Done

1. As a first step I have imported required libraries and I have imported the dataset which was scraped and stored in csv or excel format.
2. Then I did all the analysis like checking shape, value counts, info, duplicates value null values etc.
3. I have also dropped “unnamed:0”, column as I found they are useless.
4. While checking for null values I found null values in the dataset.
5. I have merge review title and review comment columns
6. Make columns to check length of strings in particular row index comments.
7. Convert all text data into lower case.
8. Remove white space and special characters.
9. Remove html tags from data
10. Remove non-ascii words from text
11. Than make **CLAENTEXT** function to prepare text data, in that function remove string punctuation, tokenize comment in words, remove stop words like(I,me,u,your,have,had,etc…),than using POS TAG(part of speech) lemmatization the words.
12. Apply this function to review comments column.
13. Again, make column to check clean text length.
14. Using TFIDF Vectorize transform text to vectors and extract features for models



2.4 Data Inputs- Logic- Output Relationships

* Since I had one input column as text format and had clean the text from that extract features or vocabulary from that as input feature to train our models.
* Plotting some graph to see the distribution of clean and non-clean text of review comments.
* As Output we get multiclass classification from models by analysing the words and as output we get rating in between 1 to 5.

2.6 Hardware and Software Requirements and Tools Used

* Device name: HP Pavilion
* Processor: AMD Ryzen 5 3550H with Radeon Vega Mobile Gfx 2.10 GHz
* RAM: 8.00 GB
* System type: 64-bit operating system, x64-based processor
* Jupyter NoteBooks Version: 6.4.3
* Python3 vesrion: 3.8.8
* **Libraries** **required :-**
* import pandas as pd
* import numpy as np
* import matplotlib.pyplot as plt
* import seaborn as sns
* import warnings

warnings.filterwarnings('ignore')

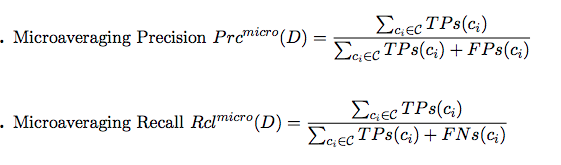
* import re # regular expression
* import string
* import nltk
* from nltk.corpus import stopwords # for stop words
* from nltk.stem import WordNetLemmatizer # for lemmatization
* from nltk.tokenize import word\_tokenize # for word tokenize
* from nltk import pos\_tag # for part of speech
* from wordcloud import WordCloud
* from nltk import FreqDist
* from sklearn.feature\_extraction.text import TfidfVectorizer
* from sklearn.naive\_bayes import MultinomialNB
* from sklearn.model\_selection import train\_test\_split,cross\_val\_score
* from sklearn.metrics import accuracy\_score,confusion\_matrix,classification\_report,roc\_curve,roc\_auc\_score,f1\_score,auc,hamming\_loss
* from sklearn.linear\_model import LogisticRegression
* from sklearn.tree import DecisionTreeClassifier
* from sklearn.ensemble import RandomForestClassifier,
* from sklearn.neighbors import KNeighborsClassifiers
* from sklearn.linear\_model import SGDClassifier
* AdaBoostClassifier, Bagging Classifier
* from xgboost import XGBClassifier
* from yellowbrick.text import FreqDistVisualizer
* import pickle

**Data Analysis and Visualization**

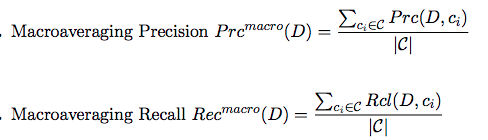
* 1. Testing of Identified Approaches (Algorithms)

Since multi-class was my target and it was a classifiaction , so this particular problem was Classfication problem. And I have used all classifier algorithms to build my model. By looking into the difference of Accuracy score and F1-score micro or weighted and also hamming loss. To get the best model we have to run through multiple models and to avoid the confusion of overfitting we select best random state. Below are the list of Classifier algorithms I have used in my project.

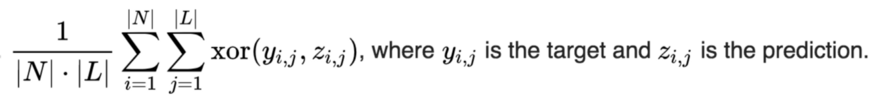
1. Logistic Regression
2. MultinomialNB
3. KNeighborsClassifier
4. SGDClassifier
5. XGB Classifier
6. Decision Tree Classifer
7. Random Forest Classifier
8. Bagging Classifier
9. AdaBoost Classifier
   1. Key Metrics for success in solving problem under consideration
10. **Micro-averaging (Label based measures):**To measure a multi-class classifier we have to average out the classes somehow. There are two different methods of doing this called micro-averaging and macro-averaging.in micro-averaging all TPs, TNs, FPs and FNs for each class are summed up and then the average is taken.

****

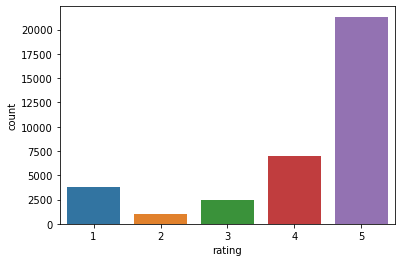
1. **Macro-averaging:** Macro-averaging method can be used when you want to know how the system performs overall across the sets of data. You should not come up with any specific decision with this average. On the other hand, micro-averaging can be a useful measure when your dataset varies in size.



1. **Hamming loss (Example base measure):**In simplest of terms, Hamming-Loss is the fraction of labels that are incorrectly predicted, i.e., the fraction of the wrong labels to the total number of labels.

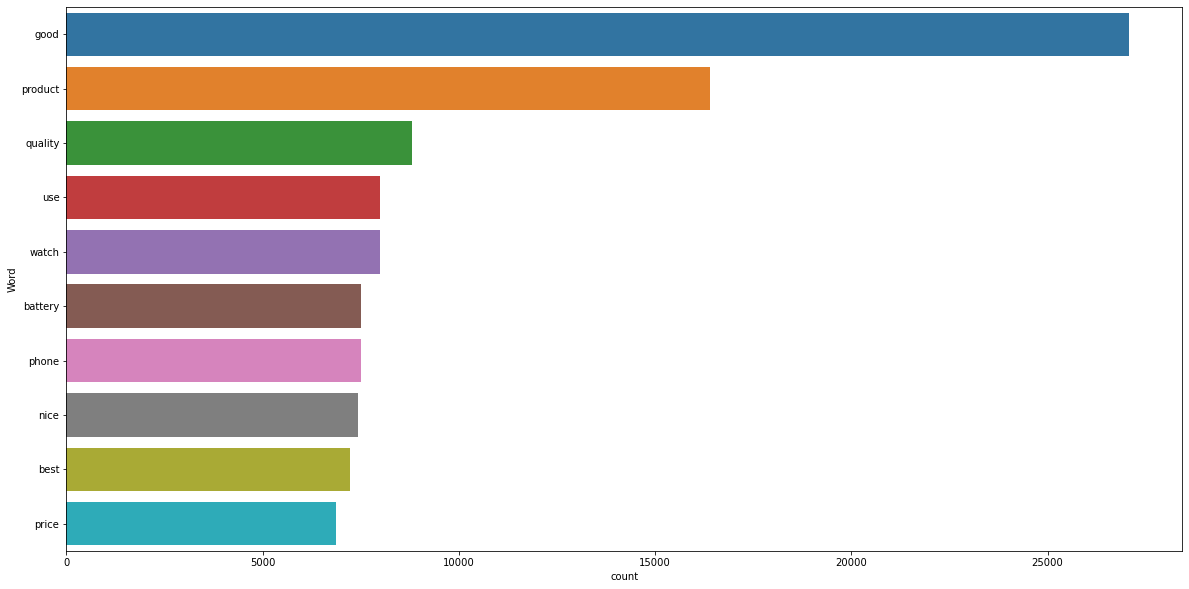
****

* 1. Visualizations



As show in fig pie chart of label counts for rating of review comments most of comments rate as 5 star rating.

* **Most frequent used words in positive manner**

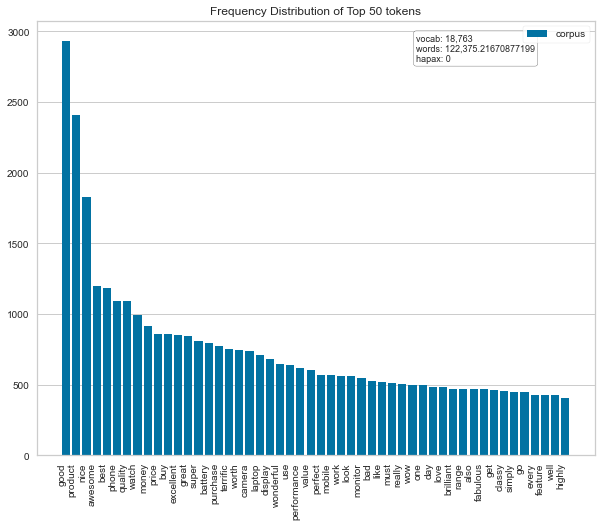


## Word cloud of rating from 1 to 5 star

## 

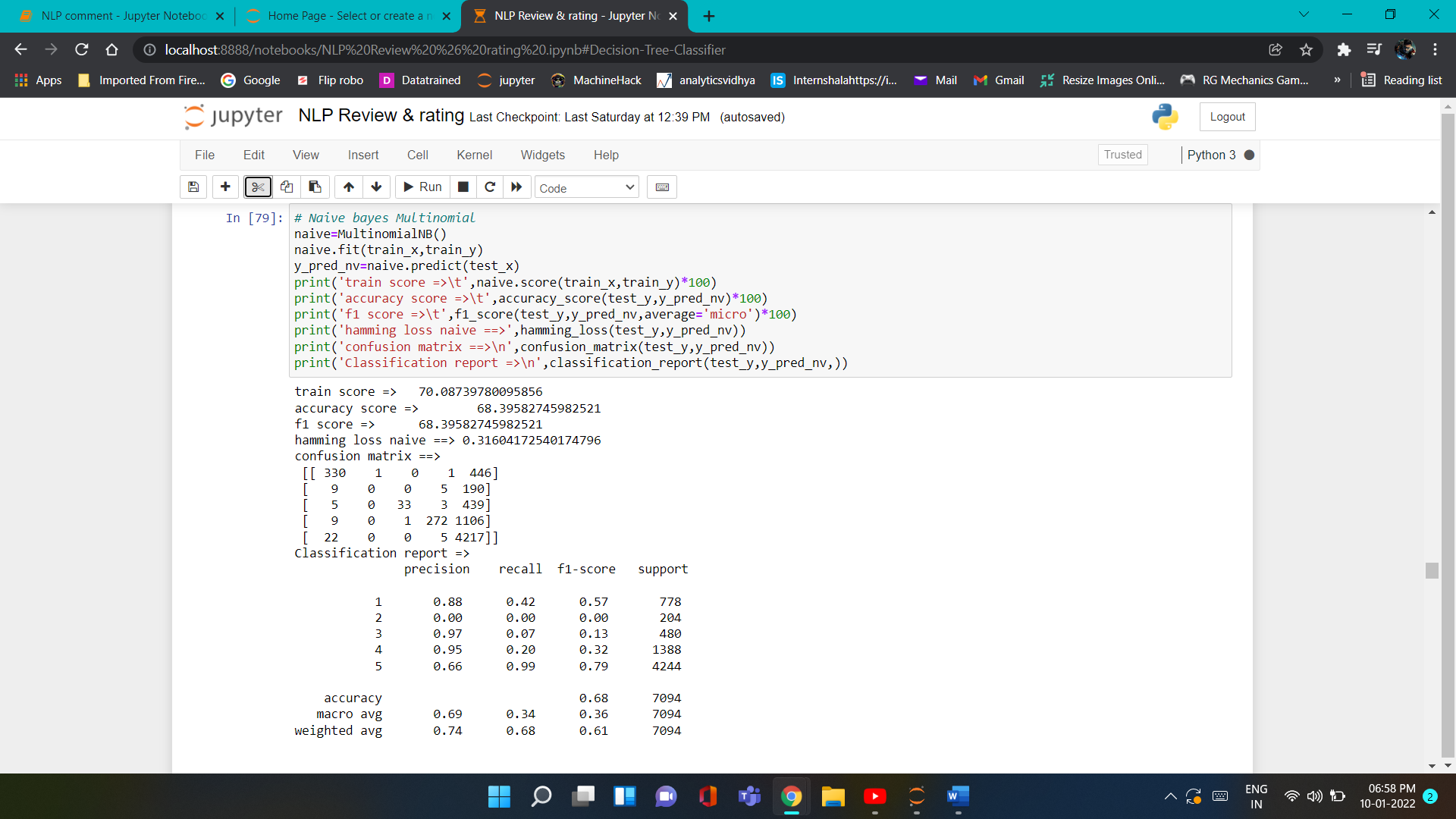
## 

**Frequency distribution of top 50 token words**

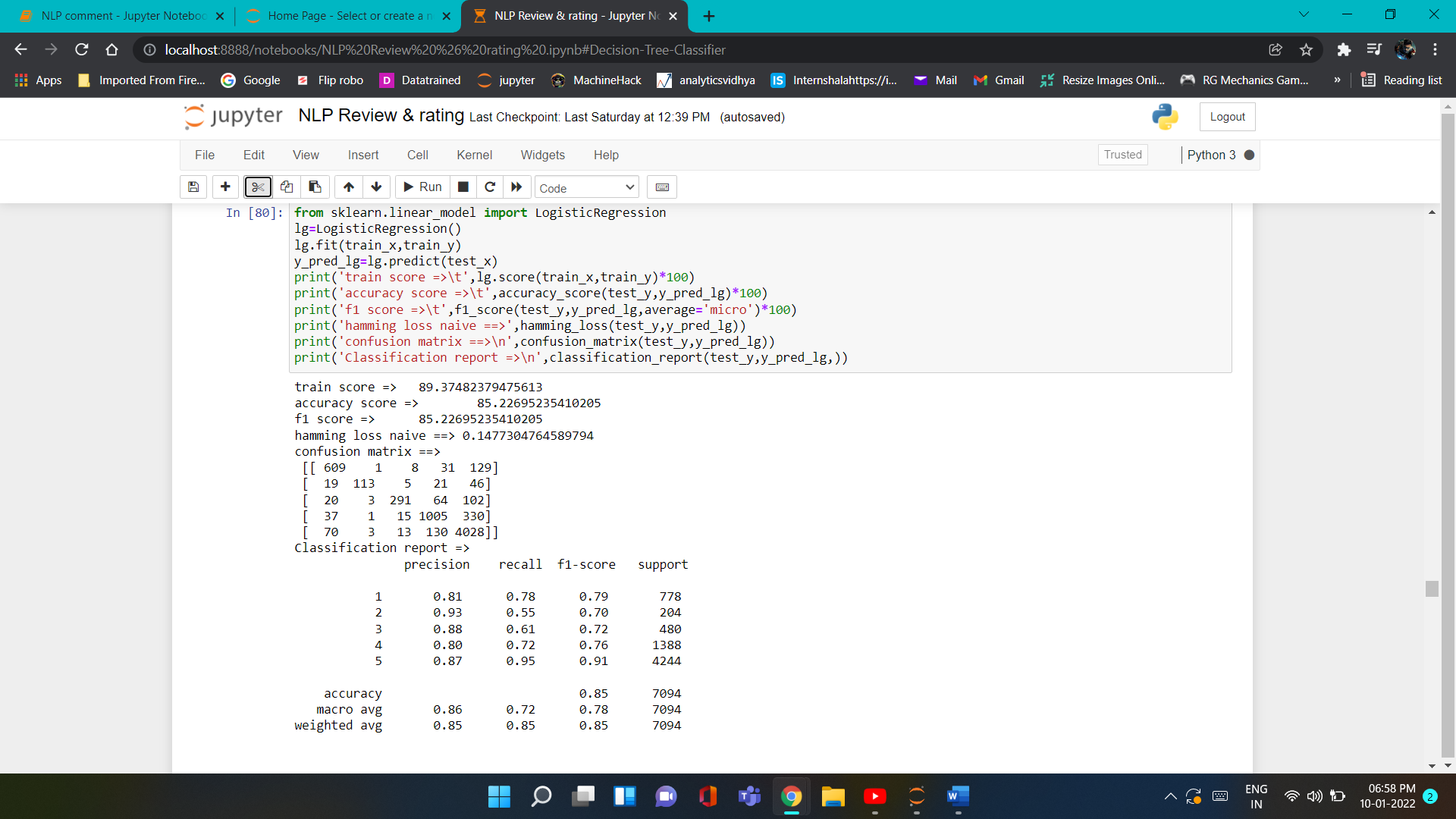
****

* 1. Run and evaluate select models

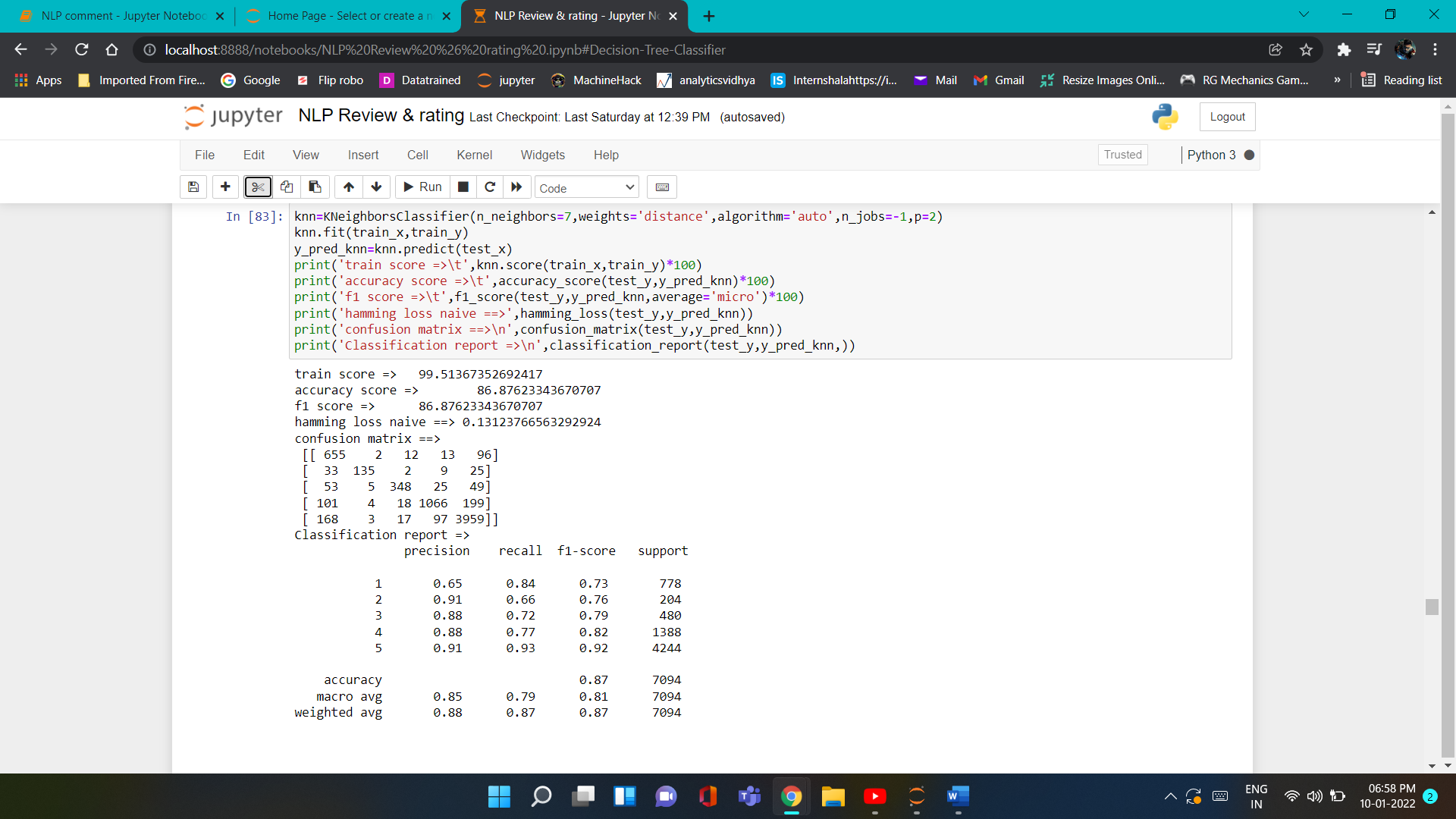
**1). MultinomialNB:**

****

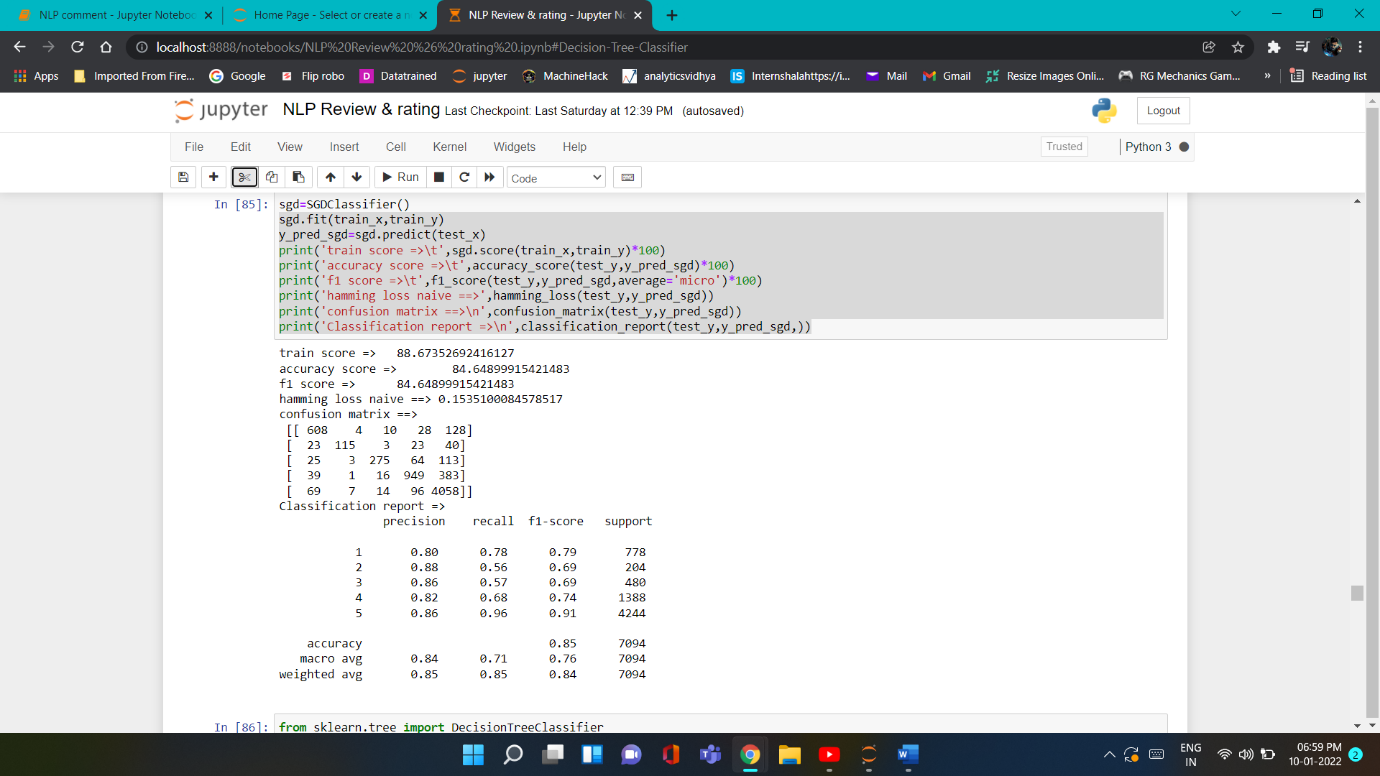
**2). Logistic Regression:**

****

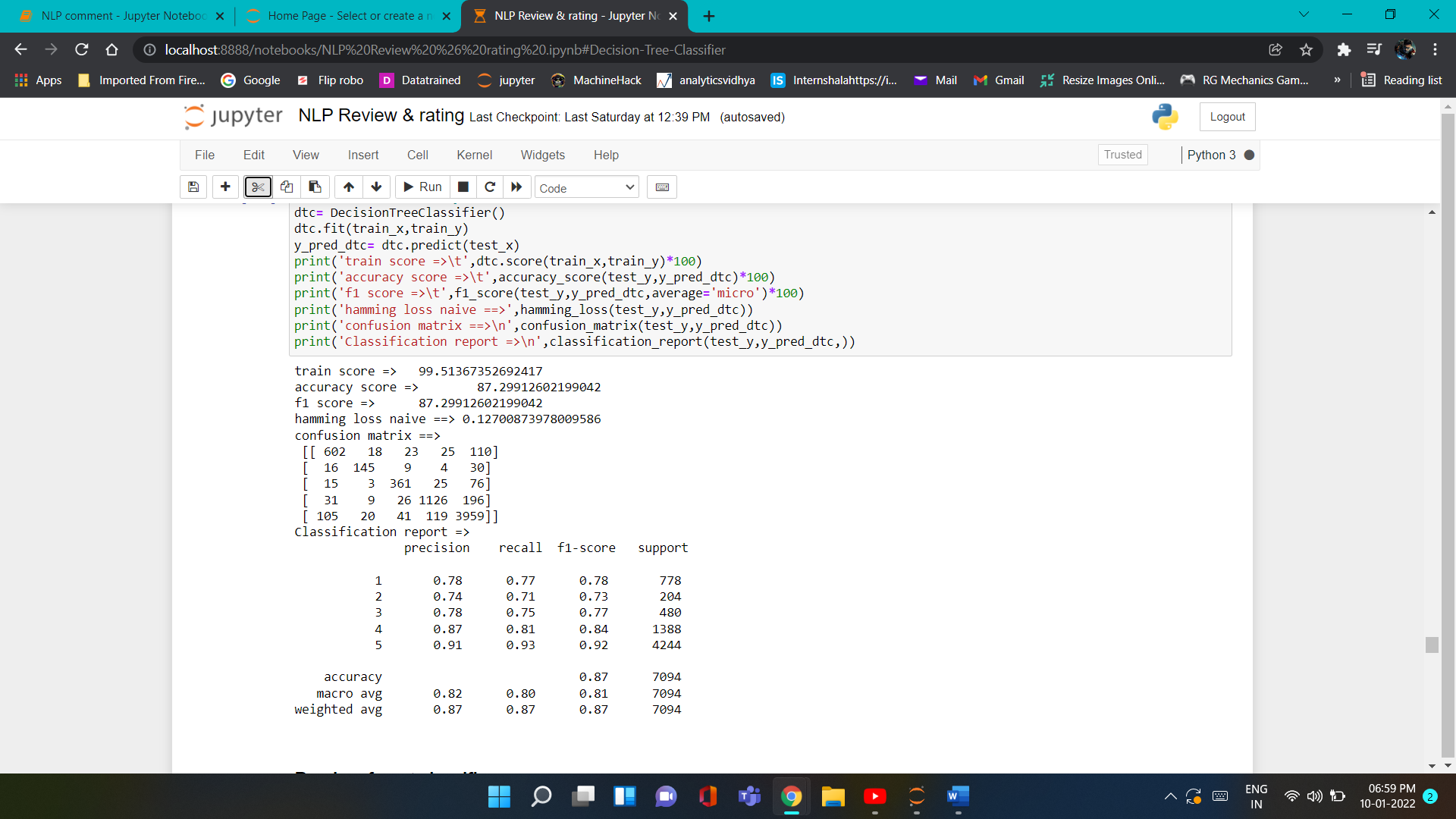
**3). Kneighbors classifier:**

****

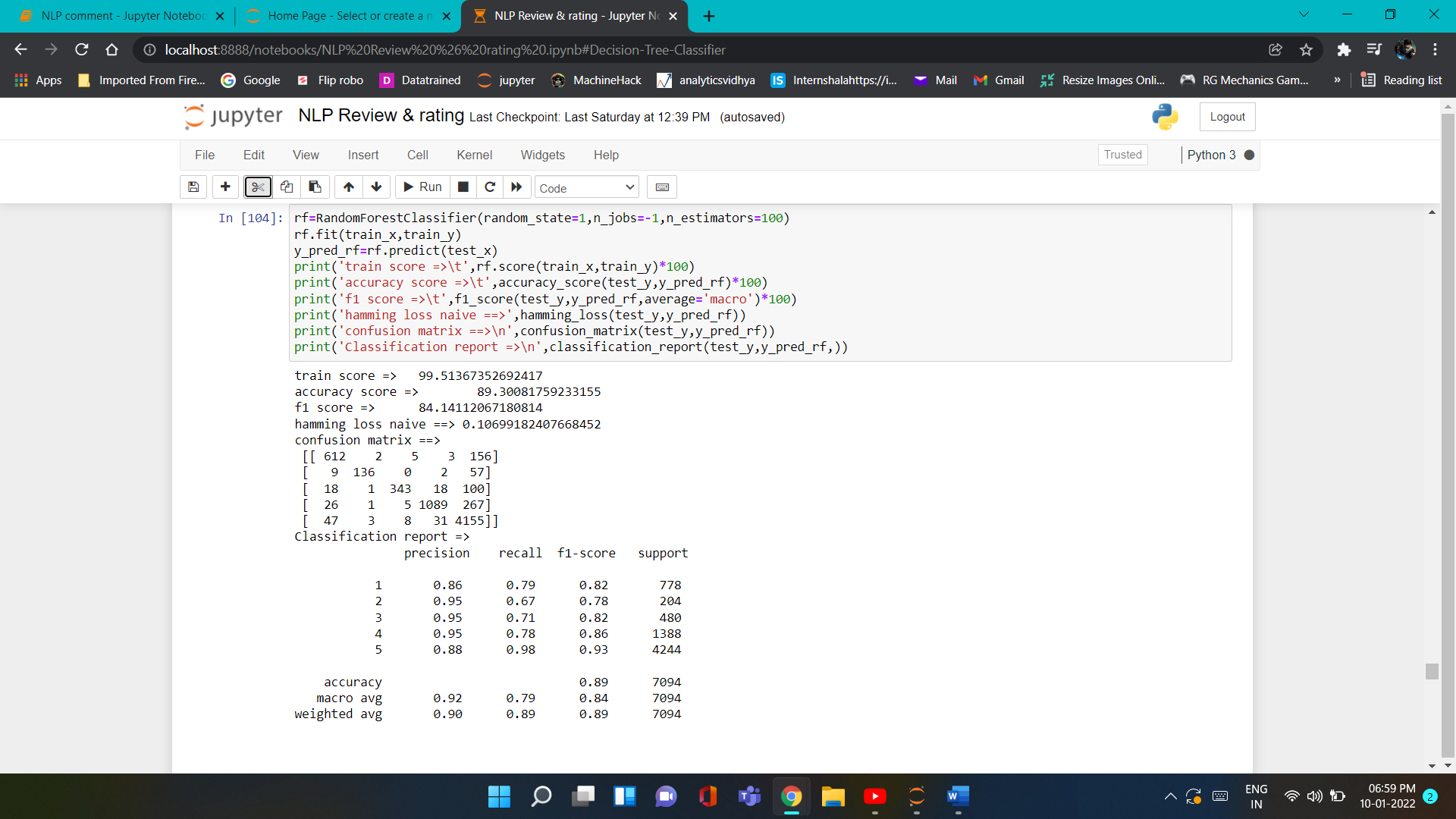
**4). SGD Classifier:**

****

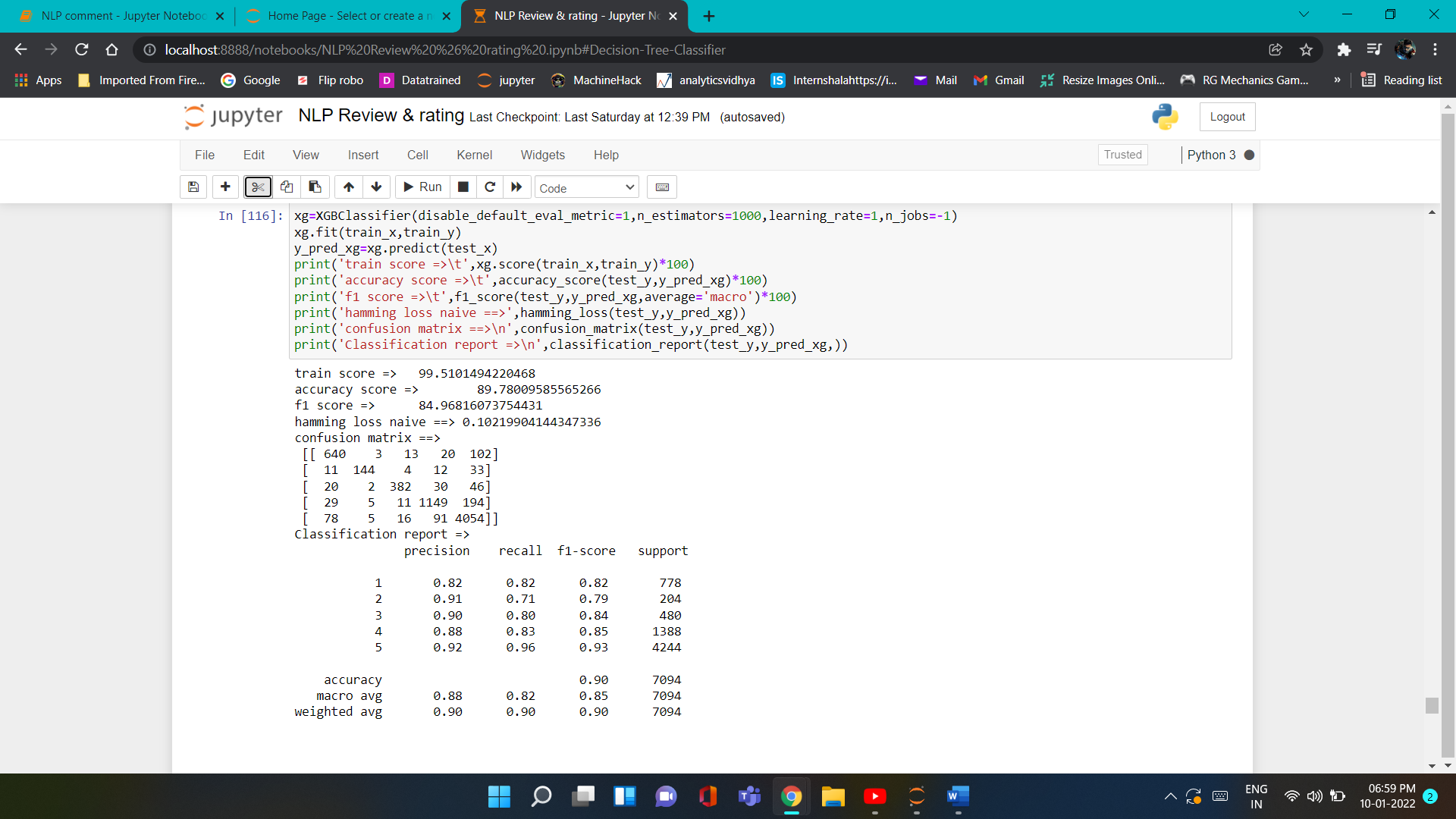
**5). Decision Tree Classifier:**

****

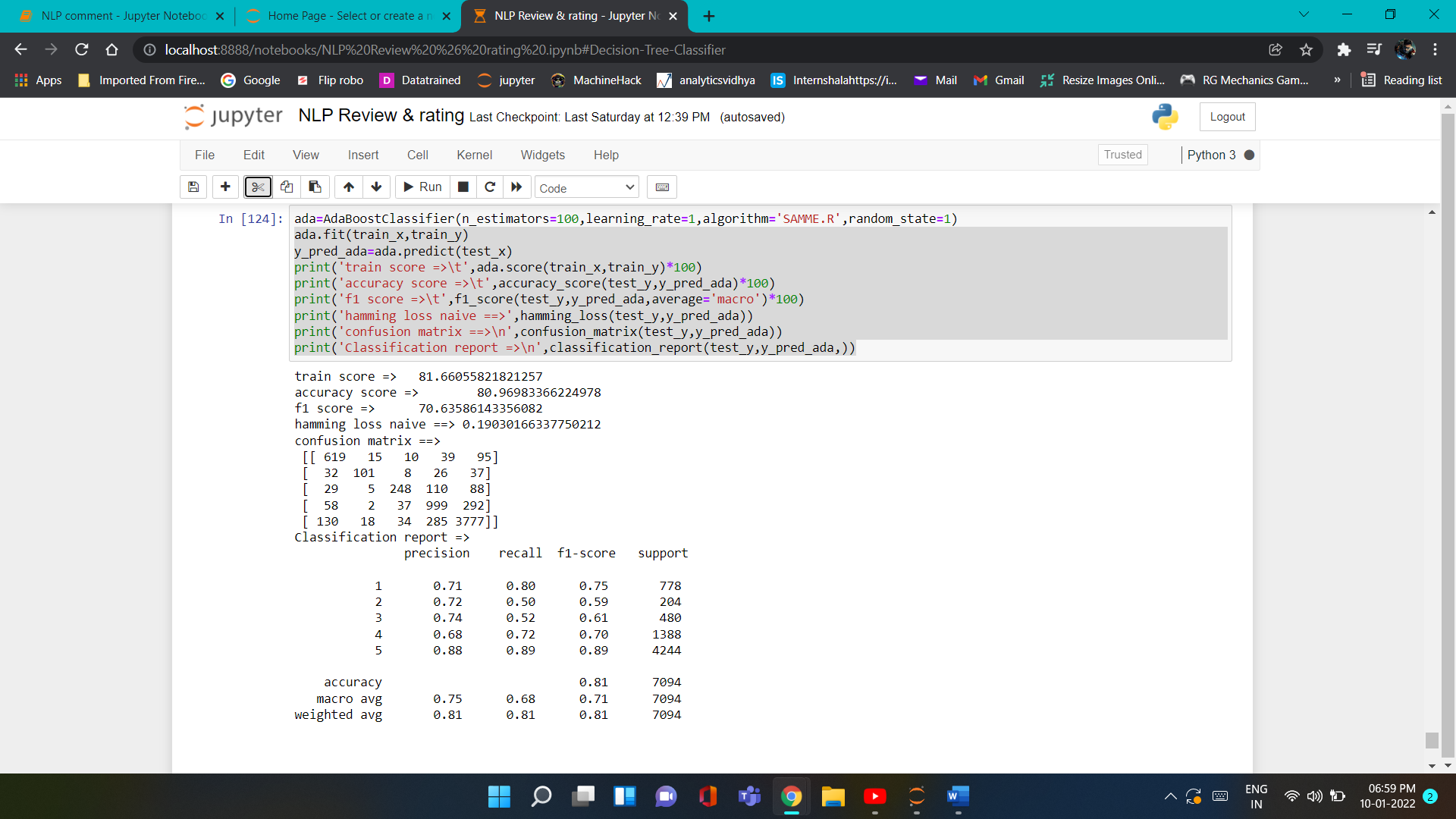
**6). Random Forest Classifier:**

****

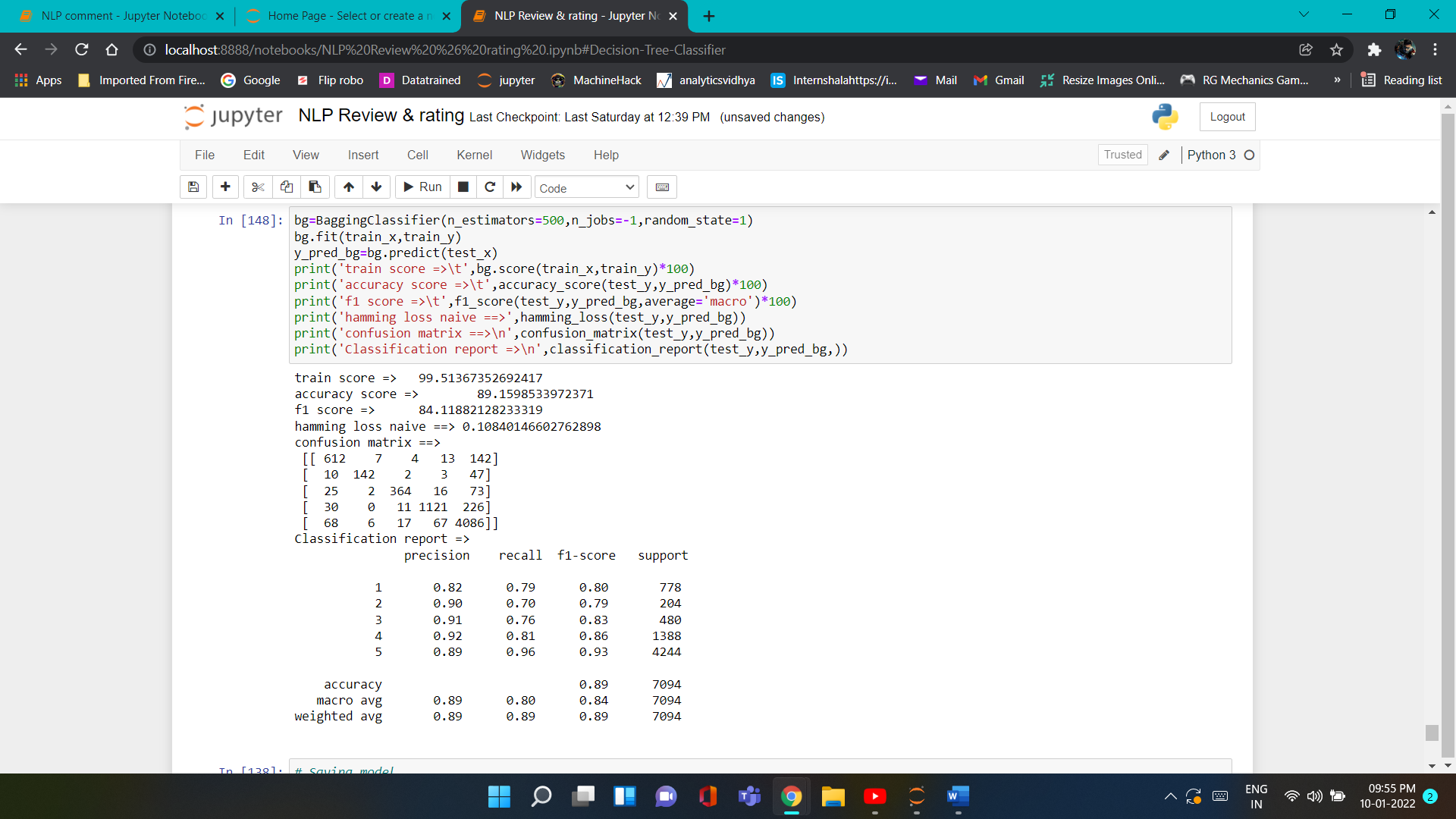
**7). XGBoost Classifier:**

****

**8). AdaBoost Classifier:**

****

**9). Bagging Classifier:**

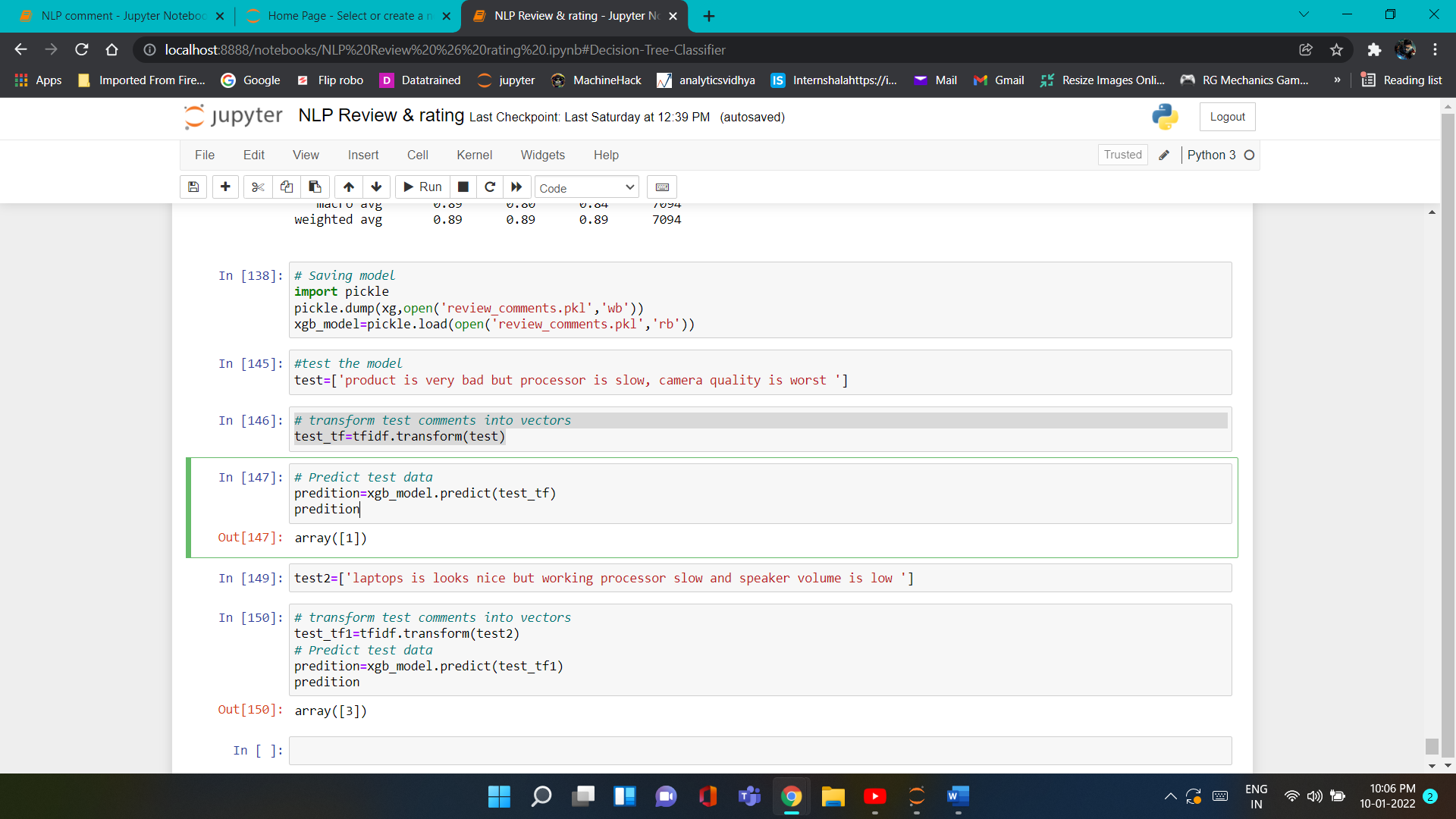
****

**9). Model Dashboard:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr.NO** | **Model name** | **Hamming loss** | **Accuracy score %** | **F1\_score Macro %** |
| 1 | MultinomialNB | 0.316 | 68.39 | 68.39 |
| 2 | Logistic Regression | 0.147 | 85.26 | 85.26 |
| 3 | KNeighbors Classifier | 0.131 | 86.87 | 86.87 |
| 4 | SGD Classifier | 0.153 | 84.64 | 84.64 |
| 5 | Decision Tree Classifier | 0.127 | 87.29 | 87.29 |
| 6 | Random forest Classifier | 0.106 | 89.30 | 84.14 |
| 7 | XGBoost Classifier | 0.102 | 89.78 | 84.96 |
| 8 | Adaboost Classifier | 0.190 | 80.96 | 70.63 |
| 9 | Bagging Classifier | 0.108 | 89.15 | 84.11 |

**10). Saving Model for Prediction**

=> I have saved my best model using .pkl as follows**. We save XGBoost Classifier with tunning parameters.**

****

* 1. Interpretation of the Results
* Firstly, the datasets were Null values.
* And proper plotting for proper type of features will help us to get better insight on the data.
* It is multi-class classification problem
* And we have to use multiple metrics like Accuracy score, F1-macro score, Hamming loss, which will help us to decide the best model.
* I found XGBoost classifier as the best model with **89.78% Accuracy score, hamming loss of 0.102.**
* At last I have predicted **Test data set** using saved model. It was good!! that I was able to get proper classification result.

**CONCLUSION**

4.1 Key Findings and Conclusions of the Study

In this project report, First we have to scrape data from different websites is very useful to get live data from sources and then clean the data and pre-process and do some EDA, we have used machine learning algorithms to predict the multi-class. We have mentioned the step-by-step procedure to analyse the dataset and feature extraction. These feature set were then given as an input to algorithms. Hence, we calculated the performance of each model using different performance metrics and compared them based on these metrics. Then we have also saved the best model and predicted the label.

4.2 Limitations of this work and Scope for Future Work

* The same problem can be solved using deep learning.
* For that our model train on small data set so we no sure for accuracy of model perform well on different reviews.
* This model can further improved by additional algorithms.