

**Comment’s Rating Classifier**

Submitted by:

Sahib Hussain

**ACKNOWLEDGMENT**

This project would not have been possible without the support of many people. Many thanks to my SME Keshav Bansal who gave suggestion how to proceed in building this model and helped make some sense of the confusion. Also thanks to my Internship Company FlipRobo Technologies Pvt. Ltd for giving me this opportunity to work on such a project which can help the industry to grow and achieve success.

**INTRODUCTION**

* Business Problem Framing

Comments and Ratings given by a purchaser can help a company lot in analysing the behaviour of customers towards the product, it can help us to find out the affects on its sale and what all can be done to modify and predict also the future of a product. This helps a company is making proper decision to control loss and improve it’s sale . In this project we would be seeing on the basis of comments how much rating the customer might have given which will help the company further in decision making.

* Motivation for the Problem Undertaken

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

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem

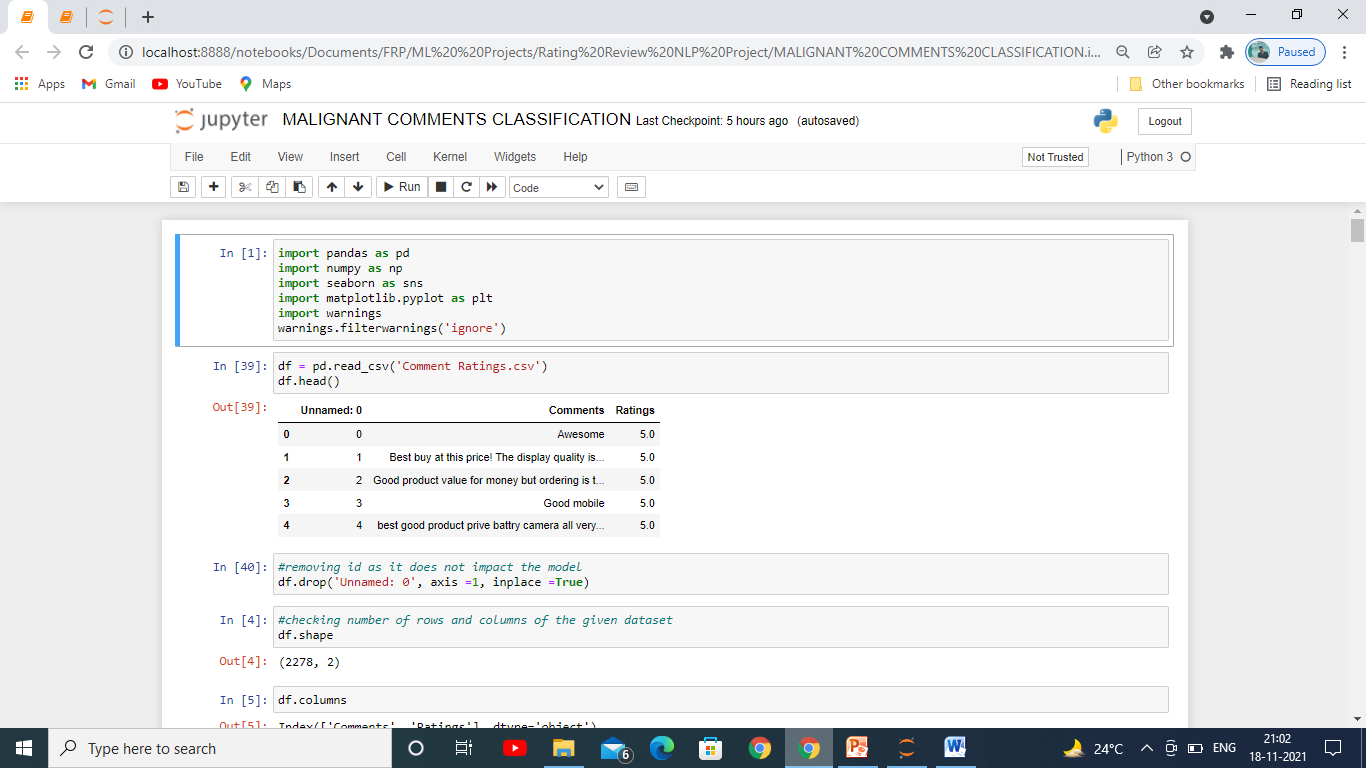
The data set contains the training set, which has approximately 2278 samples of comments along with their ratings in it.

By using shape we saw the Number of rows and columns in the dataset.

To make this model we used Naïve Bayes Classifier, Passive Aggressive Classifier, Logistic Regression, Decision Tree Classifier, Random Forest Classifier and XGB Classifier algorithm and used the best one out of these by evaluating the necessary metrics.

* Data Sources and their formats

The data was collected from Amazon ecommerce website using selenium to build this model.



Above figure shows us that there are 3 columns in it out of which Unnamed:0 is an unwanted column.

It has one independent Variable only which is our collected comments from e-commerce website Amazon and ratings is our dependent variable.

* Data Preprocessing Done

To clean the data we had to delete id as it would had matter in getting the predictions.

Then each of the comments where tokenized into words for better model building.

After tokenization, Lemmatization was performed on these words for each and every comment. It is the process of grouping together the different inflected forms of a word so that they can be analysed as a single item.

Also stop words technique was used to clear out the words which does not add any meaning to a sentence for building this model. They can safely be ignored without sacrificing the meaning of the sentence. For example, the words like the, he, have etc.

On these sentence we then performed **re** known as regular expression. It specifies a set of strings that matches it; the functions in this module let you check if a particular string matches a given regular expression (or if a given regular expression matches a particular string, which comes down to the same thing).

Finally at the end, we used **TfidVectorizer** for every comments which transforms text to feature vectors that can be used as input to estimator.

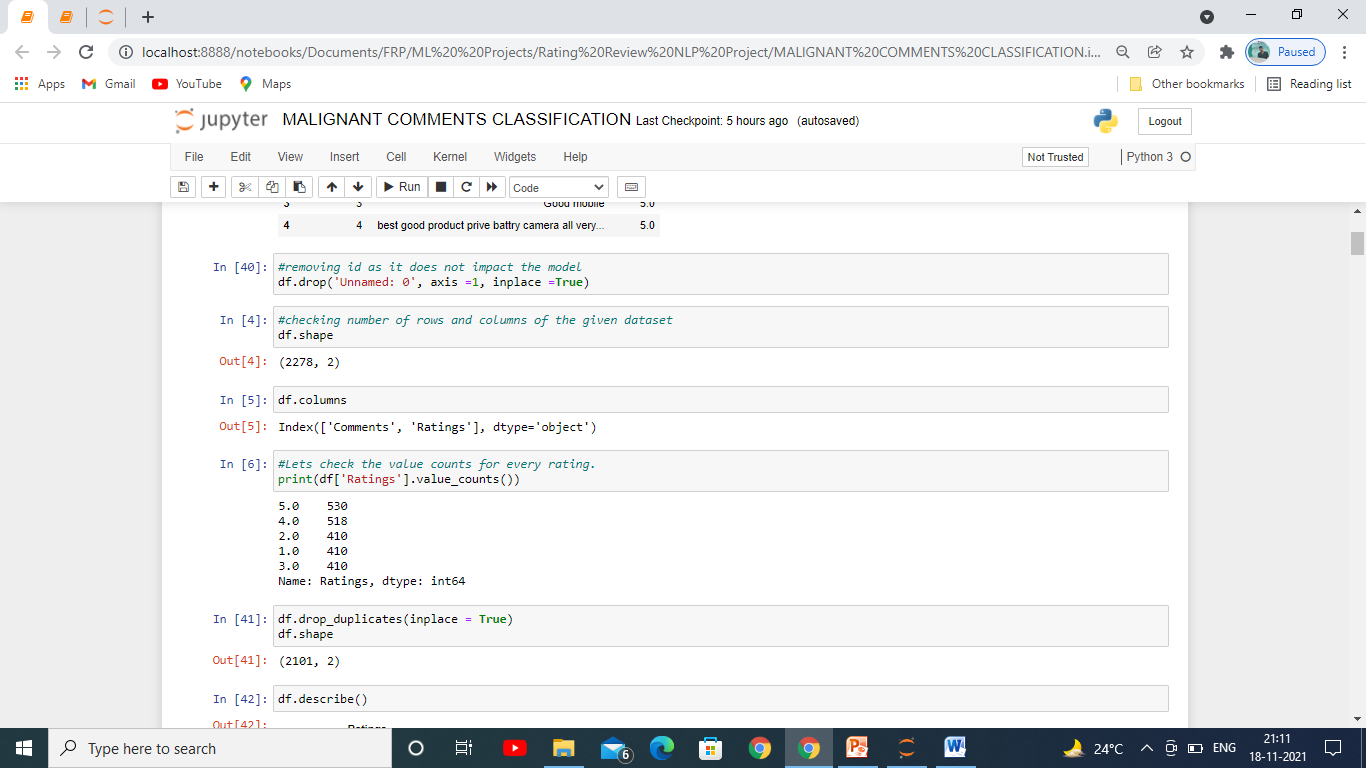
* Hardware and Software Requirements and Tools Used

**Software used** : In this we used Jupyter Notebook from the Anaconda to build the model, Miscrosoft powerpoint to make the powerpoint presentation and Miscrosoft word for documentation.

**Libraries used**: Pandas to read the dataset, re, stopwords, NLTK, word tokenizer, lemmatizer, tfidvectorizer, Matplotlib and Seaborn to analyse the data and clean the data and Scikit-Learn to build our models.

**Model/s Development and Evaluation**

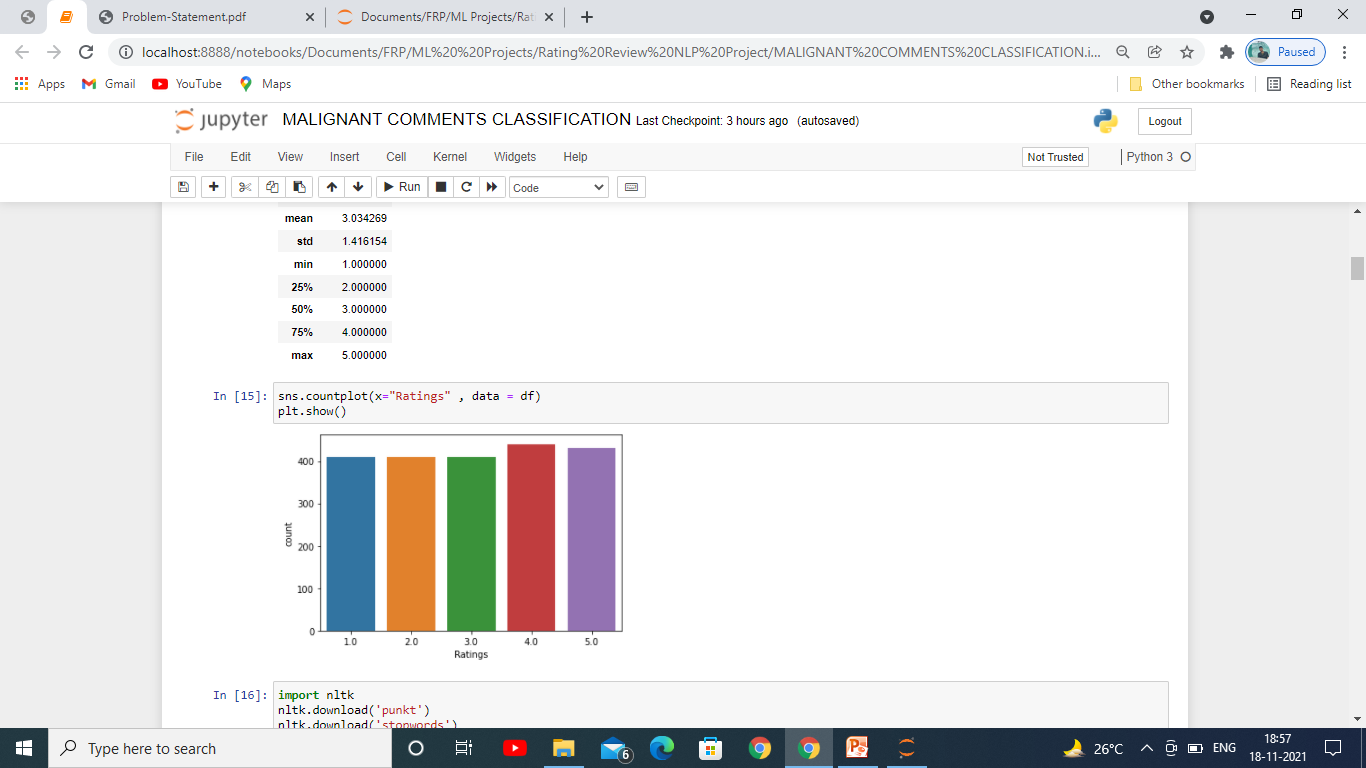
* Identification of possible problem-solving approaches (methods)



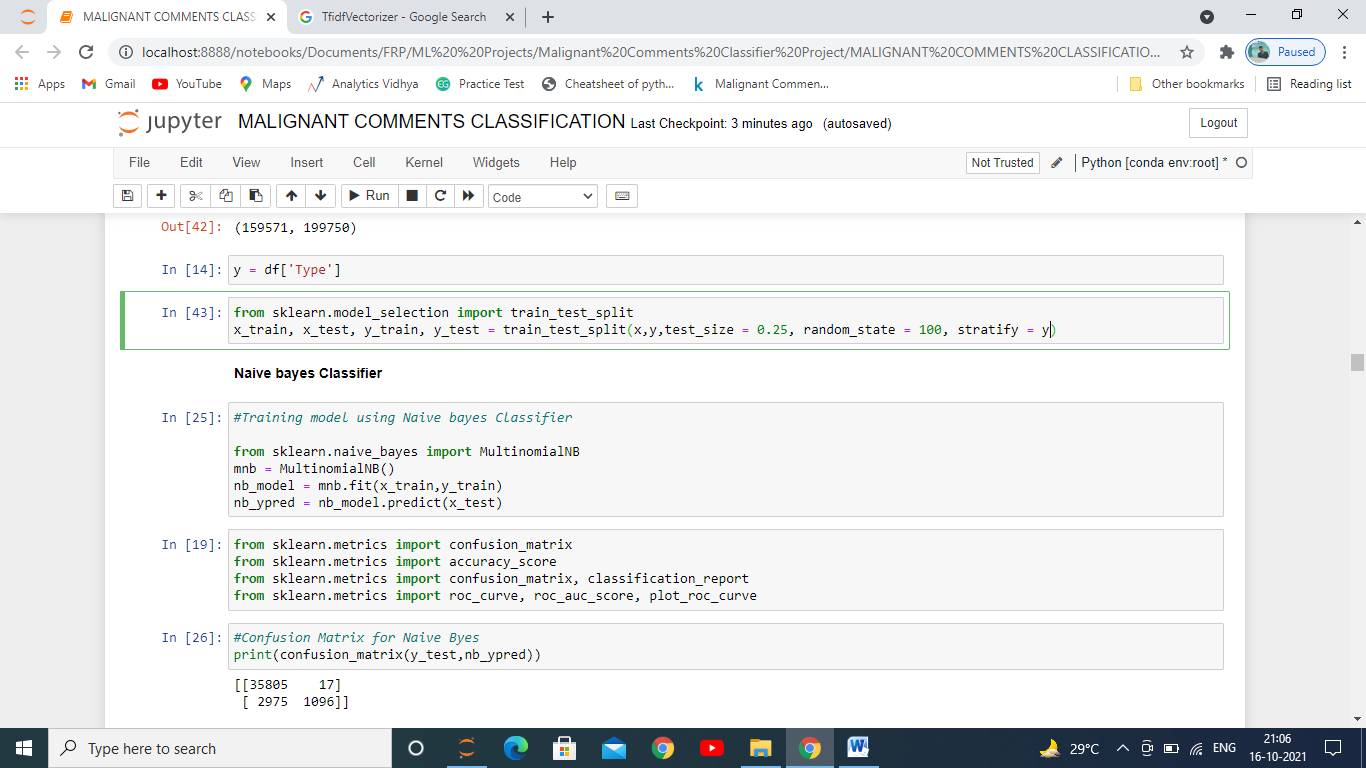
As we can see that the dataset was huge there were 2278 rows and 2 columns. 1 of the 2 columns was dependent one which is our Ratings.

The comments were than pre-processed and clean to build a better model using the steps mentioned earlier.

Also we saw that there was hardly any imbalance in the dependent variable.

****

Testing of Identified Approaches (Algorithms)

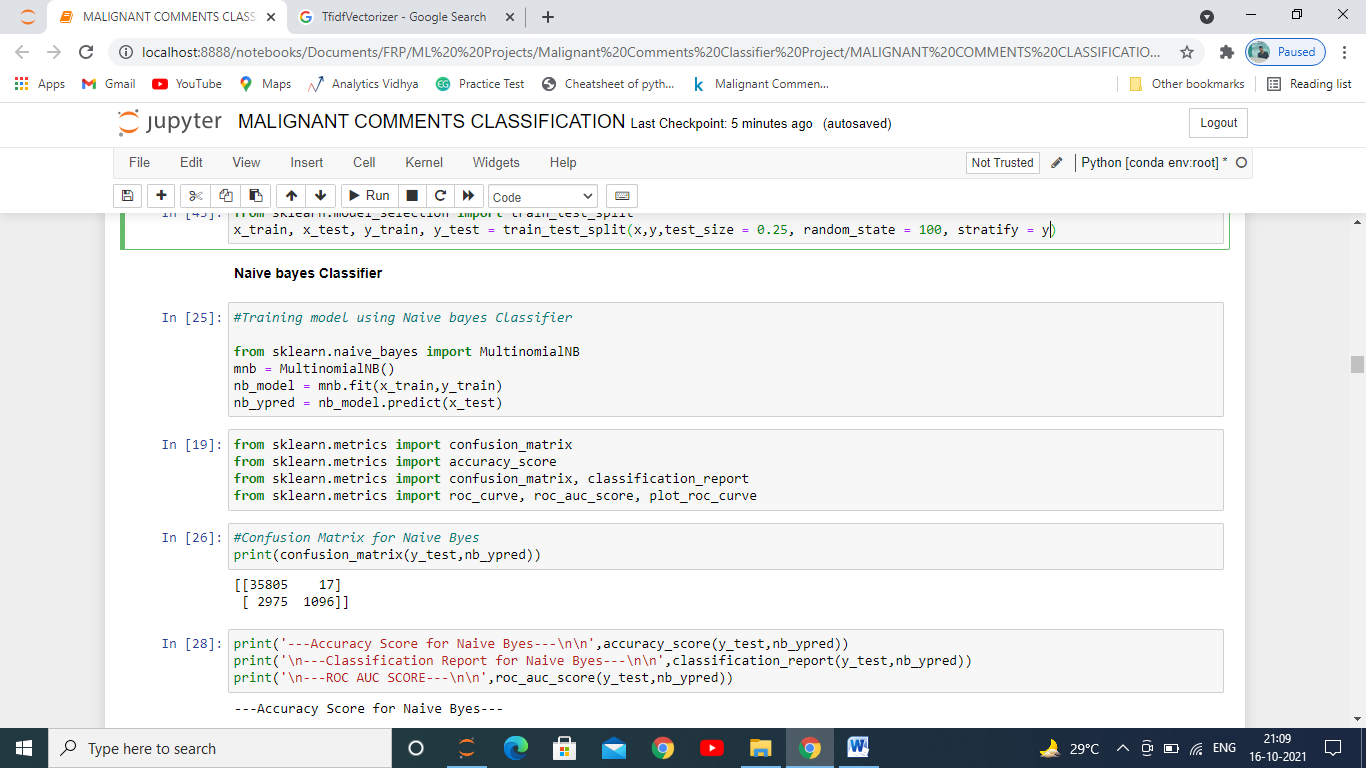


We first imported the train test split as you can see in the above image.

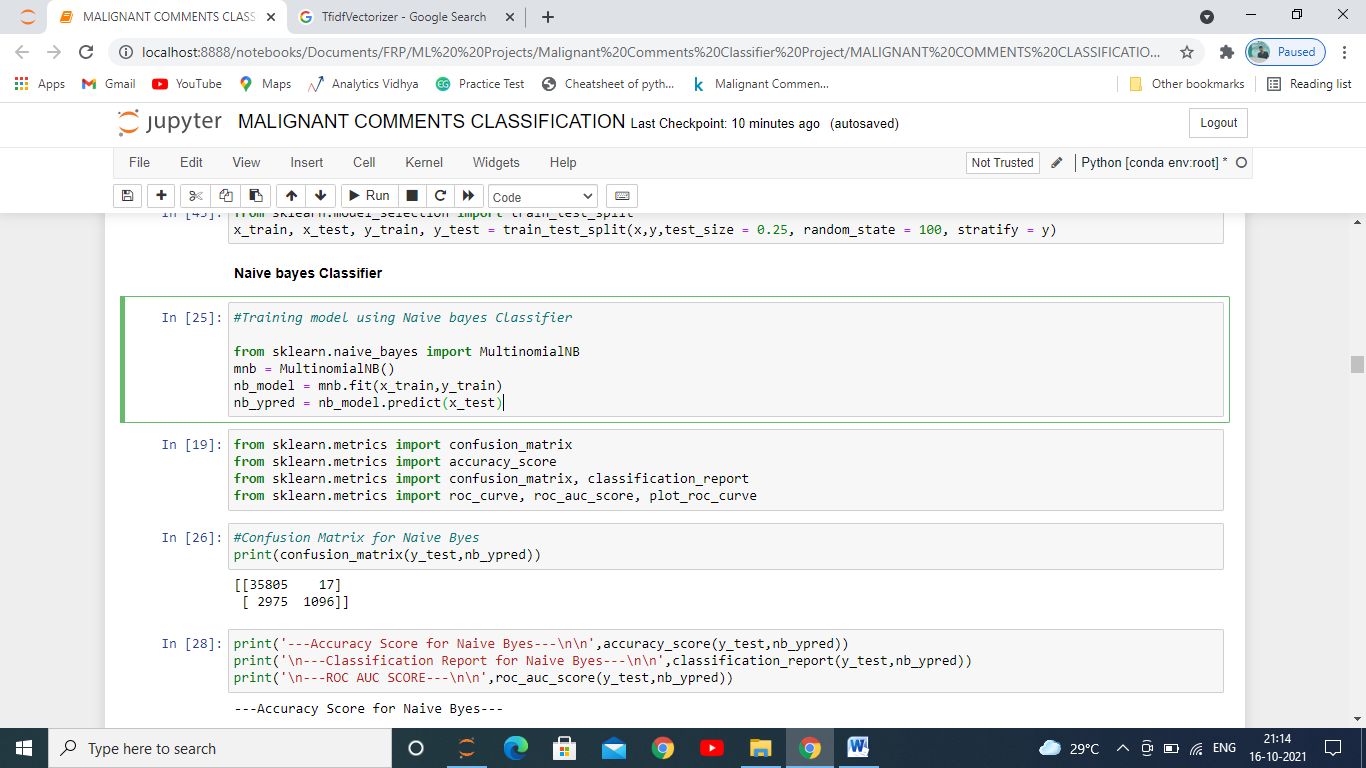
The whole dataset is broken down into 4 parts each for train and test, 2 parts for traning the model and 2 parts for testing, in it we kept 75% of the data for training and remaining 25% for testing the data. We used stratify because there was imbalance in the label. It will take equal proportion of each output to train and test the model.

* Run and Evaluate selected models

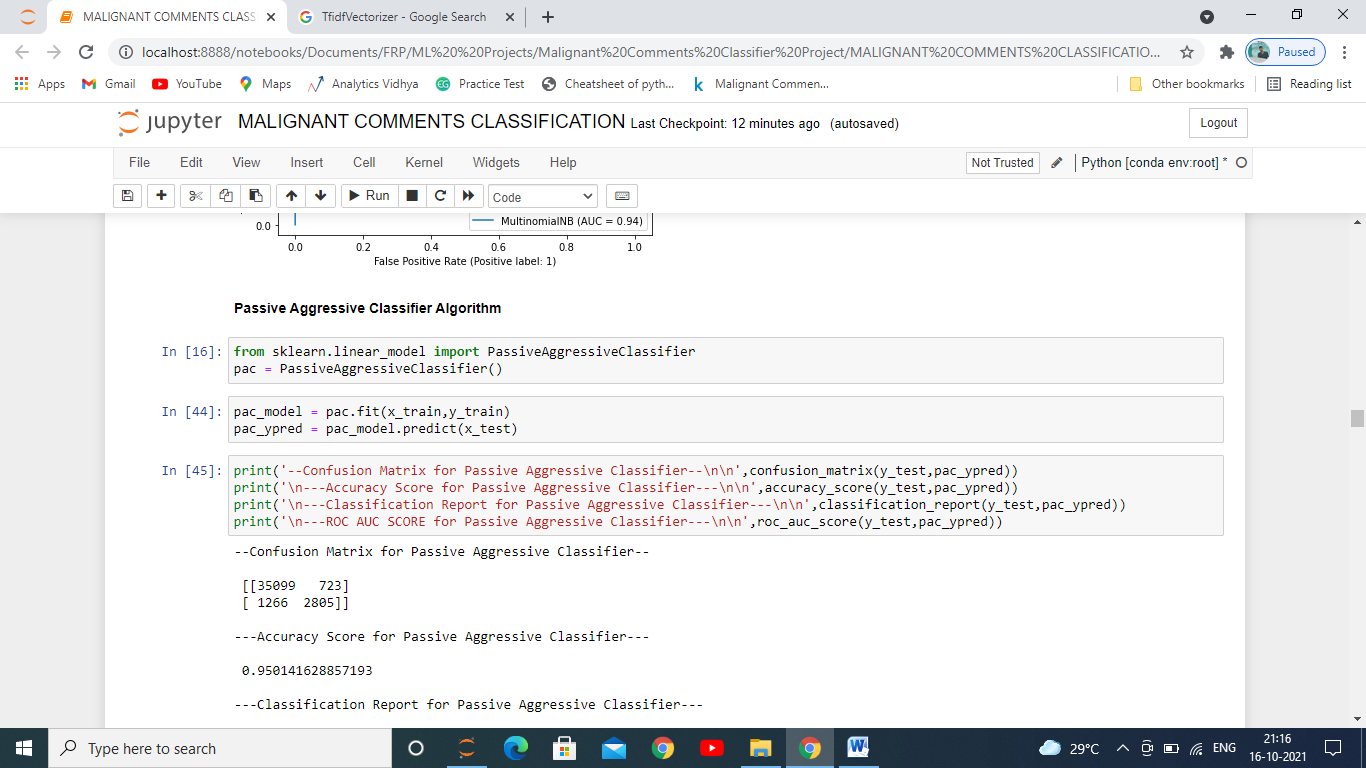
**Evaluation Matrics:**



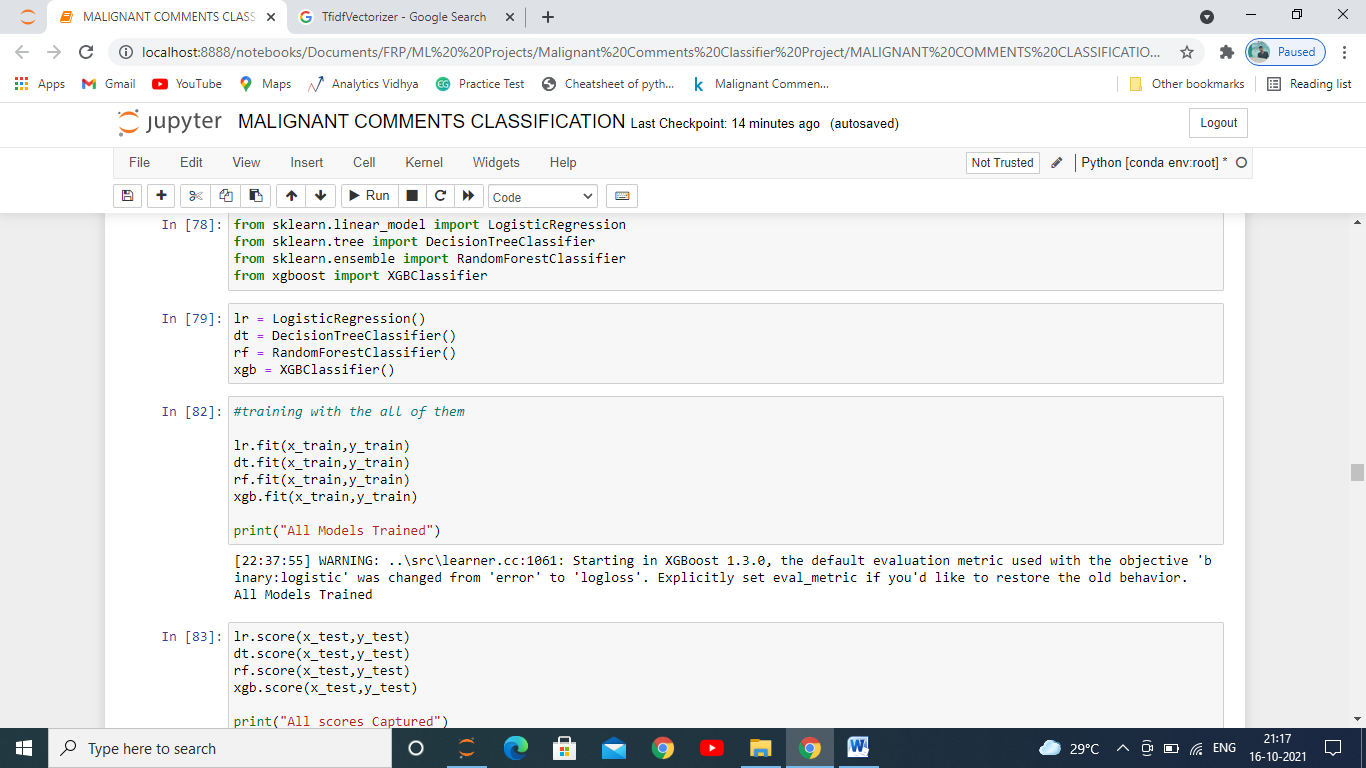
**Naïve Bayes Classifier:**



**Passive Aggressive Classifier:**



**Other Classification Models:**

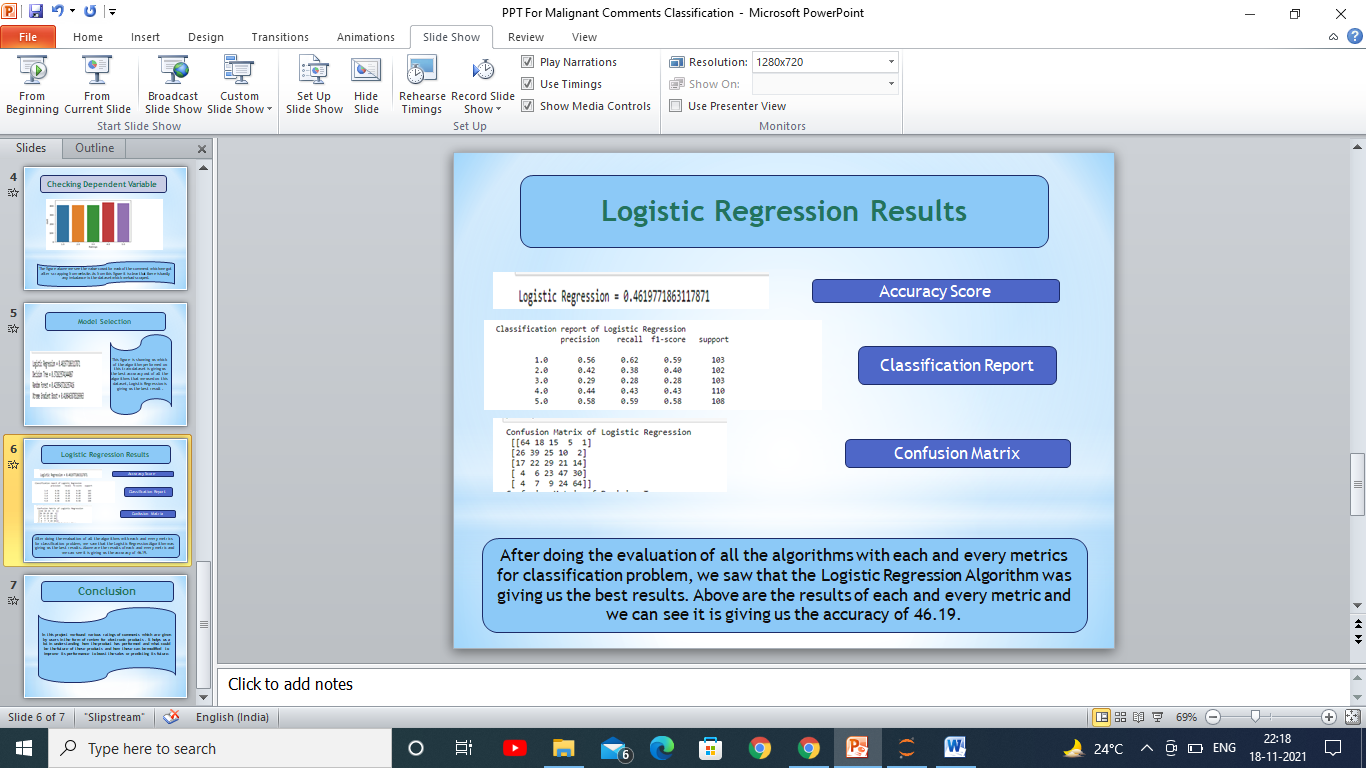


First we imported various evaluation metrics used for evaluating a classification model since the label had to predict to ratings which the user would have given on the basis of comment given out of 5, then imported various models we are used for building classification model as you can see we imported 6 models: Naïve Bayes Classifier, Passive Aggressive Classifier, Logistic Regression, Decision Tree Classifier, Random-Forest Classifier, Xtreme Gradient Boost Classifer then we used them with OneVsRest classifier as it was a multiclass problem and for each one of them which we are going to use further in building the model.We then trained it with x\_train and y\_train which we got from the dataset after splitting it in train and test set. This train set composes of 75% of the data which is going to be used to study the data by the model to predict the test set.

Then we captured the scores of each with our test set divided in 2: x\_test, y\_test.

After evaluating we saw that Logistic regression was giving us the highest accuracy.

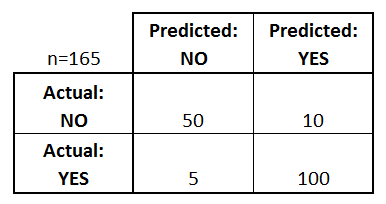
* Key Metrics for success in solving problem under consideration



In the above image we can see that the accuracy score given by the Logistic Regression Classifier is 46.19%. Informally, **accuracy** is the fraction of predictions our model got right. Formally, accuracy has the following definition:

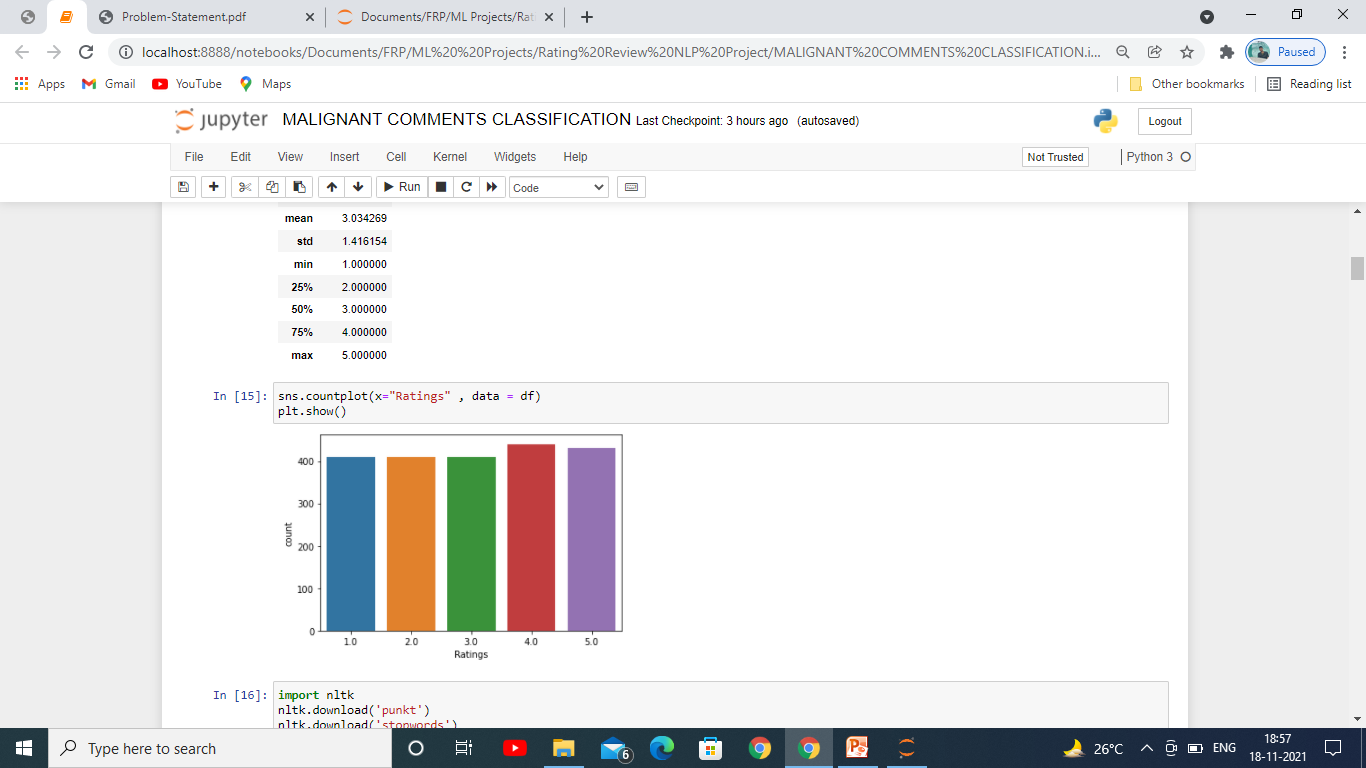
Accuracy=Number of correct predictions/ Total number of predictions

We also evaluated it with A confusion matrix is a table that is often used to **describe the performance of a classification model** on a set of test data for which the true values are known as shown in the below figure.



We also printed the Confusion metric for evaluating the model.   
A classification report is **a performance evaluation metric in machine learning**. It is used to show the precision, recall, F1 Score, and support of your trained classification model.

* Visualizations



With the help of counplot of seaborn library we checked the imbalance in dependent variable and as you can see in the above picture there is hardly any difference between the ratings given by user.

* Interpretation of the Results

The dataset is huge and it was regarding the ratings given by purchasers .

After building the we found that Logistic Regressor gave us the best result taking into consideration all the metrics scores and parameters, this model had covered most of the Area.

**CONCLUSION**

* Key Findings and Conclusions of the Study

This model helped us to analyse the comments and ratings which are given by different users.

* Learning Outcomes of the Study in respect of Data Science

While doing this project we observed that data visualization plays a key role in analysis the data and further processing it to build a model which has very less error and more accuracy. Analysis helps us to find out which are the feature mostly affecting the label and has more weightage and on this basis the data needs to be cleaned so that the data which we get to build our model gives us the best results. Shape is used to find the number of rows and columns, info for getting what are the data types of features and whether there are any null values in it or not, if there are this needs to be either filled up or removed from the dataset so that it does not build an erroneous model for us, describe has also been used to check the quartiles and mean, standard deviation, min value and maximum value of each of the features.