

**Review Rating Project**

Submitted by:

Vandana Jain

Internship 10

**ACKNOWLEDGMENT**

The internship opportunity I had with FlipRobo was a great chance for learning and professional development. Therefore, I consider myself as a very lucky individual as I was provided with an opportunity to be a part of it. I am also grateful for having a chance to meet so many wonderful people and professionals who led me though this project period.

I would like to thank our SME for suggesting this project and for his whole hearted cooperation and constant encouragement throughout the project.

**INTRODUCTION**

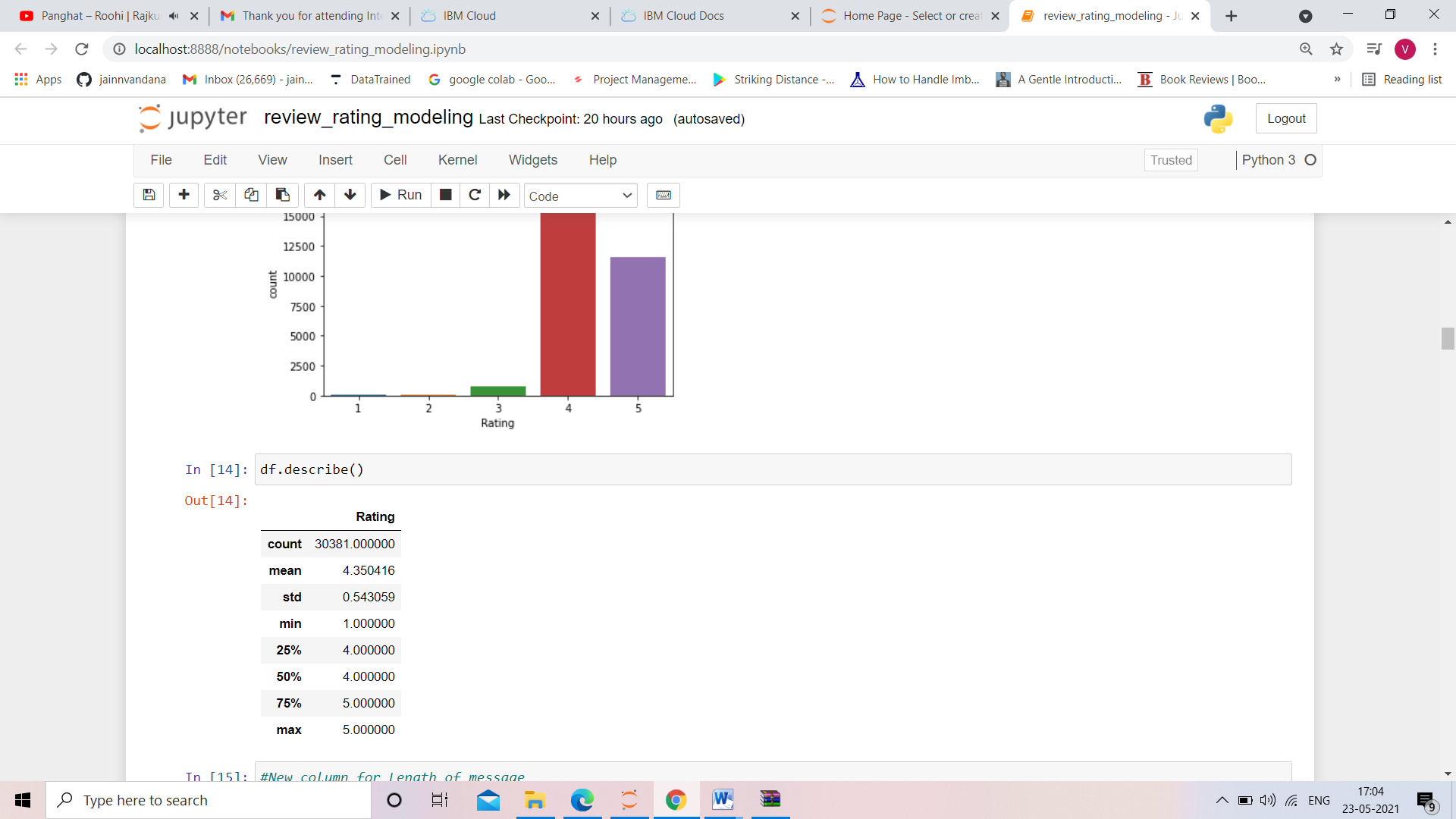
We have a client who has a website where people write different reviews for technical products. Now they are adding a new feature to their website i.e. the reviewer will have to add stars (rating) as well with the review. The rating is out 5 stars and it only has 5 options available 1 star, 2 stars, 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.

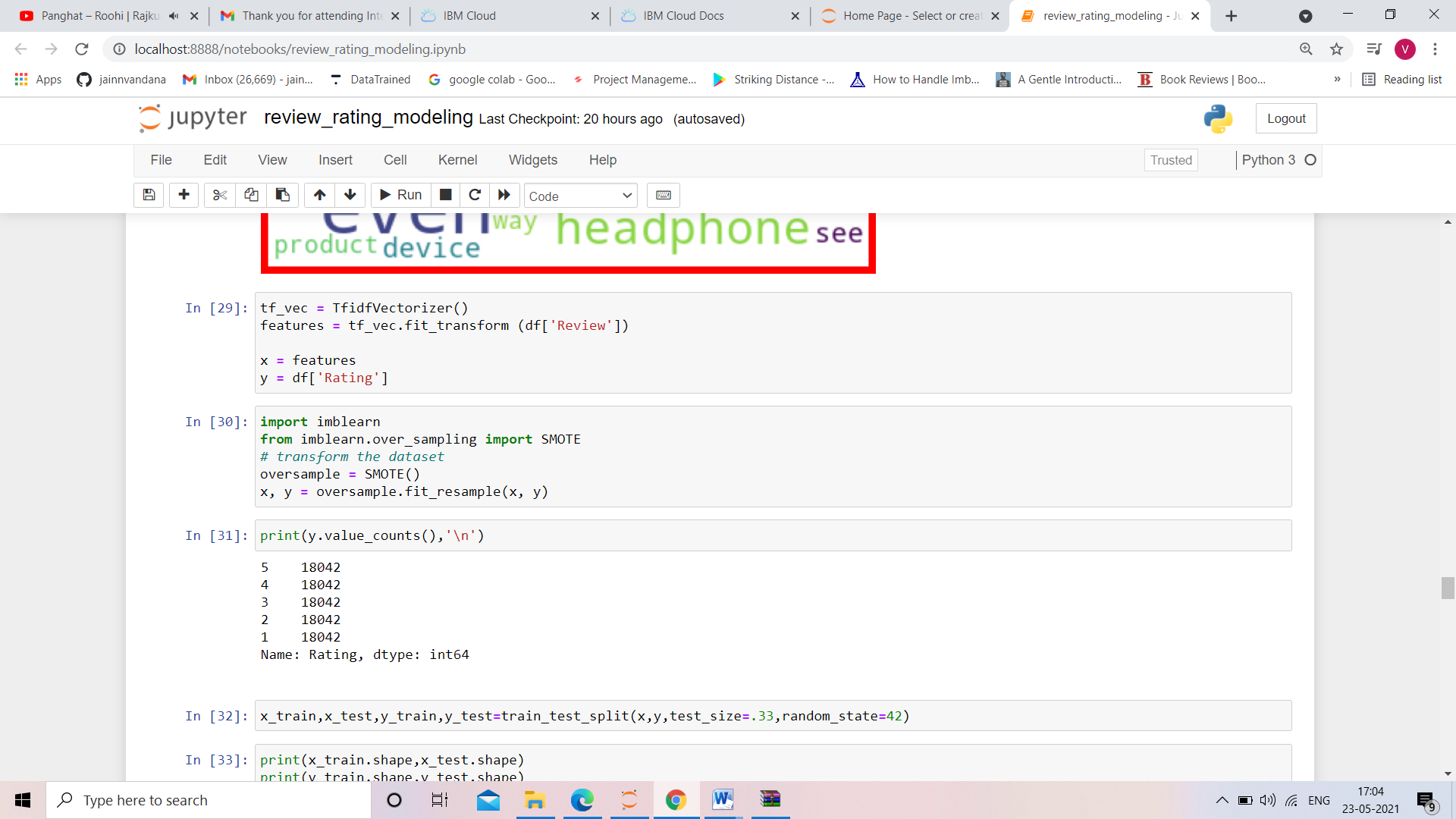
**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem

Machine Learning is defined by Tom Mitchell in his book as “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E”. Supervised learning is when the output is known for the corresponding inputs, and is also provided for the machine to learn.

* + EDA (Exploratory data analysis)
  + Data Pre-processing
  + Feature Extraction
  + Scoring & Metrics

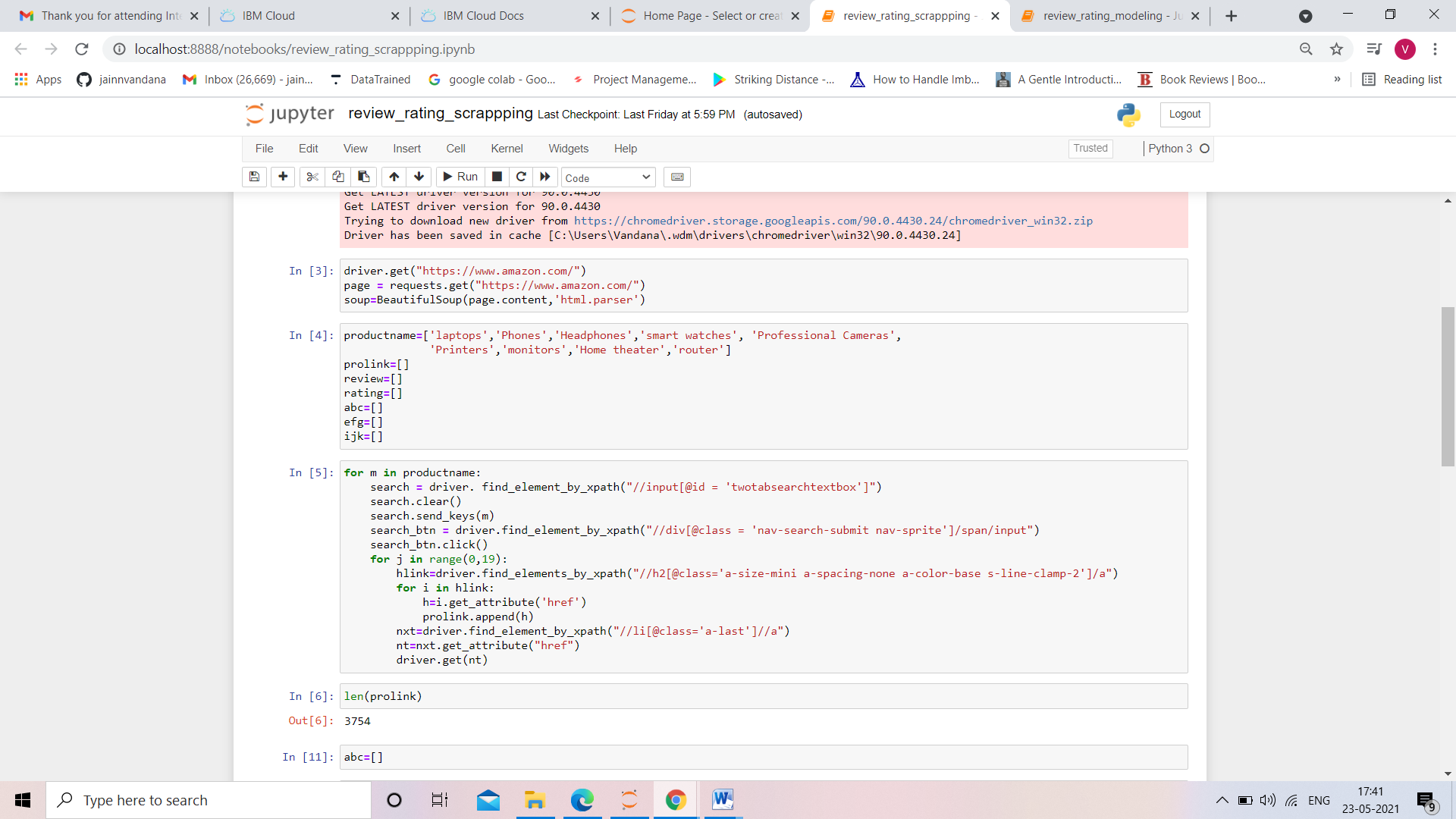


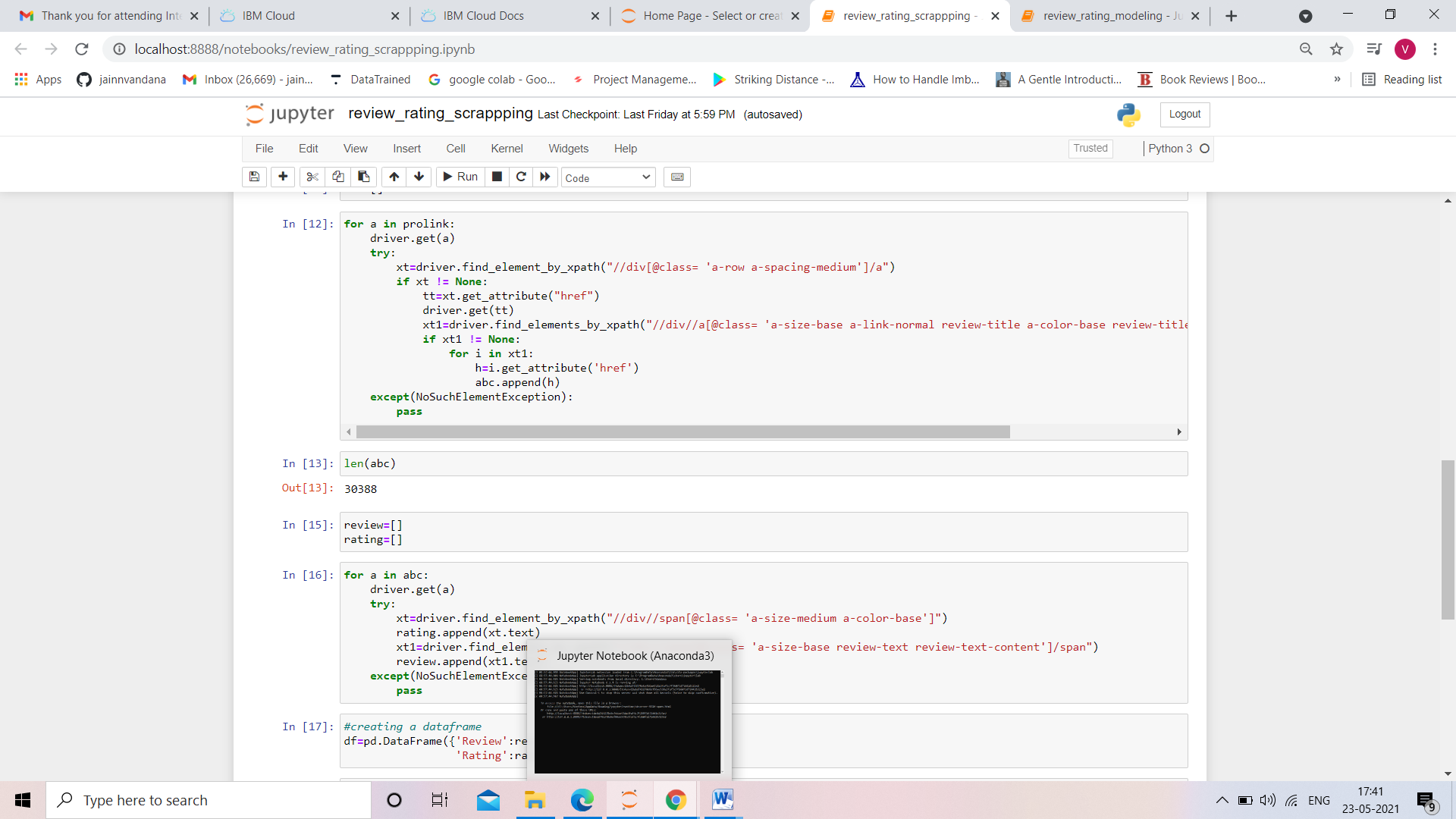


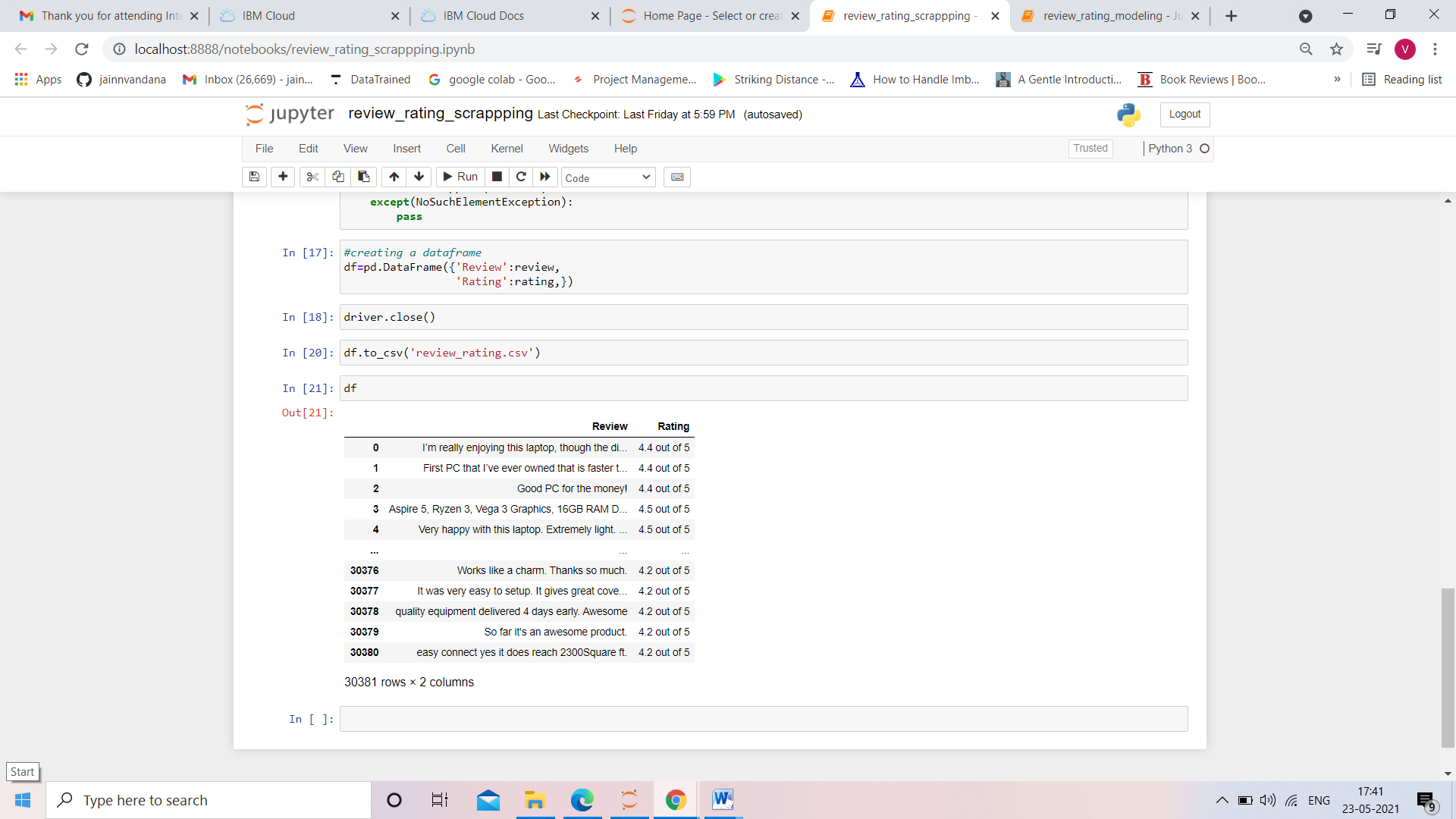
* Data Sources and their formats

The data is scrapped from the online shopping website amazon.com . The dataset consists of around 30380reviews and ratings of technical products which s provided by the user.

We scrapped the data using the selenium with python.





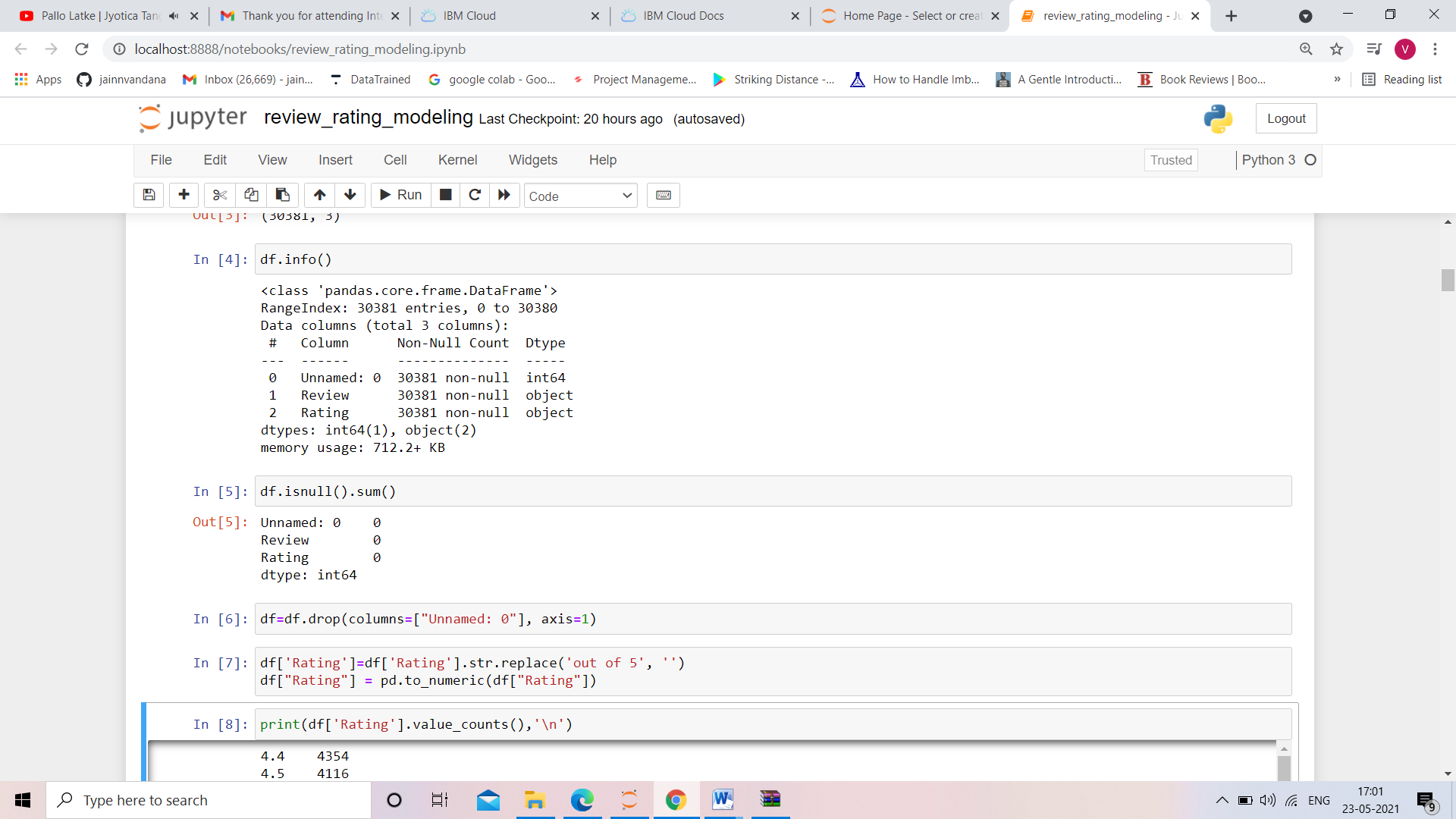


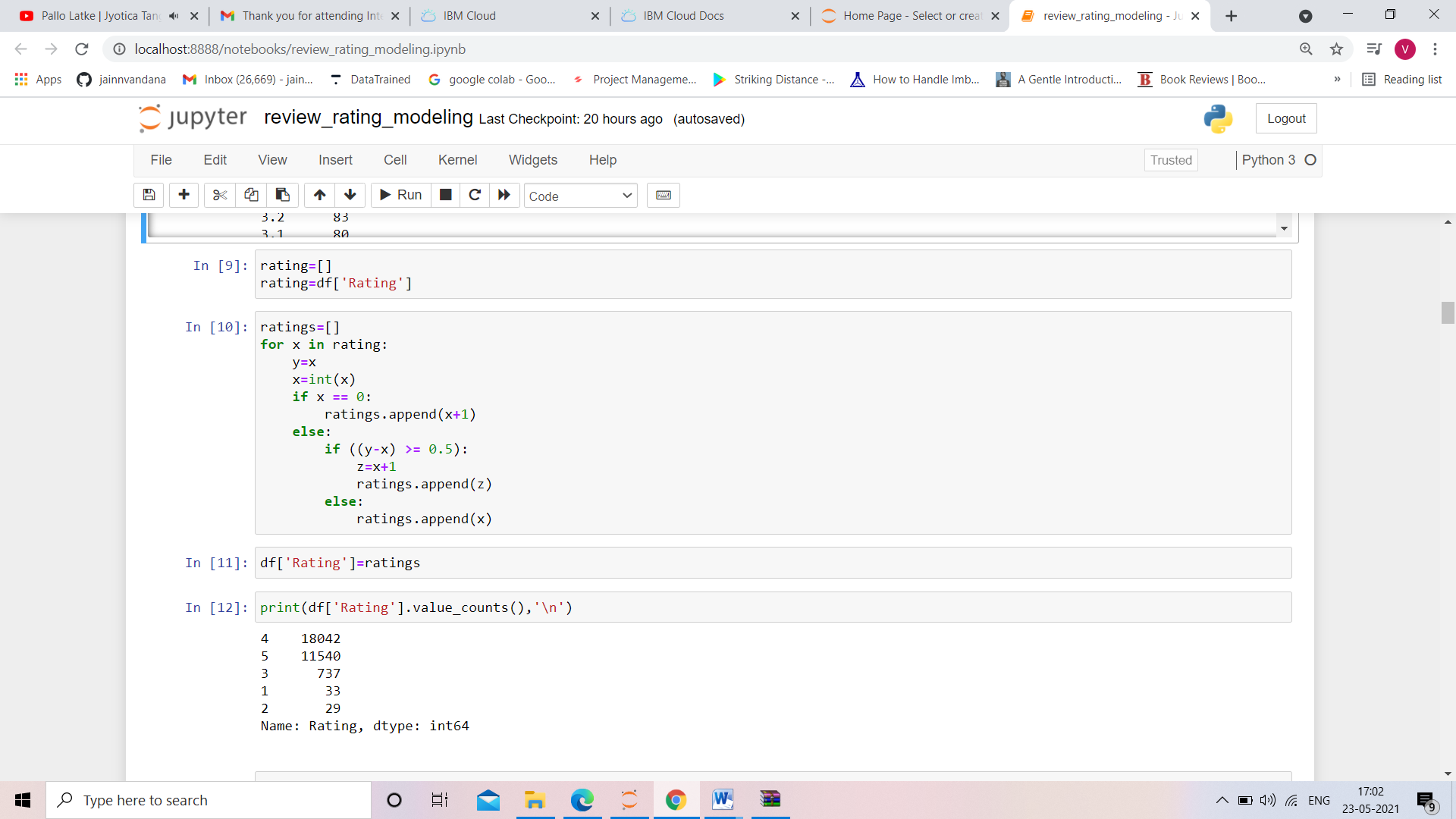
* Data Preprocessing Done

The dataset that will be used to train the model has some challenges. Text Cleaning is a very important step in machine learning because your data may contains a lot of noise and unwanted character such as punctuation, white space, numbers, hyperlink and etc.

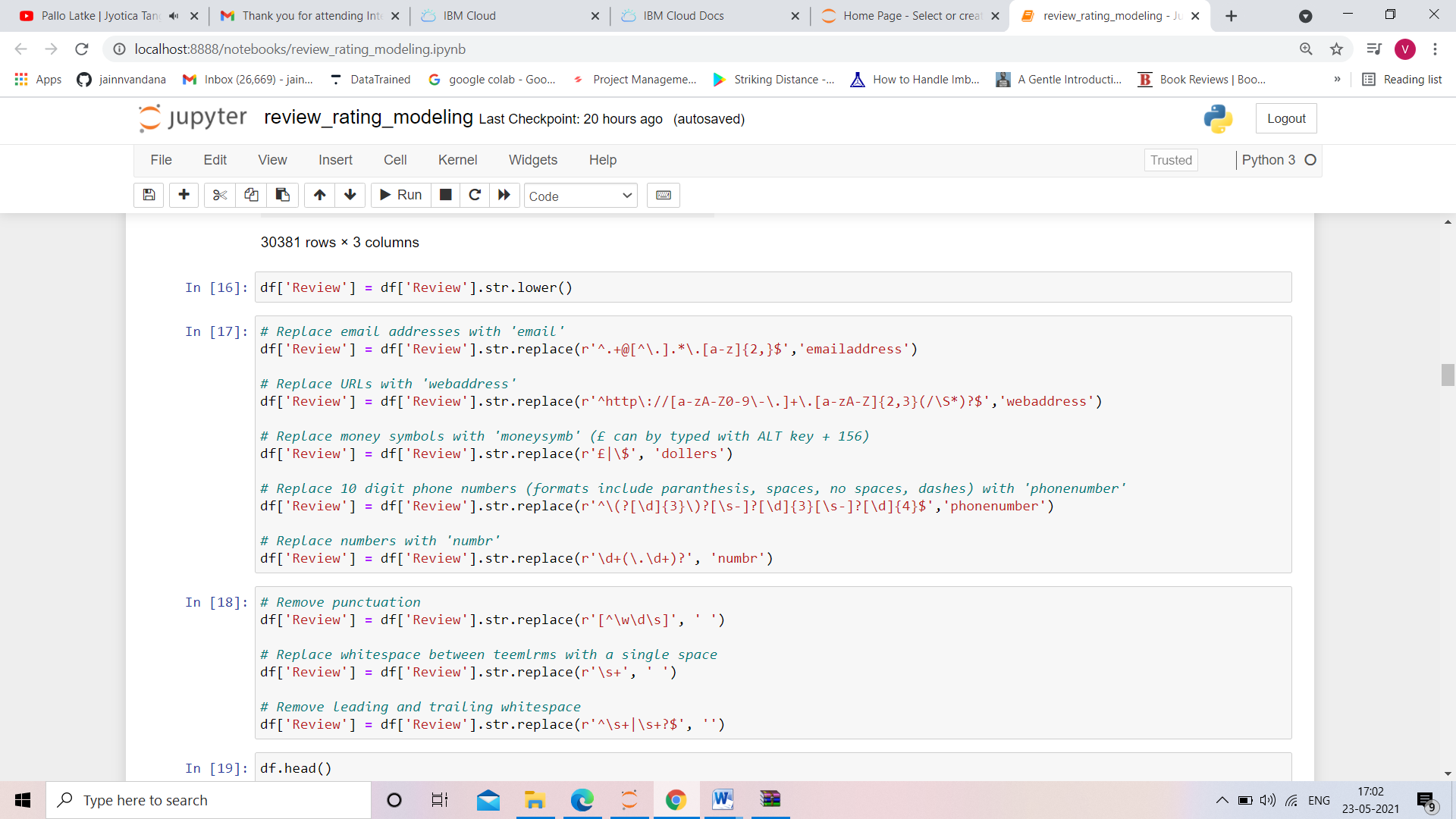
Some standard procedures are:

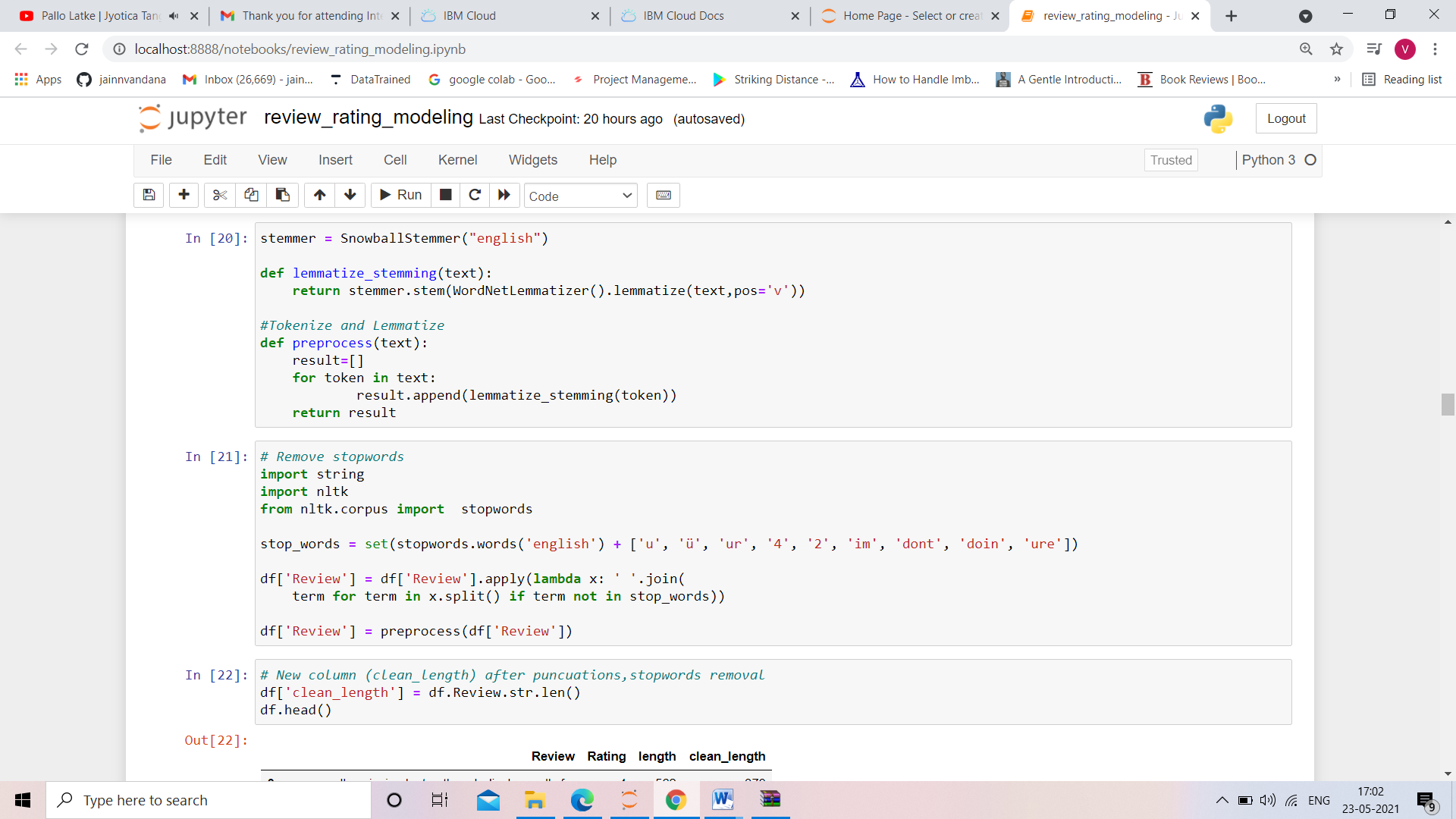
* + convert all letters to lower/upper case
  + removing numbers
  + removing punctuation
  + removing white spaces
  + removing hyperlink
  + removing stop words such as a, about, above, down, doing and the list goes on… Sometimes, the extremely common word which would appear to be of very little value in helping select documents matching user need are excluded from the vocabulary entirely.
  + Word Stemming: Stemming algorithms work by removing the end or the beginning of the words, using a list of common prefixes and suffixes that can be found in that language.
  + Word lemmatization: Lemmatization is utilizing the dictionary of a particular language and tried to convert the words back to its base form. It will try to take into account of the meaning of the verbs and convert it back to the most suitable base form.









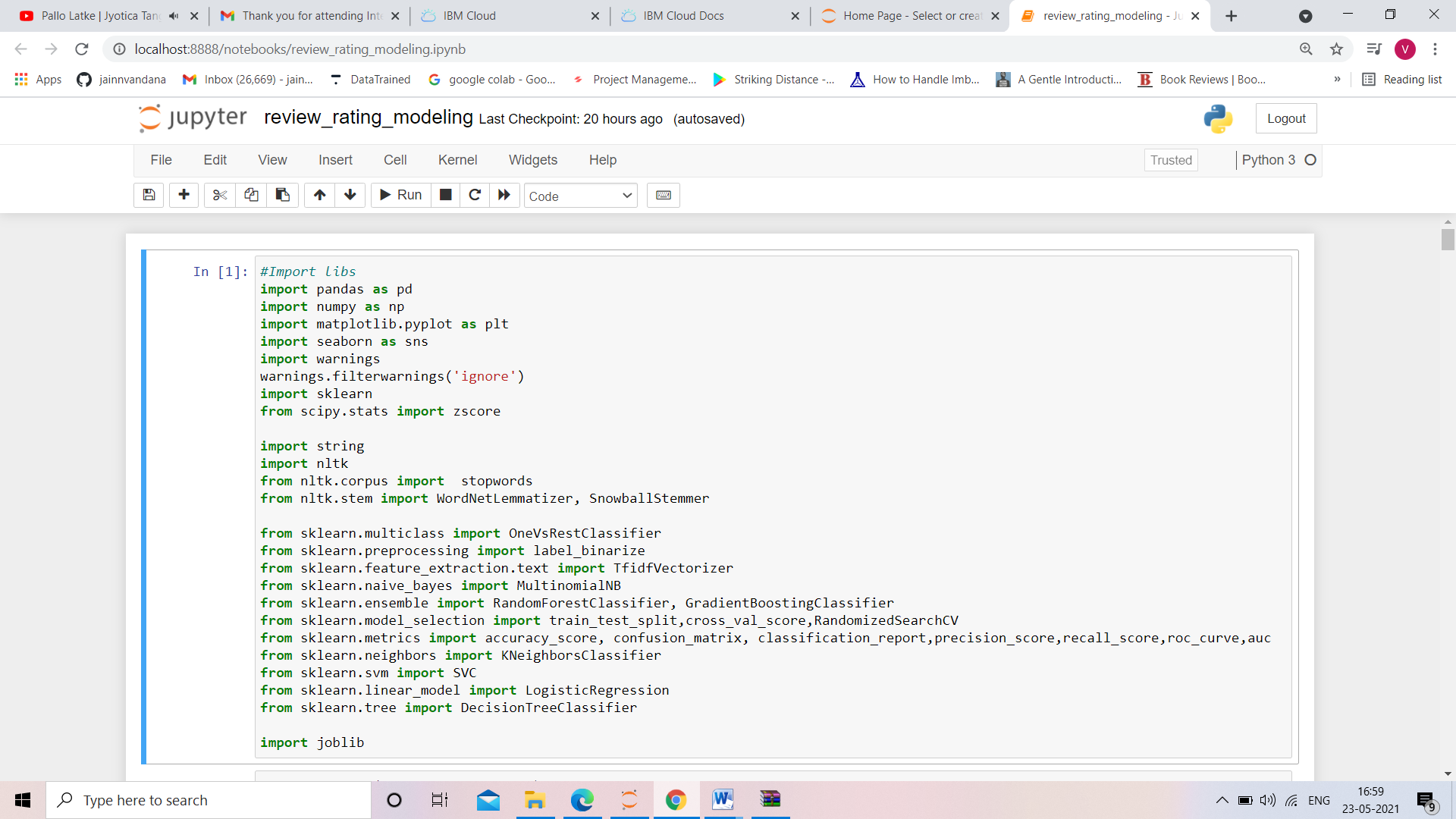


* Hardware and Software Requirements and Tools Used

Hardware : Since the computational aspect of the project is of importance to PANDA, it is important to know the hardware that was used in the evaluation process. The training and evaluation of the neural network model has been done on a Windows 10 computer using a quad-core CPU at i3.

Software : anaconda 3 , windows 10 ,Microsoft office.

Tools used : python , machine learning libraries, Nltk, Nlp libraries.



**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

1. NAÏVE BAYS CLASSIFIER

Simple probabilistic classifier that calculates a set of probabilities by counting the frequency and combination of values in a given dataset. Represent as a vector of feature values. It is very useful to classify the comments properly. The precision and recall of this method is known to be very effective.

2. RANDOM FOREST CLASSIFIER

Random forest is a supervised learning algorithm. The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result. Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

3. GRADIENT BOOSTING CLASSIFIER

In Gradient Boosting, each predictor tries to improve on its predecessor by reducing the errors. But the fascinating idea behind Gradient Boosting is that instead of fitting a predictor on the data at each iteration, it actually fits a new predictor to the residual errors made by the previous predictor. Gradient Boosting has repeatedly proven to be one of the most powerful technique to build predictive models in both classification and regression. Because of the fact that Grading Boosting algorithms can easily overfit on a training data set, different constraints or regularization methods can be utilized to enhance the algorithm's performance and combat overfitting. Penalized learning, tree constraints, randomized sampling, and shrinkage can be utilized to combat overfitting.

4. DECISION TREE CLASSIFIER

Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

5. SUPPORT VECTOR MACHINE CLASSIFIER

A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems. After giving SVM model sets of labelled training data for each category, they’re able to categorize new text.

6. K NEAREST NEIGHBORS CLASSIFIER

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. It assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. It stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. It can be used for Regression as well as for Classification but mostly it is used for the Classification problems. K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.

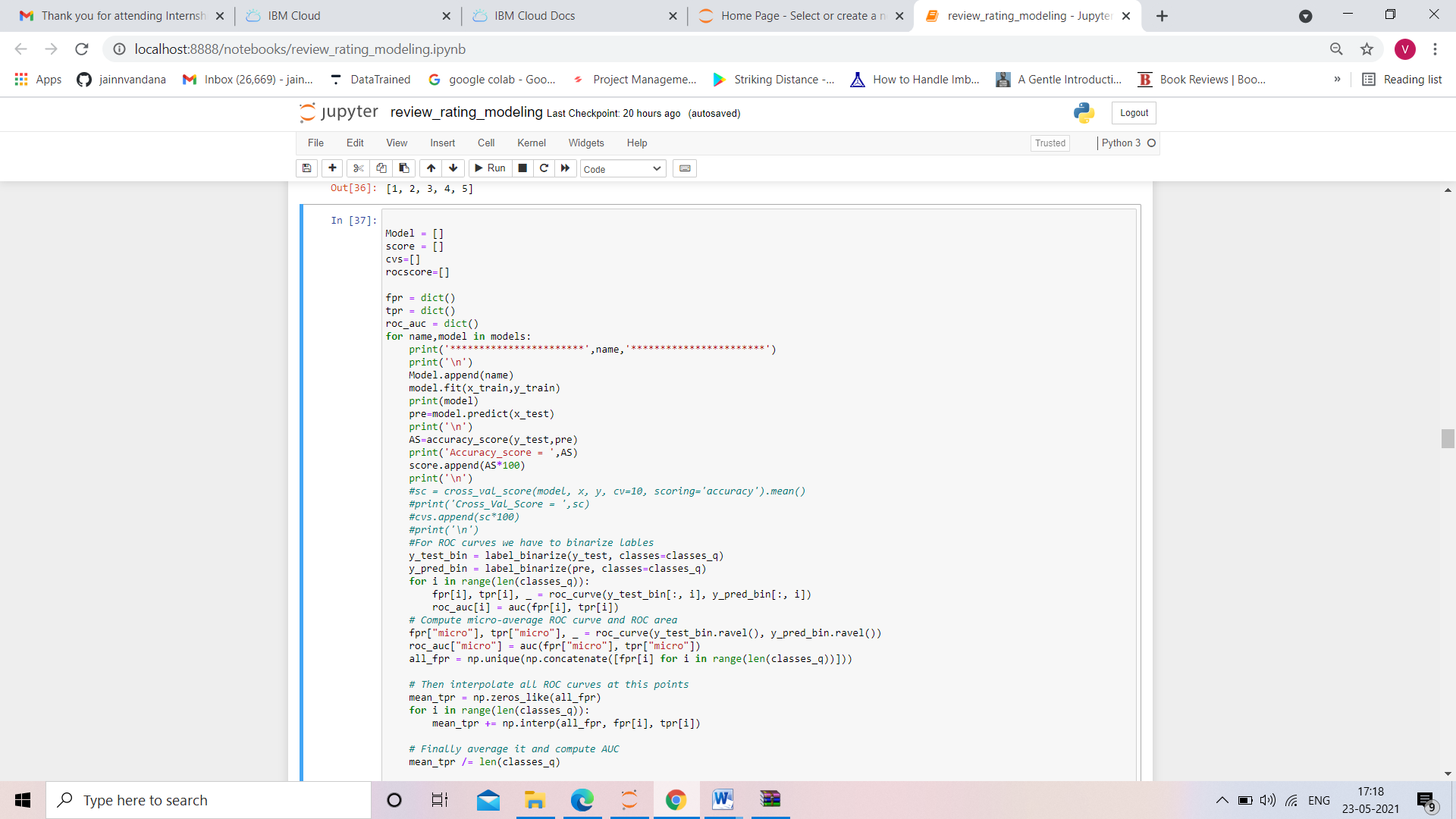
7. LOGISTIC REGRESSION

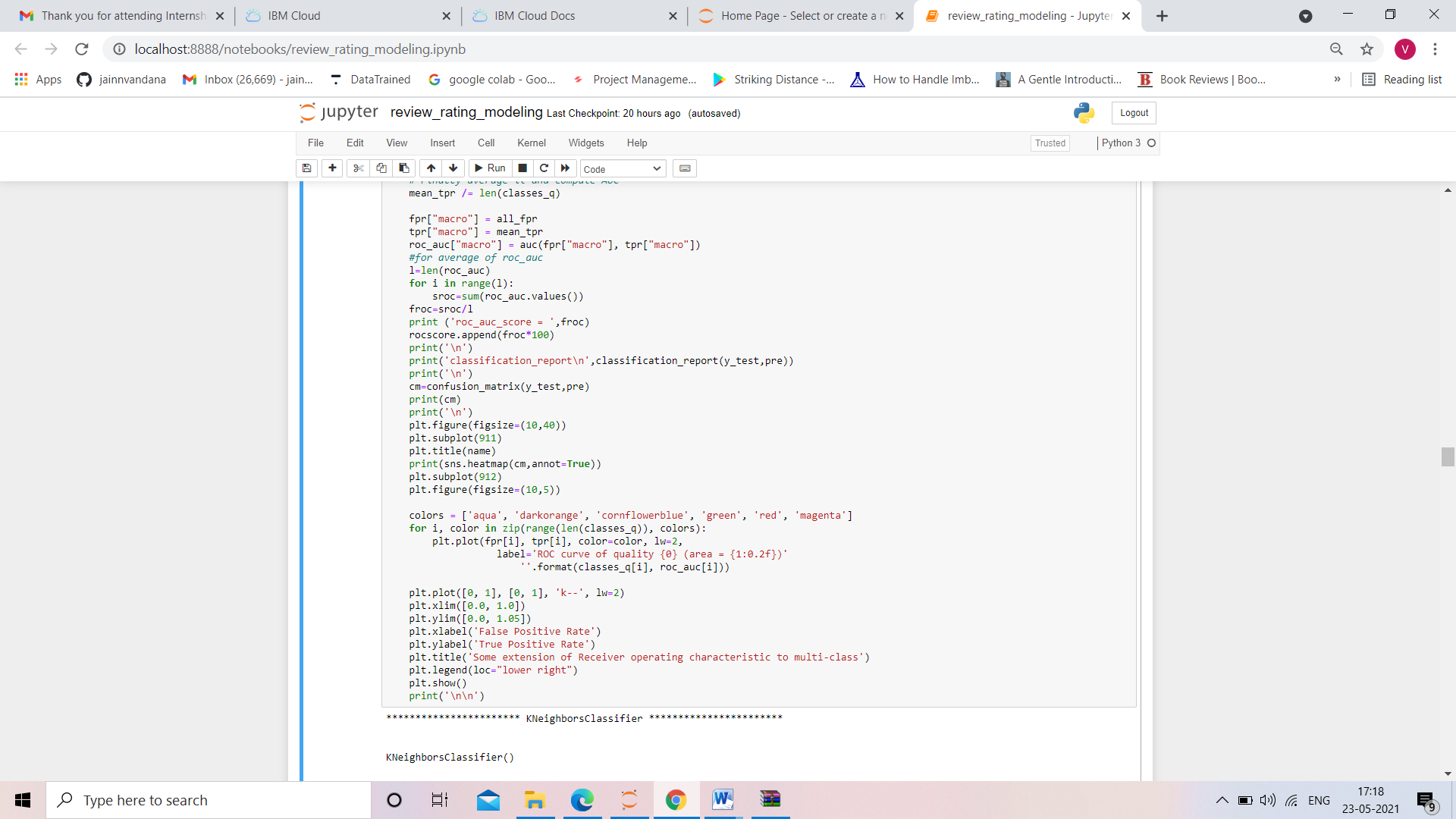
Logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

8. SMOTE

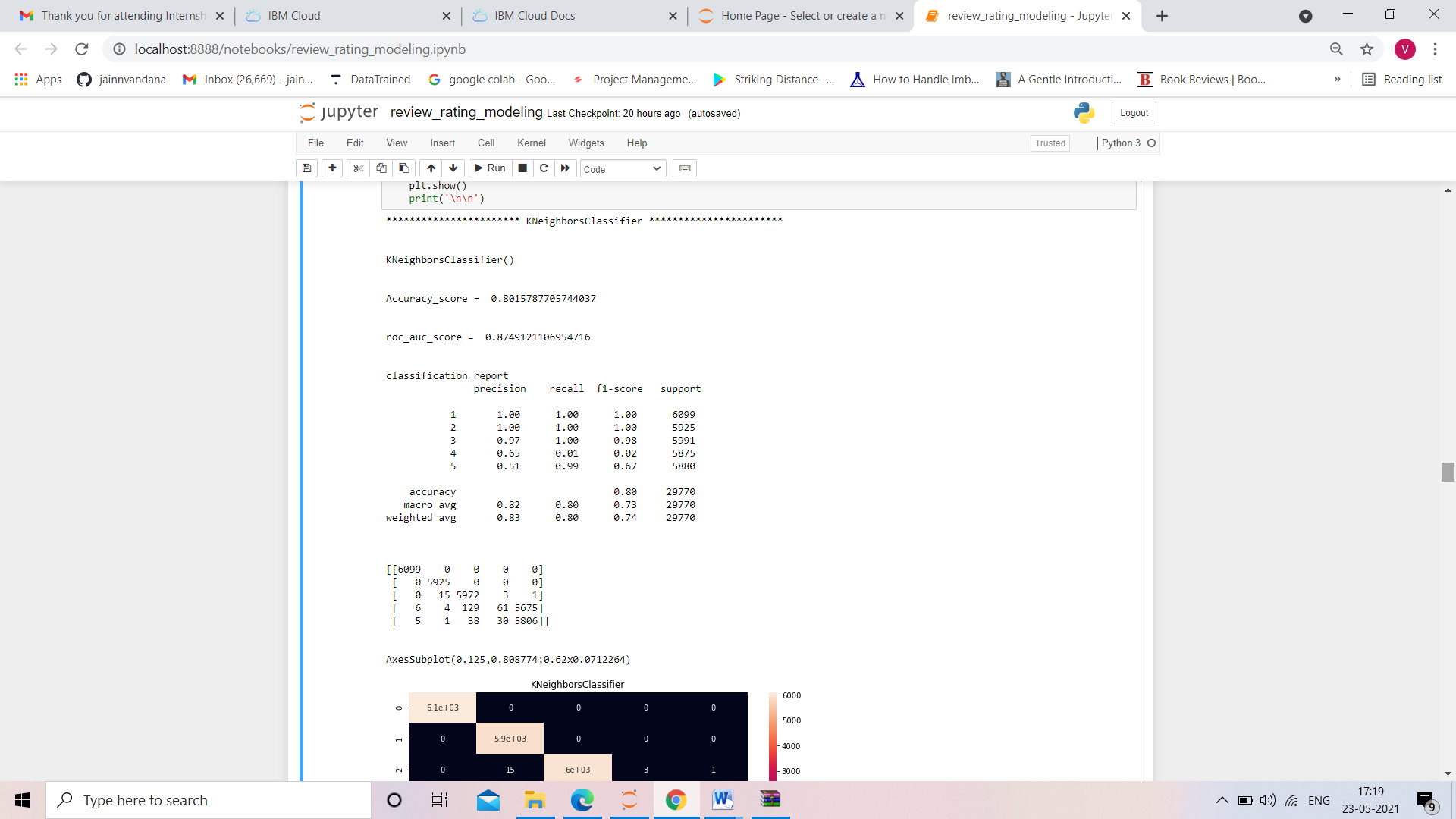
A problem with imbalanced classification is that there are too few examples of the minority class for a model to effectively learn the decision boundary. One way to solve this problem is to oversample the examples in the minority class. This can be achieved by simply duplicating examples from the minority class in the training dataset prior to fitting a model. This can balance the class distribution but does not provide any additional information to the model. An improvement on duplicating examples from the minority class is to synthesize new examples from the minority class. This is a type of data augmentation for tabular data and can be very effective. Perhaps the most widely used approach to synthesizing new examples is called the Synthetic Minority Oversampling TEchnique, or SMOTE for short. This technique was described by Nitesh Chawla, et al. in their 2002 paper named for the technique titled “SMOTE: Synthetic Minority Over-sampling Technique.” SMOTE works by selecting examples that are close in the feature space, drawing a line between the examples in the feature space and drawing a new sample at a point along that line.

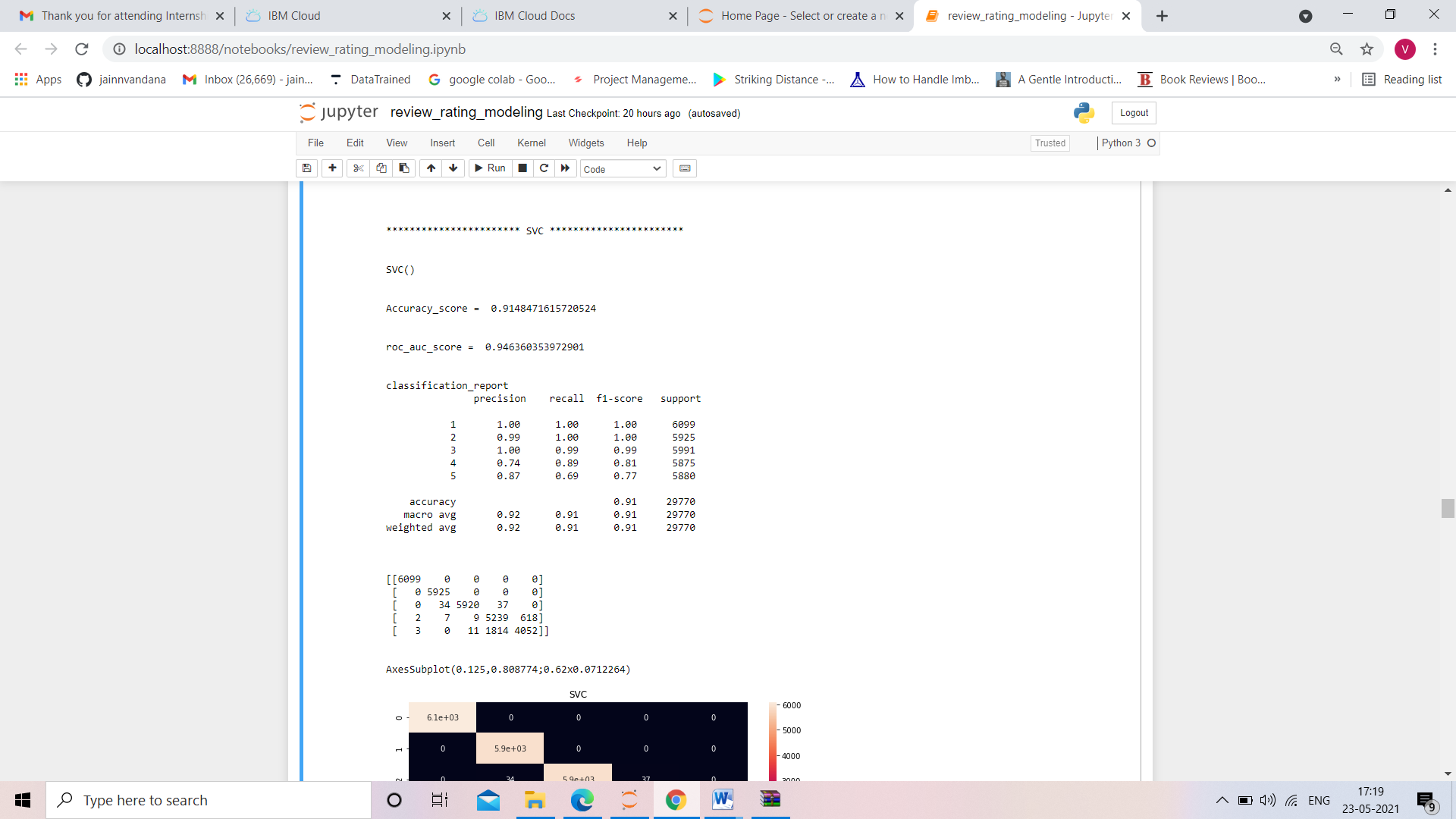
* Testing of Identified Approaches (Algorithms)

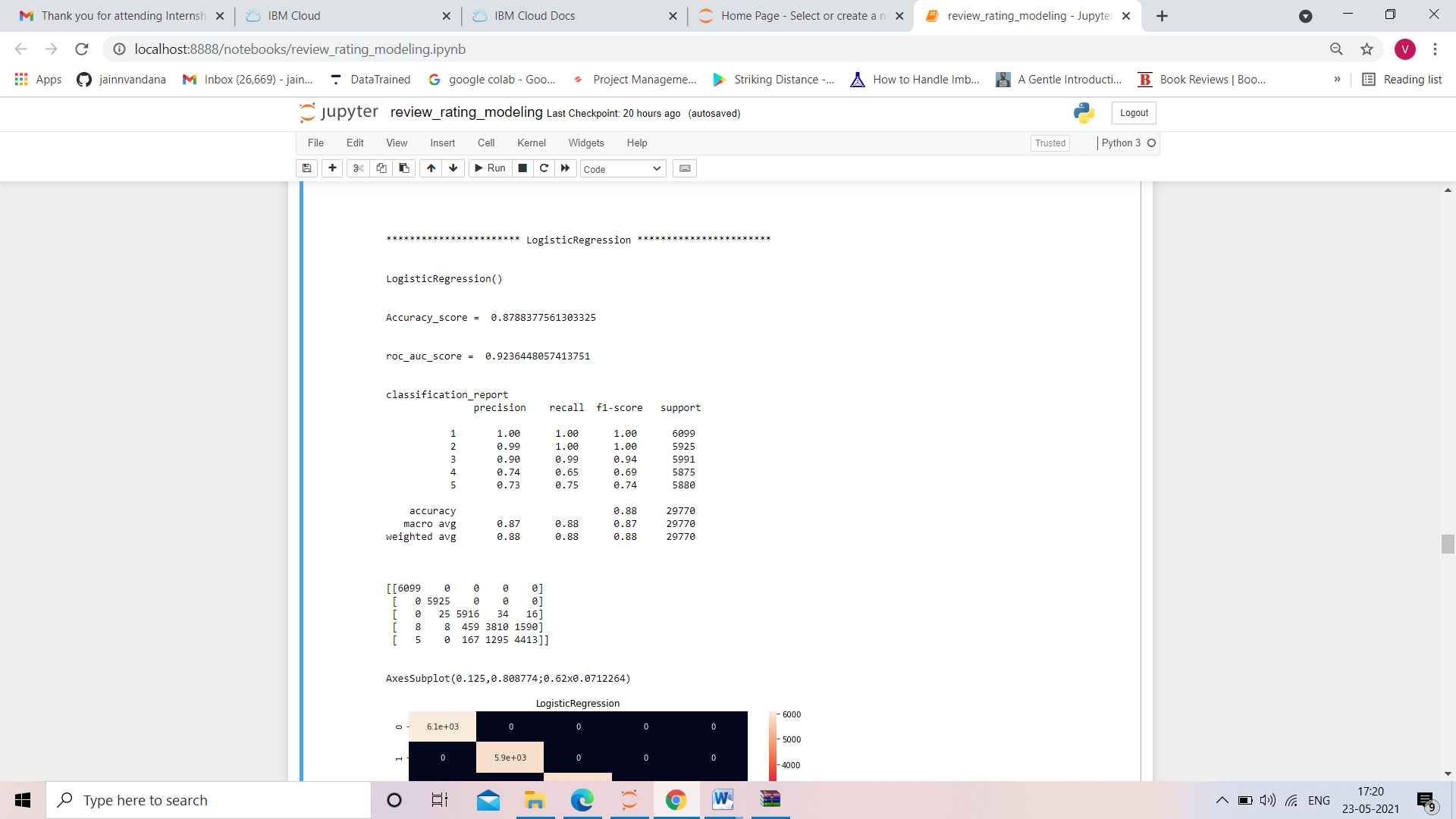


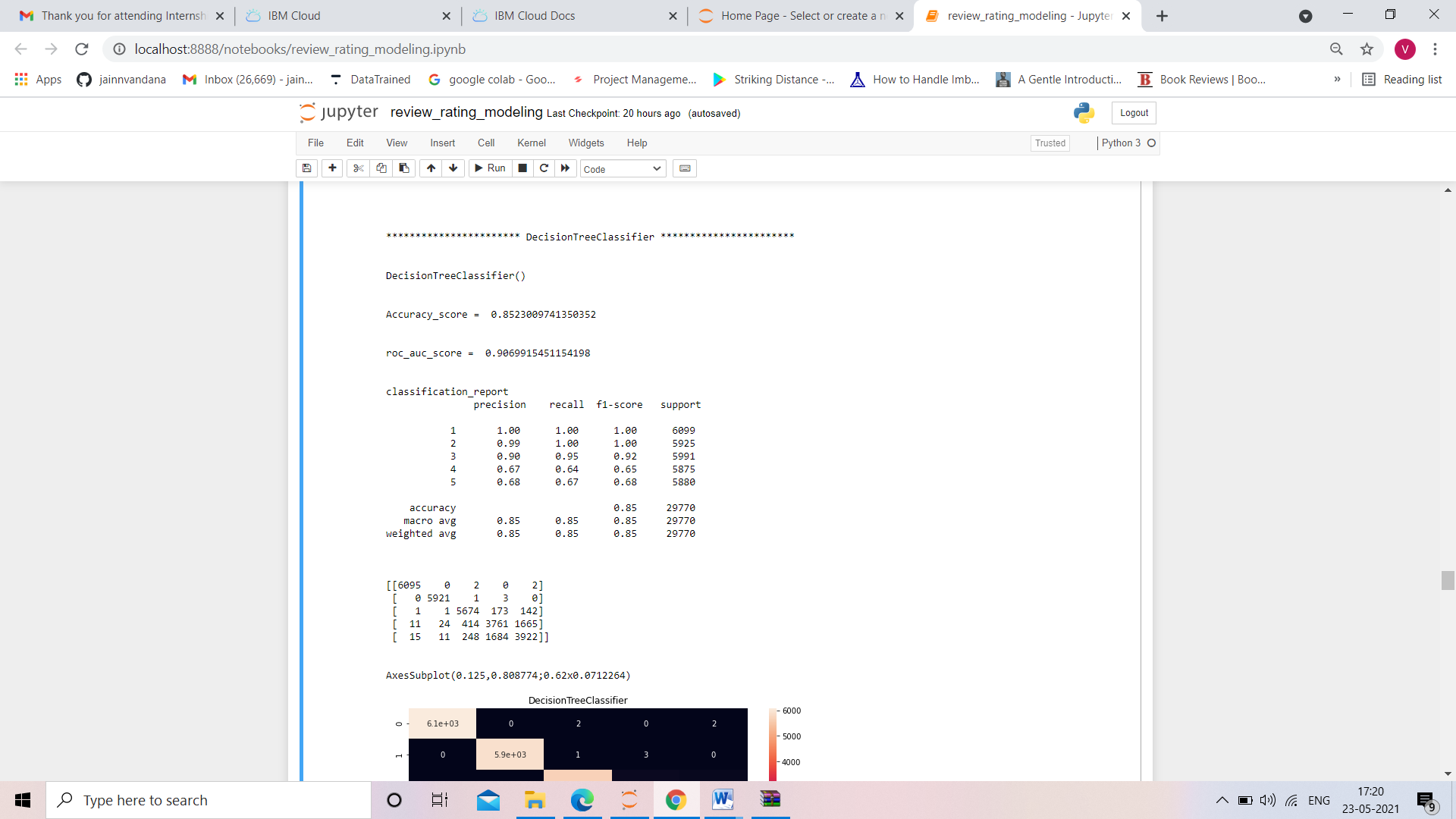


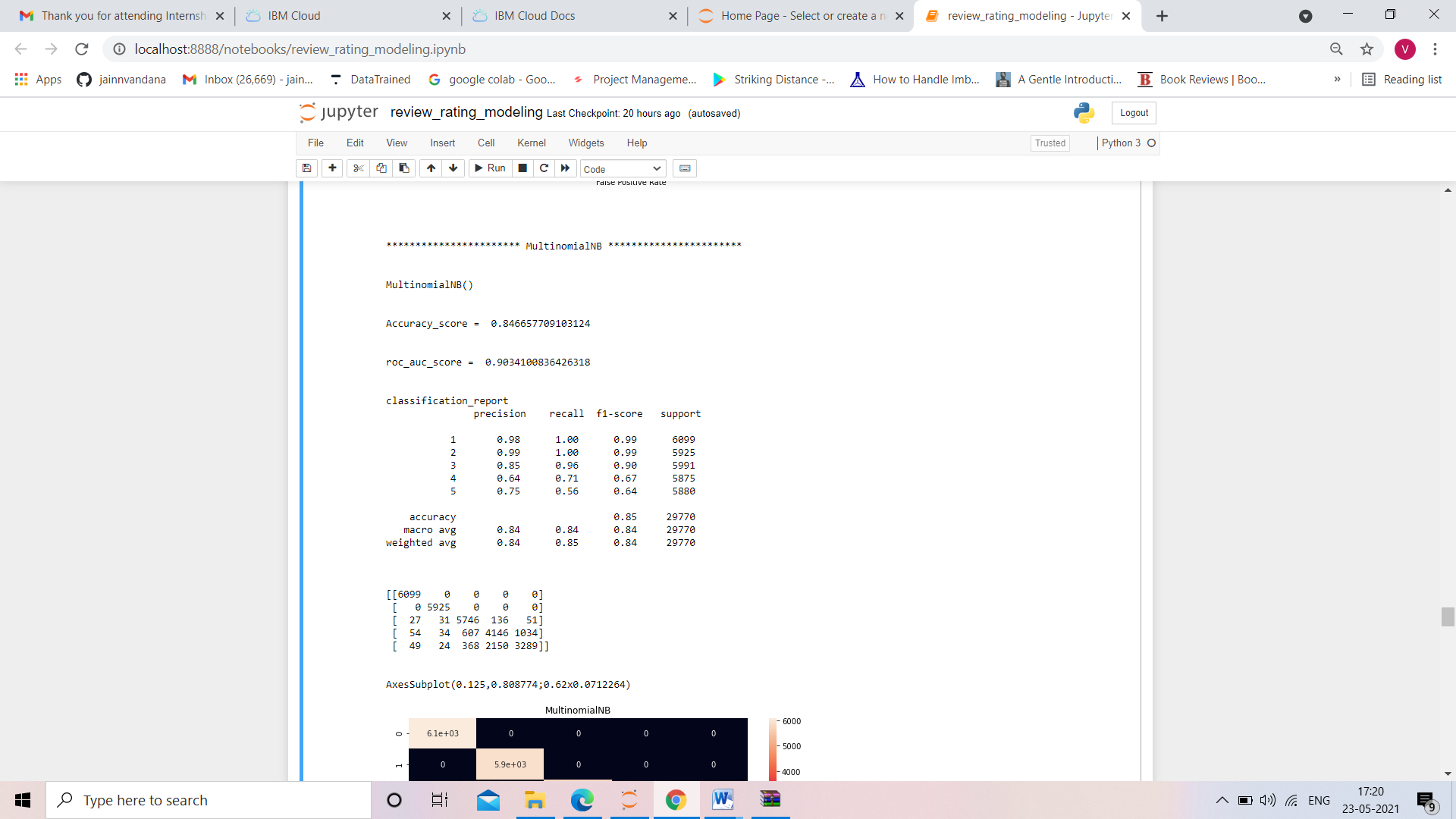
* Run and Evaluate selected models



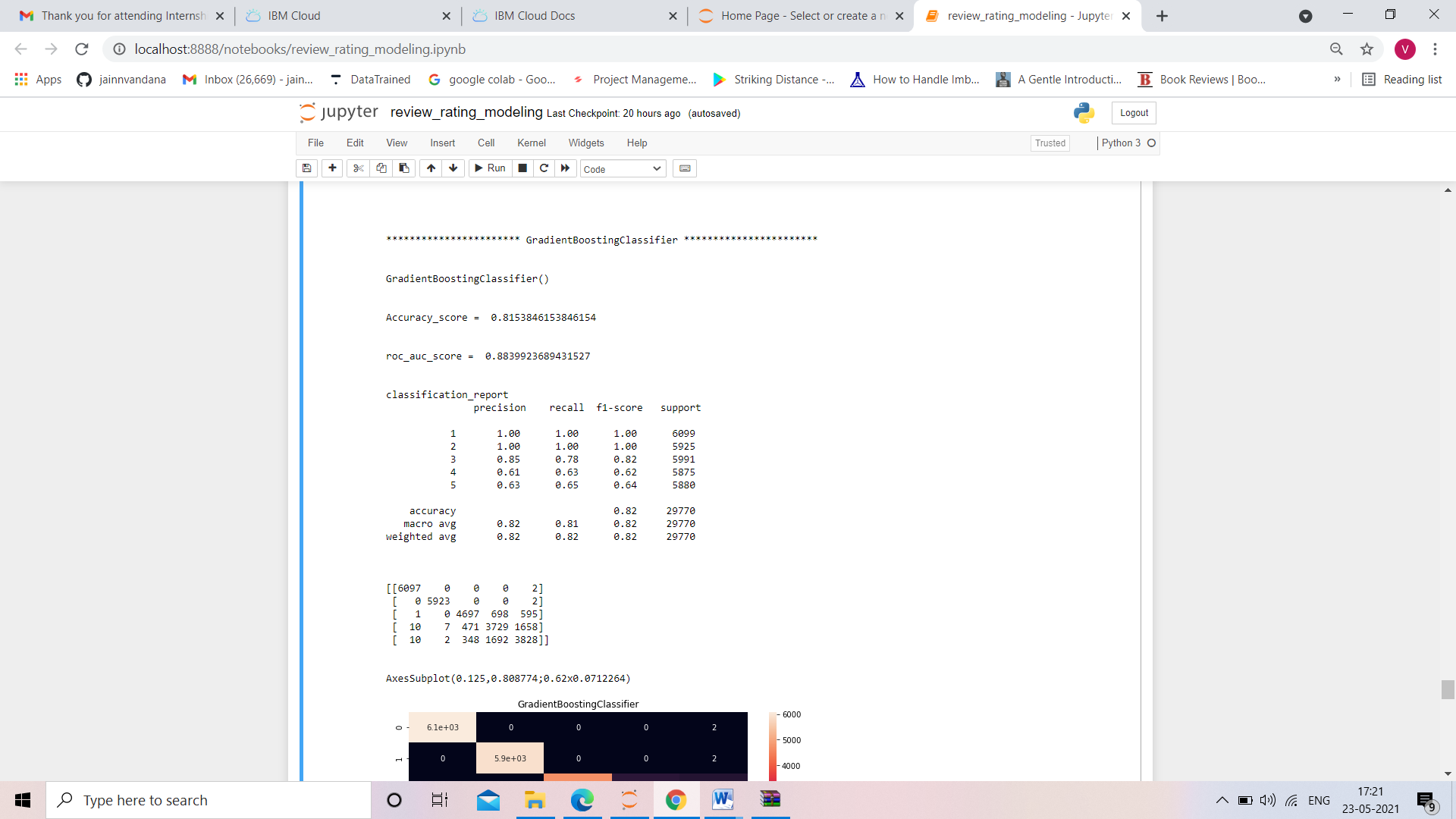












* Key Metrics for success in solving problem under consideration

When it comes to evaluation of a data science model’s performance, sometimes accuracy may not be the best indicator.

Some problems that we are solving in real life might have a very imbalanced class and using accuracy might not give us enough confidence to understand the algorithm’s performance.

In the fake news problem that we are trying to solve, fake and real news data is approximately 47% of our data. Here we can see our dataset is balanced dataset so accuracy score nearly tells the right predictions.so the problem of overfitting in this problem is nearly not to occur. So here, we are using accuracy score as to find better model.

Accuracy score

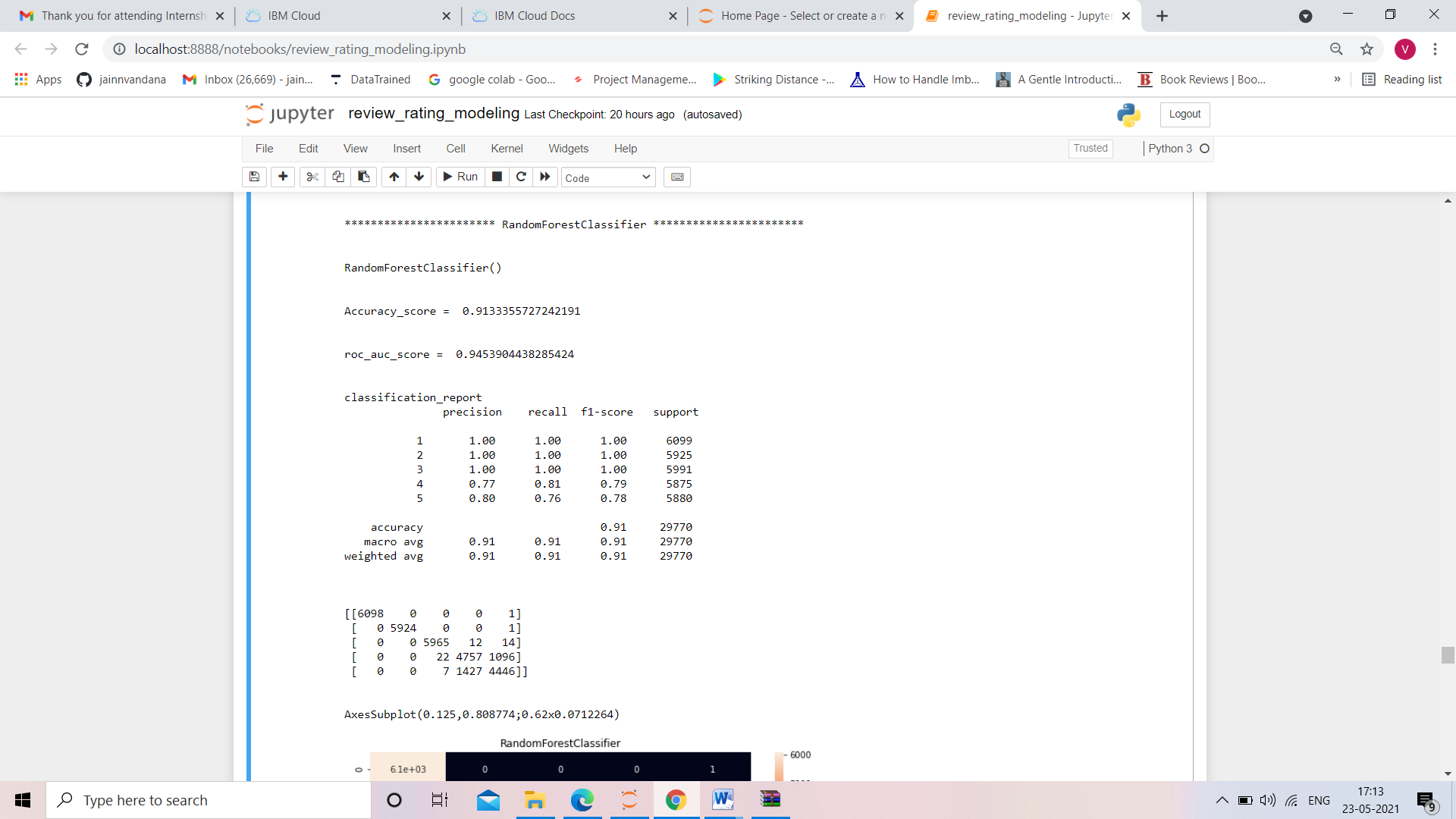
Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. One may think that, if we have high accuracy then our model is best. Yes, accuracy is a great measure but only when you have symmetric datasets where values of false positive and false negatives are almost same. Therefore, you have to look at other parameters to evaluate the performance of your model.

Confusion Matrix

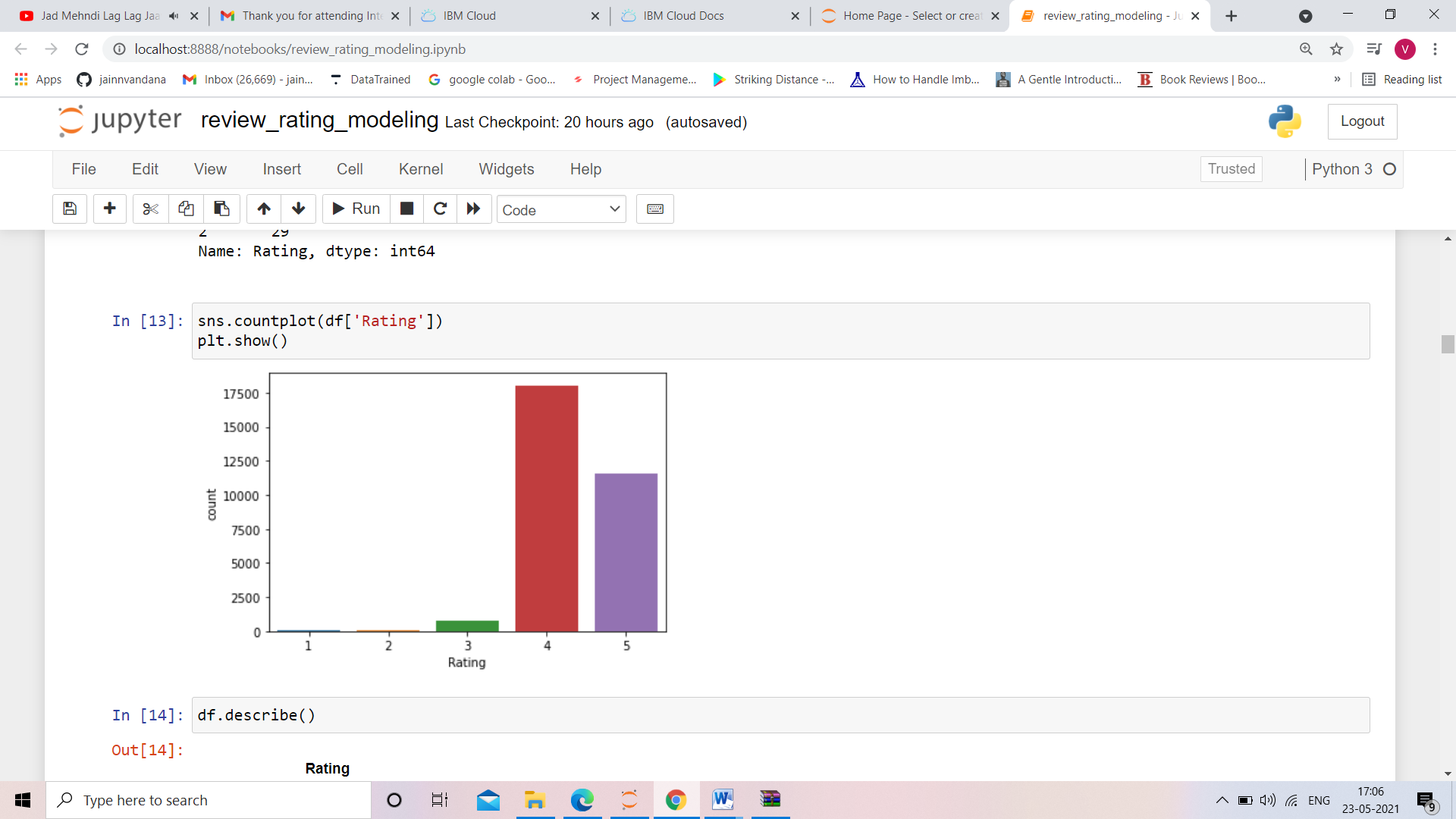
Confusion Matrix is a very good way to understand results like true positive, false positive, true negative and so on.

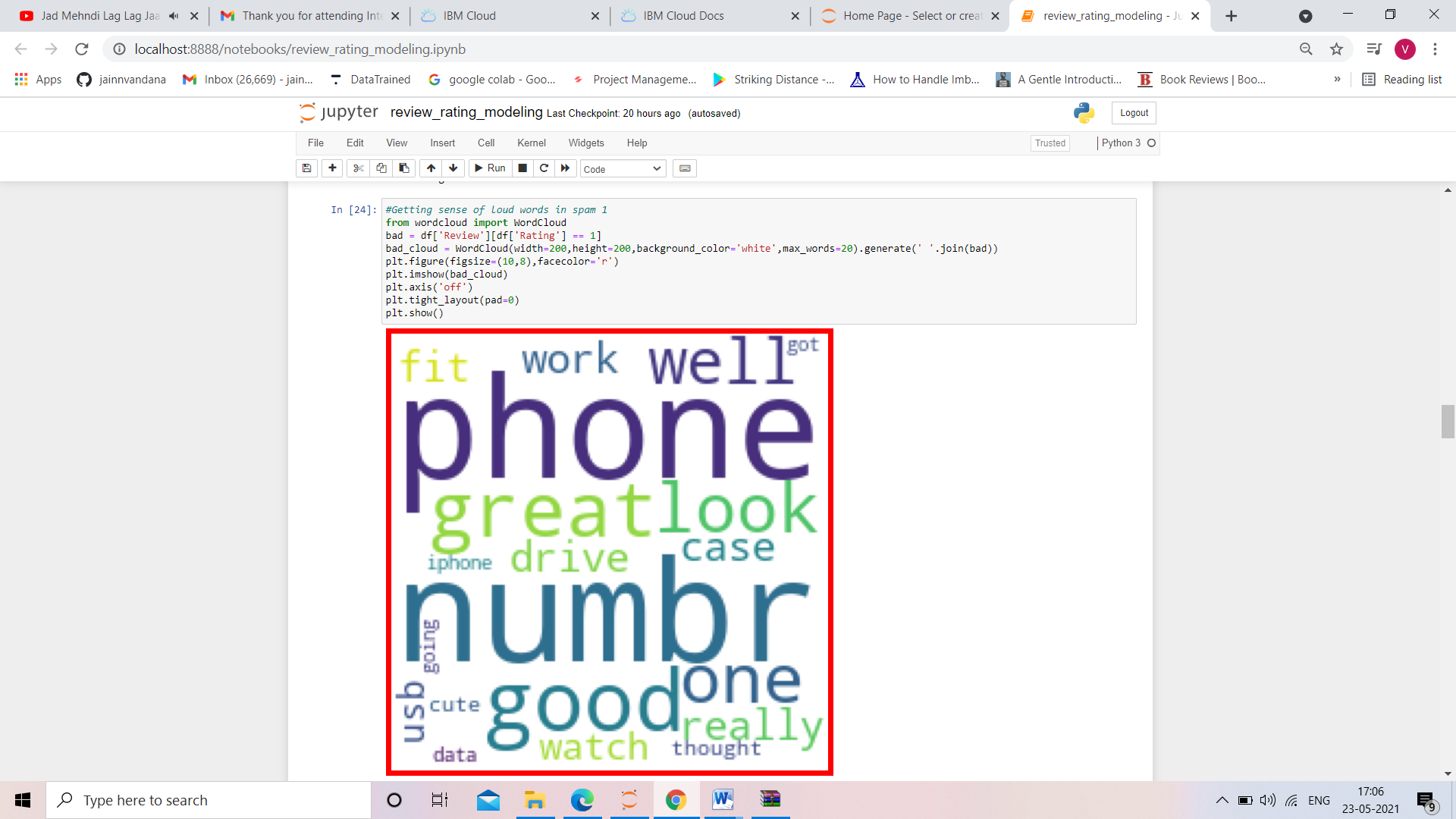
Sklearn documentation has provided a sample code of how to plot nice looking confusion matrix to visualize your result..

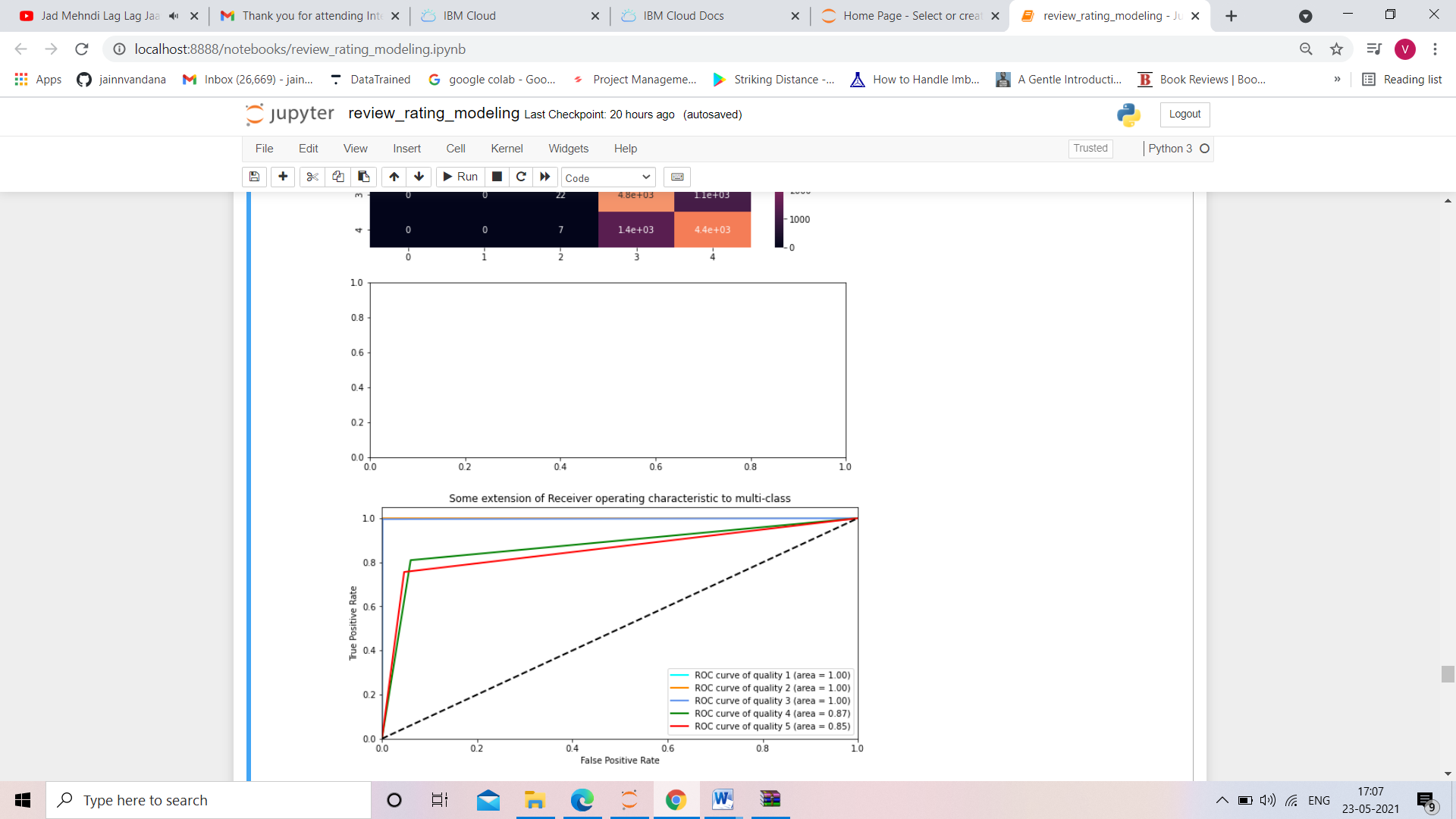
Confusion Matrix of the result



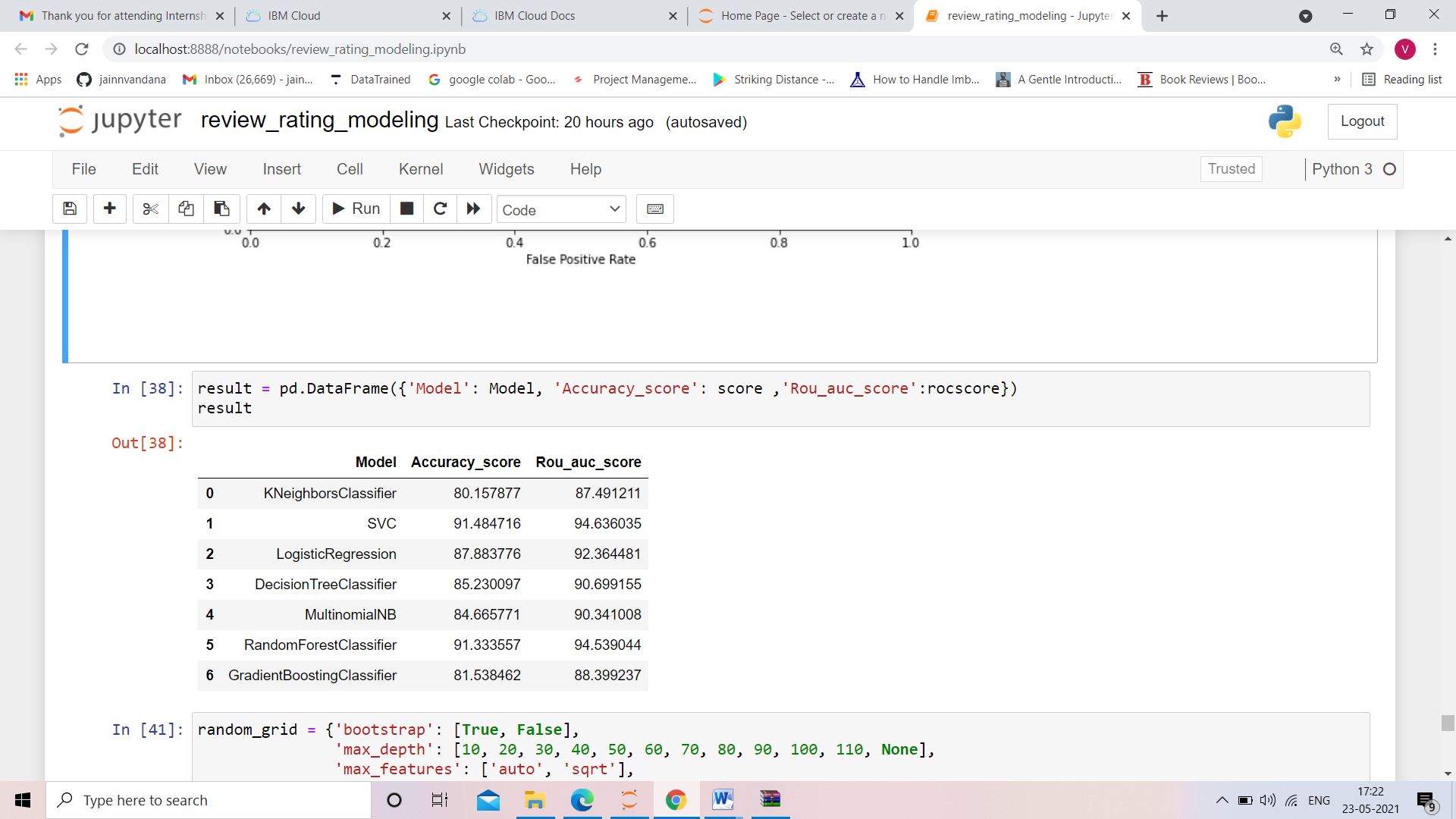
Visualizations



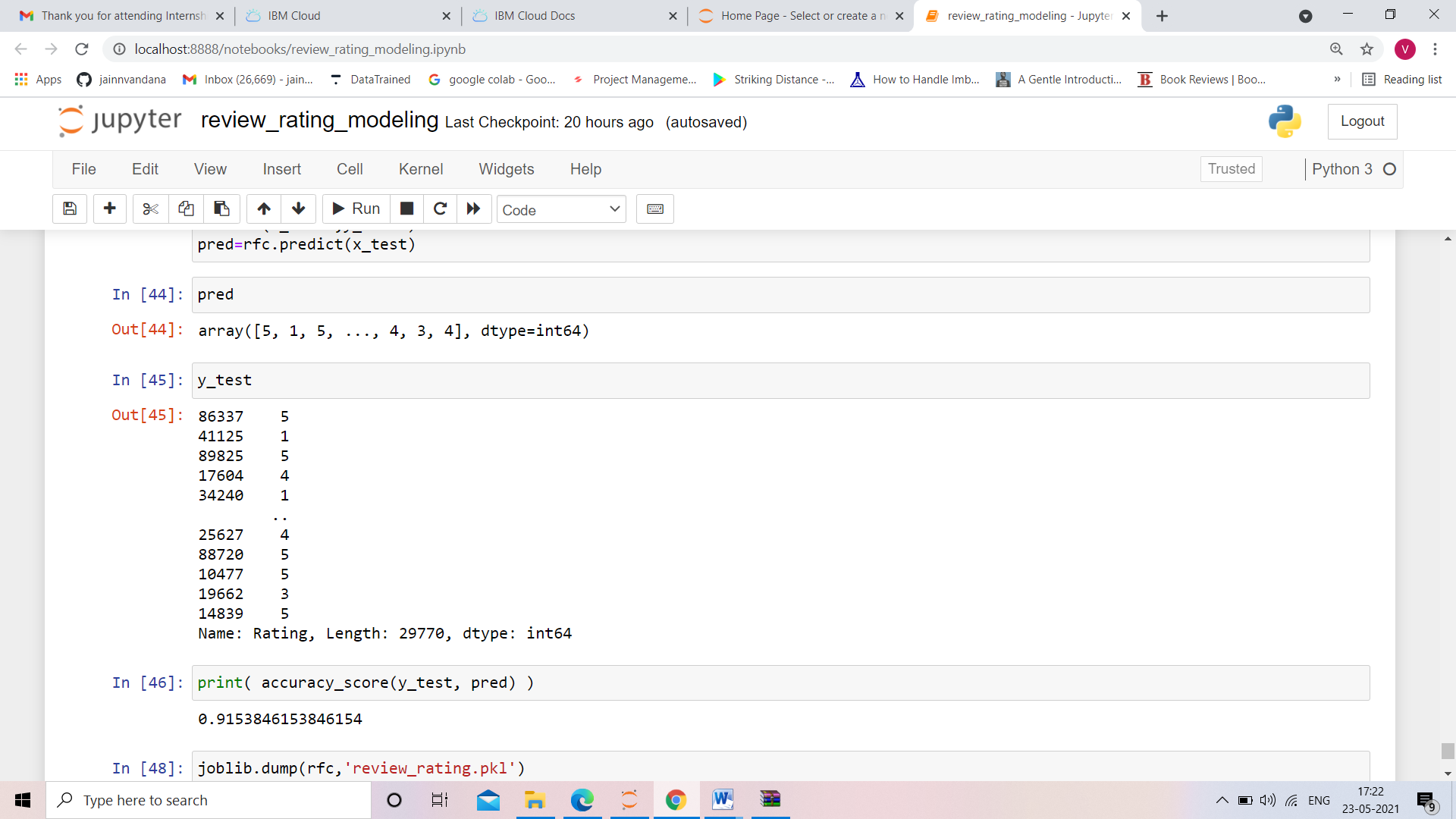




* Interpretation of the Results



**CONCLUSION**



As we can see from the pic that the predicted values are almost same as the y\_test value and accuracy score of random forest classifier is 91.53 % . which is quite acceptable. Here, in modelling we have used the smote oversampling technique to treat the inbalancing in our dataset and for hyperparameter tunning we have used Random search cv. As grid search cv takes more time and more expensive to use so we used the Random Search CV with Random Forest Classifier.