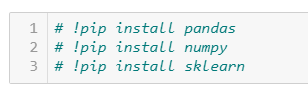
User guide for Problem 3: Customer Risk Prediction Model

**Prerequisites**

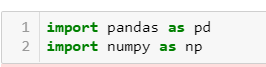
1. Python 3
2. Jupyter Notebook
3. Visual Studio Code
4. GitHub Account

**Section 1: Data Cleaning**

1. Open Jupyter notebook
2. Open Data Cleaning.ipynb
3. Uncomment and run the first cell to install pandas, numpy, sklearn.

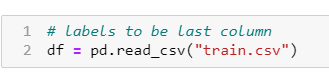


1. Run the 2nd cell to import pandas and numpy packages

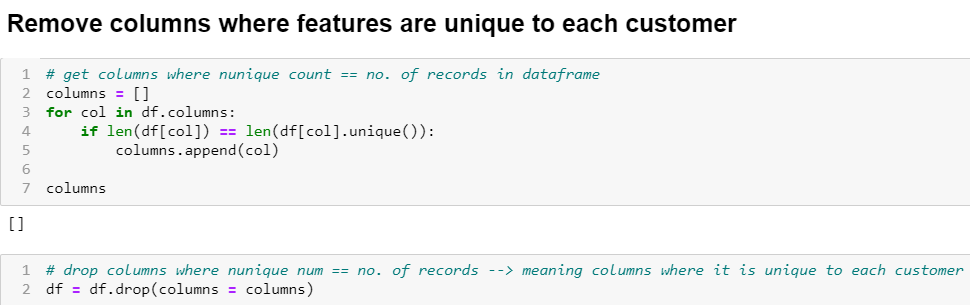


1. Import the dataset to be cleaned. Change “train.csv” to the name of the file. This can be both the training file and testing file.

Note that for training file, labels need to be the at the right most column

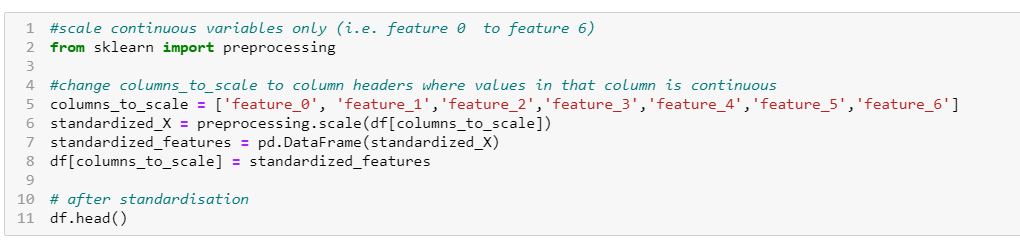


1. Continue running the cells to check how many records there are in the dataset, number of rows that contain NA values and drop these NA values.
2. Check for features that are unique to each customer using df.nunique()
   1. If there are features that are unique to each customer for eg. CustomerID, remove them by running the next 2 cell.

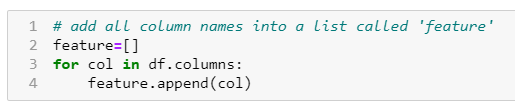


1. Check that columns that have values unique to each customer has been removed by running the next cell, df.head()
2. Next, we will be performing data standardization. Data standardization should only be applied to column where values are **continuous (**decimal for eg. 1.23) NOT discrete (whole number for eg. 0,1,2,3)

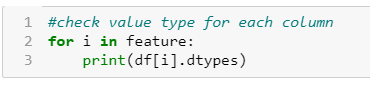
Note: Change columns\_to\_scale to column headers where its values are continuous



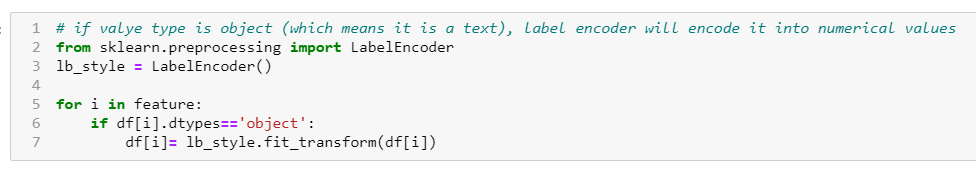
1. Next, we will perform label encoding to convert textual data to numerical data as machine learning models cannot work with text.
   1. Add all the columns name into a list called feature



* 1. Check value type of each column



* 1. If value type of column is ‘object’, it will be encoded into numbers



* 1. Check that textual values have been changed into numbers by running df.head()

1. Export the cleaned dataset. You can rename the file to any name you prefer by changing ‘train\_cleaned’. Note that the extension ‘.csv’ is required.

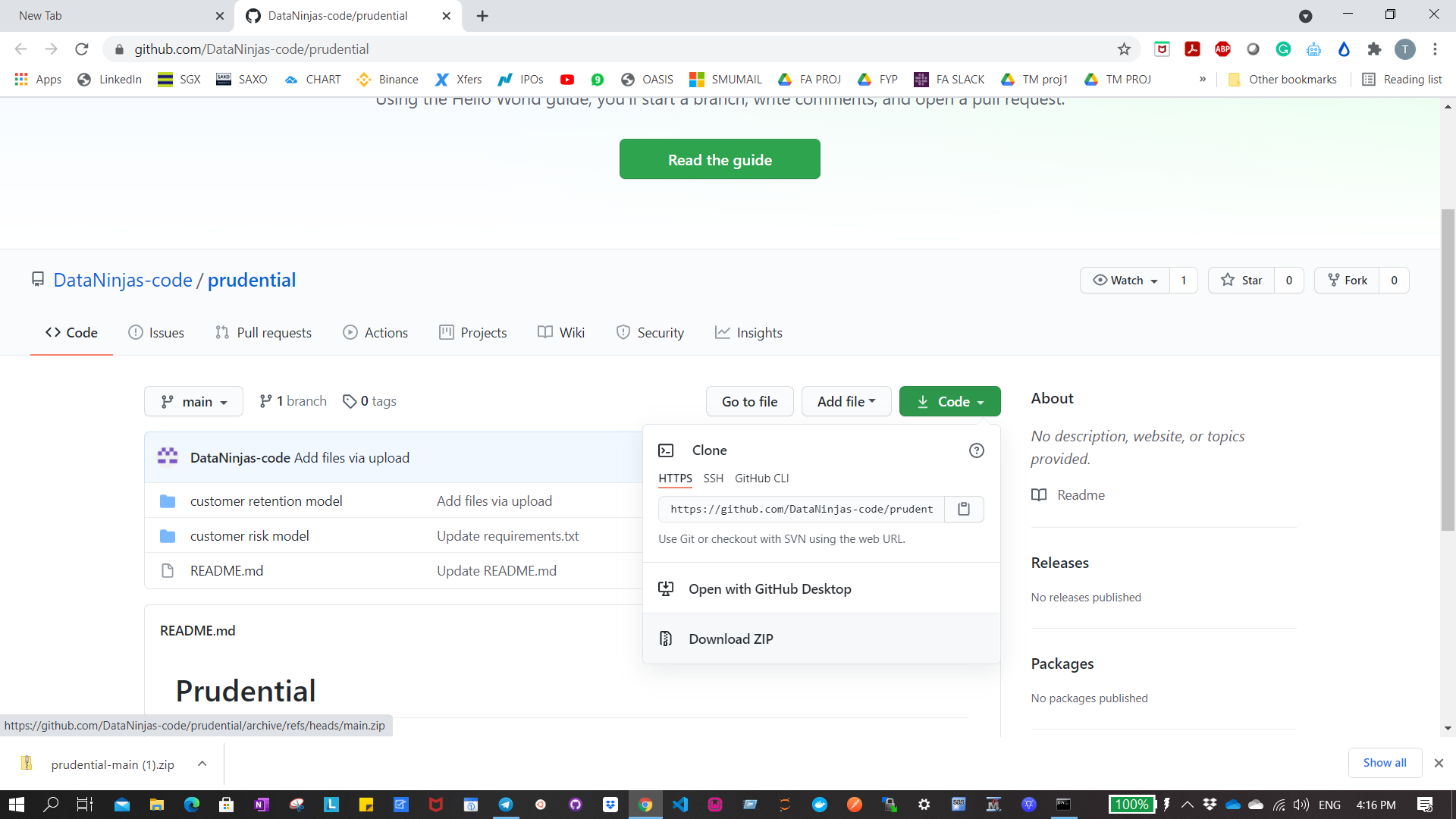


1. Perform data cleaning for both the train and test file and export them so that they can be uploaded into the flask model (next section).

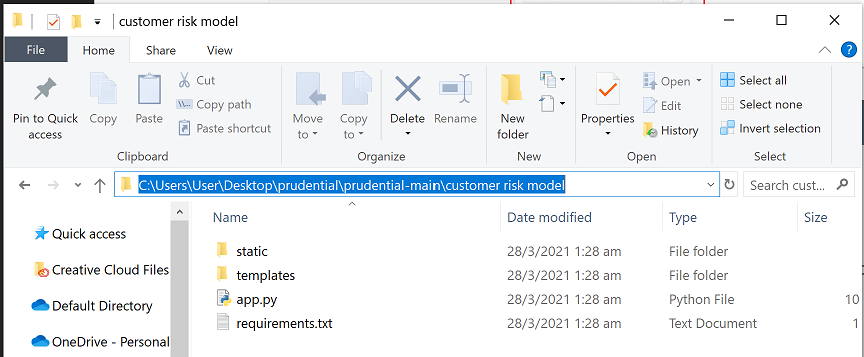
**Section 2: Customer Risk Prediction Model**

1. Go to <https://github.com/DataNinjas-code/prudential>

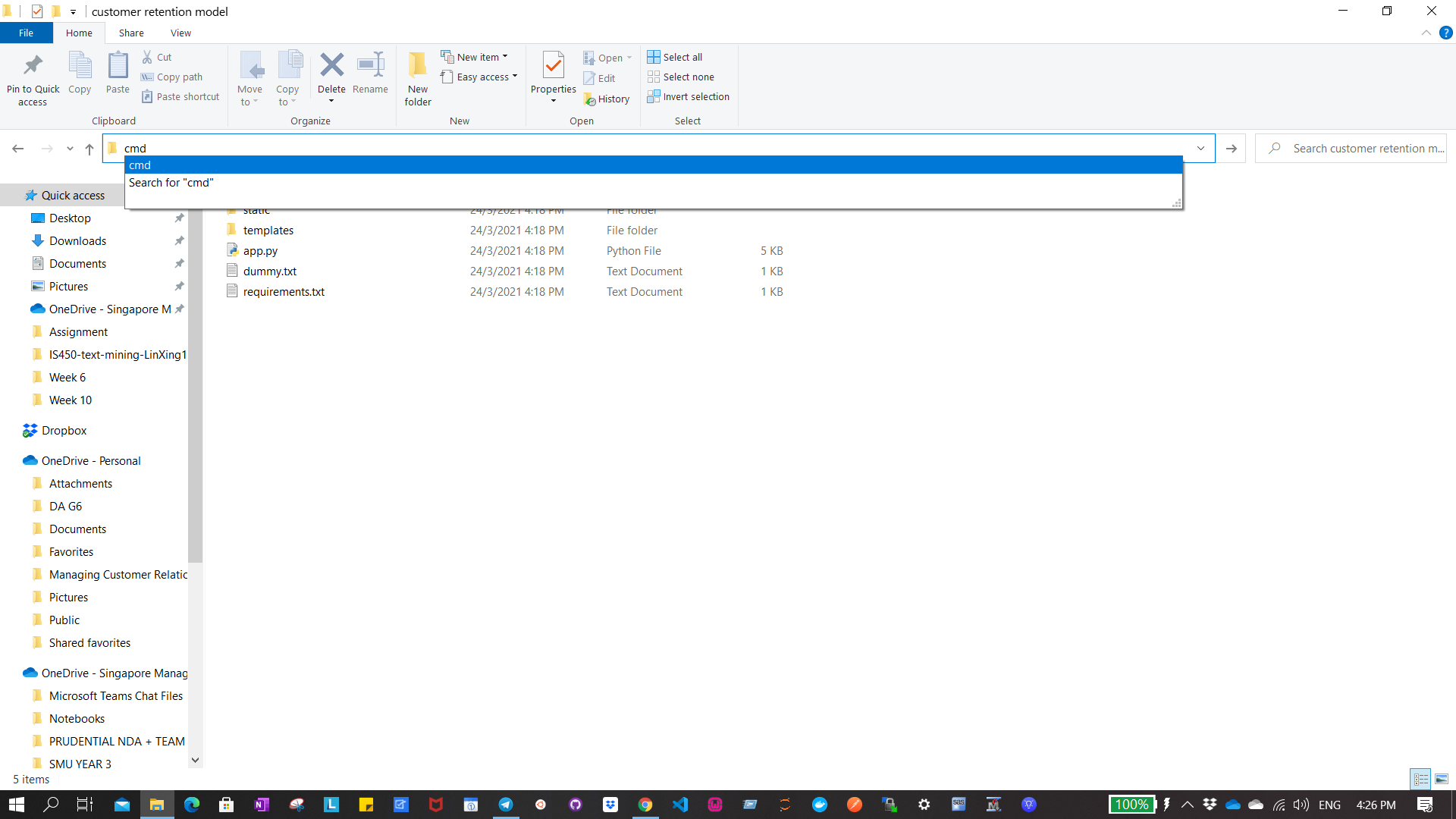
* Download the file



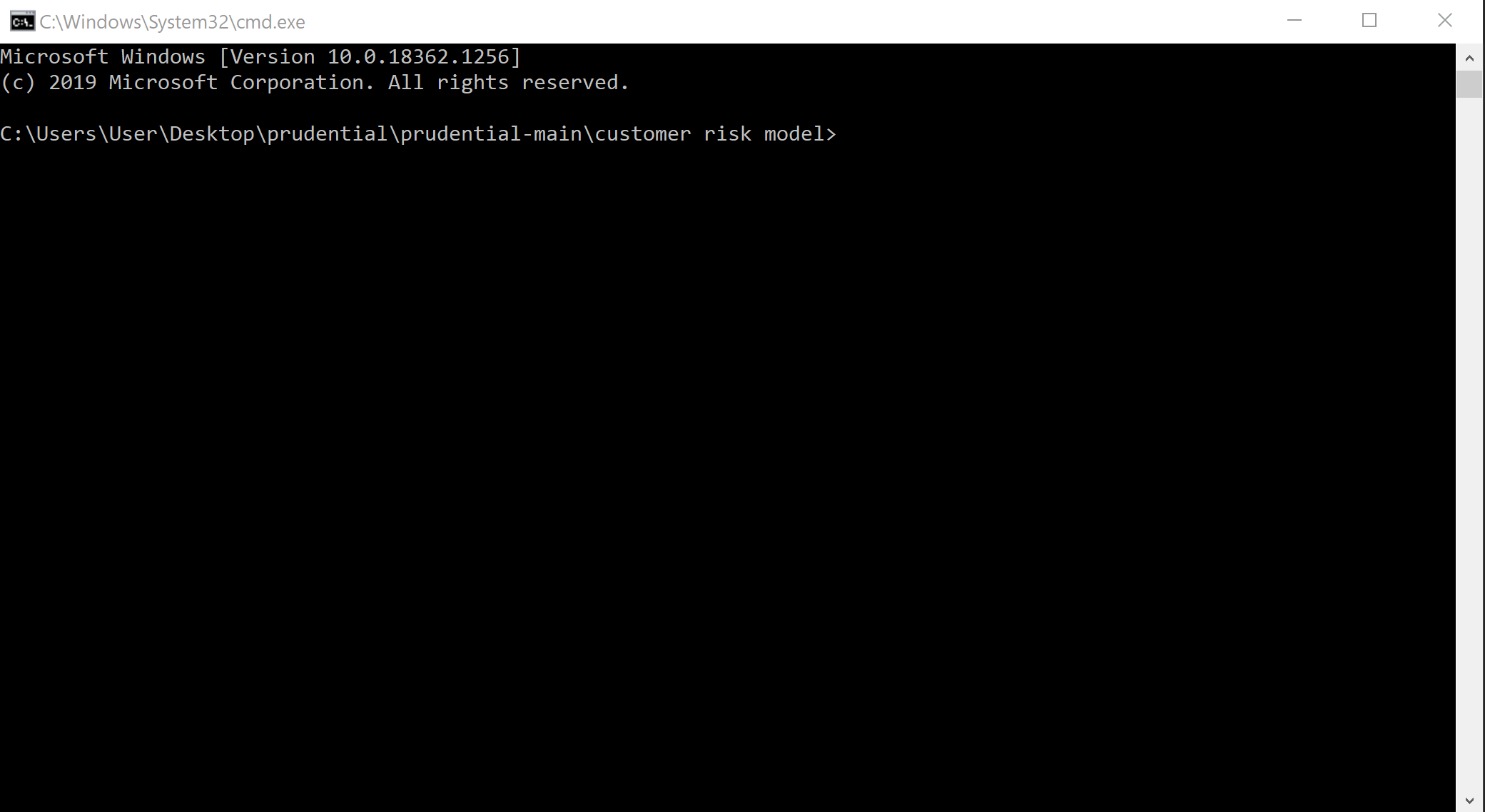
* Extract the contents in the zip file
* Click into the folder and into ‘customer retention model’
* Click on the file path (make sure your file path is similar to the screenshot)



* Type ‘cmd’ and press enter

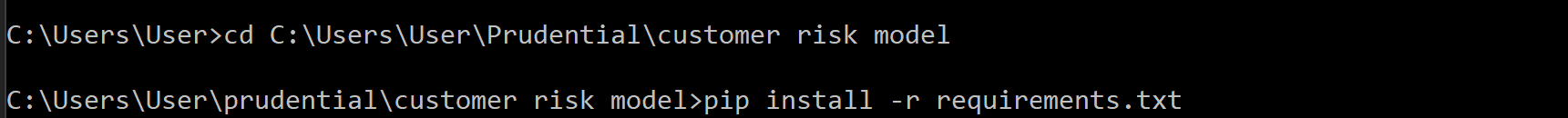


* You will see your command prompt being opened up.



1. Pip install all the required packages using requirements.txt

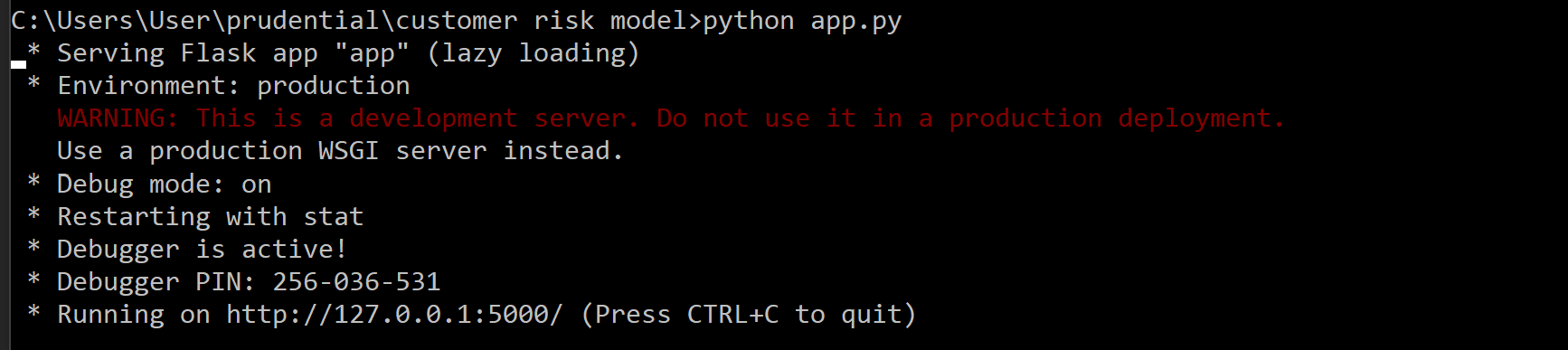
pip install -r path/to/requirements.txt



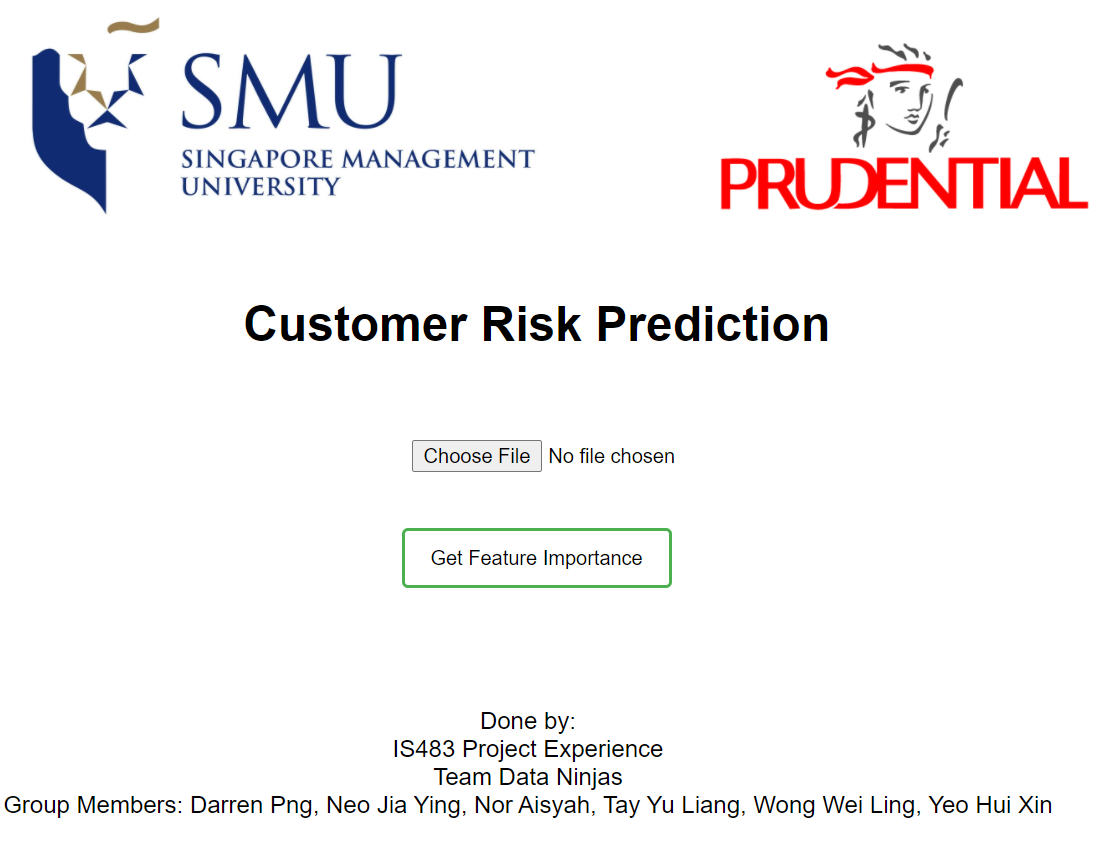
1. Run the Flask App

python app.py

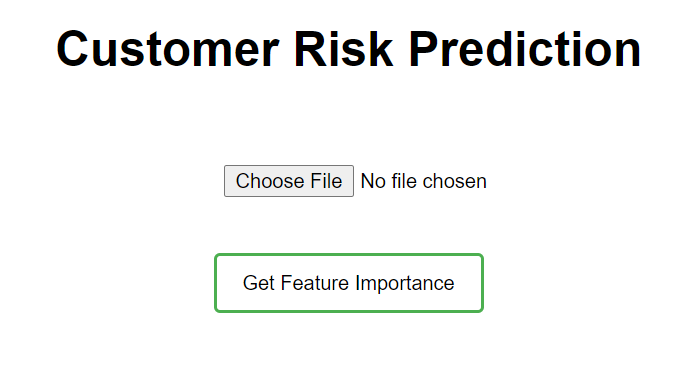
You should see this which means the flask app is live and running.



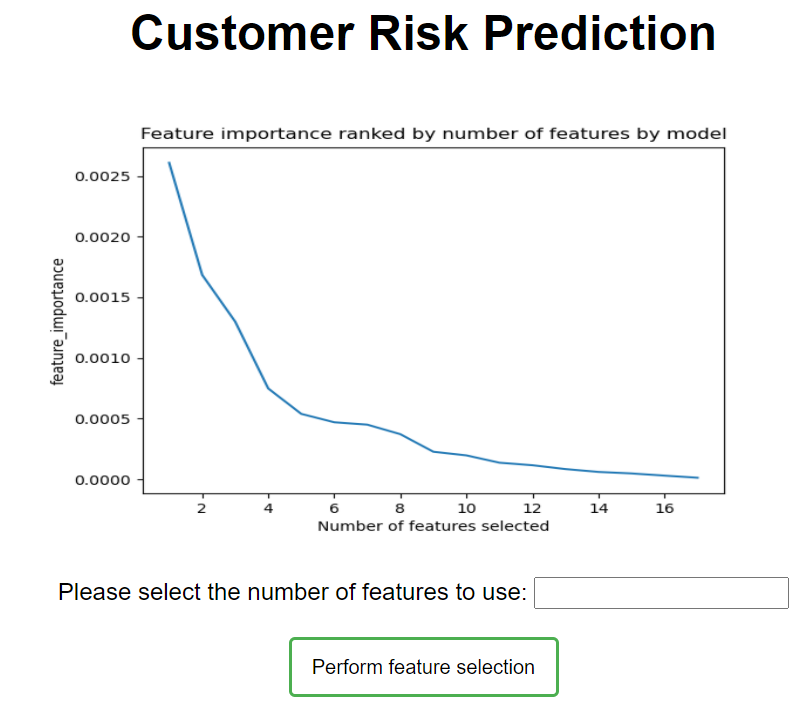
1. Go to the link: <http://127.0.0.1:5000/> and you should see this page below:



1. Upload cleaned training file by clicking on the ‘Choose File’ button.



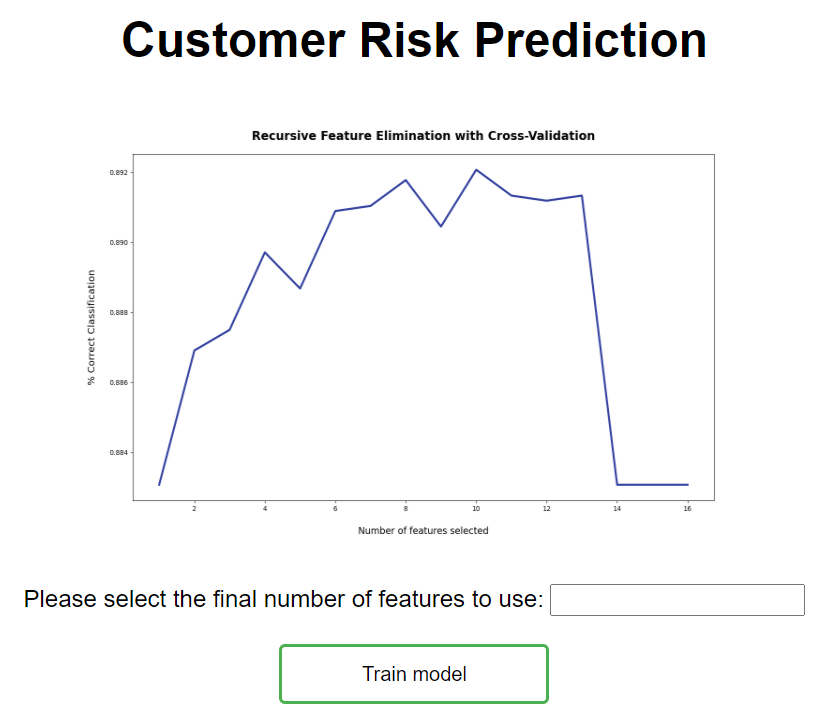
1. Click on “Get Feature Importance” button to run feature importance ranking across all the features. A feature importance graph will be shown, and you will need to select the number of features to use to run feature selection. This is because feature selection is an expensive process and it would be better to cut down unimportant features first at the start.



Input the number of features to use and click on the “feature selection” button.

Note: I*t may take awhile for the feature selection to run, depending on the number of features you have. Took around 40 minutes to cut from 120 features to 60 features.*

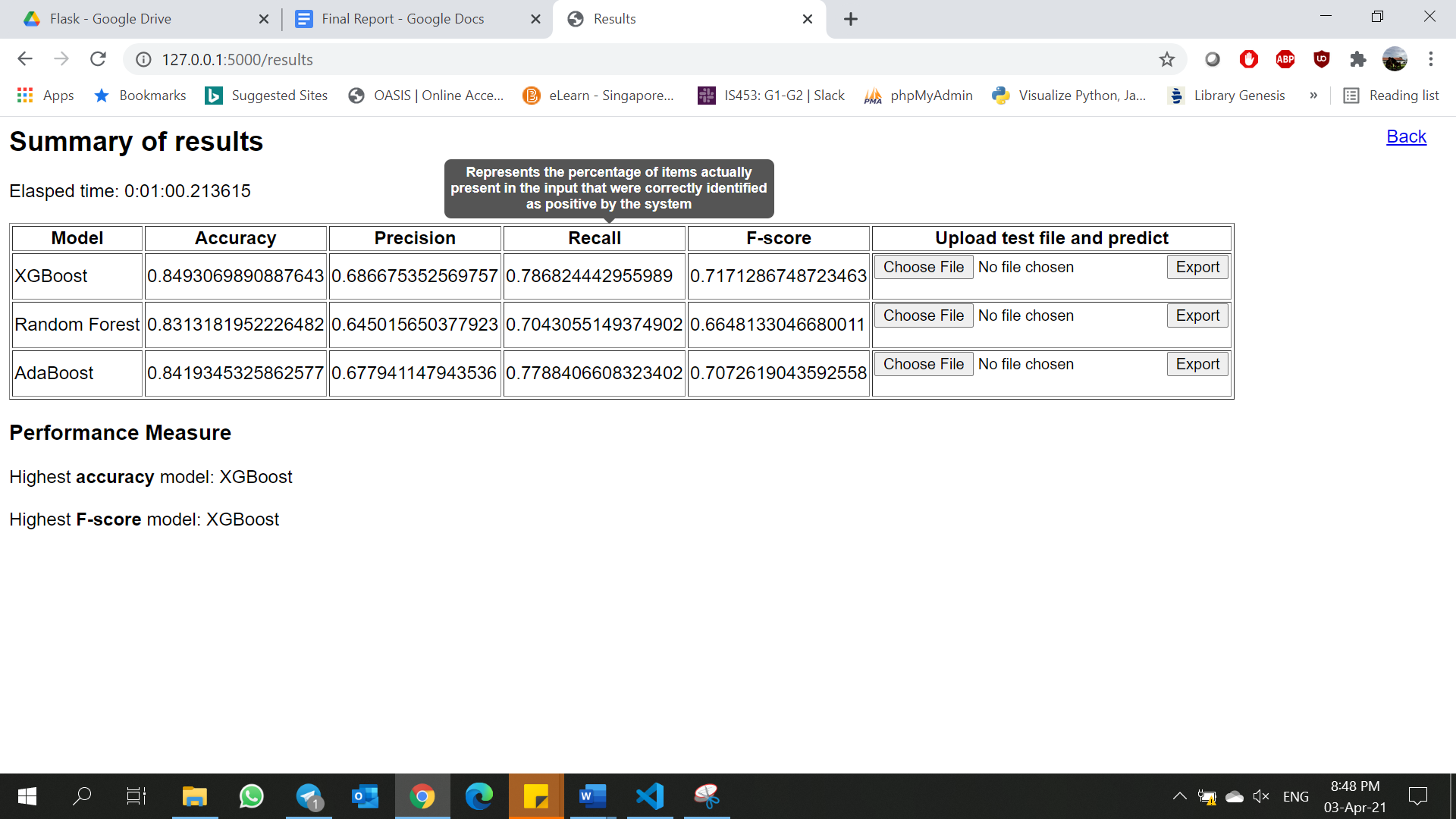
1. A feature selection graph would be shown once feature selection is done.



From the graph, you can pick the final number of features you want to use to train your model with based on the number of features that gives the highest % of correct classification. Input the final number of features to use and click on “Train model” to start model training.

1. After model has finished training, you will be bought to the results page. At this page, you can see the accuracy, precision, recall and f-score of the 3 models – AdaBoost, XGBoost and Random Forest.

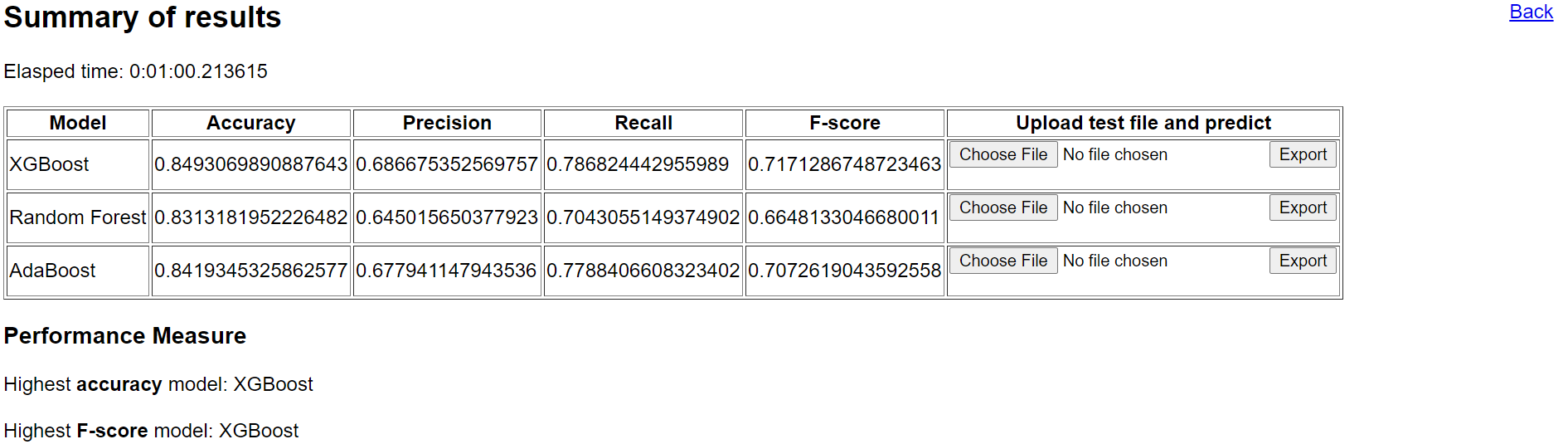
\*\* If you hover over the metric name, there is a tooltip that shows the meaning of the metric.



1. To choose the best model to predict the labels on your test file:

For the model chosen:

* + 1. Upload test file
    2. Click “export”



1. Once you click on ‘export’, a file will be downloaded. This file contains the labels predicted by the respective model on the test file you uploaded.