Programming Assignment 4: Naïve Bayes

Instructions:

- The aim of this assignment is to give you an initial hands-on regarding real-life machine learning application.
- Use separate training and testing data as discussed in class.
- You can only use Python programming language and Jupyter Notebook.
- You can only use **numpy**, **matplotlib** and are not allowed to use **NLTK, scikit-learn or any other machine learning toolkit**.
- **Submit your code as one notebook file (.ipynb) on LMS. The name of file should be your roll number.**
- Deadline to submit this assignment is: **Tuesday 21st April, 2020 11:55 p.m.**

Problem:

The purpose of this assignment is to get you familiar with multinomial sentiment classification. By the end of this assignment you will have your very own "Sentiment Analyzer". You are given with Twitter US Airline Sentiment Dataset that contains around 14,640 tweets about airlines labelled as positive, negative and neutral. Your task is to train a Naïve Bayes classifier on this dataset.

Dataset Splitting:

Use the same train/ test split as in programming assignment 3. If you haven't attempted programming assignment 3 here is how to do it:

Instead of a usual random split, you will split the dataset in a stratified fashion. Stratified splitting ensure that the train and test sets have approximately the same percentage of samples of each target class as the complete set. For example, in an 80-20 stratified split 80% samples of each class will be in train set and 20% in test set.

Implement stratified split and do the 80-20 train-test split of the provided dataset.

Implementation:

Implement Naïve Bayes keeping in view all the discussions from the class lectures. Feel free to read Chapter 4 (Section 4.1, 4.2, 4.3) of Speech and Language Processing book to get in-depth insight of the Naïve Bayes classifier.

Use the procedural programming style and comment your code thoroughly (just like programming assignment 1).

Evaluation Report:

You are required to provide a confusion matrix with values obtained by running your Naïve Bayes classifier on test set. Also report macro average (Precision, Recall, Accuracy, and F1) scores.