Instructions. There are two parts in this assignment. The first part involves a series of theory questions and the second part involves coding. The goal of this problem set is to make you understand and familiarize with Naive Bayes algorithm.

PART I: Theory Questions

MLE

• Suppose you have N samples \( x_1, x_2, \ldots, x_N \) from a univariate normal distribution with unknown mean \( \mu \) and known variance \( \sigma^2 \). Derive the MLE estimator for the mean \( \mu \).

• Consider a dataset \( (x^n, c^n), n = 1, \ldots, N \) of binary attributes, \( x^n_i \in \{0, 1\}, i = 1, \ldots, D \) and associated class label \( c^n \). The number of datapoints from class \( c = 0 \) is denoted \( n_0 \) and the number from class \( c = 1 \) is denoted \( n_1 \). Estimate \( p(x_i = 1 | c) \equiv \theta^c_i \).

• A training set consists of one dimensional examples from two classes. The training examples from class 1 are \( \{0.5, 0.1, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.35, 0.25\} \) and from class 2 are \( \{0.9, 0.8, 0.75, 1.0\} \). Fit a (one dimensional) Gaussian using Maximum Likelihood to each of these two classes. You can assume that the variance for class 1 is 0.0149, and the variance for class 2 is 0.0092. Also estimate the class probabilities \( p_1 \) and \( p_2 \) using Maximum Likelihood. What is the probability that the test point \( x = 0.6 \) belongs to class 1?

Naive Bayes

• Whizzco decide to make a text classifier. To begin with they attempt to classify documents as either sport or politics. They decide to represent each document as a (row) vector of attributes describing the presence or absence of words.

\[ x = (\text{goal, football, golf, defence, offence, wicket, office, strategy}) \]

Training data from sport documents and from politics documents is represented below in a matrix in which each row represents the 8 attributes.

\[
x_{\text{politics}} = \begin{bmatrix} 1 & 0 & 1 & 0 \ & 1 & 1 & 0 \ & 0 & 0 & 1 & 0 \ & 1 & 0 & 1 & 0 \ & 0 & 1 & 0 & 1 \ & 0 & 0 & 0 & 0 \ & 0 & 0 & 1 & 1 \ & 0 & 0 & 0 & 1 \ \end{bmatrix}
\]
Using a maximum likelihood naive Bayes classifier, what is the probability that the document $x = (1, 0, 0, 1, 1, 1, 1, 0)$ is about politics?

**PART II: Sentiment Analysis**

In this part of the assignment, you will try to predict the class of tweets that are given. You will implement a Naive Bayes classifier and verify its performance on a twitter dataset. As you learned in class, Naive Bayes is a simple classification algorithm that makes an assumption about conditional independence of features, but it works quite well in practice. You will implement the Naive Bayes algorithm to classify a news corpus into 2 different categories. You have been provided with a corpus contains tweets with positive or negative labels. You will try to implement Naive Bayes algorithm to predict the test tweet’s label.

**Corpus**

- It contains 1,578,627 classified tweets, each row is marked as 1 for positive sentiment and 0 for negative sentiment.
- Using 1/10 of the corpus for testing your algorithm is recommended, while the rest can be dedicated towards training to classify sentiment.
- Each row contains,
  - Tweet ID,
  - Label,
  - Sentiment Source,
  - Tweet information. You will use label and tweet information for classification assignment.
- You can use csv library to read corpus.

**Approach**

You will represent your data with listed approaches and use them to learn a classifier via Naive Bayes algorithm. You have to implement each feature extraction method and Naive...
Bayes algorithm.

- Features
  You will use Bag of Words (BoW) model which learns a vocabulary from all of the documents, then models each document by counting the number of times each word appears. You will implement BoW with two options:
  - Unigram: The occurrences of words in document (frequency of the word).
  - Bigram: The occurrences of two adjacent words in document.

- Suggestion for Implementation
  - You may encounter words during classification that you haven’t during training. This may be for a particular class or over all. Your code should deal with that.
    *Hint: You can use Laplace smoothing*

  - You have to use a dictionary for BoW representation. You can use all tweets for that purpose or you can make different assumptions to generate more efficient dictionary. For example, you can narrow down your dictionary by choosing specific words for positive and negative labels. In other words, your classification results can be improved by selecting a subset of extremely effective words for dictionary.
  - You can analyze TF-IDF.
  - You can use $\log(\text{frequency}(w))$ instead of $\text{frequency}(w)$
  - You can remove duplicates in each tweet.
  - Analyze existing works to find an improvement.
  - You can use different features after you implement BoW.

You will report your average result for 10-fold cross validation.

Grading

- Code (50): 30 points for baseline (Naive Classifier with BoW), 20 points for improvements.

**Notes for the report**: You should analyze features and the method you employed. How did you improve your results? Explain every step you choose.

Late Policy

You may use up to three extension days (in total) over the course of the semester for the three problem sets you will take. Any additional unapproved late submission will be weighted by 0.5.
Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else’s work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.