CS 6375- ASSIGNMENT 4

Please read the instructions below before starting the assignment.

- This assignment consists of two parts – the first one requires written answers and the second one requires programming. For the written part, you can submit typed solution or legible hand written one. If TA cannot read your solution, you will not be given any credit.

- Please place the solutions in different folders titled parti and partii

- In the code folder, please include a README file indicating how to compile and run your code. Also, mention clearly which language and packages you have used.

- You should use a cover sheet, which can be downloaded at:
  http://www.utdallas.edu/~axn112530/cs6375/CS6375_CoverPage.docx

- You are allowed to work in pairs i.e. a group of two students is allowed. Please write the names of the group members on the cover page.

- You have a total of 4 free late days for the entire semester. You can use at most 2 days for any one assignment. After that, there will be a penalty of 10% for each late day. The submission for this assignment will be closed 2 days after the due date.

- Please ask all questions through Piazza, and not through email.

Part I

1. Probability [4 points]

Suppose we have two random variables, both defined over all students in CS 6375.

- w_h = worked hard for the course
- g_a=got an A

Assume we know from previous offerings of the course that:

- $P(w_h)=0.85$
- $P(g_a)=0.95$
- $P(g_a|w_h) = 0.99$

(a) Given that a student got an A, what is the probability he or she worked hard for the course?

(b) Given that a student didn't work hard for the course, what is the probability that he or she got an A?

Show and explain your work for both (a) and (b).

2. Probability [5 points]

Suppose you are a witness to a nighttime hit-and-run accident involving a taxi in Honolulu. All taxis in Honolulu are blue or green. You swear, under oath, that the taxi was blue. Extensive testing shows that, under the dim lighting conditions, discrimination between blue and green is only 75% reliable.

Calculate the most likely color for the taxi? If so, show your calculations. If not, explain why not. You can assume prior of green taxi is 0.9 and prior of blue taxi is 0.1.
(Hint: distinguish between the proposition that the taxi *is* blue and the proposition that it *appears* blue)

3. Probability [4 points]

Jim is a CS 6375 student. Recently, his mood has been highly influenced by three factors: the weather (W), his study habits (S), and whether his neighbor is at home or not (N). We want to predict his happiness according to these three factors using previous observations. The table below shows this data.

Weather (W)	Study (S)	Neighbor (N)	Happy (H)
Bad	Fail	Home	No
Good	Fail	Out	No
Good	Fail	Out	No
Good	Fail	Out	No
Bad	Pass	Home	No
Bad	Pass	Home	Yes
Bad	Pass	Home	Yes
Good	Pass	Out	Yes

(a) On a new day when W=Good, S=Pass, and N=Out, how would we predict his happiness using a Naive Bayes classifier? Show your calculations.

(b) On the day when W=Good, S=Pass, and N=Out, how would we predict his happiness using a Bayes classifier instead? Show your calculations.

4. Probability [3 points]

Below are some statistics on the usage of programming languages in software companies:

- 50% of all programmers can program in C++.
- 40% of all programmers can program in Java.
- 1% of all programmers work for Macrosoft
- 99% of Macrosoft employees can program in C++.
- 98% of Macrosoft employees can program in Java.

Using naive Bayes reasoning, decide if a programmer who knows both C++ and Java is a Macrosoft employee. Show your calculations.

5. Logistic Regression [12 points]

Read the new chapter of Tom Mitchell's book on Logistic Regression available from:
https://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf

Solve question 3 in the exercises. Be sure to look at the hints before solving.

6. Logistic Regression [12 points]

In the class, we had derived the equation for logistic regression assuming that the data for each class follows a Gaussian distribution. Now, let's assume that data for each class follows an exponential distribution described by the following equation:

$$P(X_i|Y=j) = h\big(X_i, \theta_{ij}\big)\exp(-\theta_{ij}^T X_i + c)$$

You can assume that there are two classes - j = 0 and j = 1. You can also assume feature independence within each class.

We would like to write the equation for posterior of class 1 given data feature
X = (X_1, X_2, …, X_n) as follows:

$$P(Y=1|\boldsymbol{X}) = \frac{1}{1+\exp(-z)}$$

where

$$z = w_0 + \sum w_i X_i$$

Find out the values of w_0 and w_i in terms of distribution parameters.

Part II

Naïve Bayesian Classifier Implementation [60 points]

In this part, you will implement the naïve Bayes algorithm for text classification tasks. The version of naïve Bayes that you will implement is called the multinomial naïve Bayes (MNB). The details of this algorithm can be read from chapter 13 of the book "Introduction to Information Retrieval" by Manning et al. This chapter can be downloaded from:
http://nlp.stanford.edu/IR-book/pdf/13bayes.pdf
Read the introduction and sections 13.1 and 13.2 carefully. The MNB model is presented in Figure 13.2 Note that the algorithm uses add-one Laplace smoothing. Make sure that you do all the calculations in log-scale to avoid underflow as indicated in equation 13.4.

To test your algorithm, you will use the 20 newsgroups dataset, which is available for download from here: http://qwone.com/~jason/20Newsgroups/
You will use the "20 Newsgroups sorted by date" version. The direct link for this dataset is:
http://qwone.com/~jason/20Newsgroups/20news-bydate.tar.gz
This dataset contains folders for training and test portions, with a sub-folder for different classes in each portion. For example, in the train portion, there is a sub-folder for computer graphics class, titled "comp.graphics" and a similar sub-folder exists in the test portion. **To simplify storage and memory requirements, you can select any 5 classes out of these 20 to use for training and test portions**.

In the naïve Bayes classifier, each unique word is a distinct feature in itself. You can make the following simplifying assumptions:
- each word is independent of the other
- stop words have no role in classification and you can ignore them.

Note: You can find a list of stop words here: https://www.ranks.nl/stopwords
or you can use Python's stop-words package: https://pypi.python.org/pypi/stop-words

As always, you have to train your algorithm using the training portion and test it using the test portion.

IMPORTANT:

• To implement the multinomial Naive Bayes classifier, you may use any of the following languages: Java, Python, C, C++. You cannot use any machine learning library, but are free to use any data loading or processing library. You have to specify the libraries used and their source clearly in the README file. If you have any doubts, first contact the TA and then post the question on Piazza.

. • Your program should be able to read all files from 5 sub-folders from the training portion and create a naïve Bayes model. You will have to ignore the stop words before creating the naïve Bayes model. The model will be tested on test files from the same classes on the test portion and accuracy on the test dataset should be outputted by your program. Your program should exclude the header portion from each file. Generally, this means excluding lines from start of the file up to the line starting with "Lines: xxx"

• Your program should allow exactly two arguments to be specified in the command line invocation of your program: location of training root folder and test root folder. Your program should be able read the same 5 sub-folders from each folder. Then create the model using the training data and test the model on the data in the test sub-folders.

• Your program should output accuracy obtained on the test portion to the stdout.

• Submit a README file containing results and analysis of the results.