CS 6375- ASSIGNMENT 3

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.

- The deadline for this assignment is Sunday October 23 at 11:59 PM. No extensions are allowed.

- 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.

(a) Is it possible to calculate the most likely color for the taxi? If so, show your calculations. If not, explain why not. (Hint: distinguish between the proposition that the taxi *is* blue and the proposition that it *appears* blue.)

(b) Is it possible to calculate the most likely color for the taxi, given that 9 out of 10 taxis in Honolulu are green? If so, show your calculations. If not, explain why not.

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 Microsoft 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 [10 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.

Part II

Naïve Bayesian Classifier Implementation [24 points]

In this problem, you will implement the Naive Bayes learning algorithm for binary classification tasks (i.e. each instance will have a class value of 0 or 1). To simplify the implementation, you may assume that all attributes are binary-valued (i.e. the only possible attribute values are 0 and 1) and that there are no missing values in the training or test data.

A sample training file (train.dat) and test file (test.dat) are available from the assignment page on eLearning. The same files for Windows users are also provided (train-win.dat and test-win.dat). In these files, only lines containing non-space characters are relevant. **The first relevant line holds the attribute names**. Each following relevant line defines a single example. Each column holds this example's value for the attribute named at the head of the column. The last column (labeled "class") holds the class label for the examples.

IMPORTANT:

• To implement the Naive Bayes classifier, you may use any of the following languages:
R, Python, C, C++, C#, Visual Basic, Java. You are also free to use any package of these languages provided you specify it 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 handle any binary classification task with any number of binary-valued attributes. Consequently, both the number and names of the attributes, as well as the number of training and test instances, should be determined at runtime. In other words, these values should not be hard-coded in your program. You can assume that the class attribute would always be the last one.

• Your program should allow exactly two arguments to be specified in the command line invocation of your program: a training file and a test file. Your program should take these two arguments in the same order as they are listed above. There should be no graphical user interface (GUI). Any program that does not conform to the above specification will receive no credit.

WHAT TO DO:

1. Train the Naive Bayes learning algorithm on the training instances. Print to stdout the parameters of the classifier that were estimated from the training data (i.e., the class priors and the class-conditional probabilities). To exemplify, assume that the class attribute is named C and has two possible values, c1 and c2; furthermore, assume that there are three attributes A1, A2, and A3, where A1 has two possible values (x1 and x2), A2 has two possible values (y1 and

y2), and A3 has three possible values (z1, z2, and z3). The learned 3 parameters should be printed according to the following format:

P(C=c1)=0.32 P(A1=x1|c1)=0.33 P(A1=x2|c1)=0.67 P(A2=y1|c1)=0.25 P(A2=y2|c1)=0.75 P(A3=z1|c1)=0.26 P(A3=z2|c1)=0.49 P(A3=z3|c1)=0.25

P(C=c2)=0.68 P(A1=x1|c2)=0.22 P(A1=x2|c2)=0.78 P(A2=y1|c2)=0.91 P(A2=y2|c2)=0.09 P(A3=z1|c2)=0.16 P(A3=z2|c2)=0.74 P(A3=z3|c2)=0.10

In other words, all the probabilities associated with a particular class should appear in the same line.

2. Use the learned classifier to classify the training instances. Print to stdout the accuracy of the classifier. The accuracy should be computed as the percentage of examples that were correctly classified. For example, if 86 of 90 examples are classified correctly, then the accuracy of the classifier would be 95.6%.

Accuracy on training set (90 instances): 95.6%

3. Use the learned classifier to classify the test instances. Print to stdout the accuracy of the classifier.

Accuracy on test set (10 instances): 60.0%

Hint: An excellent resource for NB classifier in R is available at:
https://eight2late.wordpress.com/2015/11/06/a-gentle-introduction-to-naive-bayes-classification-using-r/
Hint: If you want to know how to pass command line arguments to an R script, see this:
https://www.r-bloggers.com/passing-arguments-to-an-r-script-from-command-lines/