CS 540: Introduction to Artificial Intelligence

HW #5: Probabilistic Inference

Assigned: November 21
Due: December 8

Late Policy:

Homework must be handed in by noon on the due date and electronically turned in by this same time.

1. If it is 0 – 24 hours late, including weekend days, a deduction of 10% of the maximum score will be taken off in addition to any points taken off for incorrect answers.
2. If it is 24 – 48 hours late, including weekend days, a deduction of 25% of the maximum score will be taken off in addition to any points taken off for incorrect answers.
3. If it is 48 – 72 hours late, including weekend days, a deduction of 50% of the maximum score will be taken off in addition to any points taken off for incorrect answers.
4. If it is more than 72 hours late, you will receive a '0' on the assignment.
5. In addition, there are 2 'late days' you may use any time throughout the semester. Each late day has to be used as a whole – you can't use only 3 hours of it and "save" 21 hours for later use.

Collaboration Policy:

You are to complete this assignment individually. However, you are encouraged to discuss the general algorithms and ideas with classmates, TA, and instructor in order to help you answer the questions. You are also welcome to give each other examples that are not on the assignment in order to demonstrate how to solve problems. But we require you to:

2. not explicitly tell each other the answers
3. not to copy answers or code fragments from anyone or anywhere
4. not to allow your answers to be copied
5. not to get any code or help on the Web

In those cases where you work with one or more other people on the general discussion of the assignment and surrounding topics, we suggest that you specifically record on the assignment the names of the people you were in discussion with.

Question 1: [15] Reasoning with a Joint Probability Table

The following table gives probabilities for three Boolean random variables, X, Y, and Z:

	Y		$\neg Y$	
	Z	$\neg Z$	Z	$\neg Z$
X	0.70	0.015	0.10	0.02
$\neg X$	0.08	0.01	0.07	0.005

(a) What is $P(Y \mid X)$? Show your work.
(b) What is $P(Y \mid X, Z)$? Show your work.
(c) What is $P(Y)$? Show your work.
(d) What is $P(X, Z)$? Show you work.
(e) Is the data consistent with X and Z being independent? Briefly explain.

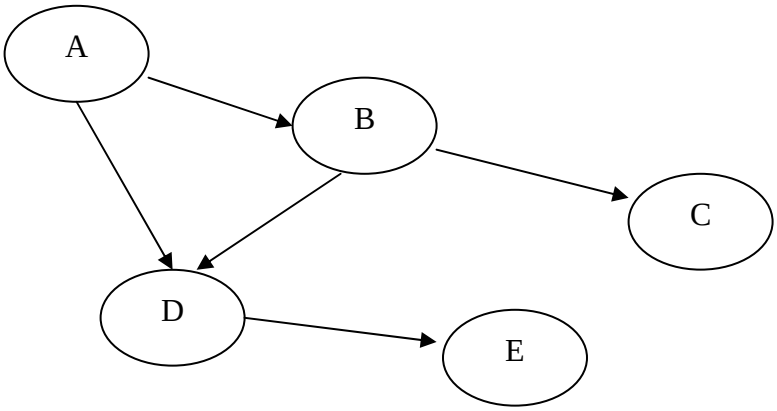
Question 2: [10] Bayes's Rule

Jack tells his professor that he forgot to print his homework assignment to hand in. From experience the professor knows that students with finished homework forget and tell her so about once in 100 times. She also knows that about half the students who have not finished their homework will tell her they forgot. She thinks about 90% of the students in this class completed their homework on time. What is the probability that Jack is telling the truth? Clearly show the way you formulate the problem, including the meaning of each random variables you introduce.

Question 3: [15] Answer questions using a Bayesian Network

Using the Bayesian Network below to answer the following questions using inference by enumeration. Show your work.

(a) $P(E \mid B, C)$
(b) $P(D \mid E, A)$



The CPTs are defined as follows:

- $P(A) = 0.8$
- $P(B \mid A) = 0.2$
- $P(B \mid \neg A) = 0.9$
- $P(C \mid B) = 0.95$
- $P(C \mid \neg B) = 0.5$
- $P(D \mid A, B) = 0.85$
- $P(D \mid A, \neg B) = 0.25$
- $P(D \mid \neg A, B) = 0.15$
- $P(D \mid \neg A, \neg B) = 0.05$
- $P(E \mid D) = 0.60$
- $P(E \mid \neg D) = 0.50$

Question 4: [20] Hidden Markov Models

Read the paper "Markov Models and Hidden Markov Models: A Brief Tutorial" by E. Fosler-Lussier (available on the class webpage) and answer the first three questions in the exercises on pages 6 and 7.

Question 5: [40] Language Identification with a Naïve Bayes Classifier

Naive Bayes is a simple, effective machine learning method that can be used to solve the problem of identifying the language of a document. You are to implement a Naive Bayes classifier that classifies a document as English, Spanish, or Japanese – all written with the 26 lower case letters and space.

The dataset for this assignment is Q5.tar.gz, available at the class website. This dataset consists of training and test documents in English, Spanish and Japanese. Both the training dataset and the test dataset contain three subdirectories: `English/`, `Spanish/`, and `Japanese/`. These subdirectories in turn contain the documents as separate ASCII text files. The data is therefore organized as follows:

```
trainingdataset/English/
trainingdataset/Spanish/
trainingdataset/Japanese/
testdataset/English/
testdataset/Spanish/
testdataset/Japanese/
```

We will be using a character-based Naïve Bayes model. You need to view each document as a stream of characters, including space. We have made sure that there are only 27 different types of characters (*a* to *z*, and *space*).

You must compute and store the prior probabilities, $P(English)$, $P(Spanish)$ and $P(Japanese)$, as well as the conditional probabilities, $P(c \mid English)$, $P(c \mid Spanish)$, and $P(c \mid Japanese)$, from the training set. Store all probabilities as logs to avoid underflow. This also means you need to do arithmetic in log-space. That is, multiplications of probabilities become additions of log probabilities. Hints are given at the end of this question.

You are required to complete the following tasks:

1. Using all the characters in the training data, build a Naive Bayes classifier for the three languages. Implement your classifier using a log-likelihood formulation of Naïve Bayes.

2. Print $P(English)$, $P(Spanish)$ and $P(Japanese)$, as well as the conditional probabilities $P(c \mid English)$, $P(c \mid Spanish)$, and $P(c \mid Japanese)$ for all 27 characters c.

3. Evaluate the performance of your classifier on the *test set* using a confusion matrix. A confusion matrix summarizes the types of errors your classifier makes, as shown in Table 1. The columns are the true language a document is in, and the rows are the classified outcome of that document. The cells are the number of test documents in that situation. For example, the cell with row = English and column = Spanish contains the number of test documents that are really Spanish, but misclassified as English by your classifier.

Table 1. Confusion Matrix

	English	Spanish	Japanese
English			
Spanish			
Japanese			

4. If someone prints out a test document, then shreds the paper, so that all characters in the document are still visible but their order is completely scrambled, will your classifier work on the scrambled text? Justify your answer.
5. Suggest a different feature representation so that Naïve Bayes should perform better. You do not have to implement it. But you need to justify why you think it will be better.

You may implement your program any way you like, but you should write a Java class with the following calling convention:

```
java hw5 trainset_directory testset_directory
```

Your program should be able to complete the above tasks and output the required probability lists in task 2 and confusion matrix in task 3 to the standard output.

Hints:

1. **Computing prior probabilities**

Count the number of English documents in the training set:
nEnglish = number of documents in the training set's `English` directory

Count the number of Spanish documents in the training set:
nSpanish = number of documents in the training set's `Spanish` directory

Count the number of Japanese documents in the training set:
nJapanese = number of documents in the training set's `Japanese` directory

Compute the total number of training documents:
$nTotal = nEnglish + nSpanish + nJapanese$

Compute the prior probability for English:
$P(English) = nEnglish / nTotal$

Compute the prior probability for Spanish:
$P(Spanish) = nSpanish / nTotal$

Compute the prior probability for Japanese:

$P(Japanese) = nJapanese / nTotal$

2. **Computing** conditional **likelihoods**

Let *nCharEnglish* be the total number of characters (including multiple occurrences of the same unique character, including spaces) contained in all English training documents, and let *nCharSpanis* and *nCharJapanese* be that for the Spanish and Japanese training documents, respectively.

For each of the 27 unique characters, c_i, compute three conditional probabilities: $P(c_i \mid English) = countEnglish(c_i) / nCharEnglish,$ where $countEnglish(c_i)$ is the number of times character c_i occurs in all English documents in the training set. Similarly, compute $P(c_i \mid Spanish)$ and $P(c_i \mid Japanese)$.

3. **From probabilities to log probabilities**

Convert all probabilities to log probabilities to avoid underflow problems. Use the natural logarithm (`log(x)` in Java). Apply the log function to all probabilities.

4. **Classifying a test document**

Consider a new document, *doc*, from the test set. Suppose it contains the sequence of characters *c1, c2, ..., ck* (note: the same character may occur multiple times).

Compute the posterior probabilities (where α is the common denominator in Bayes's rule, which can be ignored for ranking the three languages):

$$P(English \mid doc) = P(English \mid c1, c2, ..., ck)$$
$$= \alpha\, P(c1, c2, ..., ck \mid English)\, P(English)$$
$$= \alpha\, P(English)\, P(c1 \mid English)\, P(c2 \mid English) ... P(ck \mid English)$$
$$P(Spanish \mid doc) = P(Spanish \mid c1, c2, ..., ck)$$
$$= \alpha\, P(c1, c2, ..., ck \mid Spanish)\, P(Spanish)$$
$$= \alpha\, P(Spanish)\, P(c1 \mid Spanish)\, P(c2 \mid Spanish) ... P(ck \mid Spanish)$$
$$P(Japanese \mid doc) = P(Japanese \mid c1, c2, ..., ck)$$
$$= \alpha\, P(c1, c2, ..., ck \mid Japanese)\, P(Japanese)$$
$$= \alpha\, P(Japanese)\, P(c1 \mid Japanese)\, P(c2 \mid Japanese) ... P(ck \mid Japanese)$$

Using log probabilities, the product becomes a sum:

$$\log P(English \mid doc) = \log \alpha + \log P(English) + \log P(c1 \mid English) + \log P(c2 \mid English) + ... + \log P(ck \mid English)$$

and similarly for Spanish and Japanese.

Classify the document as English if $\log P(English \mid doc) > \log P(Spanish \mid doc)$ and $\log P(English \mid doc) > \log P(Japanese \mid doc)$. Similarly for the other two languages.

Question 6: [2 extra credit points] Bonus Question

This optional question is for fun, while being an exercise for your imagination and your grasp of AI in the future. In a world where AI is increasingly important, we can assume that computers will be made to behave more and more like humans. Perhaps by design, or by negligence, computers will do one thing that we humans often do: forgetting. Can you come up with the most comical scenario related to "machine forgetting" (as opposed to machine learning ☺)? Any format is welcome: jokes, short stories, cartoons, etc. Your answer will be judged by its creativeness.

What to Turn In

Hand in in class a hard copy of your written questions. Put this all together, stapled, with your name, login and the date on the top of the first page.

Electronic content: hand in your code for Question 5 using the `handin` program. Copy it along with any other java files necessary to compile and run your program on the Linux machines into your own private handin directory, say called hw-handin. Then, execute the following command from the directory containing your own handin directory:

```
handin –c cs540-SECTION –a hw5 –d hw-handin
```

where SECTION is 1 or 2, depending on your CS540 section. For more information on how to electronically hand in your code, see the course web page. Be sure your code will compile and run on the departmental Linux machines; if it does not, we will not be able to grade it.