15-381 Spring 06 Assignment 5:
Inductive Learning Methods

Questions to Sajid Siddiqi (siddiqi@cs.cmu.edu)

Out: 3/30/06   Due: 4/13/06

Name: ________________________   Andrew ID: ________________________

Please turn in your answers on this assignment (extra copies can be obtained from the class web page). This written portion must be turned in at the beginning of class at 1:30pm on April 13. The code portion must be submitted electronically by 1:30pm on April 13. Please write your name and Andrew ID in the space provided on the first page, and write your Andrew ID in the space provided on each subsequent page. This is worth 5 points: if you do not write your name/Andrew ID in every space provided, you will lose 5 points.

Code submission. To submit your code, please copy all of the necessary files to the following directory:

```
/afs/andrew.cmu.edu/course/15/381/hw5_submit_directory/yourandrewid
```

replacing `yourandrewid` with your Andrew ID. You can use any of the following programming languages: C/C++,Java,Perl,Matlab,Lisp,ML/Ocaml, or Python. All code will be tested on a Linux system, we will not accept Windows binaries. You must ensure that the code compiles and runs in the afs submission directory. Clearly document your program.

Late policy. Both your written work and code are due at 1:30pm on 3/30. Submitting your work late will affect its score as follows:

- If you submit it after 1:30pm on 4/13 but before 1:30pm on 4/14, it will receive 90% of its score.
- If you submit it after 1:30pm on 4/14 but before 1:30pm on 4/15, it will receive 50% of its score.
- If you submit it after 1:30pm on 4/15, it will receive no score.

Collaboration policy. You are to complete this assignment individually. However, you are encouraged to discuss the general algorithms and ideas in the class in order to help each other answer homework 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:

- not explicitly tell each other the answers

- not to copy answers
- not to allow your answers to be copied

In those cases where you work with one or more other people on the general discussion of the assignment and surrounding topics, we ask that you specifically record on the assignment the names of the people you were in discussion with (or “none” if you did not talk with anyone else). This is worth five points: for each problem, space has been provided for you to either write people’s names or “none”. If you leave any of these spaces blank, you will lose five points. This will help resolve the situation where a mistake in general discussion led to a replicated weird error among multiple solutions. This policy has been established in order to be fair to the rest of the students in the class. We will have a grading policy of watching for cheating and we will follow up if it is detected. For the programming part, you are supposed to write your own code for submission.

1 Bayes Rule and Independence

References (names of people I talked with regarding this problem or “none”):

____

These questions further explore the rules of probability covered in the lecture. Assume all random variables to be boolean unless stated otherwise. **For problems in Section 1, you must state what axiom or rule of probability (chain rule, Bayes rule, etc.), if any, justifies each step in your derivations.**

1.1 (2 points)

Prove the symmetry property of independence, i.e.:

$$P(A|B) = P(A) \Leftrightarrow P(B|A) = P(B)$$

1.2 (3 points)

Prove that independence of A and B implies the independence of $\neg A$ and B, i.e.

$$P(A|B) = P(A) \Rightarrow P(\neg A|B) = P(\neg A)$$

1.3 (4 points)

It is often useful to apply probability statements in contexts where we have some fixed background evidence. Prove the *conditionalized* version of the chain rule:

$$P(A, B|e) = P(A|B, e)P(B|e)$$

Credit-Card Fraud Detection

You are a fraud analyst working for FasterCard, a major credit card company. Your job is to monitor customers' credit card activity for signs of possible theft. Your department's slogan: *Bayes Rule* Rules! From previous experience, you know the following facts: There is a 1% chance that any credit card used is a stolen one. A stolen credit card has a 95% chance of being used for multiple large online purchases in a single day. In contrast, a non-stolen credit card has only a 2% chance of being used for multiple large online purchases in a single day.

1.4 (2 points)

Define variables corresponding to the **two** events described above, and write down the **three** facts presented above as probabilities involving those two variables.

1.5 (3 points)

In terms of your definitions above, calculate the probability of seeing multiple large online purchases in a single day from any credit card.

1.6 (4 points)

A customer you are monitoring buys a new SUV, a 19th-century Persian carpet and a rare Siberian tiger on eBuy, all within minutes of each other. What is the probability that the credit card used for this transaction is a stolen one? Remember to justify your steps.

2 Naive Bayes Classifiers

References (names of people I talked with regarding this problem or “none”):

____

You are stranded on a deserted island. Mushrooms of various types grow wildly all over the island, but no other food is anywhere to be found. Some of the mushrooms have been determined as poisonous and others as not (determined by your former companions’ trial and error). You are the only one remaining on the island. You have the following data to consider, which describes mushrooms by their **attributes** *IsHeavy,IsSmelly, IsSpotted* and *IsSmooth.* The **class** tells us whether a mushroom is poisonous. 0 denotes False, and 1 denotes True.

Example	IsHeavy	IsSmelly	IsSpotted	IsSmooth	IsPoisonous
A	0	0	0	0	0
B	0	0	1	0	0
C	1	1	0	1	0
D	1	0	0	1	1
E	0	1	1	0	1
F	0	0	1	1	1
G	0	0	0	1	1
H	1	1	0	0	1
U	1	1	1	1	**?**
V	0	1	0	1	**?**
W	1	1	0	0	**?**

You know whether or not mushrooms A through H are poisonous, but you do not know about U,V or W. Suppose you decide to predict IsPoisonous using a naive Bayes classifier.

2.1 (4 points)

Train your classifier on the labeled data (A-H). Calculate and write down the **eight** conditional probabilities of attributes given the class value, e.g. $P(IsHeavy|\neg IsPoisonous)$, as well as the prior probability of the class values.

2.2 (3 points)

After learning is complete, what would be the predicted labels of mushrooms U,V and W according to your classifier? Recall that predicting the class requires computing the $\arg\max_x$ of the posterior probability $P(class = x|attributes)$ by Bayes rule, so you should not need to compute the denominator $P(attributes)$ at any point.

2.3 (3 points)

Now you spot another mushroom X on top of a nearby hill, but all you can tell about it is its spottedness, i.e. $IsSpotted = 1$. However, you can still use your classifier to get a prediction for this distant mushroom and help you decide whether to climb that hill. Calculate $\arg\max_x P(IsPoisonous = x|IsSpotted = 1)$ using your classifier.

It's nice that the Naive Bayes classifier was able to give you a prediction for mushroom X despite having only one attribute. We call this property *being robust to missing data.* Many classifiers do not have this property.

3 Entropy and Decision Trees

References (names of people I talked with regarding this problem or "none"):

____

In this section, use log-base-2 everywhere that a logarithm is needed.

3.1 (3 points)

What is the entropy of the following probability distribution: $[\frac{1}{8}, \frac{1}{16}, \frac{1}{16}, \frac{1}{4}, \frac{1}{2}]$? Show your work. A calculator will not be needed.

3.2 (2 points)

You are back on the deserted island of Section 2, but this time you've decided to try your luck with decision trees. What is the entropy of IsPoisonous in the table? Only mushrooms A-H are considered.

3.3 (2 points)

Which attribute among those in the table should you choose as the root of a good decision tree for classifying mushrooms? *Hint: You can figure this out just by looking at the data.*

3.4 (3 points)

What is the information gain of the attribute you chose in the previous question?

3.5 (2 points)

Build a decision tree of **height at least** 2, with your answer to Question 3.3 as the root. The layout of the rest of the tree is arbitrary.

3.6 (3 points)

Classify mushrooms U,V and W using this decision tree.

3.7 Extra Credit (5 points)

This decision tree cannot handle mushroom X from Question 2.3, with all its missing attributes. Can you suggest and briefly describe a way (or ways) of making our decision tree classifier robust to missing data? There is no single right answer here, but you should list the pros **and** cons of whatever you propose.

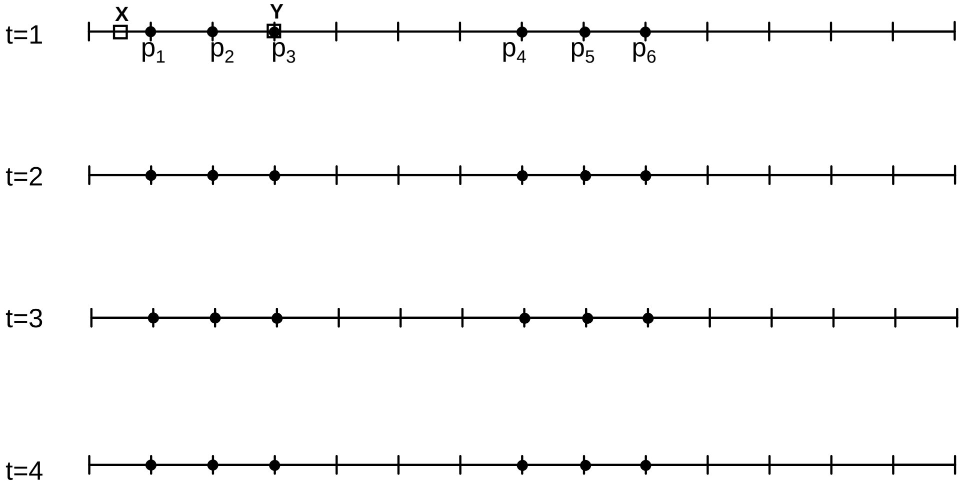
4 K-means

4.1 (4 points)

References (names of people I talked with regarding this problem or “none”):

____

In this question you will execute the K-means algorithm in a one-dimensional space, i.e. on points that lie along the real line. In the top row of the figure below, $p_1, \ldots, p_6$ are **six** such data points, and the **two** cluster centers X and Y are initialized as shown at $t = 1$, denoted by squares. For subsequent timesteps, estimate the points-to-clusters partitioning and the positions of the updated cluster centers (roughly), and write down ‘X’ and ‘Y’ at those positions. Repeat this process until convergence, adding more lines below $t = 4$ if necessary. Assume that we use the distortion metric discussed in class (which corresponds to the **square** of the euclidian metric).



4.2 (3 points)

Do the initial cluster positions in Question 4.1 lead to the best possible clustering? If they do, is there any initialization of two clusters on the real line that would lead to a sub-optimal clustering on these data points? If so, describe/illustrate.

4.3 Extra Credit (5 points)

Suppose now that we use the Manhattan distance metric instead of the Euclidian. For the one-dimensional K-means update step, prove mathematically that the cluster center location which minimizes the *distortion* (as defined in the class notes) is the **median** of all points owned by the cluster.

5 Implementing K-NN and Naive Bayes Classifiers

You will now implement K-Nearest-Neighbors and Naive Bayes classifiers and analyze their performance. In choosing a programming language, make sure you can easily read in and handle text data, as well as generate random numbers, in your chosen language. Basic functionality will be tested by running your code, but this section will mainly be evaluated by your answers for the questions below.

The Data

The data set for this problem is the Iris data set, a well-known data set for machine learning problems. The goal is to classify iris plants into one of 3 different types based on their sepal and petal lengths. A detailed description of the data set, along with the data itself, can be found at:
`http://www.ics.uci.edu/∼mlearn/databases/iris/`

5.1 K-Nearest-Neighbors

Write code to read in the data and store it internally in a convenient format. You also need to add functionality for randomly selecting subsets of the data (equal numbers from all 3 classes) to obtain training sets and test sets. Once you have this functionality, you will need to implement K-NN with the Euclidian distance metric, as described in the class notes, such that the value of K can be specified as a function parameter.

5.1.1 (5 points)

Partition the data in the following way: for each of the 3 classes, include the first 40 samples in the training set, and the remaining 10 in the test set. This gives us a training set of 120 samples and a test set of 30. Run your 1-NN code on this data. What is the classification accuracy of 1-NN on the test set with this partitioning?

Classification accuracy = _______%

5.1.2 (10 points)

The way we divide the training and test sets can greatly affect classifier performance. Therefore, the best practice is usually to calculate the mean classification accuracy over many partitionings of the data into training/test sets of fixed size.

Extend your previous experiment in this way: we still want training:test set sizes of 120:30, but this time, use your code to divide the data set into training and test sets by randomly choosing 10 samples of each class to be in the test set, and calculate the 1-NN classification accuracy in each case. What is the minimum, maximum and average classification accuracy of 1-NN over ten such random partitionings?

Mean classification accuracy = _______%
Max. classification accuracy = _______%
Min. classification accuracy = _______%

5.1.3 (10 points)

We will now see how K-NN behaves as we vary K. Repeat your previous experiment by finding the mean classification accuracy over ten partitionings of size 120 : 30, but this time do so for each of the values below. Write down the **mean classification accuracy** for each value of K below, and report the best value of K (the one with highest mean classification accuracy) according to your experiments.

Best value of K = _____

K	*Accuracy*
1	
5	
10	
15	
20	
30	
40	
50	
60	
70	

5.1.4 (5 points)

Low values of K seem to perform really well. What does this indicate about the distribution of data in different classes? (i.e. if we could plot this 4-D data and color points according to their labels, what might we notice?)

5.2 Naive Bayes

For Naive Bayes, you normally would need to learn values for $P(attrib|class)$ for all values of all the (discrete) attributes and for all three classes, as well as $P(class)$. **However**, before we can do that here, we must deal with the fact that all our attributes are real-valued! The simplest thing to do is to threshold each attribute to get a binary variable. Use the means of the attributes as the thresholds (i.e. 5.84, 3.05, 3.76 and 1.20).

The following questions will deal with the data discretized according to the thresholds above. Now that each attribute has been discretized and you have a data set with four binary attributes and a three-valued class label, train a Naive Bayes classifier for predicting the iris plant type. You will need to write functions to learn values for $P(attrib|class)$ for both values of all discrete attributes and for all three classes, as well as $P(class)$.

5.2.1 (5 points)

Partition the data into training and test sets as in Question 5.1.1. Train a Naive Bayes classifier on the training data. What is the classification accuracy on the test set?

Classification accuracy = ________%

5.2.2 (10 points)

Repeatedly partition the data into training and test sets as in Question 5.1.2. For each partitioning, train a Naive Bayes classifier on the training data, and calculate the classification accuracy on the test set. Compute the mean, maximum and minimum accuracies over the ten partitionings and write them down below.

Mean classification accuracy = ________%
Max. classification accuracy = ________%
Min. classification accuracy = ________%

5.2.3 (5 points)

How is the performance of Naive Bayes compared to K-NN? What are possible reason(s) for this?

Implementation Tips This data set is small enough that K-NN classification will be efficient even with the simplest possible search methods for finding nearest neighbors, so don't bother trying more clever search methods unless needed.

In Naive Bayes, make sure that your tables contain valid probability distributions, i.e. $0 \leq P(X = x) \leq 1$ and $\sum_x P(X = x) = 1$.

Break ties in k-NN randomly (i.e. two or three classes having the same number of nearest neighbors for a test sample).

5.3 Extra Credit (20 points)

Experiment with variants of the algorithms above. This could possibly improve your mean classification accuracies over questions 5.1.2 and 5.2.2. Read the description of the iris attributes and how they vary and correlate with the class labels, to get some idea of their relative usefulness.

K-NN For K-NN, you can try changing the distance metric. You could try using a Manhattan metric, or even more simply scaling different dimensions of your Euclidian distance metric by different quantities i.e. computing the distance between samples x and y as $\sqrt{\sum_i^4 \gamma_i (x_i - y_i)^2}$ for some constants $\gamma_1, \gamma_2, \gamma_3, \gamma_4$. The scale factor increases or decreases the importance of individual attributes in your K-NN classifier. If you set all these factors to 1, you get the regular Euclidian distance metric.

Bayes classifiers For Bayes classifiers, you could try using different subsets of the four attributes instead of using them all at once. You could also implement a Joint Bayes classifier using some or all attributes, learning a full joint probability table and using it in your Bayes classifier. Another possibly useful thing would be to discretize the real-valued attributes into n-ary attributes (where $n > 2$) instead of just binary.

Describe your experiments and report any results obtained (improved or otherwise). Discuss the results.

5.4 Submitting your Code

Grading Submit ALL your code from the experiments above. Grading will be done primarily on the basis of your answers to the questions. However, to demonstrate that you have working code, your code needs to be able to run (at least) the experiments in questions 5.1.1 and 5.2.1 from the command-line. You can submit your own input files along with your code in any format.

Documentation You need to submit a README file **only** for compilation and running instructions of your code, and to mention any known bugs. **Providing a description of your code is optional.**