CSC421 Assignment 3. Spring 2006 (10pts)

Misunderstanding of probability may be the greatest of all impediments to scientific literacy.
Gould, Stephen Jay

Student Name: Student Number:
Instructor George Tzanetakis

Question	Value	Mark
1	2	
2	2	
3	2	
4	2	
5	2	
Total	10	
Extra Credt	3	

1 Overview

The goal of this assignment is to familiarize you with probability theory, Baysian Networks, exact inference in Baysian Networks, and approximate inference using Monte-Carlo techniques. Be aware that some of the questions require much more work than others. Your deliverable will be a report documenting your answers. **Clarity and correct use of mathematical notation will count in grading**.

You can use any programming language to implement the programming parts of the assignment. Don't hesitate to contact the instructor via email or the anonymous comment form with any questions/clarifications you might need.

2 Probability Theory (2pts)

(*Variation I on cars and goats*). The master of ceremonies confronts you with three closed doors, one of which hides the car of your dreams, new and shinny and desirable (running on fuel cells for the ecologically sensitive). Behind each of the other two doors, however, is standing a pleasant but not so shiny and somewhat smelly goat. You will choose a door and win whatever is behind it. You decide on a door and announce your choise, whereupon the host opens one of the other two doors and reveals a goat. He then asks you if you would like to switch your choice to the unopened door that you did not at first choose. Being unsure about what to do you decide to toss a fair coin. If the coin falls head you switch, if it fall tails you don't switch. Suppose you win the car. What is the probability you switched doors. **(1pt)**

(*Variation II on cars and goats*) You play the car-goat game two times in independent sessions. The first time you don't switch and the second time you do switch. What is the probability that you win two goats ? Two cars ? **(1pt)**

3 Email categorization (2pts)

(Based on AIMA 13.18) Text categorization is the task of assigning a given document to one of a fixed set of categories, on the basis of text it contains. Naive Bayes models are often used for this task. In these models, the query variable is the document category, and the "effect" variables are the presence/absence of each word in the language; the assumption is that words occur independently in documents, with frequencies determined by document category.

- Explain precisely how such a model can be constructed, given as "training data" a set of documents that have been assigned to categories. Explain precisely how to categorize a new document.

  (1pt)

- In a programming language of your choice, implement an email categorization system based on the model you designed. Create training data using your own email (for example spam, school, friends). Implement a naive bayes categorization system. Document in detail what practical issues you had to solve. **(1pt but lots of work)**.

4 Baysian Networks (2pts)

Choose any application area that you like and model it using a Baysian Network formulation. Your network should have a minimum of 10 discrete random variables some of which (minimum 5) are not boolean. Write down the complete Baysian Network (topology + conditional probability tables). Provide reasonable estimates of the probabilities for your application domain and describe how you could obtain these numbers in practice.

5 Exact Inference in Baysian Networks (2pts)

Formulate 4 queries involving both evidence events and hidden variables that would be meaningful for the Baysian Network you defined in the previous question (the queries should be more complex than simple atomic events). Show the calculation of probabilities for two of the queries using exact inference using enumeration and two of the queries using variable elimination.

6 Monte-Carlo sampling (2pts)

- Monte-Carlo estimate of π using bomardment of a circle

  Consider the circle with radius 1 and center at the origin. The area of this circle is π. Notice that the upper right quadrant of the circle lies in the unit square. The idea is the following: 1) bombard the unit square with points uniformly and independelty generated 2) The proportion of these points falling within the quadrant is an estimate of the probability that a random points falls within the quadrant. This probability is the area of the quadrant which is known to be π 4. In a programming language of your choice, use the builtin random generator function to estimate π. Show estimates for 10, 100, 1000, 10000 random points. **(1pt)**

- Generating random samples from a specified distribution (AIMA 14.9)

  Consider the problem of generating a random sample from a specified distribution on a single variable. You can assume that a random number generator is available that returns a random number uniformly distributed between 0 and 1.

Let X be a discrete variable with $P(X = x_i) = p_i$ for $i \in 1, ..., k$. The **cummulative distribution** of X gives the probability $X \in x_1, ..., x_j$ for each possible j. Explain how to calculate the cummulative distribution in O(k) time and how to generate a single sample X from it. Can the latter be done in less than O(k) time ?

Now suppose we want to generate N samples of X, where N ¿¿ k. Explain how to do this with an expected runtime per sample that is *constant* (i.e., independent of k). **(1pt).**

7 (Extra Credit) Baysian Network Inference

In any programming language you like, implement a full inference system for Baysian Networks with discrete random variables. You are responsible for designing syntax, input, output conventions etc. Input the network you designed for this assignment and solve the queries you wrote using your implementation. **(2pt a lot of work)**. If you implement variable elimination in addition to enumeration then you get **(3pt even more of work)**.

8 Deliverables

Your deliverable is a report with your answers to the questions. For the questions requiring programming you must include the source code and test cases and provide enough documentation to make it easy to understand and read.