CSC 535 – Probabilistic Graphical Models

Assignment One

Due: 11:59pm, Wednesday, September 9.

Weight about 4 points

This assignment should be done individually

Part of this assignment is intended to sort out the software that you will use for this course. There is no requirement to use any particular language, but future assignments may include starter code in Matlab only. So consider Matlab if you have no particular preference. See the Matlab primer on the course webpage for a quick overview of the language. Python is a good alternative, however.

Deliverables

Deliverables are specified below in more detail. For the high level perspective, you are to provide a program to output a few numbers and to create figures. You also need to create a PDF that details results of each question, along with summary output, plots, and images. **Even if the question does not explicitly remind you to put the resulting image into the PDF, if it is flagged with ($), you should do so.** The instructor should not need to run the program to verify that you attempted the question. See

http://kobus.ca/teaching/grad-assignment-instructions.pdf

for more details about preparing write-ups. While it takes work, it is well worth getting better (and more efficient) at this. **A substantive part of each assignment grade is reserved for exposition.**

---

1. Coinflips

Suppose we flip a fair coin 10 times. What is the probability of the following events:

a) The number of heads and the number of tails are equal.
b) There are more heads than tails.
c) The ith flip and the (11-i)th flip are the same for i=1…5
d) We flip at least four consecutive heads

2. Random monkey

A monkey types on a 26-letter keyboard, using only lowercase letters. It is well-known that monkeys type uniformly at random from the alphabet. If the monkey types 1,000,000 letters, what is the expected number of times the sequence "proof" appears?

3. Random numbers

Set the random number generator seed to 0 and then produce 1000 throws of two (6-sided) die. Use the result to estimate the probability of double sixes. Report what you did and the result ($). Now run your code 9 more times, making sure that the random number generator is not reset. Report the results, and comment how many times you got the same estimate as the first time **($)**. Finally, set the seed to 0 a second time, and report whether you get the same result as the first time **($)**. Explain why it is often important to have random number sequences that are not really random, and can be controlled **($)**.

4. Random images

Download the text file http://kobus.ca/teaching/cs535/data/tiger.txt and read it in as a matrix. The number of rows in matrix should be the number of lines in the file, just as one would expect. Display the matrix as a grayscale image, and put the image into your report **($)**.

We will assume that grayscale means that each pixel has a brightness represented by 8 bits per pixel (256 shades of gray). This means that the grayscale tiger image could be thought of as a (uniform) random sample of an integer between 0 to 255, repeated for each pixel in the 236x364 grid. Create two images that are such random samples and put them into your report **($)** [Three sub-figures side by side probably works best]. Are the new images recognizable as scenes in the world like the tiger image? **($)**. Create and display some additional examples as needed (but do not put them into your report) to comment on (a) the difference (if any) between the random examples and whether a different random seed would change your conclusions, and (b) the relationship (if any) between the generated random images and everyday visual content, and (c) does this experiment tell you anything about everyday visual content ($).

Can you estimate how many times on average you would have to sample grayscale images of size (236, 364) to get the exact tiger image? [Do either version A or B as follows, or both for epsilon of extra credit] **($)**.

(A) The problem as most simply stated (as above) is a bit tricky and relies on more background then we want to assume at this point because the stream of images can have duplicates. For the purpose of this alternative (version (A)), you can instead imagine that the universe has a single copy of each of these images (like a deck of cards), and gives you one at a time. When all images have been handed out, it stops.

(B) Do the problem as stated where the system keeps handing you images including ones that are duplicates of previous ones. Thus it is possible that you could have to wait an arbitrarily long period of time to get a particular one (but you will get it eventually). Simply recognizing this as an instance of a problem that you have already studied and providing the answer without explanation does not suffice as it does not give the reader any intuition about the problem.

5. A quick check that you can provide plots and write captions

(A) Provide a plot of both sin(x) and cos(x) over the domain - $[-\pi, \pi]$. Each curve should have a different color **($)**.

(B) Provide bar plots for the histogram counts for the grey values of the tiger image and one of the random images that you used in the preceding question ($). Provide similar plots where the counts have been converted to empirical probabilities by scaling them so that they sum to one ($). Put the plots into your PDF **with an informative caption ($)**.

If you have not encountered histograms before, please look them up (e.g., on Wikipedia).

6. Integration

We often approximate continuous functions by sampling them at frequent intervals, and storing the values a vector. Derivatives and integrals can then be approximated by finite differences (http://en.wikipedia.org/wiki/Finite_difference) and Riemann sums (http://en.wikipedia.org/wiki/Riemann_sum), respectively. In this class, for example, we might use integrals to evaluate the probability of an event under a continuous probability distribution.

Assume the heights of adult men follow a normal distribution with mean 70 inches and standard deviation of two inches. Let's evaluate the probability of a man having a height between 68 and 80 inches.

The normal distribution function is defined as:

$$\mathcal{N}\left(x;\mu,\sigma^2\right) = \left(2\pi\sigma^2\right)^{-1/2} \exp\left(-\frac{\left(x-\mu\right)^2}{2\sigma^2}\right) \quad (1)$$

Here, σ is 2 inches, and μ is 70 inches. (The semi-colon in the above might confuse. It is relatively common to separate the last block of function parameters with a semi-colon when those parameters are being thought of as constants).

Begin by creating a vector of x values, evenly spaced over the range [68, 80] at increments of 2 inches. Next, compute a y vector, containing the values of $\mathcal{N}\left(x;\mu,\sigma^2\right)$. Plot the result as a bar chart together with the curve for the formula over a wide enough range that it close to zero at the extremities of the range **($)**. The bars are an approximation of the function in the region that we will be integrating. It is not smooth as we are discretizing a continuous function by sampling it at intervals of size 2. Decreasing the interval between x-values will make the function appear smoother, but will require more computing time to operate on.

Now compute the Riemann integral (http://en.wikipedia.org/wiki/Riemann_sum), using the y values as the rectangle heights, and the delta-x (2 inches) as the rectangle width. This is equivalent to summing the values of y and multiplying by delta-x. The result is an approximate integral which represents the probability of a man being between 68 and 80 inches. Report your estimate **($)**.

Now compute the integral over a much wider range of values (say [20,120]). State in your writeup what you expect the value to roughly be ($). Since you have a good idea what the result should be, you can now experiment with changing the value for delta-x and observe the change in the result. Try this for some values of delta-x that both larger, equal to, and smaller than the one just used. Plot the error of the estimate verses delta-x. Make sure your cast a wide enough net on delta so that your results can tell a story **($)**.

What to Hand In

Hand in a program hw1.<suffix> (e.g., hw1.m if you are working in Matlab) and the PDF file hw1.pdf in D2L.