COL 776: Assignment 1 (Part A)

Due Date: 11:50 pm, Thursday August 27, 2015. Total Points: 34 (+ 10 extra credits)

Notes:

- You should submit all your code as well as any graphs that you might plot (see below).
- Include a **single write-up (pdf) file** which includes a brief description for each question explaining what you did. Include any observations and/or plots required by the question in this single write-up file.
- You can use any programming language from the set C++, Java, Python, Matlab. If you would like to use any other language, please check with us before you start.
- Your code should have appropriate documentation for readability.
- You will be graded based on what you submit as well as your ability to explain your code.
- Refer to the course website for assignment submission instructions.
- This assignment is supposed to be done individually. You should carry out all the implementation by yourself.
- We plan to run Moss on the submissions. Any cheating will result in a penalty of **-10** points on your total course score (in addition to a 0 on the assignment). Stricter penalties (**including a fail grade**) may follow.

1. **(34 points + 10 Extra Credit) Conditional Independencies in Bayesian Networks:** In this problem, we will learn about (conditional) independencies in Bayesian networks. Specifically, we will implement the Bayes Ball algorithm for computing the set of *d-separated nodes* and *requisite probability nodes* (see below). You can download the paper from the course website. You are provided with a file readme.txt which specifies various input/output formats: a) how you should store a Bayesian network in a file b) the format of the queries that need to be processed c) the format of your output. You should go through this file carefully. NOTE: We will be using the specified input/output format to test your programs. So, you may lose several points if you do not stick to the specification.
    - **(6 points)** Write a program to generate a random Bayesian network structure of given size. Your program should input the number of nodes in the network (n) and max-number of children for each node (k). You can assume a topological order over the nodes. Starting with the root node, decide a number u between 1 and k, uniformly at random, and then randomly pick u children from the nodes appearing after the current node in the topological order (you should pick all the nodes if there are less than u remaining). Continue this process until you generate the children for every node in the network. Note that at the end of this process, you have fully specified the structure of the Bayesian network. You should think about how you will represent the Bayesian network (what data structures will you use). Make sure to use the right data structures which enable you to implement the programs below. Write another program to write the Bayesian network that you generated above to a file in the format specified in readme.txt. Also, write a program to read the network back from the file into the memory.
    - **(4 points)** Read the paper on "Bayes Ball" posted on the website. You can ignore the contents in the Section 4 and afterwards if you like. Write a 500 word summary of the paper in your own words. Your description should include the key ideas presented in the paper as well as important results (DO NOT just copy paste the abstract of the paper). You should understand the paper well enough to

implement the algorithms presented. For the purpose of this assignnment, you can assume there are no deterministic nodes in the network (see the paper for a characterization). Next, you will implement the Bayes Ball algorithm to compute the following two parts. For each row in the query file, which specifies a set of query nodes followed by a set of observed nodes [1], do the following.

- **(10 points)** Write a program to compute the set of nodes which are d-separated from the query nodes given the observations. Make sure to output the d-separated nodes in the topological order for easy verification.
- **(10 points)** Write a program to compute the set of requisite probability nodes (both observed as well as unobserved) whose distributions and/or states are required to calculate the probability of the query given the observed nodes. Your program should output the observed requisite nodes and unobserved requisite nodes separately. As before, you should output them in the topological order for easy verification.
- **(2 points)** What is the complexity of your program for each of the parts above? Argue briefly.
- **(2 points)** Output your results in a separate file with one row for each of the queries. Each row should contain the original set of query nodes and observed nodes followed by the set of d-separated nodes and the set of requisite probability nodes computed by your program [2]. You are given a script (display.py) which takes the output of your program in a file in the format specified above and graphically displays the network as well as the set of query,observed,d-separated and requisite nodes. Run the script to visually verify the output of your program for each Bayeisan network that you generate.
- **Extra Credits (10 points)** Implement an efficient algorithm that computes the set of all pairs of nodes (X_i, X_j) which are d-separated from each other in the graph given a set of observed nodes. Your program should input a Bayesian network from a file and a separate file containing a comma separated list of observed nodes in the network. Your program should be more efficient than simply running your code above for each node X_i in the network and calculating the set of nodes d-separated from X_i. What is the complexity of your program written for this part? Write your output to a file based on the topological order of the nodes in the network with one pair of d-separated nodes in each row (see readme.txt for exact details).

[1] see readme.txt for exact format of the query file.
[2] see readme.txt for exact output format.