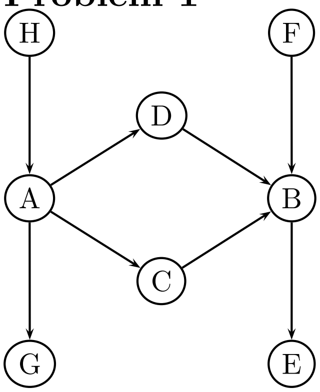
STAT 535 Final Exam Solutions
date, rules of exam, time available 1h 50min

Problem 1

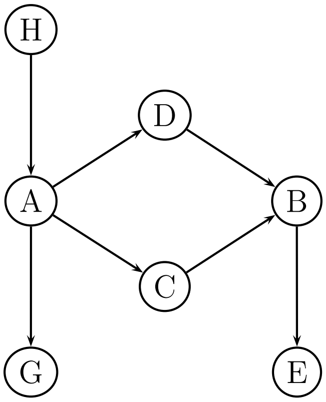


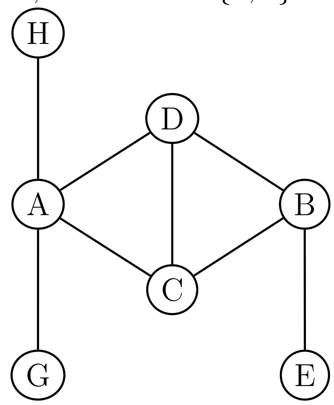
Which of the conditional independence statements below are true? For those which are not, give an unblocked path.

			unblocked path, if any:
$H \perp E$		FALSE	$H \rightarrow A \rightarrow C \rightarrow B \rightarrow E$
$H \perp F$	TRUE		
$H \perp D \mid A, G$	TRUE		
$C \perp D \mid A$	TRUE		
$C \perp D \mid A, B$		FALSE	$C \rightarrow B \leftarrow D$
$D \perp F \mid E$		FALSE	$F \rightarrow B \leftarrow D$ because of decendant E

Problem 2

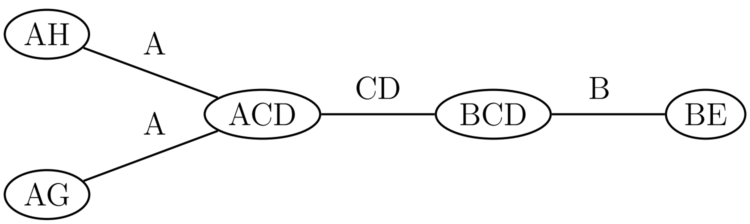
In the graphical model below all variables are discrete, with values $\{0, 1\}$.





2.1 Moralize and triangulate this graphical model: see above

2.2 Draw a junction tree for this graphical model.



2.3 Give, step by step, an *efficient* algorithm to compute the probability $P(C = 0|A = 0, B = 0)$ in this graphical model. It is assumed that $P_{ABCDEGH}$ is represented by conditional probability tables in the standard way.

Details:

1. The algorithm should be *specific* for computing $P(C = 0|A = 0, B = 0)$.
2. An algorithm for this is *efficient* if it does not do useless calculations, and if it does not create tables or potentials larger than it needs to. Your algorithm must be efficient to be acceptable as a solution, or only a little worse. Note that an algorithm that solves general inferences in this graphical model may not be efficient according to the present definition.

3. There are several possible methods to compute this probability. Give the name of the method you decide to use (e.g Variable elimination, Junction Tree, Message Passing, Sum-Product, etc).

4. Give the size of the table/potential/message/separator/clique on which every step of the algorithm operates, and an approximate number of operations for each step.

Here is a solution by variable elimination.

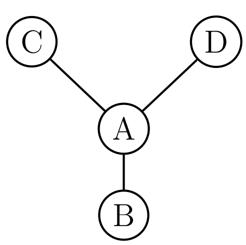
$$P_C(0) = \sum_{degh} P_H(h)P_{A|H}(0|h)P_{G|A}(g|0)P_{C|A}(0|0)P_{D|A}(d|0)P_{B|CD}(b|0,d)P_{E|B}(e|0)$$

Step	$P_C(0)$	Variable(s) eliminated, Potentials formed, number operations
1.	$\sum_{dh} P_H(h)P_{A|H}(0|h)\underbrace{\sum_g P_{G|A}(g|0)}_{1}\, P_{C|A}(0|0)$	$E, G \perp C|A, B$, potentials = 1, no operations
	$\times P_{D|A}(d|0)P_{B|CD}(b|0,d)\underbrace{\sum_e P_{E|B}(e|0)}_{1}$	their elimination gives potentials equal to 1
2.-3.	$\left[\sum_h P_H(h)P_{A|H}(0|h)\right] P_{C|A}(0|0)$,	elim. H, 2 factors$\times$2 terms=3 ops, const.$z_1 = P_A(0)$,
	$\times \left[\sum_d P_{D|A}(d|0)P_{B|CD}(b|0,d)\right]$	elim. D, 2 factors$\times$2 terms, const. z_2
4.	$z_1 z_2 P_{C|A}(0|0)$	3 factors, 2 ops

Problem 3: Rudolf the Reindeer marginalized

Characters: Rudolf the Reindeer, Santa's Little Helper

RUDOLF THE REINDEER: Look Santa's Little Helper, I decorated my horns this season using a Markov Ranodom field. The variables A, B, C, D are light bulbs which are on and off (that is 1 and 0 to you) randomly, with the joint distribution given by



$$P_{ABCD}(a,b,c,d) \;=\; \frac{1}{Z}\phi_{AB}(a,b)\phi_{AC}(a,c)\phi_{AD}(a,d)$$

$$\text{with } \phi_{AX}(a,x) \;=\; 2^{-|a-x|}$$

SLH: Brilliant! Do you realize that your MRF is a tree, and as such it has a factorization in terms of clique and separator marginals?

RtR: A tree! That's amazing, SLH. I hadn't thought of that! But I wonder: could it be true that A, B, C and D all have the same marginal probability of being on?

3.1 Prove or disprove: $P_A(x) = P_B(x)$ for $x = 0, 1$.
(Note: you can also answer this question after you answer 3.2, 3.3., 3.4)

Solution: Note that ϕ_{AX} for $X = B, C, D$ only depends on the relative values of the variables A and X (whether they are equal or not). Therefore, P_{ABCD} will also have this property. So, for any configuration $abcd$ we have that $P(abcd) = P(\bar{a}\bar{b}\bar{c}\bar{d})$ where $\bar{x} = 1 - x$. Hence, $P_X(x) = P_X(\bar{x})$ for each $X = A, B, C, D$.

SLH: I don't know how to answer your question, but I can see that if you fix A, you can write P_{ABCD} in a simpler form, especially if you use the notations $\bar{b}, \bar{c}, \bar{d}$ for $1-b, 1-c, 1-d$. Maybe this will let me compute $P_{A,B}$ and then I will be able to answer...

3.2 SLH doesn't notice it, but with his observation it's easier to calculate $P_{B|A}$. Show him how, and obtain the tables $P_{B|A=0}, P_{B|A=1}$ (numerical values required). Remember the brute force solution is not acceptable.

Solution: Note that

$$\begin{aligned} A = 0: \quad P_{ABCD}(0, b, c, d) &= \frac{1}{Z} 2^{-b-c-d} \propto 2^{-b}\text{function of variables other than } B \\ A = 1: \quad P_{ABCD}(1, b, c, d) &= \frac{1}{Z} 2^{-\bar{b}-\bar{c}-\bar{d}} \end{aligned}$$

This is consistent with the independence $B \perp C, D \mid A$. Hence, $P_{B|A=0}(b|0) \propto 2^{-b}$, ie.

$$\begin{array}{lcl} P_{B|A}(0|0) & \propto & 1 \\ P_{B|A}(1|0) & \propto & 1/2 \end{array} \Rightarrow P_{B|A}(0|0) = \frac{2}{3} \;\; P_{B|A}(1|0) = \frac{1}{3}$$

By symmetry, $P_{B|A}(b|1) = \left[\frac{1}{3}\;\frac{2}{3}\right]$.

RtR: Look, SLH, now that you have $P_{B|A}$, $P_{C|A}$, $P_{D|A}$, I see a way to compute Z, the normalization constant, by moving the sums inside the products! **3.3** Do what RtR says he can: calculate the value of Z (no brute force please).

Solution:

$$\begin{aligned} Z &= \sum_{abcd} P_{ABCD} = \left[\underbrace{\sum_{bcd} 2^{-b-c-d}}_{A=0} + \underbrace{\sum_{bcd} 2^{-\bar{b}-\bar{c}-\bar{d}}}_{A=1} \right] \\ &= \left[\sum_b 2^{-b} \sum_c 2^{-c} \sum_d 2^{-d} + \sum_b 2^{-\bar{b}} \sum_c 2^{-\bar{c}} \sum_d 2^{-\bar{d}} \right] \\ &= \left[(1+1/2)^3 + (1+1/2)^3 \right] = \frac{27}{4} \end{aligned}$$

3.4 Now find P_A and P_{AB}, using your results from 3.2,3.3 (numerical values please). (You can verify now your answer in 3.1).

Solution: By a similar reasoning as above, $P_A(0) = \frac{1}{Z}\sum_{bcd} 2^{-b-c-d} = \frac{3^3/2^3}{3^3/2^2} = \frac{1}{2}$ and

$$P_{AB} = P_A P_{B|A} = \begin{array}{cc} \begin{array}{cc} 0 & 1 \end{array} & \\ \left[\begin{array}{cc} 1/3 & 1/6 \\ 1/6 & 1/3 \end{array}\right] & \begin{array}{c} 0 \\ 1 \end{array} \end{array}$$

Problem 4

Santa has enlisted n elves, $X_{1:n}$ to deliver presents on Christmas Eve. The elves are spread all over the world, but each of them can communicate with a set of "neighbors"; we denote $i \sim j$ if X_i and X_j are neighbors. The pattern of connections forms a graph with no cycles. Santa himself is node X_0 in this graph.

4.1 By midnight, each elf X_i has delivered k_i presents. Design an efficient algorithm based on communication only between neighbors, that will allow

Santa to find out the total number of presents that were delivered until midnight?

Assume that Santa and the elves can only know who their neighbors are (the local structure of the graph) but none of them knows the global structure of the graph, except that it is connected, and that it has no cycles. They can send any kind of data in their messages, but for efficiency they try not to send too much useless data. Santa's neighborhood is not special, but you can assume Santa is able to send special messages if you think this is necessary.

Solution: Let the potential of node X_i be $\phi_i = k_i$, and the message a node X_i sends to a neighbor X_j be

$$\mu_{i \to j} = \phi_i + \sum_{j' \sim i, j' \neq j} \mu_{j' \to i}$$

Note that both the potentials and the messages are scalars. Hence, the generalized operations $\otimes, \oplus$ are

$$\begin{array}{ccc} \otimes & = + & \text{(composition)} \\ \oplus & = \text{identity} & \text{summarization} \end{array}$$

A centralized algorithm could have this form:

1. Santa sends a CollectEvidence message to its neighbors.
2. Recursively: when a node X_i receives CollectEvidence it
   (a) sends a CollectEvidence message to its neighbors other than the one from whom it receved the message (i.e the "parent")
   (b) waits for messages $\mu_{j' \to i}$ from all the neighbors other than the parent
   (c) sends his own message $\mu_{i \to j}$ to its parent X_j
3. When Santa receives messages $\mu_{j \to 0}$ from all its neighbors, he computes $k_{total} = \phi_0 + \sum_{j \sim 0} \mu_{j \to 0}$

4.2 Can you design an efficient message passing algorithm, to be run at midnight, so that *each elf* can know how many presents were delivered? (If your algorithm in 4.1 already does this, explain why.)

Solution: The above algorithm also has a distributed version, which is a standard sum-product algorithm

- Each node X_i sends messages $\mu_{i \to j}$ as above
- A node X_i can send $\mu_{i \to j}$ only when it has received messages from all its neighbors other than X_j.
- After having received messages from all its neighbors, the node computes its "marginal" by $\phi_i + \sum_{j \sim i} \mu_{j \to i}$. This value is the total number of presents delivered.

Note that the message passing can be initiated by the leaf nodes, which have only a single neighbor.

4.3 Can you design an algorithm so that Santa and all the elves can be updated regularly on the total number of presents delivered?

Solution: The above algorithm can be made parallel and asynchronous: when an elf delivers a new present, it updates its potential ϕ_i, then it sends messages to all its neighbors. In this algorithm all messages must be initialized at the beginning with 0. In addition, when an elf discovers that one of its messages has been updated, it updates its "marginal", and updates its own messages to all other neighbors.