6.824 Artificial Intelligence, Fall 1994
Problem Set 5

Due at class Friday, Oct. 21

Problem 1. Consider the Bayesian network consisting of the following linear sequence of nodes.

$P_1 \rightarrow P_2 \rightarrow \ldots \rightarrow P_T$

Suppose that each node is a Boolean variable and that for arrow $P_t \rightarrow P_{t+1}$ we have that there is a $1 - \epsilon$ chance that P_{t+1} has the same value as P_t, and hence an ϵ chance that they are different. Also suppose that the prior probability of P_1 is $\frac{1}{2}$. Let O be the observation that $P_1 = \texttt{True}$ and $P_T = \texttt{True}$.

Part a. For the observations O described above give a recurrence relation for computing $\beta_{P_t}(\texttt{True})$ and $\beta_{P_t}(\texttt{False})$ from $\beta_{P_{t+1}}(\texttt{True})$ and $\beta_{P_{t+1}}(\texttt{False})$

Part b. Use the recurrrence in part a to show (by induction on k) that $\beta_{T-k}(\texttt{True}) + \beta_{T-k}(\texttt{False}) = 1$.

Part c. Determine the limit as $t \rightarrow \infty$ of $\beta_2(\texttt{True})$ and $\beta_2(\texttt{False})$. (Hint: the limit is a fixed point of the recurrence relation you gave in part a and must also satisfy the invariant given in part b.)

Part d. Using techniques similar to those for parts a, b and c, determine the limit as $t \rightarrow \infty$ of $\alpha_t(\texttt{True})$ and $\alpha_t(\texttt{False})$.

Part e. Give the limit as $T \rightarrow \infty$ (for even values of T) for $P_r(P_{\frac{T}{2}} = \texttt{True}|O)$.

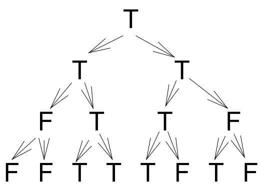
Part f. Give the limit as $T \rightarrow \infty$ of $P_r(O)$. For finite values of T is $P_r(O)$ larger or smaller than this limit?

Part g. Give a qualitative description (no formulas) of $P_r(P_t = \texttt{True}|O)$ for large values of T and various value of t.

Problem 2. Consider a Bayesian network which is an n level binary tree with arcs directed from parents to children and ending in the 2^n leaves of the tree. Each node in this tree is a Boolean variable where the probability that a node has the same value as its parent is $\frac{2}{3}$. The probability that the root node is true is $\frac{1}{2}$. Suppose we consider two observations O_1 and O_2 both of which observe some (but not necessarily all) of the leaf nodes of the tree. Furthermore suppose that O_1 and O_2 are identicle except that O_1 contains $P = \texttt{True}$ while O_2 contains $P = \texttt{False}$ for some leaf node P. Describe the nodes in the tree where the α numbers will be different under these two observations, the nodes in the tree where β numbers will be different and the nodes Q such that $P_r(Q = \texttt{True}|O_1)$ is different from $P_r(Q = \texttt{True}|O_2)$. In each case state whether the quantity being discussed is larger or smaller under O_1 as opposed to O_2. Briefly explain your answers.

Problem 3. Again consider a Bayesian network which is an n level binary tree with arcs directed from parents to children and ending in the 2^n leaves of the tree. Each node in this tree is a Boolean variable where the probability that a node has the same value as its parent is $(1 - \epsilon)$ with $0 < \epsilon < \frac{1}{2}$. The prior probability that the root node is true is $\frac{1}{2}$. We use P_n to represent a node that is n arcs above the leaf nodes.

part a. Consider the execution of the tree network shown below. What is the probability of the given execution, i.e., what is the probability of this assignment of values to nodes. Your answer should be given as a function of ϵ.



part b. Consider the case where all leaf nodes are observed to have value T. Recall that $\beta_{P_n}(\texttt{True})$ is the probability of the oservations below the node P_n given that $P_n = \texttt{True}$. $\beta_{P_n}(\texttt{False})$ is defined similarly. For each n these numbers can be written as polynomials in ϵ. Since ϵ is in the interval $(0, \frac{1}{2})$ we take the order of a polynomial in ϵ to be *smallest* exponent of ϵ in a nonzero term. For example, the order of $1 + \epsilon$ is 0 and the order of $3\epsilon^2 + \epsilon^3$ is 2. The polynomial 0, i.e., the function that is always 0, should be considered to have infinite order (note that $0 = \epsilon^\infty$). Give the orders of the polynomials for $\beta_{P_n}(\texttt{True})$ and $\beta_{P_n}(\texttt{False})$ as a function of n. (The answer is a very simple function.)

part c. Now consider a set of observations O at the leaves of a tree with n levels of arcs and root node n. We allow O to be arbitrary (some of the observed values may be F). For the fixed observation set O consider all the possible ways that the network can be executed starting with root value $P_n = \texttt{True}$ and ending in the observed values O. A given execution corresponds to a particular set of values for all the internal nodes of the tree. For a given execution we consider the number of "changes" in the execution, i.e., the number of nodes whose values are different from the value of its parent. Define $B_{P_n}(\texttt{True})$ to be the minimum over all executions of the number of changes of that execution. Circle the true statement in the following list and explain your answer.

- The order of the polynomial for $\beta_{P_n}(\texttt{True})$ is always less than $B_{P_n}(\texttt{True})$.
- The order of the polynomial for $\beta_{P_n}(\texttt{True})$ is always larger than $B_{P_n}(\texttt{True})$.

- The order of the polynomial for $\beta_{P_n}(\texttt{True})$ is always equal to $B_{P_n}(\texttt{True})$.
- For trees and data sets the order of the polynomial for $\beta_{P_n}(\texttt{True})$ is less than $B_{P_n}(\texttt{True})$ but for other trees and data sets the order of the polynomial for β_{P_n} is greater than $B_{P_n}(\texttt{True})$.

Hint: $\beta_{P_n}(\texttt{True})$ is the sum over all executions E, consistent with the oservations O, of $P_r(E \mid P_n = \texttt{True})$.

Problem 4. This problem is about using inference rules to reason about Bayesian networks. Suppose we are given a tree structured Bayesian network with observed values at all of the leaves. The tree may have nonuniform depth. We want to find the most likely execution of the network given that the root node has value T and the leaf nodes have their observed values. We assume that the Bayesian network is a binary tree. Every node other than the leaves has exactly two children. We suppose that the Bayesian network and the observations O are represented by assertions in a data base of the following forms.

- The assertion (`LEFT-CHILD` n m) means that m is the left child of n.
- The assertion (`RIGHT-CHILD` n m) means that m is the right child of n.
- The assertion (`CONDITIONAL-COMPLEXITY` X x Y y w) means that Y is a child node of the node X and $w = -\log_2 P_r(Y = y \mid X = x)$.
- The assertion (`OBSERVED` Z z) means that Z is a leaf node which has been observed to have value z.

To find the most likely execution given the data we let E_X be the set of possible executions of the nodes at or below X, i.e., the set of all possible assignments of values to X and the nodes below X. For a given node X and possible value x for X we define $\delta_X(x)$ as follows.

$$\delta_X(x) = \max_{e \in E_x} P_r(e \wedge O | X = x)$$

In English, $\delta_X(x)$ is the probability of the most likely execution e of the nodes at or below X such that e is consistent with the observations O. If Z is a leaf node observed to have value z then $\delta_Z(z) = 1$. If Z is a leaf node and w is some possible value for Z other than the observed value then $\delta_Z(w) = 0$. We define $C_X(x)$ to be

$-\log_2 \delta_X(x)$. Note that if w is a possible value for Z other than the observed value for z then $C_Z(w) = \infty$. In this case we can ignore the value w for Z. If z is the observed value for leaf X then $C_Z(z) = 0$.

We wish to infer assertions of the following form.

- The assertion ($\leq$ (C X x) w) means that $C_X(x) \leq w$

part a. Give two inference rules for infering bound assertions such as the above. The inference rules should be capable of infering an exact value for each complexity $C_X(x)$ in the network.

part b. Assume that there are n nodes in the network and that each node has d possible values. Give the order of running time for executing the rules you gave in part a according to the "Dijkstra tightest bounds" algorithm. Your answer should be in terms of n and d.