CS 188 Spring 2010 — Introduction to Artificial Intelligence — Written 6: Sampling and HMMs

Due: Thursday 4/8 in 283 Soda Drop Box by 11:59pm (no slip days)

Policy: Can be solved in groups (acknowledge collaborators) but must be written up individually

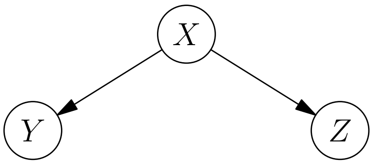
First name	
Last name	
SID	
Login	
Collaborators	

For staff use only:

Q1. Sampling	/6
Q2. Inference in Chains	/10
Total	/16

Q1. [6 pts] Sampling

Consider the following Bayesian network.



X	$\Pr(X)$
0	0.5
1	0.5

Y	X	$\Pr(Y\|X)$
0	0	0.3
1	0	0.7
0	1	0.4
1	1	0.6

Z	X	$\Pr(Z\|X)$
0	0	0.1
1	0	0.9
0	1	0.5
1	1	0.5

Define the function $f(X, Y, Z)$ by

$$f(X, Y, Z) = (X + Y + Z)^2.$$

In the questions below, perform either rejection sampling or likelihood-weighted sampling by sampling the individual variables (as required by the sampling method) in the order (X, Y, Z). Use as many values as needed from the following sequence $\{a_i\}_{1 \le i \le 15}$ of numbers generated independently and uniformly at random from $[0, 1)$ as a source of randomness.

a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}
0.142	0.522	0.916	0.792	0.859	0.036	0.656	0.249	0.934	0.679	0.758	0.743	0.392	0.655	0.171

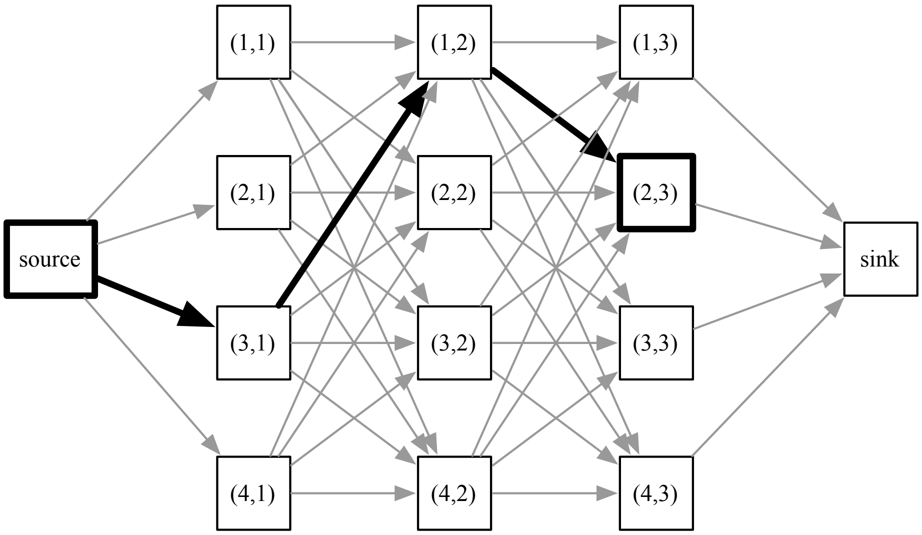
To sample a binary variable W with $\Pr(W = 1) = p$ (and $\Pr(W = 0) = 1 - p$) using a value a_i sampled uniformly at random from $[0, 1)$, choose

$$W = \begin{cases} 1 & \text{if } a_i < p; \\ 0 & \text{if } a_i \ge p. \end{cases}$$

(a) [3 pts] Estimate the expectation $\mathrm{E}[f(X, Y, Z) | Y = 0, Z = 1]$ using rejection sampling, continuing to collect samples until exactly two have been accepted. Show your work.

(b) [3 pts] Estimate the expectation $\mathrm{E}[f(X, Y, Z) | Y = 0, Z = 1]$ using two samples obtained using likelihood-weighting. Show your work. Reuse the sequence $\{a_i\}_{1 \le i \le 15}$ (starting with a_1) as a source of randomness.

Q2. [10 pts] Inference in Chains



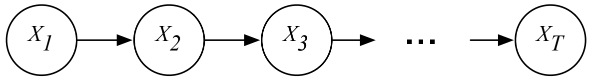
Consider the directed graph above. Vertices in this graph are denoted by $v = (n, t)$ where $t \in \{0, \dots, T+1\}$ is the horizontal axis, and $n \in \{1, \dots, N\}$ is the vertical axis on the figure ($N = 4, T = 3$ in the example above). All the vertical layers have N vertices, except for the first (leftmost) and last (rightmost) layers that have only a single element. We call the vertex in the last layer the *sink*, corresponding to $(1, T+1)$, and the vertex in the first layer, the *source*, corresponding to $(1, 0)$. Vertices in adjacent vertical layers are fully connected (i.e. there is an edge $(n, t) \to (m, t+1)$ for all n, m, t). This type of graph is called a *lattice*.

(a) [2 pts] For a pair of vertices $v = (n, t), v' = (n', t+k)$, $k > 0$, a (directed) *path* is a list of k edges that link v to v'. For example, in the figure, the edges in bold form a path from source to $(1, 2)$. We denote the set of all paths going from vertex v to vertex v' by $\mathcal{S}(v, v')$. Show that specifying a path between a pair of vertices $v = (n, t), v' = (n', t+k)$, $k > 0$ is equivalent to picking a $(k-1)$-tuple $(x_1, \dots, x_{k-1})$ (i.e. for each such path there is a unique corresponding $(k-1)$-tuple, and conversely, for each $(k-1)$-tuple $(x_1, \dots, x_{k-1})$, there is a unique corresponding path).

(b) [1 pt] Use your result above to find the number of paths going from the source to the sink, $|\mathcal{S}(\text{source}, \text{sink})|$.

Suppose now that for each edge e in this lattice, we are given an associated *edge cost* $c(e)$. This cost is extended to paths in the following way: for a given path p, the *path cost*, $c(p)$ is defined as the product of the cost of the edges in the path: $\prod_{e \in p} c(e)$.

We will now establish the relationship between the lattice and inference in the following Bayes net:



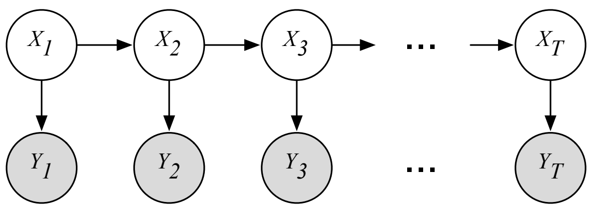
Assume each random variable X_t has domain $\{1, \ldots, N\}$. This Bayes net is called a *Markov chain*.

(c) [2 pts] If the edge costs are set to:

$$c\Big((n,t),(n',t+1)\Big) = \begin{cases} \Pr(X_{t+1} = n' | X_t = n) & \text{for } 0 < t < T; \\ \Pr(X_1 = n') & \text{for } t = 0; \\ 1 & \text{otherwise,} \end{cases}$$

find a probabilistic interpretation for the cost of a path p going from the source to the sink, $c(p)$. Is it a joint probability, a marginal, or a conditional?

Let us add observations to this Bayes net:



We still assume that the unobserved random variables X_t have domain $\{1, \ldots, N\}$, while the domain of the observed (shaded) random variables is $\{1, \ldots, M\}$.

The resulting Bayes net is called a *Hidden Markov Model* (HMM). It is one of the most widely used Bayes nets, with applications in speech recognition, cryptanalysis, gene prediction, machine translation, and many other fields.

(d) [2 pts] Show how to redefine the edge costs $c(e)$ so that the cost of a path going through vertices $(x_1, 1), (x_2, 2), \ldots, (x_T, T)$ coincides with the joint probability $\Pr(X_1 = x_1, X_2 = x_2, \ldots, X_T = x_T, Y_1 = y_1, Y_2 = y_2, \ldots, Y_T = y_T)$. Use the same N by T lattice as in the previous questions; the values $y_1, y_2, \ldots, y_T$ at the observed nodes $Y_1, Y_2, \ldots, Y_T$ are known and fixed once and for all.

$$c\Big((n,t),(n',t+1)\Big) = \begin{cases} \\ \\ \end{cases}$$

(e) [2 pts] For a given node (n, t), we also define $s(n, t)$, the sum of the costs of all the paths going from the source to (n, t):

$$s(n, t) = \sum_{p \in \mathcal{S}(\text{source}, (n,t))} c(p),$$

where $c(p)$ refers to the modified cost function you defined in the previous question. Find a probabilistic interpretation of $s(\text{sink})$ in the HMM model. Is it a joint probability, a marginal, or a conditional?

(f) [1 pt] Explain how computing $s(\text{sink})$ can in turn be used to evaluate the posterior distribution

$$\Pr(X_1 = x_1, X_2 = x_2, \ldots, X_T = x_T | Y_1 = y_1, Y_2 = y_2, \ldots, Y_T = y_T).$$

(g) **[Extra Credit: 3 pts]** Prove the following identity:

$$s(n, t+1) = \sum_{m=1}^{N} s(m, t) c\Big((m, t), (n, t+1)\Big). \tag{1}$$

Equation (1) is extremely important in practice. It enables computing marginals in HMMs in polynomial time, even though $s(p)$ is defined by summing over a set of exponential size. Starting from $t = 1$, at each step, $t = 1 \ldots T$, the algorithm maintains the table $s(n, \cdot)$. Equation (1) is used to compute the new table $s(\cdot, t+1)$ from the old one $s(\cdot, t)$, in time $O(N^2)$ (there are N entries in the table, and each entry requires computing the right-hand side of Equation (1), which is dominated by the summation over N terms). There are T such steps, for a total running time of $O(TN^2)$.