TTIC 31180 — Probabilistic Graphical Models

Problem Set #2 Solutions

Due Date: May 15, 2020

1 Variable Elimination: Marge Innovera Auto Repair [20pts]

You run an auto repair shop, and would like to use a Bayesian network to help you track down issues with the cars that your customers bring in. You decide to relate the reliability of a vehicle's make/model, whether it has been regularly maintained, its general condition, the working state of different parts (e.g., the starter, battery, and alternator), and things that you can readily observe (e.g., whether the engine and lights turn on). Suppose that you represent these as boolean random variables and model their interactions via the Bayesian network in Figure 1.

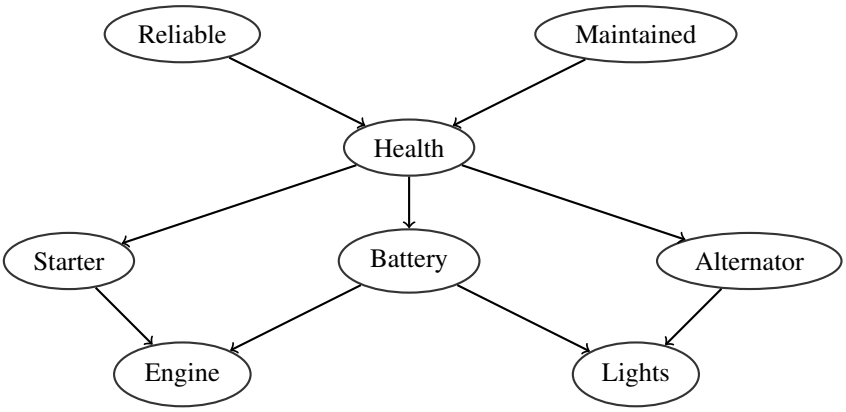


Figure 1: Bayesian network for automotive diagnostics.

(a) **[3 pts]** Suppose that we are interested in calculating the marginal probability $P(E)$ that a car's engine will start. Begin with the expression for $P(E)$ in terms of the unfactorized joint distribution. Then, write the expression for $P(E)$ in terms of the factorized distribution and simplify the computation by pushing in the summations as much as possible.

Solution:

$$
\begin{aligned}
P(E) &= \sum_G \sum_M \sum_H \sum_S \sum_B \sum_A \sum_L P(G, M, H, S, B, A, E, L) \\
&= \sum_G \sum_M \sum_H \sum_S \sum_B \sum_A \sum_L P(E \mid S, B) P(L \mid B, A) P(S \mid H) \\
&\qquad \cdot P(B \mid H) P(A \mid H) P(H \mid G, M) P(G) P(M) \\
&= \sum_S \sum_B P(E \mid S, B) \sum_H P(S \mid H) P(B \mid H) \\
&\qquad \cdot \sum_G P(G) \sum_M P(H \mid G, M) P(M) \overbrace{\sum_A P(A \mid H)}^{1} \overbrace{\sum_L P(L \mid B, A)}^{1} \\
&= \sum_S \sum_B P(E \mid S, B) \sum_H P(S \mid H) P(B \mid H) \sum_G P(G) \sum_M P(H \mid G, M) P(M)
\end{aligned}
$$

■

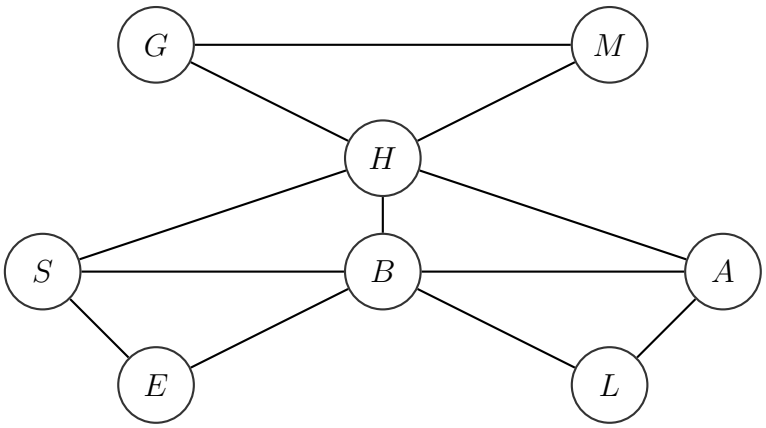
(b) **[1 pts]** What elimination ordering $\prec$ does the above expression correspond to?

Solution:

L, A, M, G, H, B, S (though, note that M and G can be marginalized in parallel to L and A, since they are independent of one another.) ■

(c) **[2 pts]** Draw the moralized graph for the Bayesian network in Figure 1.

Solution:



■

(d) **[6 pts]** Work through the entire process of variable elimination to arrive at $P(E)$ using the above ordering. Use the moralized graph determined above to define the set of initial factors Φ in terms of the original conditional probability distributions. You should then

follow the procedure that we went through in class, whereby in each step you define (i) the intermediate factor ψ as a product of the initial factors ϕ and the new factors (messages) τ that are created; (ii) the variables involved in the computation; and (iii) the new factor (message) that is created along with its scope.

Solution:

First, we define the following factors:

$$\begin{aligned}
\phi_1(M, G, H) &= P(H \mid G, M)P(G)P(M) \\
\phi_2(H, S, B) &= P(S \mid H)P(B \mid H) \\
\phi_3(H, B, A) &= P(A \mid H) \\
\phi_4(S, B, E) &= P(E \mid S, B) \\
\phi_5(B, A, L) &= P(L \mid B, A)
\end{aligned}$$

Next, we perform variable elimination according to the ordering $\prec = L, A, M, G, H, B, S$ (as noted above, we can use any ordering that is consistent with the fact that M and G can be marginalized in parallel to L and A),

(i) Marginalize L:

$$\begin{aligned}
\psi_1(B, A, L) &= \phi_5(B, A, L) \\
\tau_1(B, A) &= \sum_L \psi_1(B, A, L)
\end{aligned}$$

(ii) Marginalize A:

$$\begin{aligned}
\psi_2(H, B, A) &= \phi_3(H, B, A) \cdot \tau_1(B, A) \\
\tau_2(H, B) &= \sum_A \psi_2(H, B, A)
\end{aligned}$$

(iii) Marginalize M and G:

$$\begin{aligned}
\psi_3(M, G, H) &= \phi_1(M, G, H) \\
\tau_3(H) &= \sum_G \sum_M \psi_3(M, G, H)
\end{aligned}$$

(iv) Marginalize H:

$$\begin{aligned}
\psi_5(S, B, H) &= \phi_2(H, S, B) \cdot \tau_2(H, B) \cdot \tau_3(H) \\
\tau_4(S, B) &= \sum_H \psi_5(S, B, H)
\end{aligned}$$

(v) Marginalize B:

$$\begin{aligned}
\psi_6(E, S, B) &= \phi_4(S, B, E) \cdot \tau_4(S, B) \\
\tau_5(S, E) &= \sum_B \psi_6(E, S, B)
\end{aligned}$$

(vi) Marginalize S:

$$\psi_7(S, E) = \tau_5(S, E)$$
$$\tau_6(E) = \sum_S \psi_7(S, E)$$

The scope of each intermediate factor (and message) defines the variables involved in each elimination step. ■

(e) **[2 pts]** What is the scope of the largest intermediate factor that is created using this elimination ordering? Assuming that each variable can take one of k values (i.e., $|\text{Val}(X)| = k$), how many entries does this factor have? What is the computational complexity of variable elimination using this ordering as a function of the total number of variables n?

Solution:

The largest intermediate factors involve three variables, and correspond to $\psi_1, \psi_2, \psi_3, \psi_5$ and ψ_6. If each variable can take one of k variables, these factors each have $\underline{k^3}$ entries.

The computational complexity is $\mathcal{O}(nk^3)$. ■

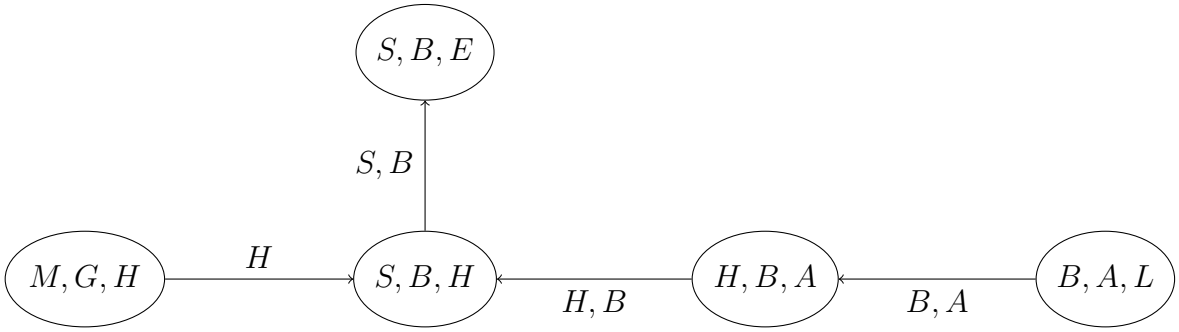
(f) **[2 pts]** Draw the induced graph that is created by variable elimination using the ordering identified above.

Solution:

The induced graph is the same as above. This elimination ordering does not result in any fill edges. ■

(g) **[2 pts]** Clique trees (also known as junction trees) provide a convenient data structure for variable elimination. Draw the clique tree associated with the Bayesian network in Figure 1 that is induced by the most efficient elimination ordering.

Solution:

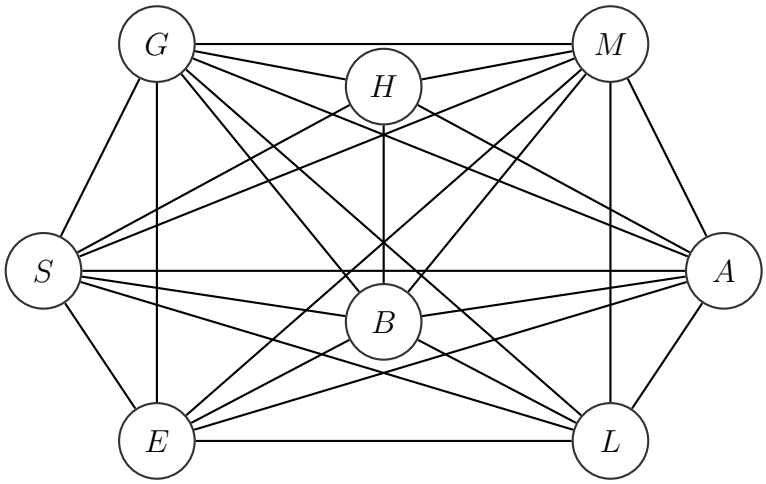


■

(h) **[2 pts]** What is the elimination ordering that results in the greatest computational complexity? What is the computational complexity of variable elimination using this ordering (assuming each variable takes on one of k values)?

Solution:

$\prec = H, B, G, M, S, A, L, E$



The scope of the largest factor that results from this ordering is 7. The overall complexity is $\mathcal{O}(k^7 + 2k^6 + k^5 + k^4 + k^3 + k^2)$, which is upper bounded by $\mathcal{O}(nk^7)$. ■

2 Equivalence of Cluster Trees and Clique Trees [20pts]

A tree $\mathcal{T}$ is a clique tree for an undirected graph $\mathcal{H}$ if the following two conditions hold:

- Each node in $\mathcal{T}$ corresponds to a clique in $\mathcal{H}$ and each maximal clique in $\mathcal{H}$ is a node in $\mathcal{T}$
- Each sepset $\boldsymbol{S}_{ij}$ separates $\boldsymbol{W}_{<(i,j)}$ and $\boldsymbol{W}_{<(j,i)}$ in $\mathcal{H}$, where $\boldsymbol{S}_{i,j}$ is the set of random variables in the intersection of the scope of adjacent cliques $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$ and $\boldsymbol{W}_{<(i,j)}$ and $\boldsymbol{W}_{<(j,i)}$ denote the set of random variables exclusive to the $\boldsymbol{C}_i$-side and $\boldsymbol{C}_j$-side of the tree, respectively.

This exercise asks you to show that the cluster tree associated with variable elimination is actually a clique tree, which involves showing that the above two conditions hold.

(a) **[5 pts]** Provide the sketch of a proof that the cluster tree $\mathcal{T}$ satisfies the running intersection property (i.e., if a variable X_i is in the scope of $\boldsymbol{C}_i$ and $\boldsymbol{C}_k$, then it is in the scope of the unique path from $\boldsymbol{C}_i$ to $\boldsymbol{C}_k$) if and only if $\boldsymbol{W}_{<(i,j)}$ and $\boldsymbol{W}_{<(j,i)}$ are separated given $\boldsymbol{S}_{ij}$ in the chordal graph $\mathcal{H}_\Phi$.

Solution:

Assume that $\mathcal{T}$ satisfies the running intersection property, but that $\boldsymbol{W}_{<(i,j)}$ and $\boldsymbol{W}_{<(j,i)}$ are not separated given $\boldsymbol{S}_{ij}$ in the chordal graph. That means that there must be another path between any variable $X \in \boldsymbol{W}_{<(i,j)}$ and any variable $Y \in \boldsymbol{W}_{<(j,i)}$ that does not pass through any nodes in $\boldsymbol{S}_{ij}$. Each edge along this path is associated with a maximal clique in H_Φ and,

in turn, a node in $\mathcal{T}$. These nodes in $\mathcal{T}$ form another path from $\boldsymbol{C}_i$ to $\boldsymbol{C}_k$, which violates the running intersection property (and the fact that $\mathcal{T}$ is a tree). Thus, $\boldsymbol{W}_{<(i,j)}$ and $\boldsymbol{W}_{<(j,i)}$ must be separated given $\boldsymbol{S}_{ij}$.

Next we prove the other direction, i.e., that if $\boldsymbol{W}_{<(i,j)}$ and $\boldsymbol{W}_{<(j,i)}$ are separated given $\boldsymbol{S}_{ij}$, then $\mathcal{T}$ obeys the running intersection property. We will do so by contradiction, assuming that $\mathcal{T}$ does not obey the running intersection property. Specifically, assume that there are cliques $\boldsymbol{C}_i$ and $\boldsymbol{C}_k$ that both contain X_i, but there is at least one clique $\boldsymbol{C}_j$ along the path from $\boldsymbol{C}_i$ to $\boldsymbol{C}_k$ that does not include X_i in its scope. Consider the sepset $\boldsymbol{S}$ associated with one of these nodes and its immediate neighbor in $\mathcal{T}$. Since X_i is not in this sepset, then there is another path from the nodes associated with $\boldsymbol{C}_i$ to the nodes associated with $\boldsymbol{C}_k$ that does not pass through any nodes in this sepset, which violates our separation assumption. ■

(b) **[5 pts]** Provide a sketch of why every node in $\mathcal{T}$ is a clique in a chordal graph that includes $\mathcal{H}$ and that every maximal clique in this chordal graph is a node in $\mathcal{T}$.

Solution:

Triangulating a graph may involve adding edges, but does not remove edges. As a result, every clique in H is also a clique in the chordal graph for H. Thus, per the definition above, since every node in $\mathcal{T}$ corresponds to a clique in H, every node also corresponds to a clique in the chordal graph for H.

Now, we just need to show that every maximal clique in the chordal graph for H is also a node in $\mathcal{T}$. Recall that any elimination ordering associated with a clique tree $\mathcal{T}$ has a corresponding induced graph. By Theorem 9.8 in Koller & Friedman, every induced graph is chordal. Next, we use Theorem 9.6 in Koller & Friedman, which states that every maximal clique in the induced graph (and, in turn, the chordal graph) corresponds to an intermediate factor in variable elimination. Since $\mathcal{T}$ exactly expresses these factors, we conclude that every maximal clique in the chordal graph for H is a node in the clique tree $\mathcal{T}$. ■

(c) **[5 pts]** Using the independencies associated with the sepset from above, prove that when a clique tree is calibrated, we can represent the joint distribution as

$$\tilde{P}(\boldsymbol{X}) = \frac{\prod_i P(\boldsymbol{C}_i)}{\prod_{ij} P(\boldsymbol{S}_{i,j})}$$

Hint: It may be helpful to use the fact that that the subgraphs on the $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$ side are each trees.

Solution:

Consider two adjacent nodes i and j in $\mathcal{T}$ with sepset $\boldsymbol{S}_{ij}$. The set of random variables is

then $\boldsymbol{X} = \{\boldsymbol{W}_{<(i,j)}, \boldsymbol{S}_{ij}, \boldsymbol{W}_{<(j,i)}\}$. We can write the joint distribution as

$$\begin{aligned}
P(\boldsymbol{X}) &= P(\boldsymbol{W}_{<(i,j)}, \boldsymbol{S}_{ij}, \boldsymbol{W}_{<(j,i)}) \\
&= P(\boldsymbol{W}_{<(i,j)}, \boldsymbol{W}_{<(j,i)} \mid \boldsymbol{S}_{ij}) P(\boldsymbol{S}_{ij}) \\
&= P(\boldsymbol{W}_{<(i,j)} \mid \boldsymbol{S}_{ij}) P(\boldsymbol{W}_{<(j,i)} \mid \boldsymbol{S}_{ij}) P(\boldsymbol{S}_{ij}) \\
&= \frac{P(\boldsymbol{W}_{<(i,j)}, \boldsymbol{S}_{ij})}{P(\boldsymbol{S}_{ij})} \frac{P(\boldsymbol{W}_{<(j,i)}, \boldsymbol{S}_{ij})}{P(\boldsymbol{S}_{ij})} P(\boldsymbol{S}_{ij}) \\
&= \frac{1}{P(\boldsymbol{S}_{ij})} P(\boldsymbol{W}_{<(i,j)}, \boldsymbol{S}_{ij}) P(\boldsymbol{W}_{<(j,i)}, \boldsymbol{S}_{ij})
\end{aligned}$$

The sets $\{\boldsymbol{S}_{ij}, \boldsymbol{W}_{<(i,j)}\}$ and $\{\boldsymbol{W}_{<(j,i)}, \boldsymbol{S}_{ij}\}$ correspond to disjoint subtrees of $\mathcal{T}$. Continuing with recursion, we get the desired expression for the joint distribution. ■

(d) **[5 pts]** Theorem 10.6 in the text states that, if $\mathcal{T}$ is a clique tree for a set of factors Φ, then there exists a clique tree $\mathcal{T}'$ such that

- each clique in $\mathcal{T}'$ is also a clique in $\mathcal{T}$; and
- there is no pair of cliques $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$ in $\mathcal{T}'$ such that $\boldsymbol{C}_i \subset \boldsymbol{C}_j$.

Show that given a valid clique tree, the variable elimination step yields a valid clique tree, i.e., that it is a tree, that it satisfies the running intersection property, and that it satisfies the family preservation property.

Solution:

TBD ■

3 Variable Elimination within Clique Trees [20pts]

Consider an undirected graphical model in the form of a Markov chain $X_1 - X_2 - \cdots - X_n$ where each X_i can take on k possible values. We can form a clique tree $\mathcal{T} : \boldsymbol{C}_1 - \boldsymbol{C}_2 - \cdots - \boldsymbol{C}_{n-1}$ for this graphical model by setting $\boldsymbol{C}_i = \{X_i, X_{i+1}\}$. Let's assume that the clique tree has been calibrated. Suppose that we would like to solve for $P(X_i, X_j)$ for $j > i$. When $j = i + 1$, we can readily compute the marginal from the belief at each clique, but the process is less straightforward when $j \neq i + 1$.

(a) **[6 pts]** Describe how we can use variable elimination to solve for $P(X_i, X_j)$ for some $j > i$ in time that is linear in the length of the sequence between i and j, given a calibrated clique tree.

Solution:

Consider some index k. Per Bayes' rule, we can compute the following conditional distri-

bution as

$$P(X_k \mid X_{k+1}) = \frac{P(X_k, X_{k+1})}{P(X_{k+1})},$$

where we can calculate $P(X_k, X_{k+1})$ and $P(X_{k+1})$ from the potential associated with $\boldsymbol{C}_k$ since the clique tree is calibrated.

Let $i, i+1, \ldots, j-1, j$ index the path from X_i to X_j in the original Markov chain (and, in turn, the path in the clique tree). Now, we can recursively compute the following conditional distributions:

$$\begin{aligned}
P(X_i \mid X_{i+2}) &= \sum_{X_{i+1}} P(X_i, X_{i+1} \mid X_{i+2}) \\
&= \sum_{X_{i+1}} P(X_i \mid X_{i+1}) P(X_{i+1} \mid X_{i+2}) \\
P(X_i \mid X_{i+3}) &= \sum_{X_{i+2}} P(X_i, X_{i+2} \mid X_{i+3}) \\
&= \sum_{X_{i+2}} P(X_i \mid X_{i+2}) P(X_{i+2} \mid X_{i+3}) \\
&\vdots \\
P(X_i \mid X_j) &= \sum_{X_{j-1}} P(X_i, X_{j-1} \mid X_j) \\
&= \sum_{X_{j-1}} P(X_i \mid X_{j-1}) P(X_{j-1} \mid X_j)
\end{aligned}$$

The desired joint distribution then follows from the last conditional as

$$P(X_i, X_j) = P(X_i \mid X_j) P(X_j)$$

Now, each of the joint distributions above can be computed directly from the beliefs associated with the clique tree since it is calibrated.

Each of the $j-i$ operations requires k^3 multiplications and $k^2(k-1)$ additions, and thus involves $\mathcal{O}(k^3)$ computations. This gives rise to an overall complexity of $\mathcal{O}\left((j-i)k^3\right)$, which is linear in the length of the sequence. ■

(b) **[4 pts]** What is the running time of this algorithm if you'd like to compute $P(X_i, X_j)$ for all n choose 2 combinations of i and j?

Solution:

The complexity of performing the above computation for all $\binom{n}{2}$ combinations is:

$$\begin{aligned}
\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}(j-i)k^3 &= \sum_{i=1}^{n-1} i(n-i)k^3 \\
&= \left(n\sum_{i=1}^{n-1} i - \sum_{i=1}^{n-1} i^2\right)k^3 \\
&= \left(n\frac{n(n-1)}{2} - \frac{n(n-1)(2n-1)}{6}\right)k^3 \\
&= \frac{3n^3 - 3n^2 - 2n^3 + 3n^2 - n}{6}k^3 \\
&= \mathcal{O}(n^3)
\end{aligned}$$

where the first term in the third line follows from the expression of the arithmetic series and the second from Faulhaber's formula. ■

(c) **[4 pts]** Consider the case where $n = 4$. Show that if you cache $P(X_1, X_3)$, you can compute $P(X_1, X_4)$ more efficiently than by directly applying variable elimination as described in (a) above.

Solution:

$$\begin{aligned}
P(X_1, X_4) &= \sum_{X_3} P(X_1, X_3, X_4) \\
&= \sum_{X_3} P(X_1, X_4 \mid X_3)P(X_3) \\
&= \sum_{X_3} P(X_1 \mid X_3)P(X_4 \mid X_3)P(X_3) \\
&= \sum_{X_3} \frac{P(X_1, X_3)P(X_3, X_4)}{P(X_3)}
\end{aligned}$$

This involves $\mathcal{O}(k^3)$ operations. ■

(d) **[6 pts]** Based on the intuition that you used to compute $P(X_1, X_4)$ in (c), design a dynamic programming algorithm (caching partial results) that computes $P(X_i, X_j)$ for all n choose 2 combinations of i and j in time that is (asymptotically) much lower than the complexity identified in (b). What is the asymptotic running time of this algorithm?

Solution:

From the calibrated clique tree, we can readily determine $P(X_i, X_{i+1}) \forall i \in \{1, \ldots, n-1\}$. For $i \in \{1, \ldots, n-2\}$, we can compute $P(X_i, X_{i+2})$ as

$$\begin{aligned} P(X_i, X_{i+2}) &= \sum_{X_{i+1}} P(X_i, X_{i+1}, X_{i+2}) \\ &= \sum_{X_{i+1}} P(X_i \mid X_{i+1}) P(X_{i+2} \mid X_{i+1}) P(X_{i+1}) \\ &= \sum_{X_{i+1}} \frac{P(X_i, X_{i+1}) P(X_{i+1}, X_{i+2})}{P(X_{i+1})} \end{aligned}$$

where we can determine $P(X_{i+1})$ from $P(X_i, X_{i+1})$.

Similarly, as we did in part (c), we can compute $P(X_i, X_{i+3})$ as

$$\begin{aligned} P(X_i, X_{i+3}) &= \sum_{X_{i+2}} P(X_i, X_{i+2}, X_{i+3}) \\ &= \sum_{X_{i+2}} P(X_i \mid X_{i+2}) P(X_{i+3} \mid X_{i+2}) P(X_{i+2}) \\ &= \sum_{X_{i+2}} \frac{P(X_i, X_{i+2}) P(X_{i+2}, X_{i+3})}{P(X_{i+3})} \end{aligned}$$

where we use cached values for $P(X_i, X_{i+2})$ determined above and compute $P(X_{i+3})$ from $P(X_{i+2}, X_{i+3})$.

We continue in this way, computing $P(X_i, X_{i+l})$ for $i \in \{1, \ldots n-l\}$ and $l \in \{4, \ldots, n-i\}$, each time reusing the cached values for $P(X_i, X_{i+l-1})$.

The complexity of each operation is $\mathcal{O}(k^3)$ due to resusing cached computations. As a result, the cost of computing the joint likelihood for all $\mathcal{O}(n^2)$ pairs is $\mathcal{O}(n^2 k^3)$, which is more efficient than the method used in part (b).

■

4 Gaussian Belief Propagation [10pts]

Consider a Gaussian Markov random field over $x_1, x_2, \ldots, x_n$ with the pairwise and unary potentials defined as

$$\begin{aligned} \phi_{ij}(x_i, x_j) &= \exp\left(-\Lambda_{ij} x_i x_j\right) \\ \phi_i(x_i) &= \exp\left(-\frac{1}{2}(\Lambda_{ii} x_i^2 + 2\eta_i x_i)\right), \end{aligned}$$

where Λ and η are the information matrix and information vector, respectively.

This problem asks you to derive an expression for the belief associated with x_i that results from belief propagation.

(a) **[2 pts]** Consider two univariate Gaussians $p(x) = \mathcal{N}^{-}1(x; \eta_1, \lambda_1)$ and $p(x) = \mathcal{N}^{-1}(x; \eta_2, \lambda_2)$, where η_i and λ_i are the scalar parameters of the information form. Show that the product of these two Gaussians is itself a scaled Gaussian, i.e.,

$$\mathcal{N}^{-1}(x; \eta_1, \lambda_1) \cdot \mathcal{N}^{-1}(x; \eta_2, \lambda_2) = \mathcal{N}^{-1}(x; \bar{\eta}, \bar{\lambda})$$

Solution:

TBD ■

(b) **[6 pts]** Derive an expression for the the message $m_{j \to i}(x_i)$ sent to node i from each neighbor j. Hint: first compute the messages that are sent to node j from its neighbors (excluding node i), and use these messages and the factor associated with node j in defining $m_{j \to i}(x_i)$.

Solution:

■

(c) **[2 pts]** Write out an expression for the resulting belief $\beta(x_i)$ associated with node i.

Solution:

TBD ■

5 Inference in Hidden Markov Models [30pts]

The Hidden Markov Model (HMM) is a simple, yet highly useful dynamic Bayesian network that has been used to model a wide variety of problems ranging from speech recognition to activity recognition to machine translation to analyzing the stock market.

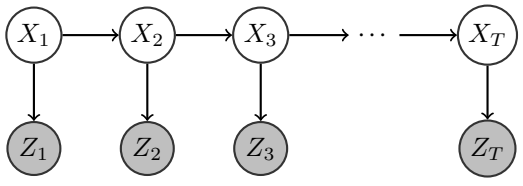


Figure 2: A hidden Markov model.

HMMs (Fig. 2) consist of a sequence of (latent) variables $\boldsymbol{X} = \{X_1, X_2, \ldots, X_T\}$ (states) that each generate an observed variable (emissions) $\boldsymbol{Z} = \{Z_1, Z_2, \ldots Z_T\}$. The latent variables form a Markov chain, meaning that the random variable at each point in time X_t is conditionally independent of all prior states given the state at the previous point in time X_{t-1}. Associated with

an HMM are the following conditional probability distributions: $P(X_1)$, $P(X_t \mid X_{t-1})$ for $t \in \{2, 3, \ldots, T\}$, and $P(Z_t \mid X_t)$ for $t \in \{1, 2, \ldots, T\}$. The joint probability factorizes as

$$P(\boldsymbol{X}, \boldsymbol{Z}) = P(X_1) \prod_{t=2}^{T} P(X_t \mid X_{t-1}) \prod_{t=1}^{T} P(Z_t \mid X_t),$$

5.1 Belief Propagation for HMMs [15pts]

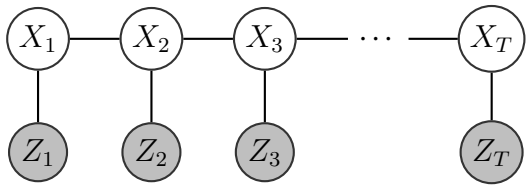
With HMMs, we are often interested in performing inference over the sequence of latent random variables (e.g., transmitted bit sequence) based upon the sequence of noisy observations $Z_t = z_t$ for $t \in \{1, \ldots, T\}$ (e.g., observed bit sequence), i.e., $P(X_t \mid z_1, z_2, \ldots, z_T)$. We can perform inference using belief propagation, whereby we transmit messages upstream (forward) and downstream (backward) in a clique tree. The following questions ask you to derive these messages.

To that end, we convert the dynamic Bayesian network to a clique tree by first converting it to an undirected graph and defining the factors in terms of the conditional probability distributions. We can then associate a factor with each clique of the graphical model.

(a) **[1 pts]** Draw the corresponding undirected graph and define the associated factors in terms of the conditional probability distributions from the original Bayesian Network.

Solution:

The undirected graph follows by changing every directed edge to an undirected edge and marying the parents. Since each random variable has no more than one parent, the undirected graph follows simply as



The corresponding factors are, for $t \in \{1, 2, \ldots, T\}$:

$$\begin{aligned} \phi(X_1, Z_1) &= P(Z_1 \mid X_1) P(X_1) \\ \phi(X_t, X_{t+1}) &= P(X_{t+1} \mid X_t) \\ \phi(X_t, Z_t) &= P(Z_t \mid X_t) \end{aligned}$$

■

(b) **[2 pts]** Draw the clique tree associated with the undirected graph and assign each factor to a node in the clique tree. Indicate the scope for each clique as well as each sepset. Hint: Since $Z_t = z_t$ is observed, we can treat it as a constant, which should simplify the structure of the clique tree.

Solution:

Since $Z_t = z_t$ is observed, we can treat it as a fixed parameter, which allows us to express the factors as functions of X_t only. The corresponding clique tree is then a chain.

■

The factors associated with each node are

$$\begin{aligned} 1: &\ \phi(X_1, z_1) \\ 2: &\ \phi(X_1, X_2), \phi(X_2, z_2) \\ t: &\ \phi(X_{t-1}, X_t), \phi(X_t, z_t) \\ &\ \vdots \\ T: &\ \phi(X_{T-1}, X_T), \phi(X_T, z_T) \end{aligned}$$

where, we use $Z_t = z_t$ to make it clear that we are using the instantiated values for Z_t.

Alternatively, it is possible to formulate this as a clique tree where there is a separate node for each X_t, Z_t (e.g., below each X_{t-1}, X_t) and assign $\phi(X_t, Z_t)$ to each.

(c) **[4 pts]** Derive the expressions for the messages passed in the forward (left-to-right) direction, starting with $t = 1$. Show how these messages relate to a marginal under the original distribution.

Solution:

$$
\begin{aligned}
m_{1\to 2}(X_1) &= \phi(X_1, z_1) \\
&= P(z_1 \mid X_1)P(X_1) \\
m_{2\to 3}(X_2) &= \sum_{X_1} \phi(X_1, X_2)\phi(X_2, z_2)m_{1\to 2}(X_1) \\
&= \sum_{X_1} P(z_2 \mid X_2)P(X_2 \mid X_1)P(z_1 \mid X_1)P(X_1) \\
&= \sum_{X_1} P(z_1, z_2, X_1, X_2) \\
&= P(z_1, z_2, X_2) \\
&\vdots \\
m_{t\to t+1}(X_t) &= \sum_{X_{t-1}} \phi(X_{t-1}, X_t)\phi(X_t, z_t)m_{t-1\to t}(X_{t-1}) \\
&= \sum_{X_{t-1}} P(z_t \mid X_t)P(X_t \mid X_{t-1})P(z_1, z_2, \ldots, z_{t-1}, X_{t-1}) \\
&= P(z_1, z_2, \ldots, z_t, X_t) \qquad \vdots \\
m_{T-1\to T}(X_{T-1}) &= \sum_{X_{T-2}} \phi(X_{T-2}, X_{T-1})\phi(X_{T-1}, z_{T-1})m_{T-2\to T-1}(X_{T-2}) \\
&= P(z_1, z_2, \ldots, z_{T-1}, X_{T-1})
\end{aligned}
$$

Thus, we see that the message sent from t to $t+1$ corresponds to the marginal over the history of previous observations and the current bit value

$$m_{t\to t+1}(X_t) = P(z_1, z_2, \ldots, z_t, X_t)$$

■

(d) **[4 pts]** Derive the expressions for messages passed in the opposite (backward) starting with $t = T$. Show how this relates to a conditional distribution over the observations as a function of the latent state(s).

Solution:

$$
\begin{aligned}
m_{T\to T-1}(X_{T-1}) &= \sum_{X_T} \phi(X_{T-1}, X_T)\phi(X_T, z_T) \\
&= \sum_{X_T} P(X_T \mid X_{T-1})P(z_T \mid X_T) \\
&= \sum_{X_T} P(z_t, X_T \mid X_{T-1}) \\
&= P(z_T \mid X_{T-1}) \\
m_{T-1\to T-2}(X_{T-2}) &= \sum_{X_{T-1}} \phi(X_{T-2}, X_{T-1})\phi(X_{T-1}, z_{T-1})m_{T\to T-1}(X_{T-1}) \\
&= \sum_{X_{T-1}} P(X_{T-1} \mid X_{T-2})P(z_{T-1} \mid X_{T-1})P(z_T \mid X_{T-1}) \\
&= \sum_{X_{T-1}} P(z_{T-1}, z_T, X_{T-1} \mid X_{T-2}) \\
&= P(z_{T-1}, z_T \mid X_{T-2}) \\
&\vdots \\
m_{t+2\to t+1}(X_{t+1}) &= \sum_{X_{t+2}} \phi(X_{t+1}, X_{t+2})\phi(X_{t+2}, z_{t+2})m_{t+3\to t+2}(X_{t+2}) \\
&= \sum_{X_{t+2}} P(X_{t+2} \mid X_{t+1})P(z_{t+2} \mid X_{t+2})P(z_{t+3}, z_{t+4}, \ldots, z_T \mid X_{t+2}) \\
&= \sum_{X_{t+2}} P(z_{t+2}, z_{t+2}, \ldots z_T, X_{t+2} \mid X_{t+1}) \\
&= P(z_{t+2}, z_{t+2}, \ldots z_T \mid X_{t+1})
\end{aligned}
$$

Thus, we see that the message sent from $t+2$ to $t+1$ corresponds to the conditional likelihood of the future measurements given X_{t+1}

$$m_{t+2\to t+1}(X_{t+1}) = P(z_{t+2}, z_{t+2}, \ldots z_T \mid X_{t+1})$$

■

(e) **[4 pts]** Derive an expression for the desired marginal distribution $P(X_t \mid z_1, z_2, \ldots, z_T)$ in terms of the above messages and initial factors.

Solution:

The distribution follows from Bayes' rule

$$
\begin{aligned}
P(X_t \mid z_1, z_2, \ldots, z_T) &= \frac{P(X_t, z_1, \ldots, z_T)}{P(z_1, z_2, \ldots, z_T)} \\
&= \frac{P(z_{t+1}, \ldots, z_T \mid X_t) P(X_t, z_1, \ldots, z_t)}{P(z_1, \ldots, z_T)} \\
&= \frac{\sum_{X_{t+1}} P(z_{t+1}, z_{t+2}, \ldots, z_T, X_{t+1} \mid X_t) P(X_t, z_1, \ldots, z_t)}{P(z_1, \ldots, z_T)} \\
&= \frac{\sum_{X_{t+1}} P(z_{t+2}, \ldots, z_T \mid X_{t+1}) P(z_{t+1} \mid X_{t+1}) P(X_{t+1} \mid X_t) P(X_t, z_1, \ldots, z_t)}{P(z_1, \ldots, z_T)} \\
&= \frac{\sum_{X_{t+1}} m_{t+2 \to t+1}(X_{t+1}) P(z_{t+1} \mid X_{t+1}) P(X_{t+1} \mid X_t) m_{t \to t+1}(X_t)}{P(z_1, \ldots, z_T)} \\
&\propto \sum_{X_{t+1}} m_{t+2 \to t+1}(X_{t+1}) P(z_{t+1} \mid X_{t+1}) P(X_{t+1} \mid X_t) m_{t \to t+1}(X_t),
\end{aligned}
$$

which is exactly the operations that we would do as part of sum-product variable elimination to determine the belief at node $t+1$. The denominator expresses the likelihood of the observations and can be determined by marginalizing X_t from the numerator (i.e., normalization).

Note that depending on how you chose to assign factors to clique nodes, you may arrive at a different expression for the forward and backward messages and, in turn, the expression for this conditional likelihood as a function of the messages. ■

5.2 Forward-Backward Algorithm [15pts]

The forward-backward algorithm is an efficient inference algorithm within an HMM to compute $P(X_t \mid z_1, \ldots, z_T)$, which is also referred to as smoothing. The α, β algorithm is one of the first variants of the forward-backward algorithm and involves recognizing and exploiting repeated calculations to improve computational efficiency. In particular, it utilizes the fact that $P(X_t \mid Z_1, \ldots, Z_T)$ is proportional to $P(X_t, Z_1, \ldots, Z_T)$, along with the fact that $\{Z_1, \ldots, Z_t\}$ is d-separated from $\{Z_{t+1}, \ldots, Z_T\}$ given X_t.

(a) **[4 pts]** Let $\alpha_t(X_t) = P(Z_1, \ldots, Z_t, X_t)$ be the forward messages. Derive a recursive expression for computing these forward messages starting with $\alpha_1(X_1)$.

Solution:

We derive a recursive expression for the forward messages by exploiting the independencies

conveyed by the graph, particularly the Markov independencies.

$$
\begin{aligned}
\alpha(X_1) &= P(Z_1, X_1) \\
&= P(Z_1 \mid X_1) P(X_1) \\
\alpha(X_2) &= P(Z_1, Z_2, X_2) \\
&= \sum_{X_1} P(Z_1, Z_2, X_2, X_1) \\
&= \sum_{X_1} P(Z_2 \mid X_2) P(X_2 \mid X_1) P(Z_1, X_1) \\
&= \sum_{X_1} P(Z_2 \mid X_2) P(X_2 \mid X_1) \alpha(X_1) \\
&\vdots \\
\alpha(X_t) &= P(Z_1, Z_2, \ldots, Z_t, X_t) \\
&= \sum_{X_{t-1}} P(Z_1, Z_2, \ldots, Z_t, X_t, X_{t-1}) \\
&= \sum_{X_{t-1}} P(Z_t \mid X_t) P(X_t \mid X_{t-1}) P(Z_1, Z_2, \ldots, Z_{t-1}, X_{t-1}) \\
&= \sum_{X_{t-1}} P(Z_t \mid X_t) P(X_t \mid X_{t-1}) \alpha(X_{t-1})
\end{aligned}
$$

Thus, we get a recursive expression for $\alpha(X_t)$ in terms of $\alpha(X_{t-1})$:

$$\alpha(X_t) = \sum_{X_{t-1}} P(Z_t \mid X_t) P(X_t \mid X_{t-1}) \alpha(X_{t-1})$$

These terms are computed by working *forwards* in time, hence the term "forward" in forward-backward algorithm. ■

(b) **[4 pts]** Let $\beta_t(X_t) = P(Z_{t+1}, \ldots, Z_T \mid X_t)$ be the backward messages. Derive a recursive expression for computing these backward messages starting with $\beta_T(X_T) = 1$.

Solution:

$$
\begin{aligned}
\beta(X_{T-1}) &= P(Z_T \mid X_{T-1}) \\
&= \sum_{X_T} P(Z_T, X_T \mid X_{T-1}) \\
&= \sum_{X_T} P(Z_T \mid X_T) P(X_T \mid X_{T-1}) \\
&= \sum_{X_T} P(Z_T \mid X_T) P(X_T \mid X_{T-1}) \beta(X_T) \quad \text{Since } \beta(X_T) = 1 \\
\beta(X_{T-2}) &= P(Z_{T-1}, Z_T \mid X_{T-2}) \\
&= \sum_{X_{T-1}} P(Z_{T-1}, Z_T, X_{T-1} \mid X_{T-2}) \\
&= \sum_{X_{T-1}} P(Z_{T-1}, Z_T \mid X_{T-1}) P(X_{T-1} \mid X_{T-2}) \\
&= \sum_{X_{T-1}} P(Z_{T-1} \mid X_{T-1}) P(X_{T-1} \mid X_{T-2}) P(Z_T \mid X_{T-1}) \quad \text{Since } Z_{T-1} \perp Z_T \mid X_{T-1} \\
&= \sum_{X_{T-1}} P(Z_{T-1} \mid X_{T-1}) P(X_{T-1} \mid X_{T-2}) \beta(X_{T-1}) \\
&\vdots \\
\beta(X_t) = &= P(Z_{t+1}, \ldots, Z_{T-1}, Z_T \mid X_t) \\
&= \sum_{X_{t+1}} P(Z_{t+1}, \ldots, Z_{T-1}, Z_T, X_{t+1} \mid X_t) \\
&= \sum_{X_{t+1}} P(Z_{t+1}, \ldots, Z_{T-1}, Z_T \mid X_{t+1}) P(X_{t+1} \mid X_t) \\
&= \sum_{X_{t+1}} P(Z_{t+1} \mid X_{t+1}) P(X_{t+1} \mid X_t) P(Z_{t+2}, \ldots, Z_{T-1}, Z_T \mid X_{t+1}) \\
&= \sum_{X_{t+1}} P(Z_{t+1} \mid X_{t+1}) P(X_{t+1} \mid X_t) \beta(X_{t+1})
\end{aligned}
$$

This gives rise to a recursive expression for $\beta(X_t)$ in terms of $\beta(X_{t+1})$

$$\beta(X_t) = \sum_{X_{t+1}} P(Z_{t+1} \mid X_{t+1}) P(X_{t+1} \mid X_t) \beta(X_{t+1})$$

These terms are computed by working *backwards* in time, giving rise to the term "backward" in the forward-backward algorithm. ■

(c) **[4 pts]** Derive an expression for the desired marginal distribution $P(X_t \mid Z_1, Z_2, \ldots, Z_T)$ in terms of $\alpha_t(X_t)$ and $\beta_t(X_t)$.

Solution:

$$
\begin{aligned}
P(X_t \mid Z_1, Z_2, \ldots, Z_T) &= \frac{P(X_t, Z_1, Z_2, \ldots, Z_T)}{P(Z_1, Z_2, \ldots, Z_T)} \\
&= \frac{P(Z_{t+1}, \ldots, Z_T \mid X_t, Z_1, \ldots, Z_t) P(X_t, Z_1, \ldots, Z_t)}{P(Z_1, Z_2, \ldots, Z_T)} \\
&= \frac{P(Z_{t+1}, \ldots, Z_T \mid X_t) P(X_t, Z_1, \ldots, Z_t)}{P(Z_1, Z_2, \ldots, Z_T)} \\
&= \frac{\beta(X_t)\alpha(X_t)}{P(Z_1, Z_2, \ldots, Z_T)}
\end{aligned}
$$

where the third line follows from the fact that $(Z_{t+1}, \ldots, Z_T) \perp (Z_1, \ldots Z_t) \mid X_t$. We can compute the denominator by marginalizing over any α, β pair, the most convenient being the pair at time T

$$
P(Z_1, Z_2, \ldots, Z_T) = \sum_{X_T} \alpha(X_T)\beta(X_T) = \sum_{X_T} \alpha(X_T).
$$

■

(d) **[3 pts]** How do the messages derived above as part of belief propagation relate to α and β?

Solution:

For node t, the message that is sent forward $m_{t \to t+1}(()X_t) = P(Z_1, Z_2, \ldots, Z_t, X_t)$ is exactly $\alpha(X_t)$. The message that is received from the node on the right $m_{t+1 \to t}(X_t) = P(Z_{t+1}, Z_{t+2}, \ldots, Z_T \mid X_t)$ is $\beta(X_t)$.

If we consider the messages coming into node t from the left (forward) and right (backward), we have

$$
\begin{aligned}
m_{t-1 \to t}(X_{t-1}) &= P(Z_1, Z_2, \ldots, Z_{t-1}, X_{t-1}) = \alpha(X_{t-1}) \\
m_{t+1 \to t}(X_t) &= P(Z_{t+1}, Z_{t+2}, \ldots Z_T \mid X_t) = \beta X_t
\end{aligned}
$$

which explains the difference between the expression for the conditional distribution $P(X_t \mid Z_1, Z_2, \ldots, Z_T$ in terms of forward and backward messages, as determined in 4.1(e) and the expression in terms of α and β in 4.2(c). ■