Problem Set 2 Solutions

Calvin Walker

1.

(a)

$$\begin{aligned}
P(E) &= \sum_{r,m,h,s,b,a,l} P(r,m,h,s,b,a,l,E) \\
&= \sum_{r,m,h,s,b,a,l} P(r)P(m)P(h|r,m)P(s|h)P(a|h)P(b|h)P(E|s,b)P(l|b,a) \\
&= \sum_b \sum_s P(E|s,b) \sum_h P(s|h)P(b|h) \sum_a P(a|h) \sum_l P(l|b,a) \sum_r P(r) \sum_m P(h|r,m)P(m)
\end{aligned}$$

(b) The corresponding elimination ordering is $\prec = M, R, L, A, H, S, B$

(c)

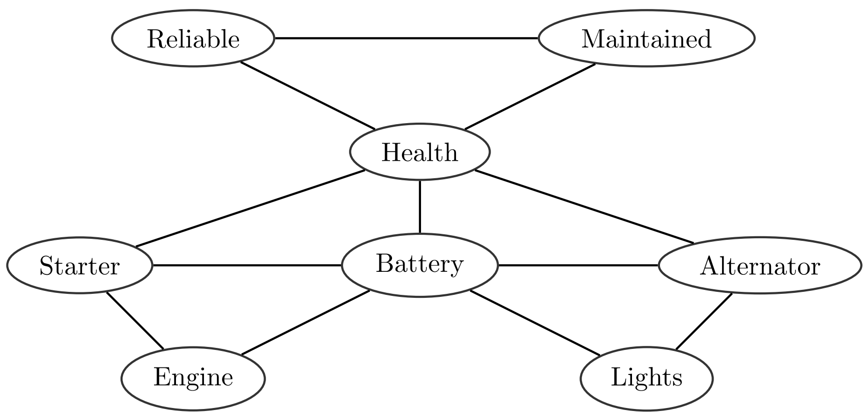

Figure 1: Moralized graph for the Bayesian Network

(d) The initial set of factors is $\Phi = \{\phi(H,M,R), \phi(S,B,H), \phi(H,B,A), \phi(S,B,E), \phi(B,A,L)\}$

Step	Variable Eliminated	Intermediate Factor	Variables Involved	New Factor
1	M	$\psi_1(H,M,R) = \phi(H,M,R)$	H,M,R	$\tau_1(H,R) = \sum_m \psi_1(H,m,R)$
2	R	$\psi_2(H,R) = \tau_1(H,R)$	H,R	$\tau_2(H) = \sum_r \psi_2(H,r)$
3	L	$\psi_3(B,A,L) = \phi(B,A,L)$	B,A,L	$\tau_3(B,A) = \sum_l \psi_3(B,A,l)$
4	A	$\psi_4(A,B,H) = \tau_3(B,A)\phi(H,B,A)$	H,B,A	$\tau_4(B,H) = \sum_a \psi_4(a,B,H)$
5	H	$\psi_5(S,B,H) = \tau_2(H)\tau_4(B,H)\phi(S,B,H)$	S,B,H	$\tau_5(S,B) = \sum_h \psi_5(S,B,h)$
6	B	$\psi_6(S,B,E) = \tau_5(S,B)\phi(S,B,E)$	S,B,E	$\tau_6(S,E) = \sum_l \psi_6(S,b,E)$
7	S	$\psi_6(S,E) = \tau_6(S,E)$	S,E	$\tau_7(E) = \sum_s \psi_7(s,E)$

Table 1: Variable Elimination Procedure for $P(E)$

(e) The largest induced scope is 2, and assuming each variable can take k possible values, the computational complexity would be $O(nk^2)$, since the factor has k^2 entries.

(f) The induced graph is the same as the moralized netowrk in part (c), since there are already edges between all variables appearing in some ψ generated by the variable elimination.

(g) Since the ordering in part (*b*) produced the most efficient variable elimination in part (*d*), the associated clique tree is that of part (*d*), shown below:

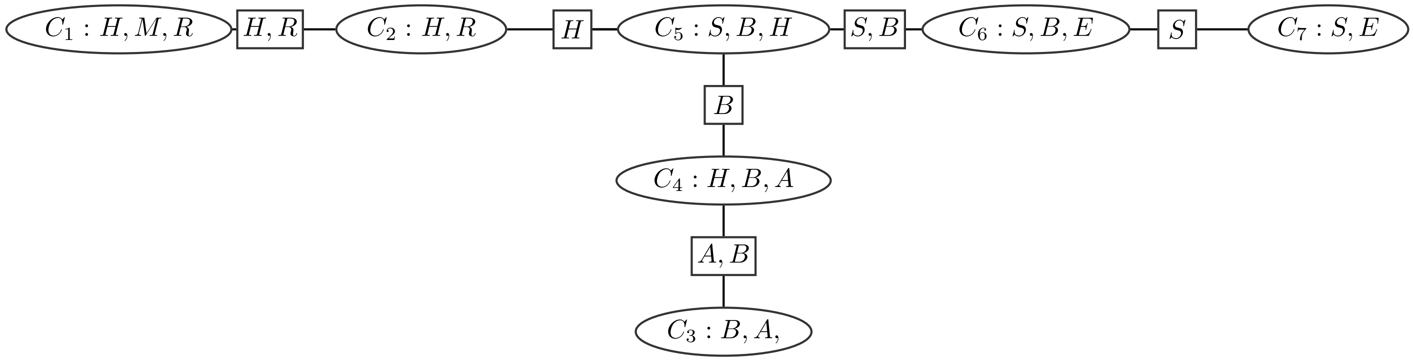

Figure 2: Clique Tree for the Bayesian Network

(h) If we first eliminate H, there is an induced scope of 5, as we create a new factor $\tau_1(R, M, S, B, A)$, next we could eliminate B, to induce a scope of 6, as we create a new factor $\tau_2(R, M, S, A, E, L)$ with the remaining 6 random variables. So any elimination ordering starting $\prec = H, B \ldots$ would lead to a computational complexity of $O(nk^6)$

2.

(a) Consider two clusters C_i and C_k in $\mathcal{T}$, a cluster tree for the graph $\mathcal{H}$, and a variable X that is eliminated at C_k. If $W_{<(i,j)}$ and $W_{<(j,i)}$ are seperated given $S_{i,j}$ in $\mathcal{H}$, then no clusters upstream from C_k can contain X, since X is not in the sepset of any upsteam cluster of C_k. We know X must be present in the message $m_{i \to j}$ to the next cluster from C_i, and this message is multiplied into the ensuing messages (so these clusters have X in their scope) until we reach C_k. Thus, X is in the scope of the unique path from C_i to C_k, and $\mathcal{T}$ satisfies the running intersection property.

Assume that the cluster tree $\mathcal{T}$ satisfies the running intersection property, so for some clusters C_i and C_k with a variable X that is eliminated at C_k, X must be in every cluster in the unique path from C_i to C_k. Thus, it is also in the sepset between each clique along this unique path. If X is not in the sepset between two cliques C_i and C_j, then X cannot be upstream of C_j, so $W_{<(i,j)}$ and $W_{<(j,i)}$ must be seperated given $S_{i,j}$ if $\mathcal{T}$ satisfies the running intersection property.

(b) Variable elimination produces an induced chordal graph $\mathcal{I}_{\Phi,\prec}$ that contains $\mathcal{H}$. We know by the definition of $\mathcal{I}_{\Phi,\prec}$ that the scope of every generated factor ψ, which are the nodes of $\mathcal{T}$, is a clique in $\mathcal{I}_{\Phi,\prec}$. Now, let $\mathbf{X} = \{X_1 \ldots X_n\}$ be a maximal clique in $\mathcal{I}_{\Phi,\prec}$, and X_1 the first variable from $\mathbf{X}$ in $\prec$. Then there are edges between X_1 and all $X_i \in \mathbf{X}$ before eliminating X_1, so there exist factors $\phi_{X_1}(X_1, X_i)$ for all $X_i \in \mathbf{X}$. To eliminate X_1, we must create a factor ψ that contains $X_1 \ldots X_n$ and no other variables outside the clique (or otherwise they would be in $\mathbf{X}$), So every maximal clique in $\mathcal{I}_{\Phi,\prec}$ is the scope of some factor ψ, and thus a node in $\mathcal{T}$.

(c) Consider the joint distribution between the variables in three cliques $\mathbf{Y} = \{C_i, C_j, C_k\}$ such that C_i neighbors C_j and C_j neighbors C_k:

$$P(\mathbf{Y}) = P(C_i \setminus \{S_{ij}\}, C_j \setminus \{S_{ij}, S_{jk}\}, C_k \setminus \{S_{jk}\}, S_{ij}, S_{jk})$$

Where the vairables not in the sepsets are conditionally independent in two neighboring cliques, so

$$\begin{aligned} P(\mathbf{Y}) &= P(C_i \setminus \{S_{ij}\} | S_{ij}) P(C_j \setminus \{S_{ij}, S_{jk}\} | S_{jk}, S_{ij}) P(S_{ij}) P(C_k \setminus \{S_{jk}\} | S_{jk}) P(S_{jk}) \\ &= \frac{P(C_i)}{P(S_{ij})} \frac{P(C_j)}{P(S_{ij}) P(S_{jk})} \frac{P(C_k)}{P(S_j k)} P(S_{ij}) P(S_{jk}) \\ &= \frac{P(C_i) P(C_j) P(C_k)}{P(S_{ij}) P(S_{jk})} \end{aligned}$$

So we can extend this process over the clique tree to compute the joint distribution as:

$$P(\mathbf{X}) = \frac{\prod_i P(C_i)}{\prod_{ij} P(S_{ij})}$$

(d) Either $\mathcal{T}' = \mathcal{T}$ or there are $C_i, C_j \in \mathcal{T}$ such that $C_i \subseteq C_j$. If we eliminate C_i, $\mathcal{T}'$ is family preserving since $\text{Scope}[\phi_i] \subseteq C_i \subseteq C_j$. We connect all neighbors of C_i to C_j when removing C_i, so any path through C_i is preserved via C_j since there can be no $X \in C_i$ not in C_j. Thus, if $\mathcal{T}$ satisfies the running intersection property, so does $\mathcal{T}'$

3.

(a) We can follow the algorithim for out of clique inference in a clique tree to solve for $P(X_i, X_j)$, given a calibrated clique tree $\mathcal{T}$. First we define $\mathcal{T}'$ to be the subtree of $\mathcal{T}$ such that $(X_i, X_j) \subseteq Scope[\mathcal{T}']$, then define a new set of factors (letting C_j be the root):

$$\Phi = \{\beta_j\} \cup \{\phi_k = \frac{\beta_k}{\mu_{k,k+1}} \mid k \in \mathcal{V}_{\mathcal{T}'} - \{j\}\}$$

Which can be done in linear time. Then we perform sum-product variable elimination on a ordering $\prec = X_{i+1} \dots X_{j-1}$ of $\mathbf{Z} = Scope[\mathcal{T}'] - \{X_i, X_j\}$. The running time of the variable elimination is $O(nk)$ since the largest induced width is one.

(b) Naively, we must run the $O(nk)$ process for all $\binom{n}{2}$ combinations of i and j, so the running time would be $O(n^3k)$

(c) Following the procedure in part (a), perform variable elimination on the ordering $\prec = X_2, X_3$ and set of factors $\Phi = \{\beta_3(X_3, X_4), \frac{\beta_1(X_1,X_2)}{\mu_{1,2}(X_2)}, \frac{\beta_2(X_2,X_3)}{\mu_{2,3}(X_3)}\}$. So

$$\begin{aligned} P(X_1, X_4) &= \sum_{X_3} \frac{\beta_3(X_3, X_4)}{\mu_{2,3}(X_3)} \sum_{X_2} \frac{\beta_1(X_1, X_2)\beta_2(X_2, X_3)}{\mu_{1,2}(X_2)} \\ &= \sum_{X_3} \frac{\beta_3(X_3, X_4)}{\mu_{2,3}(X_3)} P(X_1, X_3) \end{aligned}$$

Thus, if we cache $P(X_1, X_3)$, we can compute $P(X_1, X_4)$ more efficiently since we avoid re-computing $P(X_1, X_3)$

(d) From part (c), we can observe the simple recurrence relation:

$$P(X_i, X_j) = \sum_{X_j-1} \frac{\beta_{j-1}(X_{j-1}, X_j)}{\mu_{j-2,j-1}(X_{j-1})} P(X_i, X_{j-1})$$

A dynamic programming algorithm would be, for each $i \in [n-2]$, cache the result of $P(X_i, X_{i+2})$, and then use this to iterativley compute $P(X_i, X_{i+3}), P(X_i, X_{i+3}), \dots, P(X_i, X_n)$ using the above recurrence relation (and caching the last computed result). This process would take $O(nk)$ time for each i, so the total running time of the algorithim is $O(n^2k)$.

4.

(a) x

(b)

$$\begin{aligned} m_{j\to i} &= \sum_{x_j} \psi_j(x_i, x_j) \prod_{k \in \text{Nb}(j)\backslash\{i\}} m_{k\to j} \\ &= \sum_{x_j} \phi(x_j)\phi(x_i, x_j) \prod_{k \in \text{Nb}(j)\backslash\{i\}} \sum_{x_k} \exp(-\frac{1}{2}(\Lambda_{kk}x_k^2 + 2\eta_k x_k) - \Lambda_{kj}x_k x_j) \prod_{l \in \text{Nb}(k)\backslash\{j\}} m_{l\to k}(x_k) \\ &= \sum_{x_j} \exp(-\frac{1}{2}(\Lambda_{jj}x_j^2 + 2\eta_j x_j) - \Lambda_{ji}x_j x_i) \prod_{k \in \text{Nb}(j)\backslash\{i\}} \sum_{x_k} \mathcal{N}^{-1}([x_k, x_j]; \eta_k, \Lambda_k)\ \mathcal{N}^{-1}(x_k; \bar{\eta}, \bar{\Lambda}) \\ &= \sum_{x_j} \mathcal{N}^{-1}([x_j, x_i]; \eta_j, \Lambda_j)\ \mathcal{N}^{-1}(x_j; \bar{\eta}_k, \bar{\Lambda}_k) \\ &= \mathcal{N}^{-1}(x_i; \bar{\eta}_j, \bar{\Lambda}_j) \end{aligned}$$

(c) The belief at node i is the product of the incoming messages:

$$\beta(x_i) = \prod_{j \in \text{Nb}_i} m_{j\to i}$$

Which we can see from part (b) is a product of Gaussians with node i's neighbors marginalized out. So,

$$\beta(x_i) = \mathcal{N}^{-1}(x_i; \bar{\eta}, \bar{\Lambda})$$

5.1

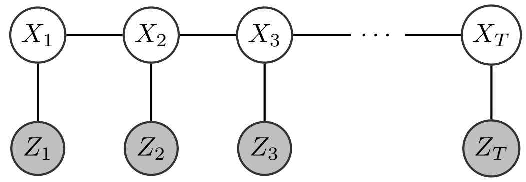

Figure 3: Undirected Hidden Markov model

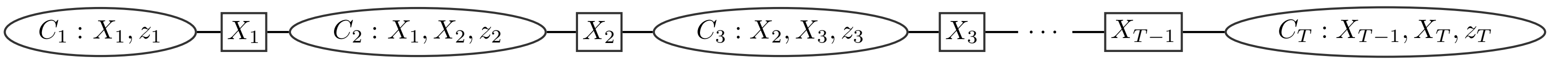

Figure 4: Clique Tree for the Undirected HMM

(a) $\Phi = \{\phi(X_1) = P(X_1),\ \phi(X_t, Z_t) = P(Z_t|X_t),\ \phi(X_{t-1}, X_t) = P(X_t|X_{t-1})\}$

(b)

(c) We can easily compute the first two messages:

$$m_{1\to 2}(X_1) = \phi(X_1, z_1) = P(X_1, z_1),$$
$$m_{2\to 3}(X_2) = \sum_{X_1} \phi(X_1, X_2)\phi(X_2, z_2)m_{1\to 2}(X_1) = \sum_{X_1} P(X_2|X_1)P(z_2|X_2)P(X_1, z_1)$$

So a general expression for $t \to t+1$ is:

$$\begin{aligned} m_{t\to t+1}(X_t) &= \sum_{X_{t-1}} \phi(X_{t-1}, X_t)\phi(X_t, z_t)m_{t-1\to t}(X_{t-1}) \\ &= \sum_{X_{t-1}} P(X_t|X_{t-1})P(z_t|X_t)m_{t-1\to t}(X_{t-1}) \end{aligned}$$

Where $m_{t-1\to t}(X_{t-1}) = P(X_{t-1}, z_1, z_2, \ldots, z_{t-1})$. Thus, $m_{t\to t+1}(X_t) = P(X_t, z_1, z_2, \ldots, z_t)$

(d) Starting with the first message:

$$\begin{aligned} m_{T\to T-1}(X_{T-1}) &= \sum_{X_T} \phi(X_{T-1}, X_T)\phi(X_T, z_T) \\ &= \sum_{X_T} P(X_T|X_{T-1})P(z_T|X_T) \end{aligned}$$

So a general expression for $t+1 \to t$ is:

$$\begin{aligned} m_{t+1\to t}(X_t) &= \sum_{X_{t+1}} \phi(X_t, X_{t+1})\phi(X_{t+1}, z_t)m_{t+2\to t+1}(X_{t+1}) \\ &= \sum_{X_{t+1}} P(X_{t+1}|X_t)P(z_{t+1}|X_{t+1})m_{t+2\to t+1}(X_{t+1}) \\ &= \sum_{X_{t+1}} P(X_{t+1}|X_t)P(z_{t+1}|X_{t+1})P(z_{t+2}, z_{t+3}, \ldots, z_T|X_{t+1}) \\ &= P(z_{t+1}, z_{t+2}, \ldots, z_T|X_t) \end{aligned}$$

(e)

$$\begin{aligned} P(X_t|z_1, z_2, \ldots, z_T) &= \frac{P(X_t, z_1, z_2, \ldots, z_T)}{P(z_1, z_2, \ldots, z_T)} \\ &= \frac{P(z_{t+1}, z_{t+2}, \ldots, z_T|X_t)P(X_t, z_1, z_2, \ldots, z_t)}{\sum_X P(\mathbf{X}, \mathbf{z})} \\ &= \frac{m_{t+1\to t}(X_t)m_{t\to t+1}(X_t)}{\sum_X \phi(X_1, z_1)\prod_{t=2}^{T} \phi(X_{t-1}, X_t)\phi(X_t, z_t)} \end{aligned}$$

5.2

(a)

$$\begin{aligned}
\alpha_t(X_t) = P(Z_1, \ldots, Z_t, X_t) &= \sum_{X_{t-1}} P(Z_1, \ldots, Z_t, X_{t-1}, X_t) \\
&= \sum_{X_{t-1}} P(Z_t|Z_1, \ldots, Z_{t-1}, X_t, X_{t-1}) P(X_t|Z_1, \ldots, Z_{t-1}, X_{t-1}) P(Z_1, \ldots, Z_{t-1}, X_{t-1}) \\
&= \sum_{X_{t-1}} P(Z_t|X_t) P(X_t|X_{t-1}) \alpha_{t-1}(X_{t-1})
\end{aligned}$$

Where $\alpha_1(X_1) = P(Z_1|X_1)P(X_1)$

(b)

$$\begin{aligned}
\beta_t(X_t) = P(Z_{t+1}, \ldots, Z_T|X_t) &\sum_{X_{t+1}} P(Z_{t+1}, \ldots, Z_T, X_{t+1}|X_t) \\
&= \sum_{X_{t+1}} P(Z_{t+2} \ldots Z_T|X_{t+1}, X_t, Z_{t+1}) P(Z_{t+1}|X_{t+1}, X_t) P(X_{t+1}|X_t) \\
&= \sum_{X_{t+1}} P(Z_{t+2} \ldots Z_T|X_{t+1}) P(Z_{t+1}|X_{t+1}) P(X_{t+1}|X_t) \\
&= \sum_{X_{t+1}} \beta_{t+1}(X_{t+1}) P(Z_{t+1}|X_{t+1}) P(X_{t+1}|X_t)
\end{aligned}$$

(c) Similar 5.1 (e) we have that $P(X_t|Z_1, \ldots, Z_T) \propto P(X_t, Z_1, \ldots, Z_T) = P(Z_{t+1}, \ldots, Z_T|X_t)P(X_t, Z_1, \ldots, Z_t) = \beta_t(X_t)\alpha_t(X_t)$

(d) $\beta_t(X_t) = P(Z_{t+1}, \ldots, Z_T|X_t) = m_{t+1\to t}(X_t)$ and $\alpha_t(X_t) = P(X_t, Z_1, \ldots, Z_t) = m_{t\to t+1}(X_t)$

6

(a) I did not collaborate

(b)