67800 | Probabilistic Methods in Artificial Intelligence | Ex 3

Guy Lutsker 207029448

Q1 - CT-BP Representation

1. We need to prove that $\mu_{ij}(S_{i,j}) = p_{\Phi}(S_{ij})$. So lets us look at $\mu_{ij}(S_{i,j})$ by definition:

$$\mu_{ij}(S_{i,j}) = m_{i\to j}(S_{i,j}) \cdot m_{j\to i}(S_{i,j})$$

We have shown in the lecture that $m_{i\to j}(S_{i,j}) = \sum_{V_{<(i,j)}} \prod_{\phi\in F_{<(i,j)}} \phi$
Where $V_{<(i,j)}$ are all the variable that are in i's side on C_i and not j's .
And so, we can write this as:

$$= \sum_{V_{<(i,j)}} \prod_{\phi\in F_{<(i,j)}} \phi \cdot \sum_{V_{<(j,i)}} \prod_{\phi\in F_{<(j,i)}} \phi$$

Notice that $V_{<(i,j)} \cup V_{<(j,i)} = \mathcal{X}/S_{i,j}$, and so we can combine the arguments to:

$$= \sum_{\mathcal{X}/S_{i,j}} \prod_{\phi\in F_{<(i,j)}} \phi \cdot \prod_{\phi\in F_{<(j,i)}} \phi$$

In addition we can see that $F_{<(i,j)}$and $F_{<(j,i)}$ are a direct sum of all Φ:

$$= \sum_{\mathcal{X}/S_{i,j}} \prod \phi = P_{\Phi}(S_{ij})$$

2.We need to prove that$p_{\Phi}(\mathcal{X}) = \frac{\prod_i \beta_i(C_i)}{\prod_{i,j} \mu_{ij}(S_{ij})}$
Lets start by opening the expression by definition of β_i :

$$\beta_i(C_i) = \Psi_i(C_i) \cdot \prod_{k\in ne(i)} (m_{k\to i})$$

In addition we have already proved in Q1 that $\mu_{ij}(S_{i,j}) = m_{i\to j}(S_{i,j}) \cdot m_{j\to i}(S_{j,i})$ so we can replace that too and get:

$$= \frac{\prod_i \Psi_i(C_i) \cdot \prod_{k\in ne(i)} (m_{k\to i})}{\prod_{i,j} m_{i\to j}(S_{i,j}) \cdot m_{j\to i}(S_{i,j})}$$

Rearranging the enumerator gets us:

$$= \frac{\prod_i \Psi_i(C_i) \cdot \prod_{i,j} m_{i\to j} \cdot m_{j\to i}}{\prod_{i,j} m_{i\to j} \cdot m_{j\to i}} = \prod_i \Psi_i(C_i)$$

$$= \prod \phi = P_{\Phi}(\mathcal{X})$$

Q2- Clique-Tree Induced by Variable Elimination

We need to prove that the resulting graph as described in the question is a valid CT.
According to the lecture a CT has 4 characteristics:

1. Its a tree
2. Each in holds several variables
3. Family Preserving
4. RIP

Let us do it step by step. We know that the clique tree must be, as its name implies, a tree. And so lets see:
We know that the VE process holds that τ_i is used not more than one time for the calculation of ψ_j. Now let us assume by way of contradiction that we had a circle in our CT that would mean that we would have had some τ_l used twice which is a contradiction - and so we get that our resulting CT is an acyclic connected graph - a tree.
Each intersection holds several variables is met directly from the way the VE algorithm flows - $C_i = scope[\psi_i]$
Family preserving is met since for each z_i we define $C_i = scope[\psi_i]$, and so
$\forall \phi_i \in \{\phi | \exists j \; z_i \in scope[\phi_i]\} \; \exists C_i \; s.t \; scope[\phi_i] \subset C_i$.
And we get that family preserving is met.
RIP is also met. Let z be a variable and us have two cliques wlog C_1, C_2 such that z is in their intersection. In addition let us define C_{sum} (being later on in the graph than C_1, C_2 - $1, 2 < sum$) which corespondents with τ_{sum}- the one we sum in the VE algorithm. Notice that since we are working with a tree, there is a unique path from $C_1 \rightarrow C_{sum}, C_2 \rightarrow C_{sum}$ and so calculation of the next node wlog C_1 - ψ_k will be τ_1, and since $z \in scope[C_1]$ that implies that $z \in \tau_1$. Of course this logic translates to each node along the path from $C_1 \rightarrow C_{sum}$ inductively. According to the algorithm we sum over z in C_{sum} and so each node on the path has z in its scope. A direct results of that is that z is iin each sepset in that path. We have already stated that we calculates this for $C_1 - C_{sum}$ wlog but what about $C_1 - C_2$? Again, since we are talking about a tree stricture we get that the path between $C_1 - C_2$ will go through a common node in the path to C_{sum} and so z is in every sepset along the path.

Q3 - Exact Inference on a Chain

1.We would like to do VE over $X_{i-1}, ...X_{j+1}$. (assuming here $i < j$)
Such that:
we would iterate over the suggested interval $t \in [i-1, j+1]$
and for each t set ϕ_t as all factors that have x_t in their scope.
then sum out x_t such that $\tau_t = \sum_{val(x_t)} \prod_{\phi \in \phi_t} \phi$
set $\Phi = \phi_{\neg x_t} \cup \tau_t$
and after all iteration set our result as $\tau_{j+a}(X_i, X_j)$
The time complexity is $O(\overbrace{n}^{iterations} \cdot \overbrace{L^3}^{Summing\ L\ times\ L \times L\ matrix})$

2.
Let us compose such an algorithm:
let us calculate $\tau_{forward}[i] = \sum_{val(x_i)} \tau_{forward}[i-1] \cdot \phi(x_{i-1}, x_i)$ for each $i \in [1, n-1]$.
let us calculate $\tau_{backwards}[i] = \sum_{val(x_{i+1})} \tau_{backwards}[i+1] \cdot \phi(x_i, x_{i+1})$ for each $i \in [n, 1]$(in backwards order).
and set our result as the pairwise multiplication of $\tau_{forward} \cdot \tau_{backwards}$.
Basically we do VE in both directions and then multiply the results.
The time complexity is $O(n)$ and in L reference it is $O(L^3)$.

Q4 - Sampling on a Tree

1. Can we compute p (x | e) exactly, in a tractable way?

We cannot. To do so we need to sum over all values of x to compute $p(e)$ which is necessary for the calculation of $p(x|e)$ as per the definition of CPD.

Can we sample directly from the distribution p (X | e)?

Again, we cannot, for the same reasons as mentioned above, we need to compute $p(e)$.

Can we compute p˜(x | e) = p (x, e) exactly, in a tractable way?

Finally, yes! If we were to sample using a topological order on the graph, we would do so in a linear time complicity ($O(n)$ - n=#variables). We would then multiple them all together to get the joint distribution.

2.

a.Firstly we will build ourselves a CT from our network, like we learned in class but - with each variable x_i, and its parents assigned to a clique C_i. In addition its factor is : $\phi(X_i, pa(i)) = q(X_i pa(i))$. And the edges are are between cliques that share a variable. We can also see that by design this CT has the same backbone as τ. This also upholds the family preserving quality since each factor is in one clique, and it upholds RIP since the edges are providing us with an acyclic graph. Let us now eliminate all $\mathcal{E}_i \in E$ from our cliques. If

$$x_i, pa(i) \in E$$

and

$$\phi(pa(i)) = \phi(x_i = e(x_i), pa(i))$$

and

$$\phi(x_i = e(x_i), pa(i) = e(pa(i))$$

This is still a valid CT.

Lets calculate $q(x_1|e) \propto q(c_1) \propto b(c_1)$. and $q(e) = \sum_{val(x_1)} b(c_1)$ and sample x_1 of the values from $q(x_1|e) = b(c_1) \cdot q/q(e)$.

Let is define t as a part of e , and now we shall sample from $q(x_i|pa(i) = t) = \phi(x_i, pa(i) = t)$.

Since we know our CT has the same backbone as τ we can now eliminate cliques in E. And so we sampled from the correct distribution because we sample at the i'th time from $q(x_i|X_{i-1} = x_{i-1})$.

b.We can reweigh our samples by using normed IS:

$$w[m] = \frac{p(m, e)}{q(m, e)}$$

c. Setting j as the number of samples we get the the estimator is:

$$\hat{p}(x, e) = \mathbb{E}_q[X] = \frac{\sum_{i=1}^{j} w[m] \cdot \mathbf{1}_{\boldsymbol{m=x}}}{\sum_{i=1}^{j} w[m]}$$