COS 513: FOUNDATIONS OF PROBABILISTIC MODELING
LECTURE 3

DAVID SHUE, JOHN VALENTINO

1. REVIEW OF CONDITIONAL INDEPENDENCE

Recall the graphical model that we have used in previous lectures. In this section we will determine whether or not two conditional independence relationships hold in the graph by using the "Bayes Ball" Algorithm.

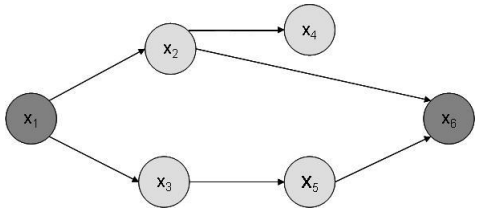

FIGURE 1. The Graphical Model in Question

Question 1: Is $X_2 \perp\!\!\!\perp X_3 \mid \{X_1, X_6\}$?
We can use the following process to test for conditional independence:

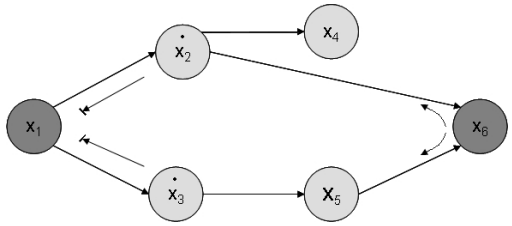

FIGURE 2. Testing $X_2 \perp\!\!\!\perp X_3 \mid \{X_1, X_6\}$

(1) Shade X_1, X_6
(2) Start the balls at X_2 and X_3
(3) Can a ball starting at X_2 reach X_3 through X_1? No ("Shoe-Size")
(4) Can a ball starting at X_2 reach X_5 through X_6? Yes ("Aliens")
(5) Can a ball starting at X_6 reach X_3 through X_5? Yes ("Markov")

Conclusion: Because the "ball" can "bounce" from X_2 to X_3, X_2 is not conditionally independent of X_3 given X_1 and X_6

Question 2: Is $X_2 \perp\!\!\!\perp X_3 \mid \{X_1\}$?

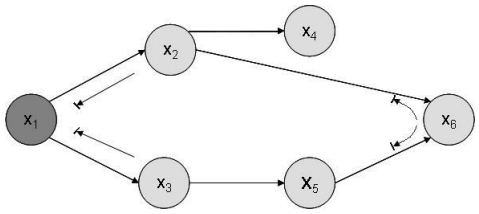

FIGURE 3. Testing $X_2 \perp\!\!\!\perp X_3 \mid \{X_1\}$

The same process as above is followed except that now X_6 is "blocked," which leaves no paths through the graph and implies conditional independence.

2. UNDIRECTED GRAPHICAL MODELS

The joint distribution of an undirected graphical model is defined in terms of ***potential functions*** over ***cliques*** of the graph.

$$p(x) = \frac{1}{Z} \prod_{c \in C} \Psi_c(X_c)$$

Clique: A fully connected sub-node of the graph.
Potential Functions: Arbitrary, positive, and real-valued.
Z: The normalizing / scaling constant.

The above product doesn't have to include every clique in the graph, but the included cliques should cover the entire set of variables. Vertices can occur multiple times, but no edge should be ignored!

In a DAG we are assured the values will sum to 1, which gives a valid joint probability distribution. In the undirected case, we use Z as a scaling constant to "fix" the potential by ensuring that $p(x)$ will sum to 1.

(1) All conditional independencies can be found with graph separation.

(2) Some joints may be represented with Undirected Graphical models but not with directed graphical models. (UDGM is a superset of DGM.)

Advantages of **UDGMS:** Potentially more expressive than DGMS.
DGMS: Z is always one.
May represent an intuitive causal structure.

3. PROBABILISTIC INFERENCE

The problem of computing conditional and marginal probabilities from a joint distribution of random variables.

$$\textbf{Goal:} \mathrm{p}(x_f|x_E) = \frac{\mathrm{p}(x_f, x_E)}{\mathrm{p}(x_E)}$$

Let f (query node) be a node index, e.g. $\{1,2,3,4,5,6\}$, E be a set of evidence nodes and R be the remaining nodes $\notin E$ and $\neq f$

(1) Compute marginal

$$\mathrm{p}(x_f, x_E) = \sum_{x_R} \mathrm{p}(\underbrace{x_f, x_E, x_R}_{\text{all r.v's in the model}})$$

(2) Compute another marginal

$$\mathrm{p}(x_E) = \sum_{x_f} \mathrm{p}(x_f, x_E)$$

(3) Take the ratio

$$\mathrm{p}(x_f|x_E) = \frac{\sum_{x_R} \mathrm{p}(x_f, x_E, x_R)}{\mathrm{p}(x_E)} \sum_{x_f} \mathrm{p}(x_f, x_E)$$

The complexity problem: step 1 is potentially exponential in the number of random variables: $O(k^R)$. If R is large then the naive summation will be extremely expensive to compute. The goal of elimination is to take advantage of local structure to reduce computational complexity.

3.1. **Example inference:** compute $\mathrm{p}(x_1|x_6)$

Let $R = \{2, 3, 4, 5\}$, $f = 1$, and $E = \{6\}$

The book defines: $\overline{x}_6$ = a clamped value of x_6, which conditions the evidence nodes by the observed values. Using the δ function we can construct an equivalent marginalization sum for conditioned variables which allows for reordering of conditioned factors in the joint distribution.

$$g(\overline{x}_6) = \sum_{x_6} g(x_6)\delta(x_6, \overline{x}_6)$$

Note that δ = 1 only when x_6 is equal to $\overline{x}_6$, otherwise it is equal to 0

$$
\begin{aligned}
\mathrm{p}(x_1, \overline{x}_6) &= \underbrace{\sum_{x_2}\sum_{x_3}\sum_{x_4}\sum_{x_5}\sum_{x_6}}_{\text{all of } R \text{ and } x_6} \mathrm{p}(x_1)\mathrm{p}(x_2|x_1)\mathrm{p}(x_3|x_1)\mathrm{p}(x_4|x_2)\mathrm{p}(x_5|x_3)\mathrm{p}(x_6|x_2,x_5)\delta(x_6,\overline{x}_6) \\
&\quad \text{Note that the naive summation is O}(k^6) \text{ or more generally O}(k^R) \\
&= \mathrm{p}(x_1)\sum_{x_2}\mathrm{p}(x_2|x_1)\sum_{x_3}\mathrm{p}(x_3|x_1)\sum_{x_4}\mathrm{p}(x_4|x_2)\sum_{x_5}\mathrm{p}(x_5|x_3)\underbrace{\sum_{x_6}\mathrm{p}(x_6|x_2,x_5)\delta(x_6,\overline{x}_6)}_{\text{Define as } \mathrm{m}_6(x_2,x_5)} \\
&= \mathrm{p}(x_1)\sum_{x_2}\mathrm{p}(x_2|x_1)\sum_{x_3}\mathrm{p}(x_3|x_1)\sum_{x_4}\mathrm{p}(x_4|x_2)\underbrace{\sum_{x_5}\mathrm{p}(x_5|x_3)\mathrm{m}_6(x_2,x_5)}_{\mathrm{m}_5(x_2,x_3)} \\
&= \mathrm{p}(x_1)\sum_{x_2}\mathrm{p}(x_2|x_1)\sum_{x_3}\mathrm{p}(x_3|x_1)\sum_{x_4}\mathrm{p}(x_4|x_2)\mathrm{m}_5(x_2,x_3) \\
&\quad \text{Note that } \mathrm{m}_5 \text{ does not depend on } x_4 \text{ so we move it out} \\
&= \mathrm{p}(x_1)\sum_{x_2}\mathrm{p}(x_2|x_1)\sum_{x_3}\mathrm{p}(x_3|x_1)\mathrm{m}_5(x_2,x_3)\underbrace{\sum_{x_4}\mathrm{p}(x_4|x_2)}_{\mathrm{m}_4(x_2)} \\
&\quad \text{Technically, any non-query/evidence ancestor terms will sum out to 1} \\
&= \mathrm{p}(x_1)\sum_{x_2}\mathrm{p}(x_2|x_1)\mathrm{m}_4(x_2)\underbrace{\sum_{x_3}\mathrm{p}(x_3|x_1)\mathrm{m}_5(x_2,x_3)}_{\mathrm{m}_3(x_1,x_2)} \\
&= \mathrm{p}(x_1)\underbrace{\sum_{x_2}\mathrm{p}(x_2|x_1)\mathrm{m}_4(x_2)\mathrm{m}_3(x_1,x_2)}_{\mathrm{m}_2(x_1)} \\
&= \mathrm{p}(x_1)\mathrm{m}_2(x_1)
\end{aligned}
$$

Finally, we compute the marginals of interest and compute the conditional probability as the ratio

$$
\begin{aligned}
\mathrm{p}(x_1, \overline{x}_6) &= \mathrm{p}(x_1)\mathrm{m}_2(x_1) \\
\mathrm{p}(\overline{x}_6) &= \sum_{x_1}\mathrm{p}(x_1)\mathrm{m}_2(x_1) \\
\mathrm{p}(x_1|\overline{x}_6) &= \frac{\mathrm{p}(x_1, \overline{x}_6)}{\mathrm{p}(\overline{x}_6)}
\end{aligned}
$$

By shifting the summation terms and computing intermediate functions, the complexity of the computation drops from O(k^6) to O(k^3)

4. ELIMINATION ALGORITHM

Elimination is a simple approach to probabilistic inference on graphical models. It is limited in scope since it only computes a single marginal probability for a designated query node, hence it is rarely used in practice. However, it lends considerable insight into the general process underlying inferential computation.

The idea at each step is to sum over a product of functions:

(1) conditional probabilities p($x_i|x_{\pi_i}$)
(2) delta functions (for conditioned evidence variables): $\delta(x_i, \overline{x}_i)$
(3) intermediate function m$_i(x_{S_i})$ generated by the previous steps

Given a graph $G = \{V, E\}$, evidence E, and query node f:

INITIALIZE

Choose a node ordering I such that f is last
Place p($x_i|x_{\pi_i}$) on an active list of functions
Place $\delta(x_i, \overline{x}_i)$ on the active list of functions for evidence nodes

ELIMINATE

for $i \in I$
do
remove all functions from the active list containing x_i
construct m$_i(x_{S_i}) = \sum_{x_i} \prod$ selected functions of x_i
where $S_i =$ union of all arguments to the functions of node x_i
add m$_i(x_{S_i})$ to the active list

Note that no $j < i$ in I can appear in S_i since they have already been summed out

At the end, NORMALIZE:

p($x_f, \overline{x}_E$) = $\phi(x_f)$
$\phi(x_f) =$ product of functions left on the active list, which are all functions ofx_f
p($x_f|\overline{x}_E$) = $\frac{\phi(x_f)}{\sum_{x_f} \phi(x_f)}$(normalization)

Using the elimination order: 6,5,4,3,2,1 the algorithm effectively grouped the factorized CPT's by their dependencies and shifted them to the proper summations, in essence performing the same algebraic manipulation of the marginalized sums as previously derived.

5. What is the Complexity of Eliminate?

The complexity of eliminate is controlled by the number of arguments in the intermediate functions $\underline{m}$, which, depend on the chosen elimination ordering: O($K^{\max(\text{nargs})+1}$))

(1) Draw the GM as an undirected graph. (Not a UDGM conversion)
(2) Moralize the graph by connecting the parents of each child node.
(3) Remove the nodes in I order, connecting the nodes not previously attached in order to construct the "reconstituted graph"

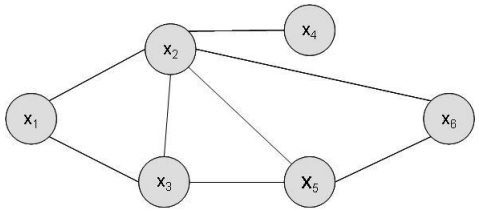

FIGURE 4. A Reconstituted Version of the Graphical Model shown in Figure 1

The complexity of eliminate (with ordering I) is exponential in the largest clique of the reconstituted graph. In our graph (see above), the largest clique is of size 3.

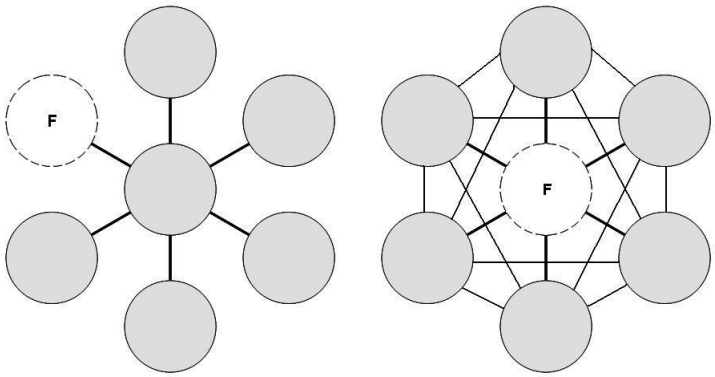

FIGURE 5. Star Example: 'F' represents the first node removed in the elimination. If 'F' is a leaf node then the clique sizes will be much smaller than if 'F' is the center node.

5.1. **Does Ordering Matter?** If we remove the leafs first, the largest clique size is 2, which leads to a complexity of O(K^2). If we remove the center first, the largest clique size is the number of leaves, or O(K^6).

Finding the optimal ordering is an NP-Hard problem.