

LECTURE 20:

HUGIN INFERENCE ALGORITHM

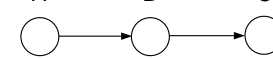
March 27, 2006

- The goal of the junction tree algorithm is to adjust the clique potentials  $\psi_c$  given some evidence  $\bar{x}_E$  so that they are unnormalized marginals:

$$\psi_C(\mathbf{x}_C) = p(\mathbf{x}_C, \bar{x}_E)$$

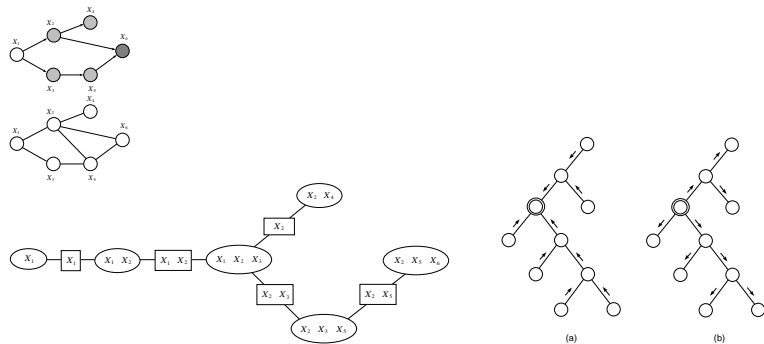
- Normally clique potentials *do not* correspond to marginals; some of them must be conditionals.

- Example:  $\psi_{AB} = p(\mathbf{x}_A, \mathbf{x}_B)$  ;  $\psi_{BC} = p(\mathbf{x}_C | \mathbf{x}_B)$



- We can always adjust the potentials to make them marginals (e.g. above we could multiply by  $p(\mathbf{x}_B)$ ) but then the product of potentials will not be proportional to the joint probability.
- What can we do? Extend the representation!

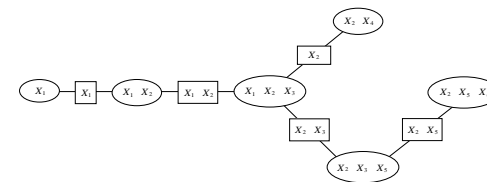
- Three main steps:
  1. Pre-processing (compiling) the graphical model to prepare for inference: building the clique junction tree.
  2. Conditioning on the evidence.
  3. Marginalizing out the non-query nodes efficiently.



- On each edge of the clique tree, we place a potential  $\phi_S$  over the variables in the intersection of the two adjacent cliques it joins.
- These intersections are called *separator sets* and are themselves cliques (fully connected in the underlying graph) although of course they are no longer maximal.
- Now our representation of the joint probability is defined as:

$$p(\mathbf{X}) = \frac{\prod_C \psi_C(\mathbf{x}_C)}{\prod_S \phi_S(\mathbf{x}_S)}$$

where the normalizer is absorbed into a special separator  $\phi_\emptyset$ .



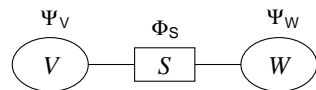
- Now we can have clique potentials proportional to marginals *and* a representation of the joint distribution at the same time.
  - e.g.  $p(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C) = \frac{p(\mathbf{x}_A, \mathbf{x}_B)p(\mathbf{x}_B, \mathbf{x}_C)}{p(\mathbf{x}_B)}$
- ```

    graph LR
      A((A)) --> B((B))
      B --> C((C))
    
```
- Furthermore, this extended representation still obeys the Hammersly-Clifford theorem, i.e. it still represents exactly the same family of distributions with the correct conditional independencies.
  - Initialization? Set all separator potentials to be unity.
  - What about division by zero? It will turn out that a separator is only zero if both clique potentials it is connected to are also zero. In this case we define the ratio to be zero.

- Since cliques overlap, some variables appear in more than one clique. If we sum out non-intersection variables, any pair of cliques must give same marginals for nodes they have in common.
- Let us focus on *local consistency*: how to make two adjacent clique agree on marginals over separator variables.
- Consider the following updates:

$$\phi_S^* = \sum_{V \setminus S} \psi_V \quad \psi_W^* = \frac{\phi_S^*}{\phi_S} \psi_W \quad \psi_V^* = \psi_V$$

$$\phi_S^{**} = \sum_{W \setminus S} \psi_W^* \quad \psi_W^{**} = \psi_W^* \quad \psi_V^{**} = \frac{\phi_S^{**}}{\phi_S^*} \psi_V^*$$



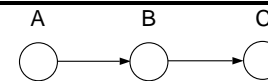
- After performing the updates on the previous slide, we are guaranteed that  $\psi_V^{**}$  and  $\psi_W^{**}$  are consistent with respect to  $S$ :

$$\sum_{V \setminus S} \psi_V^{**} = \sum_{V \setminus S} \frac{\phi_S^{**}}{\phi_S^*} \psi_V^* = \frac{\phi_S^{**}}{\phi_S^*} \sum_{V \setminus S} \psi_V^* = \frac{\phi_S^{**}}{\phi_S^*} \phi_S^* = \phi_S^{**} = \sum_{W \setminus S} \psi_W^{**}$$

- But the updates leave the joint distribution  $p(\mathbf{x}_W, \mathbf{x}_V)$  unchanged:

$$\frac{\psi_V^* \psi_W^*}{\phi_S^*} = \frac{\psi_V \psi_W \phi_S^*}{\phi_S \phi_S^*} = \frac{\psi_V \psi_W}{\phi_S}$$

$$\frac{\psi_V^{**} \psi_W^{**}}{\phi_S^{**}} = \frac{\psi_V^* \psi_W^* \phi_S^{**}}{\phi_S^* \phi_S^{**}} = \frac{\psi_V^* \psi_W^*}{\phi_S^*} = \frac{\psi_V \psi_W}{\phi_S}$$



With no evidence.

initially:  $\psi_{ab} = p(a, b) \quad \psi_{bc} = p(c|b) \quad \phi_b = 1$

updates:

$$\phi_b^* = \sum_a p(a, b) = p(b)$$

$$\psi_{bc}^* = \frac{p(b)}{1} p(c|b) = p(b, c)$$

With evidence  $a = 1$ .

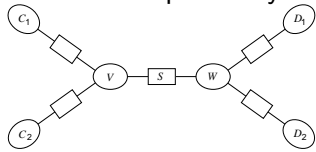
initially:  $\bar{\psi}_{ab} = p(\bar{a} = 1, b) \quad \psi_{bc} = p(c|b) \quad \phi_b = 1$

updates:

$$\phi_b^* = p(\bar{a} = 1, b)$$

$$\psi_{bc}^* = p(\bar{a} = 1, b) p(c|b) = p(\bar{a} = 1, b, c)$$

- What happens when we have a tree of cliques instead of just a pair? How can we achieve *global consistency* so that all cliques containing a variable  $x_i$  agree on its marginal  $p(x_i)$ ?
- Two things:
  1. Arrange the cliques into a *junction tree* so that *local consistency implies global consistency*.  
We don't need to consider all pairs of cliques that share variables.
  2. Order updates to ensure that updates between  $V$  and  $W$  do not ruin consistency between  $V$  and  $U$  previously achieved.



When can one node safely pass a message to another?

- The key property of the junction tree is that *local consistency implies global consistency*.
- In other words, consider a variable  $x_i$  which appears in two cliques. In a junction tree, it will also appear in every clique on the path between those two and nowhere else.
- If the cliques along that path are *pairwise consistent* with respect to  $x_i$  then they will also be *jointly consistent* with respect to  $x_i$ .
- Thus, running the pairwise message passing on a junction tree of cliques will achieve local *and* global consistency. We can get the same answer for  $x_i$  by consulting any clique node containing  $x_i$ .
- Furthermore, this answer will be exactly the correct marginal.

A clique can send a message to a neighbour only when it has received messages from all its other neighbours.

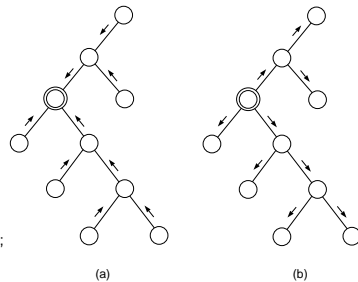
- This protocol maintains consistency.<sup>1</sup>
- Protocol is also realizable: designate one node of junction tree as root. Pass messages inward to root and then back out to leaves.

```

Formally:
COLLECTEVIDENCE(root);
DISTRIBUTEVIDENCE(root);

COLLECTEVIDENCE(node):
  foreach child(node) {
    UPDATE(node, COLLECTEVIDENCE(child));
  }
  return node;

DISTRIBUTEVIDENCE(node):
  foreach child(node) {
    UPDATE(child, node); DISTRIBUTEVIDENCE(child);
  }
    
```



<sup>1</sup>Consider a message  $W \rightarrow V$ . If  $V$  has already sent its message to  $W$ , then it must have received all its other messages. The current message  $W \rightarrow V$  will achieve consistency and no more messages will be exchanged. If  $V$  has not sent to  $W$  yet, it will wait until it has received all other messages and then send a final message to achieve consistency.

- **Compilation.**  
Moralization: For directed graphs, join parents and drop directions of links.  
Triangulation: Many possible algorithms. Hard step.  
Identification of maximal cliques: easy in triangulated graphs.  
Construction of Junction Tree: maximal spanning tree over cliques using intersection size as weights.
- **Introduction of evidence.**  
At every node where we have observed some data, take appropriate slice of potential.
- **Initialization.**  
Each potential of original graph (possibly sliced) is multiplied onto exactly one clique of junction tree.  
Separators are initialized to unity.
- **Propagation of probabilities.**  
Pass messages according to MPP: designate root of clique tree and call COLLECTEVIDENCE and DISTRIBUTEVIDENCE from root. For a message from  $V \rightarrow W$ :

$$\phi_S^* = \sum_{V \setminus S} \psi_V \quad \psi_W^* = \frac{\phi_S^*}{\phi_S} \psi_W$$

At termination, clique potentials and separator potentials are proportional to marginal probabilities of cliques/separator sets given evidence. Further marginalization can be performed for singletons or subsets.

- It is possible to develop an alternate (but equivalent) algorithm in which we don't explicitly store the separator potentials  $\phi$ .
- Instead, we work out what the multiplicative update to a clique potential  $\psi$  would have been (by expressing separator potentials in terms of clique potentials), and perform that update explicitly.
- The updates are expressed as "messages" from clique  $i$  to  $j$ :

$$\mu_{ij} = \sum_{C_i \setminus C_j} \psi_{C_i} \prod_{k \neq i} \mu_{ki}$$

- Once a clique has received messages from all its neighbours, we can compute its marginal as the product of messages and evidence:

$$p(C_i) \propto \psi_{C_i} \prod_{k \neq i} \mu_{ki}$$

Together, these two equations are the Shafer-Shenoy Algorithm.

- Two cases:
  - If the first update on the link  $vw$  was from  $v$  to  $w$ , then we want  $\mu_{vw} = \phi^*/\phi$ . If all the other messages coming into  $i$  are correct, then using  $\phi^* = \sum_{V \setminus S} \psi_V$  and the fact that the initial separator potential is unity, we can see that  $\mu_{vw}$  will be correct.
  - Otherwise, we want  $\mu_{vw} = \phi^{**}/\phi^*$ . Using the fact that  $\phi_S^{**} = \sum_{W \setminus S} \frac{\phi_S^*}{\phi_S} \psi_W$  we can see again that the message will be correct.

- Shafer-Shenoy looks a lot like belief propagation (sum-product).
- But we can relate it to the Hugin algorithm as follows:  
Consider cliques  $V$  and  $W$ .  
What is  $\mu_{vw}$  were the "update factor" in the direction from  $v \rightarrow w$ ?
- At the end of Shafer-Shenoy, the marginal would be the product of the original potential, multiplied by all the update factors – just like in the Hugin algorithm.
- So we just need to check that  $\mu_{vw}$  defined as in SS:

$$\mu_{vw} = \sum_{C_v \setminus C_w} \psi_{C_v} \prod_{k \neq v} \mu_{kv}$$

are really equal to the update factors from Hugin.

- Generalized Viterbi: replace sum with max in  $\psi, phi$  updates
- Computational Complexity
- Proofs:
  - MPP on junction tree gives correct marginals.
  - Many async. updates without MPP still achieve consistency.
  - Extended representation obeys Hammersley-Clifford.
  - Separator potentials are zero only when both cliques are.
  - Decomposable  $\Leftrightarrow$  Triangulated  $\Leftrightarrow \exists$  Junction tree.
  - Elimination performs triangulation.
  - Maximal spanning tree with  $w = |S|$  is junction tree.