University of Washington
Department of Electrical Engineering
EE512 Spring, 2006
Graphical Models

Jeff A. Bilmes <bilmes@ee.washington.edu>

Lecture 11 Slides
May 2^{nd}, 2006
Announcements

• READING:
  – M. Jordan: Chapters 4,10,12,17,18

• Reminder: TA discussions and office hours:
  – Office hours: Thursdays 3:30-4:30, Sieg Ground Floor Tutorial Center
  – Discussion Sections: Fridays 9:30-10:30, Sieg Ground Floor Tutorial Center Lecture Room

• Reminder: take-home Midterm: May 5th-12th, you must work alone on this.

• Note: I am gone all next week (May 8th-12th). Subsequent weeks we will have makeup classes on Mondays at 6:00pm.
Class Road Map

- L1: Tues, 3/28: Overview, GMs, Intro BNs.
- L2: Thur, 3/30: semantics of BNs + UGMs
- L3: Tues, 4/4: elimination, probs, chordal I
- L4: Thur, 4/6: chrdal, sep, decomp, elim
- L5: Tue, 4/11: chdl/elim, mcs, triang, ci props.
- L7: Tues, 4/18: Mobius, HC-thm, (F)=(G)
- L8: Thur, 4/20: phylogenetic trees, HMMs
- L9: Tue, 4/25: HMMs, inference on trees
- L10: Thur, 4/27: Inference on trees, start poly

- L11: Tues, 5/2: polytrees, start JT inference
- L12: Thur, 5/4
- L13: Tues, 5/9
- L14: Thur, 5/11
- L15: Tue, 5/16
- L16: Thur, 5/18
- L17: Tues, 5/23
- L18: Thur, 5/25
- L19: Tue, 5/30
- L20: Thur, 6/1: final presentations
Final Project Milestone Due Dates

- L1: Tues, 3/28:
- L2: Thur, 3/30:
- L3: Tues, 4/4:
- L4: Thur, 4/6:
- L5: Tue, 4/11:
- L6: Thur, 4/13:
- L7: Tues, 4/18:
- L8: Thur, 4/20: Team Lists, short abstracts I
- L9: Tue, 4/25:
- L10: Thur, 4/27: short abstracts II

- L11: Tues, 5/2: today
- L12: Thur, 5/4: abstract II + progress
- L13: Tues, 5/9
- L14: Thur, 5/11: 1 page progress report
- L15: Tue, 5/16
- L16: Thur, 5/18: 1 page progress report
- L17: Tues, 5/23
- L18: Thur, 5/25: 1 page progress report
- L19: Tue, 5/30
- L20: Thur, 6/1: final presentations
- L21: Tue, 6/6 4-page papers due (like a conference paper).

- Team lists, abstracts, and progress reports must be turned in, in class and using paper (dead tree versions only).
- Final reports must be turned in electronically in PDF (no other formats accepted).
- Progress reports must report who did what so far!!
Summary of Last Time

• Inference on trees
• Inference on undirected trees
• Example: voting tallying by message passing in trees
• Begin inference on poly trees
Outline of Today’s Lecture

- inference on poly trees
- Begin exact inference on junction trees
Books and Sources for Today

- M. Jordan: Chapters 4, 10, 12, 17, 18
Bottom up Inference in undirected trees

- Therefore, we get that:

\[
p(x|\bar{y}_E, \bar{z}_E) = \frac{p(\bar{z}_E|x)p(x|\bar{y}_E)}{\sum_x p(\bar{z}_E|x)p(x|\bar{y}_E)} = \frac{\lambda(x)\pi(x)}{\sum_x \lambda(x)\pi(x)}
\]

- Where we define:

  Bottom up  \[\lambda(x) \triangleq p(\bar{z}_E|x)\]

  Top down  \[\pi(x) \triangleq p(x|\bar{y}_E)\]

- Bottom up case: Suppose \( X \) has some number of children, say 2 for now, and in each child, it partitions \( Z \) into \( Z_1 \) and \( Z_2 \), and it partitions the evidence in \( Z \) into two pieces, \( \bar{z}_{E1} \) and \( \bar{z}_{E2} \).

\[
\lambda(x) = p(\bar{z}_E|x) = p(\bar{z}_{E1}, \bar{z}_{E2}|x) = p(\bar{z}_{E1}|x)p(\bar{z}_{E2}|x)
\]

and this follows since \( Z_1 \perp Z_2|X \).
Inference in undirected trees

• Note that the π’s now depend on the λ values.
• Implicitly defines a message order due to dependencies in the equations.
• For the π message, we cannot send a π message down to the child until we have received messages from all other neighbors in the graph other than the child.
• For the λ message, we cannot send a λ message up to the parent until we have received a message from all children.
• Key: For all $X \in V$, at any point once a node $X$ has received a message from all of its neighbors (in the moralized tree) then it is able to compute $p(x|\bar{y}_E, \bar{z}_E)$. Recall again, the current definition of $Y$ and $Z$ is dependent on the current selection of the variable named $X$. 
Inference in undirected trees

- Resulting implied order using dependencies in equations.
Inference in undirected trees

- This can be related to undirected trees. When we moralize, we lose no conditional independence properties since there are no $V$-structures.

Consider the undirected tree.

- We can consider $\phi_{X_i}(x_i)$ to be a form of “charge” on node $X_1$. Also, let $\psi_{X_i,X_j}(x_i,x_j)$ to be a potential on the edge $(i,j) \in E$ of $G$.

- Initialization: Consider perfect ordering of the nodes in the BN $\sigma = (\sigma_1, \ldots, \sigma_n)$. Then $\forall (i,j) \in E$ s.t. $j > i$, $\psi(x_i,x_j) = p(x_j|x_i)$.

- For all parentless $i$, for one of $i$’s children $j$, $\psi(i,j) = \psi(i,j) * p(x_i)$.

- For all $i$, $\phi(x_i) = 1$.

- Therefore, $p(x_{1:N}) = \prod_{e \in E} \psi(x_{e_1}, x_{e_2})$
Inference in undirected trees

- We consider a generic message passing procedure, that builds up a set of messages from neighboring nodes.

\[
\phi_{u \rightarrow v}(v) = \sum_{u_1} \psi_{u_1, v}(u_1, v) \phi_{u_1}(u_1)
\]

\[
\phi_{u_2 \rightarrow v}(v) = \sum_{u_2} \psi_{u_2, v}(u_2, v) \phi_{u_2}(u_2)
\]

\[
\phi_{v \rightarrow u}(u) = \sum_{v} \phi_{u, v}(u, v) \phi_{u_1 \rightarrow v}(v) \phi_{u_2 \rightarrow v}(v)
\]

- Q: what set of messages and in what order is (at this point as far as we know) sufficient to make sure this works as advertised?

- Note that in undirected tree case, we don’t really make a distinction in the messages between \( \lambda \) and \( \pi \) messages. Key thing is to ensure that a message goes out only when all other edge’s messages have come in. This implicitly defines an ordering, as in tree on page 23.
Inference in polytrees

- What if graph has V-structures? (Directed trees with V-structures are called polytrees), “poly” since a node has multiple possible parents.
- Again, goal is \( p(x|\text{evidence}) = p(x|\bar{y}_E, \bar{z}_E) \).

\[
\propto p(x, \bar{y}_E, \bar{z}_E) \\
\propto p(\bar{z}_E|x)p(x|\bar{y}_E) = \lambda(x)\pi(x)
\]

- Evidence decomposed into parent/child specific evidence.

\[
\bar{y}_E = \bar{y}_{E_1} \cup \bar{y}_{E_2} \quad \bar{z}_E = \bar{z}_{E_1} \cup \bar{z}_{E_2}
\]

- Bottom up is (initially) fairly straight-forward.

\[
\lambda(x) = p(\bar{z}_E|x) = p(\bar{z}_{E_1}, \bar{z}_{E_2}|x) = p(\bar{z}_{E_1}|x)p(\bar{z}_{E_2}|x)
\]
Inference in polytrees

- top down at $x$.

$$
\pi(x) = p(x|\bar{y}_E) = p(x|\bar{y}_{E_1}, \bar{y}_{E_2}) \\
= \sum_{y_1, y_2} p(x, y_1, y_2|\bar{y}_{E_1}, \bar{y}_{E_2}) = \sum_{y_1, y_2} p(x|y_1, y_2)p(y_1, y_2|\bar{y}_{E_1}, \bar{y}_{E_2}) \\
= \sum_{y_1, y_2} p(x|y_1, y_2)p(y_1|\bar{y}_{E_1})p(y_2|\bar{y}_{E_2}) \\
= \sum_{y_1, y_2} p(x|y_1, y_2)\pi(y_1)\pi(y_2)
$$

- Note: computation is now $O(r^3)$ as expected, for two parent case.

- In general, with $I$ children and $J$ parents, we have that

$$
p(x|\bar{y}_E, \bar{z}_E) \propto p(\bar{z}_{E_1}|x)p(\bar{z}_{E_2}|x) \sum_{y_1, y_2} p(x|y_1, y_2)\pi(y_1)\pi(y_2)
$$
Required Messages

- So, posterior at $X$ requires messages from all directions.

$$
\pi(x) = \sum_{y_1,y_2} p(x|y_1,y_2) \pi(y_1) \pi(y_2)
$$
Inference in polytrees

• Note: computation is now $O(r^3)$ as expected, for two parent case.
• But in general, with $I$ children and $J$ parents, we have that

$$p(x | \bar{y}_E, \bar{z}_E) \propto \left( \prod_{i=1}^{I} p(\bar{z}_{E_i} | x) \right) \left( \sum_{y_{1:J}} p(x | y_{1:J}) \prod_{j=1}^{J} \pi(x_j) \right)$$

• Here, we see that the computation is $O(r^{J+1})$, so computation is exponential in number of parents.
• What if we moralize and (do a good job to) triangulate graph, how big are the cliques?
Inference in polytrees

• How do we get $p(\bar{z}_{E_1} | x)$? Divide $\bar{z}_{E_1}$ into evidence via $Z_3$ (namely $\bar{z}_{E_3}$) and evidence via $Z_{4,5}$ (namely $\bar{z}_{E_{4,5}}$). Then, we have:

\[
p(\bar{z}_{E_1} | x) = p(\bar{z}_{E_3}, \bar{z}_{E_{4,5}} | x) = \sum_{z_1, z_3} p(\bar{z}_{E_3}, \bar{z}_{E_{4,5}}, z_1, z_3 | x)
\]

\[
= \sum_{z_1, z_3} p(\bar{z}_{E_3}, \bar{z}_{E_{4,5}} | z_1, z_3, x)p(z_1, z_3 | x)
\]

\[
= \sum_{z_1, z_3} p(\bar{z}_{E_3} | z_3)p(\bar{z}_{E_{4,5}} | z_1)p(z_1, z_3 | x)
\]

\[
= \sum_{z_1, z_3} p(\bar{z}_{E_{4,5}} | z_1) \frac{p(z_3 | \bar{z}_{E_3})}{p(z_3)} p(z_1 | z_3, x)p(z_3 | x)
\]

\[
= \sum_{z_1, z_3} p(\bar{z}_{E_{4,5}} | z_1)p(z_3 | \bar{z}_{E_3})p(z_1 | z_3, x)
\]

\[
= \sum_{z_1} \lambda(z_1) \sum_{z_3} p(z_1 | z_3, x)p(z_3 | \bar{z}_{E_3})
\]

• Again, note the $O(r^3)$ computation.
Required Messages

- Message to $X$ from $Z_1$ requires all other messages to $Z_1$.

$$
p(\bar{z}_{E_1} | x) = \sum_{z_1} \lambda(z_1) \sum_{z_3} p(z_1 | z_3, x) p(z_3 | \bar{z}_{E_3})
$$

- $p(\bar{z}_{E_2} | x)$ is of course similar.

$$
\lambda(z_1) = p(\bar{z}_{E_4} | z_1) p(\bar{z}_{E_5} | z_1)
$$
Inference in polytrees

- We still need $p(y_i | \bar{y}_{E_i})$. For example, $p(y_2 | \bar{y}_{E_2})$, but note that this is just like $p(y_2 | \text{all evidence})$, but without the evidence found at subnetwork connected to the terminal (head) side of the link $Y_2 \rightarrow X$.

$$p(y_2 | \bar{y}_{E_2}) = p(y_2 | \bar{y}_{E_3}, \bar{y}_{E_4}, \bar{y}_{E_5}) \propto p(\bar{y}_{E_3}, \bar{y}_{E_4}, \bar{y}_{E_5}, y_2)$$

$$\propto p(\bar{y}_{E_5} | y_2) p(y_2 | \bar{y}_{E_3}, \bar{y}_{E_4})$$

$$= p(\bar{y}_{E_5} | y_2) \sum_{y_3, y_4} p(y_2 | y_3, y_4) p(y_3, y_4 | \bar{y}_{E_3}, \bar{y}_{E_4})$$

$$= p(\bar{y}_{E_5} | y_2) \sum_{y_3, y_4} p(y_2 | y_3, y_4) p(y_3 | \bar{y}_{E_3}) p(y_4 | \bar{y}_{E_4})$$

$$= p(\bar{y}_{E_5} | y_2) \sum_{y_3, y_4} p(y_2 | y_3, y_4) \pi(y_3) \pi(y_4)$$

$$= p(y_2 | \text{evidence}) / p(\bar{y}_{E_X} | y_2)$$
Inference in polytrees

- Computing $p(y_2 | \bar{y}_{E_2})$, but note that this is just like $p(y_2 | \text{all evidence})$, but without the evidence found at subnetwork connected to the terminal (head) side of the link $Y_2 \rightarrow X$. 
Inference in polytrees: summary

• Summary: In each case, to send a message from a node \( X \) to a node \( Y \), we need first to have all messages from other neighbors of \( X \) (besides \( Y \)) before we can send to \( Y \).
• If we wish to send a message from \( X \) to \( Y \) and we already have a message sent from \( Y \) to \( X \), we can just divide it out (which gives the same update as if we hadn’t divided out the message). Note this will soon be seen to be the difference between what is called Hugin propagation and Shenoy/Shafer propagation.
• In all cases, cost is \( O(r^{J+1}) \) where \( J \) is the maximum number of parents in any node. So, there seems to be no way to do exact inference without paying exponential cost in number of parents. In case when \( J = 1 \), we get the standard cost of a tree, namely \( O(Nr^2) \).
• Note, this is still (poly)trees. What happen if there is an undirected cycle (i.e., a cycle when we moralize the graph, or alternatively when we have a general DAG for the BN).
What if cycles exist?

- An undirected cycle in a BN means that if we moralize the graph, there is a cycle not involving child and its parents.
  - There must be a V-structure. Why?
  - Problem: nobody is ready to send a message. deadlock.
  - Problem: Even the messages themselves are not valid, as the conditional independence properties we used to derive the messages are no longer true.
  - Problem: If we use the messages as is (assuming invalid independence properties), the messages no matter what the ordering of them might not converge to anything.
  - Problem: Even if it does converge to something, it might not converge to anything probabilistically meaningful.
Ex: messages not valid

\[ p(x_1 | \bar{x}_E) = p(x_1 | \bar{y}_{E_1}, \bar{y}_{E_2}, \bar{x}_{E_2}, \bar{x}_{E_3}) \]
\[ \propto p(x_1, \bar{y}_{E_1}, \bar{y}_{E_2}, \bar{x}_{E_2}, \bar{x}_{E_3}) \]
\[ \propto p(\bar{x}_{E_2}, \bar{x}_{E_3} | x_1) p(x_1 | \bar{y}_{E_1}, \bar{y}_{E_2}) \]

no longer factors

Moreover, in the above \( X \) might not even be a separator, so the 2nd to 3rd line would be invalid.

Also,

\[ \sum_{y_1, y_2} p(x_1 | y_1, y_2) p(y_1, y_2 | \bar{y}_{E_1}, \bar{y}_{E_2}) \]

no longer factors
Inference

- Pearl’s algorithm is in fact a special case of what is now known as the junction tree algorithm.
- As we know, triangulated graphs are generalizations of trees. The basic idea of message passing can be just as easily defined on junction trees, which are “covers” of the original graph, and therefore do not violate any properties of such graphs.
- We can see the JT as a data-structure on which to perform inference.
- JT algorithm can also be seen as a form of elimination, where we are eliminating (summing out) multiple nodes at the same time.
- Key is locality: 1) we build data structure that we perform local operations in the right order on the right data structure. Ensuring this locality will guarantee globally correct solution.
- JT is key to why local operations imply global correctness.
- Local operations are less expensive computationally, for this set of queries (which will be $p(x_c|x_E)$ for each $c \in C$).
Inference: outline

1) If starting with a BN, convert it to a MRF using moralization.
2) Inference can be (correctly, as we will see) done in terms of achieving local consistency on the right data structure for a MRF, by performing operations in the right order.
3) JT is that right data-structure, to ensure that local consistency implies global consistency.
4) We have a JT iff G is triangulated iff decomposable iff all min $(\alpha, \beta)$ seps are complete. Therefore, we must triangulate the graph after moralization. This will cover the original graph.
5) Complexity is exponential in the clique size. Need to triangulate while minimizing largest clique size (NP-complete optimization problem).
Inference: moralization

- **BN:** \( p(x) = \prod_v p(x_v | x_{\text{pa}(v)}) \) by (DF)
- **MRF:** \( p(x) = \prod_{c \in C} \psi_c(x) \) by (F), where \( \psi_c(x) \) are clique potentials.
- After moralization and triangulation of a BN, we can treat nodes and their former parents as complete sets in the graph, which can be used to form cliques (but not nec. maxcliques). Therefore, for each child/parent set, we find a maxclique in the triangulated graph whose potential function can “hold” it. How?

1) Initialize clique potentials to unity. \( \psi_c(x) = 1 \) for all \( c \).

2) For each \( p(x_v | x_{\text{pa}(v)}) \), chose a clique \( c \) s.t. \( x_v \cup x_{\text{pa}(v)} \in c \), and update via \( \psi_c(x) \leftarrow \psi_c(x)p(x_v | x_{\text{pa}(v)}) \).

When this is done, we will have that:

\[
p(x) = \prod_v p(x_v | x_{\text{pa}(v)}) = \prod_{c \in C} \psi_c(x)
\]
Examples: assigning prob to clique potentials

- Cliques are: \{A, B, C\}, \{C, D, E\}, \{D, E, F\}.
  \[ \psi_{A,B,C} = p(A)p(B)p(C|A, B), \quad \psi_{C,D,E} = p(D|C)p(E|C), \]
  \[ \psi_{D,E,F} = p(F|D, E). \]

So, \[ p(A, B, C, D, E, F) = p(A)p(B)p(C|A, B)p(D|C)p(E|C)p(F|D, E) = \]
\[ \psi_{A,B,C}\psi_{C,D,E}\psi_{D,E,F} \]

- Note: clique functions are not unique. For example:

\[ p(x) = \prod_{c \in \mathcal{C}} \psi_c(x) = \prod_{c \in \mathcal{C}} \psi'_c(x) \]

with \[ \psi'_c(x) = \alpha_c \psi_c(x), \] with \[ \alpha_c \] appropriately chosen constants to maintain \[ p(x) \] in the product.
Examples: assigning prob to clique potentials

- Of course, with MRFs, we can allow the cliques to be arbitrary non-negative and write:

\[ p(x) = \frac{1}{Z} \prod_{c \in C} \psi_c(x) \]

with

\[ Z = \int \prod_{c \in C} \psi_c(x) dx \]

The constant \( Z \) can be absorbed into any of the clique functions. Key property we need is that the result when multiplied together yields \( p(x) \).

- Q: again, why is moralization valid? Why is it needed? Make sure this is clear.
Evidence

- Evidence: We treat it the same way as before, but there are several ways of viewing it.
- \( V = E \cup H \) is a partition of the nodes into hidden and evidence nodes. For each \( e \in E \), we have a \( \delta(x_e, \bar{x}_e) \) for evidence value \( \bar{x}_e \).
- Each clique potential is partitioned in the same way. I.e.,
  \[ \psi_c(x) = \psi_c(x_e) = \psi_c(x_{c \cap H}, x_{c \cap E}). \]
- There are actually many different forms of evidence. For example:
  \[
  \begin{align*}
  E &= e \\
  E &\in \{e_1, e_2, \ldots, e_n\} \\
  E &\not\in \{e_1, \ldots\}.
  \end{align*}
  \]

These forms of evidence are all similar and can be treated in the same way.
- Assume we find out that \( X_E = \bar{x}_e \) and we want to calculate:
  \[
  p(x_H|\bar{x}_E) = \frac{p(x_H, \bar{x}_E)}{p(\bar{x}_E)}
  \]
\[ p(x_H | \bar{x}_E) = \frac{1}{p(\bar{x}_E)} \frac{1}{p(x_H, \bar{x}_E)} = \frac{\prod_c \psi_c(x_H, \bar{x}_E)}{Z} \frac{Z}{\sum_{x_H} \prod_c \psi_c(x_H, \bar{x}_E)} = \frac{\prod_c \psi_c(x_H, \bar{x}_E)}{\sum_{x_H} \prod_c \psi_c(x_H, \bar{x}_E)} \]

- Therefore, we can treat \( \prod_c \psi_c(x_H, \bar{x}_E) \) as a goal since it is an unnormalized version of both \( p(x_H, \bar{x}_E) \) and \( p(x_H | \bar{x}_E) \) (we just do appropriate normalization).
- To incorporate evidence \( \bar{x}_E \) starting from clique potential \( \psi_c(x_c) \), we just do: \( \psi_c(x_c) \leftarrow \psi_c(x_c) \prod_{i \in (E \cap c)} \delta(x_i, \bar{x}_i) \)
- To clarify functional notation: \( \psi_c(x) = \psi_c(x_c) \). Also, \( \psi_c(x_H, \bar{x}_E) = \psi_c(x_H \cap c, \bar{x}_E \cap c) \) (i.e., this says what \( \psi \)'s arguments are over).
- Example: \( p(A = 1) = 0.8, p(B = 1 | A = 1) = 0.7, p(B = 1 | A = 0) = 0.4 \).
Example potentials

Single clique \{A, B\} potential \(p(A)p(B|A)\) gives:

\[
\psi_{A,B} = \begin{bmatrix} 0.12 & 0.08 \\ 0.24 & 0.56 \end{bmatrix}
\]

(columns are \(B = 0, B = 1\) respectively, and rows are \(A = 0, A = 1\)).

Given evidence \(B = 1\), we zero out inconsistent entries:

\[
\psi_{A,B} = \begin{bmatrix} 0.0 & 0.08 \\ 0.0 & 0.56 \end{bmatrix}
\]

And perhaps renormalize if we want:

\[
\psi_{A,B} = \begin{bmatrix} 0.0 & 0.125 \\ 0.0 & 0.875 \end{bmatrix} = p(A|B = 1)
\]

Summary: incorporating evidence means we zero out the entries in clique potentials that are inconsistent with the evidence.
Goal

- Goal: Once process of inference is over, we wish to have the clique potentials be marginals, i.e., goal is for $\psi_c(x_c) = p(x_c)$, or with evidence $\bar{x}_E$ we have marginals of the form $\psi_c(x_{c \cap H}) = p(x_{c \cap H}, \bar{x}_E)$
- This is sufficient for example to ask many queries, form the joint, or do learning.
- But under (F) property, we can not get that the local potential functions are marginals. I.e., we can’t have $\psi_c(x) = p(x_c)$ where $p(x) = \prod_c \psi_c(x)$.
- But if we have a decomposable graph, then we get:

$$p(x) = \frac{\prod_{c \in C} p(x_c)}{\prod_{s \in cal S} p(x_s)} = \frac{\text{prod. of clique marginals}}{\text{prod. of sep marginals}} = \frac{\prod_{c \in C} \psi_c(x_c)}{\prod_{s \in cal S} \phi_s(x_s)}$$

With $\psi_c(x_c) = p(x_c)$ and $\phi_s(x_s) = p(x_s)$.

- Example: If $A \rightarrow B \rightarrow C$, then

$$p(A, B, C) = p(A)p(B|A)p(C|B) = p(A, B)p(B, C)/p(C).$$

So the above is a generalization of one of the definitions of conditional independence.
Potentials as marginals

- Again, $\psi_c(x_c)$ and $\phi_s(x_s)$ can be positively scaled while maintaining the above global interpretation. Thus, the above shows that it is possible to have potentials as marginals, but it is not necessarily so.
- The JT algorithm will adjust potential functions, starting from any initial valid configuration (or charge), so that they become actual marginals. valid means that the above global interpretation (i.e., result is $p(x)$) is true.
- But why do we want marginals? First, this is what is minimally necessary for learning. For example, it is easy to calculate prob. of individual nodes.

$$p(x_{c'}) = \sum_{c \setminus c'} \psi_c(x_c) \quad \text{for any } c' \subseteq c$$

- Subsets of nodes in neighboring cliques:

$$p(x_{c_1'}, x_{c_2'}) = p(x_{c_1'}|x_{c_2'})p(x_{c_2'}) = p(x_{c_1'}|x_s)p(x_{c_2'}) = \frac{p(x_{c_1'})p(x_{c_2'})}{p(x_s)}$$

$$p(x_{c_1'}) = \sum_{c_1 \setminus c_1'} \psi_{c_1}(x_{c_1'})$$
Potentials don’t’ start as marginals

• So goal is to get potentials as marginals, but there are several reasons why they don’t start out as such.
  1) After introduction of evidence, some clique potentials are unchanged, those with \( c \cap E = \emptyset \), and will not be able to produce \( p(x_c|\bar{x}_E) \) (but this is what we need).
  2) Not all the evidence \( \bar{x}_E \) is present at (probably all) of the clique potentials.
  3) Clique potentials might be initialized by conditional probabilities from directed graph, so might have, say, \( \psi_{DEF} = p(F|D,E) \) (not a marginal over \( DEF \)).
  4) Clique potentials might be specified arbitrarily at first (e..g, MRFs using log-linear models), so potentials are just some unnormalized values.
Potentials don’t’ start as marginals

- So, assume we start with:

\[
p(x_H, \bar{x}_E) = \prod_c \psi_c(x_c) = \frac{\prod_c \psi_c(x_H, \bar{x}_E)}{\prod_s \phi_s(x_H, \bar{x}_E)}
\]

with \( \psi_s = 1 \) for now, and \( \psi_c \) specified according to the clique factorization of the joint distribution.

- At the very least, for \( \psi_c \) the clique potentials to be marginals, they must agree on the nodes that they have in common to each other. agree means “be consistent with”, i.e., give the same marginals over intersection. I.e., given cliques \( C_1 \) and \( C_2 \), with \( S = C_1 \cap C_2 \), we must have:

\[
\sum_{x_{C_1 \setminus S}} \psi_{C_1}(x_{C_1}) = \sum_{x_{C_2 \setminus S}} \psi_{C_2}(x_{C_2})
\]