Inference and Representation

DS-GA-1005, CSCI-GA.2569

Joan Bruna
Courant Institute of Mathematical Sciences
Center for Data Science
NYU

Announcements

- Project Proposal is available, due 10/23

- PS3 released. Due 10/9 (two weeks from now).

Undirected Graphical Models

- Factors only contain nodes that are fully-connected — this is called a *clique*.
- Since a clique of size *m* contains all cliques of smaller sizes, we can reduce ourselves to *maximal cliques* (cliques that cannot be extended while being fully connected).
  - If X_C form a maximal clique, arbitrary functions $\psi(x_C)$ capture all possible dependencies within the clique.
- So, by considering

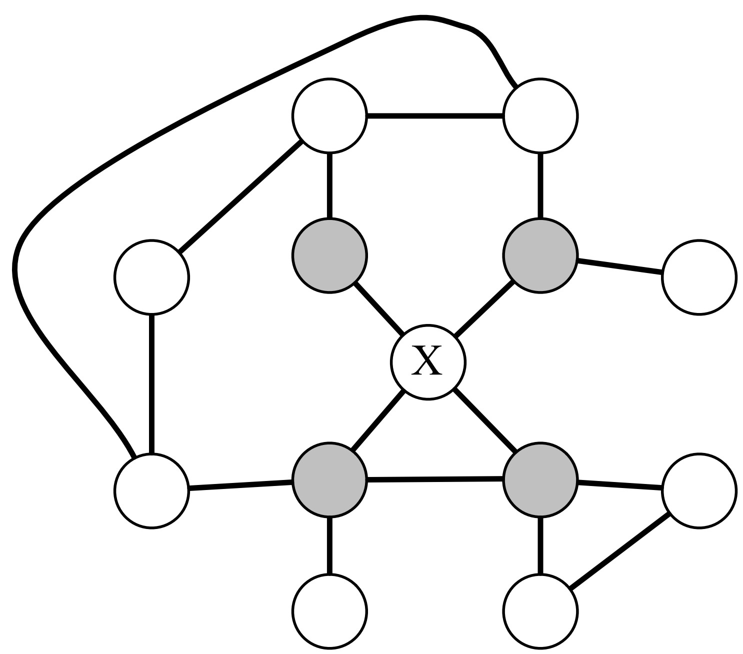
$$\mathcal{C} = \text{ set of maximal cliques of } G$$

$$\psi_C(x_C) : \text{non-negative potential function (not necessarily normalized)}$$

- We have $p(x) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \; , \qquad \underbrace{Z = \int dx \prod_{C \in \mathcal{C}} \psi_C(x_C)}_{\text{partition function}} \; .$

Markov Blanket

- A set $A \subseteq \mathcal{X}$ is a Markov Blanket of X if $X \notin A$ and if A is a minimal set of nodes such that $X \perp (\mathcal{X} \setminus (A \cup X)) \mid A$.
- In undirected graphical models, the Markov Blanket of a variable is precisely its neighbors in the graph:



- X is independent of the rest of nodes conditioned on its neighbors.

Ising Model

$$p(X_1, \ldots, X_n) = \frac{1}{Z} \exp\left(- \sum_{i<j} w_{i,j} X_i X_j - \sum_i u_i X_i \right) .$$

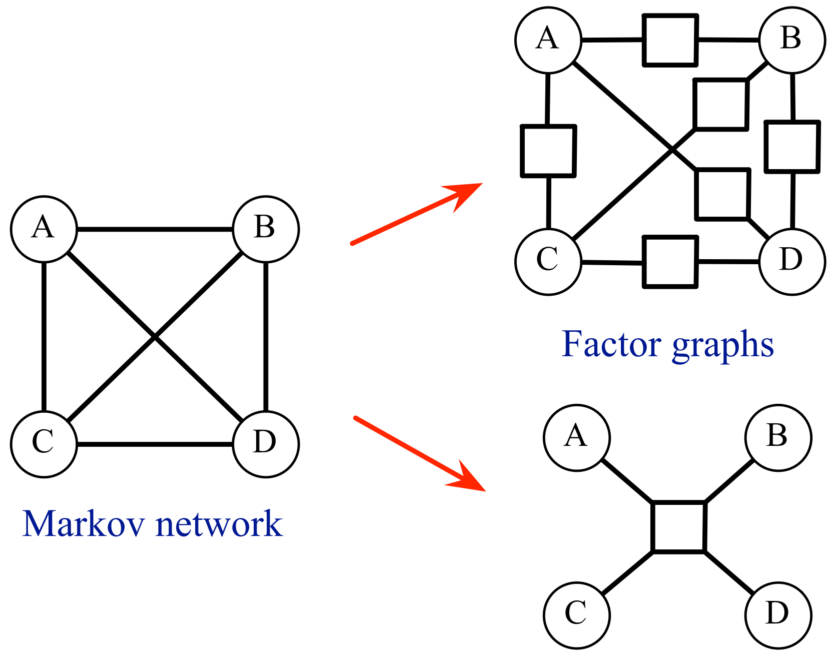
- Undirected graphical model with graph given by (1d/2d) lattice.
  - $w_{i,j} > 0$: ferromagnetic interactions (why?)
  - $w_{i,j} < 0$: anti-ferromagnetic interactions (why?)
  - u_i: external magnetic field
  - only neighbors in the lattice contribute to the interaction terms.
- From statistical mechanics, we can interpret the exponent

$$H(X) = - \sum_{i<j} w_{i,j} X_i X_j - \sum_i u_i X_i$$

as an energy quantity (in fact, it is the Hamiltonian of the system).

Factor Graphs

- A *factor graph* is a bipartite graph where
  - nodes correspond to **both** random variables $\{X_i\}_{i \leq n}$ and potential factors $\{\psi_C\}_{C \in \mathcal{C}}$.
  - edges can only be drawn between variable and factor nodes (if variable X_i appears in factor ψ_C).



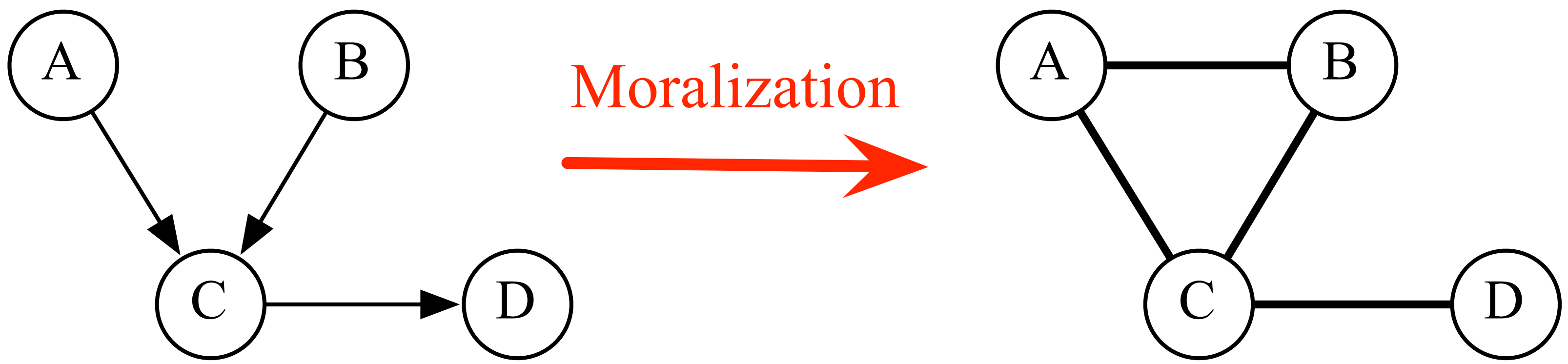
- Factor graphs do not have the clique vs maximal clique ambiguity (why?).
- Same probabilistic model, different graphical representation.

Lecture 4 Objectives

- The Hammersley-Clifford Theorem
- From Inference to Approximate Inference
- Belief Propagation

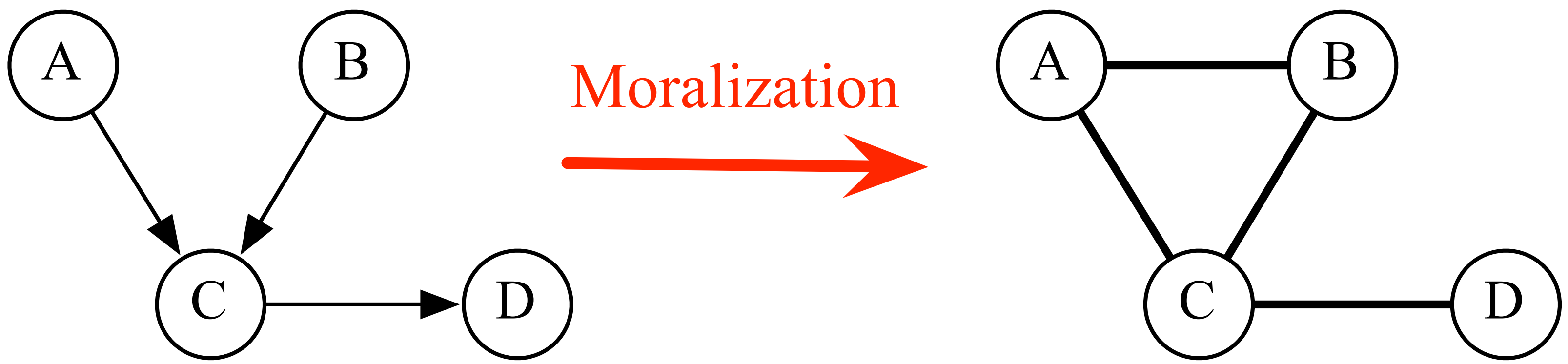
Moralization

- Algorithm to map a Bayesian Network to a Markov Network.
- Given $G = (V, E)$ DAG, we define $M(G)$ an undirected graph over V, with edge between X_i and X_j whenever
  - $X_j \to X_i$ or $X_i \to X_j$ in G.
  - X_i and X_j are parents of the same node.



Moralization

- Algorithm to map a Bayesian Network to a Markov Network.
- Given $G = (V, E)$ DAG, we define $M(G)$ an undirected graph over V, with edge between X_i and X_j whenever
  - $X_j \to X_i$ or $X_i \to X_j$ in G.
  - X_i and X_j are parents of the same node.



- In $M(G)$, we can no longer tell that $A \perp B$.
  - V-structures disappear, but we can still model "explaining away" with e.g. sparsity priors.

Moralization

- Equivalently, this rule is obtained by mapping factorization of joint distribution.

Bayesian Net

$$p(x_1, \ldots, x_n) = \prod_i p(x_i \mid x_{Pa(i)})$$

MRF

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C)$$

Moralization

- Equivalently, this rule is obtained by mapping factorization of joint distribution.

Bayesian Net ⟷ **MRF**

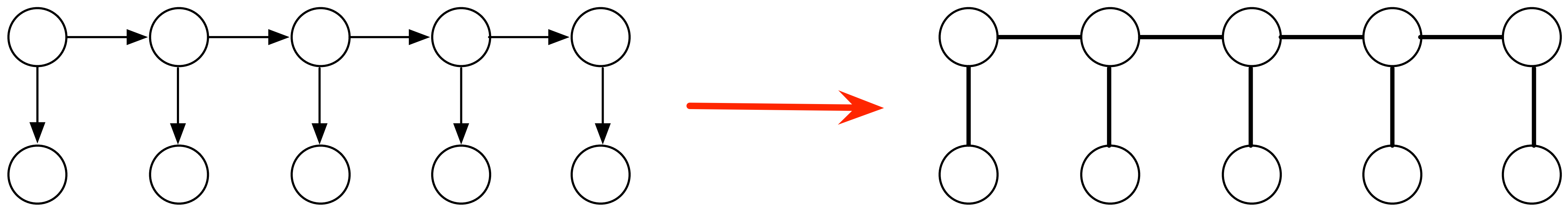
$$p(x_1, \ldots, x_n) = \prod_i p(x_i \mid x_{Pa(i)})$$

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C)$$

- Each node generates a factor in the resulting factor graph:

$$\psi_{C_i}(x_{C_i}) := p(x_i \mid x_{Pa(i)}) \ , \ C_i = \{i\} \cup Pa(i) \ .$$

- Ex: Hidden Markov Model:



Hammersley-Clifford Theorem

- We saw earlier that some distributions cannot be modeled as Bayesian Networks.
- Now we ask: which distributions can be written as Markov Fields using an appropriate graph?

Hammersley-Clifford Theorem

- We saw last week that some distributions cannot be modeled as Bayesian Networks.
- Now we ask: which distributions can be written as Markov Fields using an appropriate graph?
- $p(x)$ is a *Gibbs distribution over* G if it can be written as

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \ , \ \mathcal{C} = \text{cliques of } G$$

Hammersley-Clifford Theorem

- We saw last week that some distributions cannot be modeled as Bayesian Networks.
- Now we ask: which distributions can be written as Markov Fields using an appropriate graph?
- $p(x)$ is a *Gibbs distribution over* G if it can be written as

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \;, \; \mathcal{C} = \text{cliques of } G$$

- We saw earlier that

  If p is a Gibbs distribution for G, then $I(G) \subseteq I(p)$.

  – i.e. if Y separates X and Z in G, then $X \perp Z \mid Y$.

Hammersley-Clifford Theorem

- We saw last week that some distributions cannot be modeled as Bayesian Networks.
- Now we ask: which distributions can be written as Markov Fields using an appropriate graph?
- $p(x)$ is a *Gibbs distribution over* G if it can be written as

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \ , \ \mathcal{C} = \text{cliques of } G$$

- We saw earlier that

If p is a Gibbs distribution for G, then $I(G) \subseteq I(p)$.

  - i.e. if Y separates X and Z in G, then $X \perp Z \mid Y$.
- Converse true?

Hammersley-Clifford Theorem

- We saw last week that some distributions cannot be modeled as Bayesian Networks.
- Now we ask: which distributions can be written as Markov Fields using an appropriate graph?
- $p(x)$ is a *Gibbs distribution over* G if it can be written as

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \ , \ \mathcal{C} = \text{cliques of } G$$

- We saw earlier that

$$\text{If } p \text{ is a Gibbs distribution for } G\text{, then } I(G) \subseteq I(p).$$

  – i.e. if Y separates X and Z in G, then $X \perp Z \mid Y$.
- Converse true?
  – Not in general.

Hammersley-Clifford Theorem

- However, if we assume that p is positive, i.e. $p(x) > 0$ for all x,
- Then we have

Theorem [H-C]: An undirected graph G is an I-map for a positive distribution $p(x)$ iff p is a Gibbs distribution that factorizes over G.

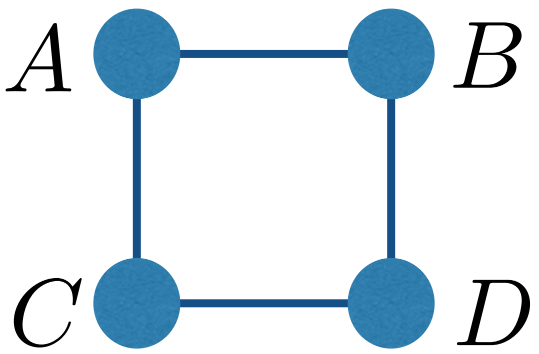
- It provides a parametrization for any distribution that complies with a series of conditional independence assumptions (Markov Property).

- Positivity condition is needed!

Global Markov but not Factorizing

- Consider 4 binary random variables A, B, C, D, and the following distribution:

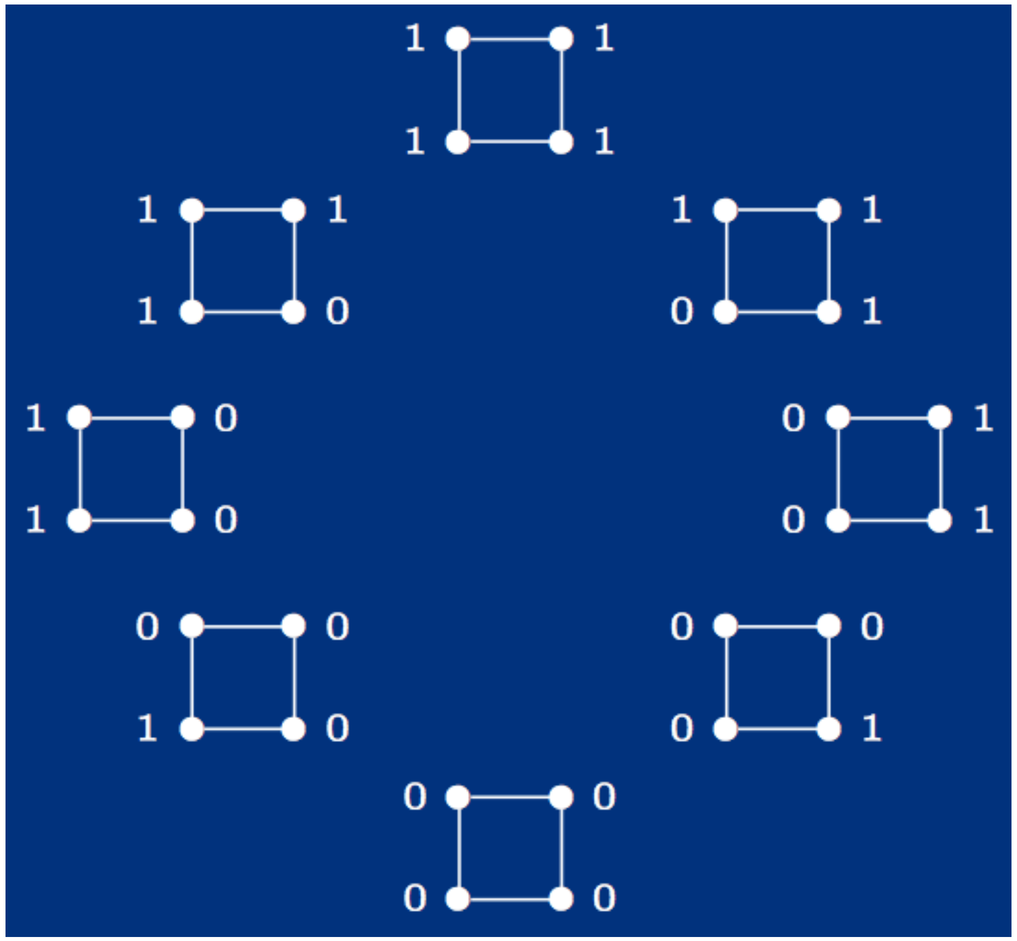
$$p(A=1,B=1,C=1,D=1)=\frac{1}{8}\ ,\ p(A=1,B=1,C=0,D=1)=\frac{1}{8}$$
$$p(A=0,B=1,C=0,D=1)=\frac{1}{8}\ ,\ p(A=0,B=0,C=0,D=1)=\frac{1}{8}$$
$$p(A=0,B=0,C=0,D=0)=\frac{1}{8}\ ,\ p(A=0,B=0,C=1,D=0)=\frac{1}{8}$$
$$p(A=1,B=0,C=1,D=0)=\frac{1}{8}\ ,\ p(A=1,B=1,C=1,D=0)=\frac{1}{8}$$

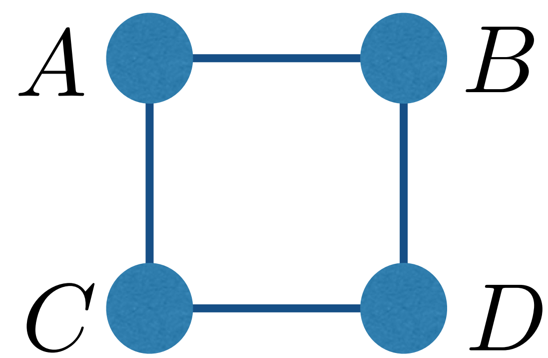


- Do we have $I(G) \subseteq I(p)$?

Global Markov but not Factorizing

- Consider 4 binary random variables A, B, C, D, and the following distribution:





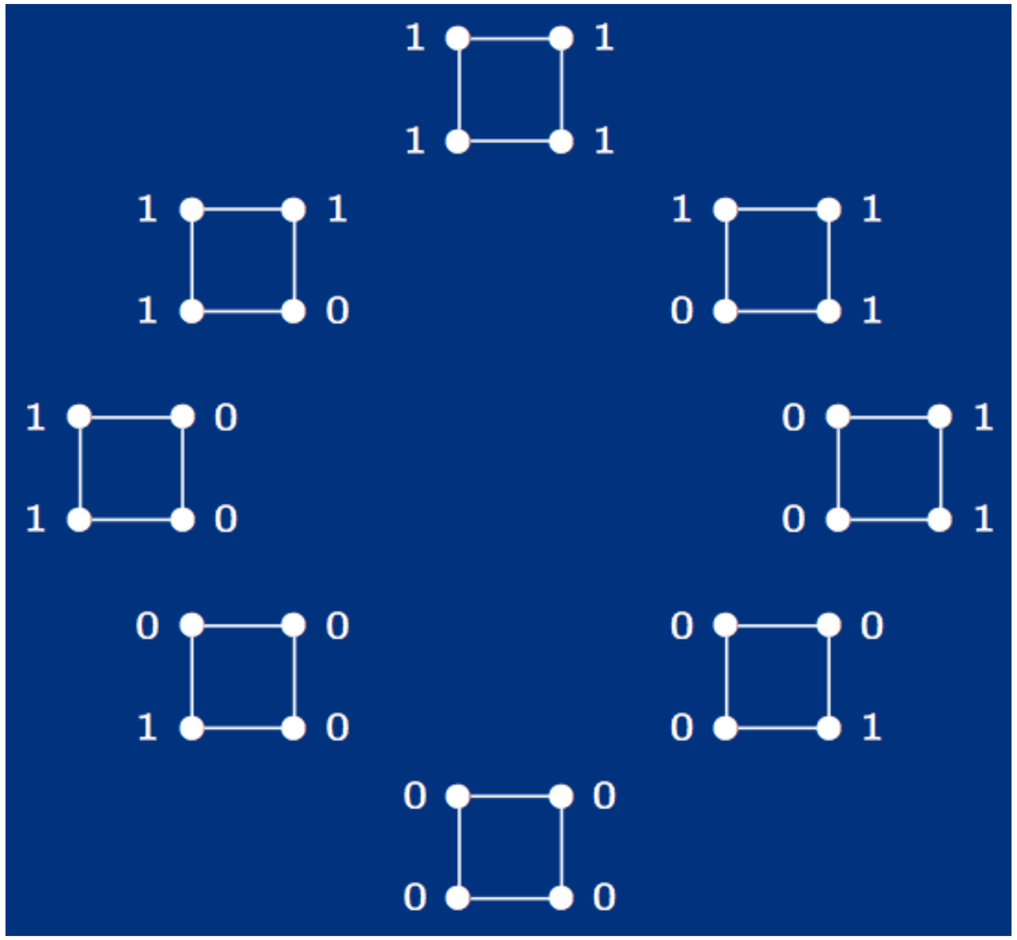
- Do we have $I(G) \subseteq I(p)$?

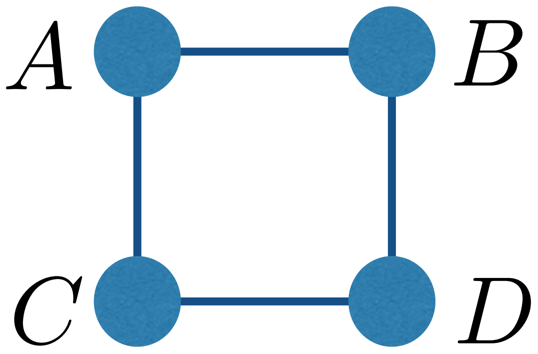
$$A \perp D \mid \{B, C\} \quad B \perp C \mid \{A, D\}$$

  – Observe that conditioning on opposite corners always yields one corner deterministic, and $X \perp Y$ **whenever** X **or** Y **are deterministic.**

Global Markov but not Factorizing

- Consider 4 binary random variables A, B, C, D, and the following distribution:

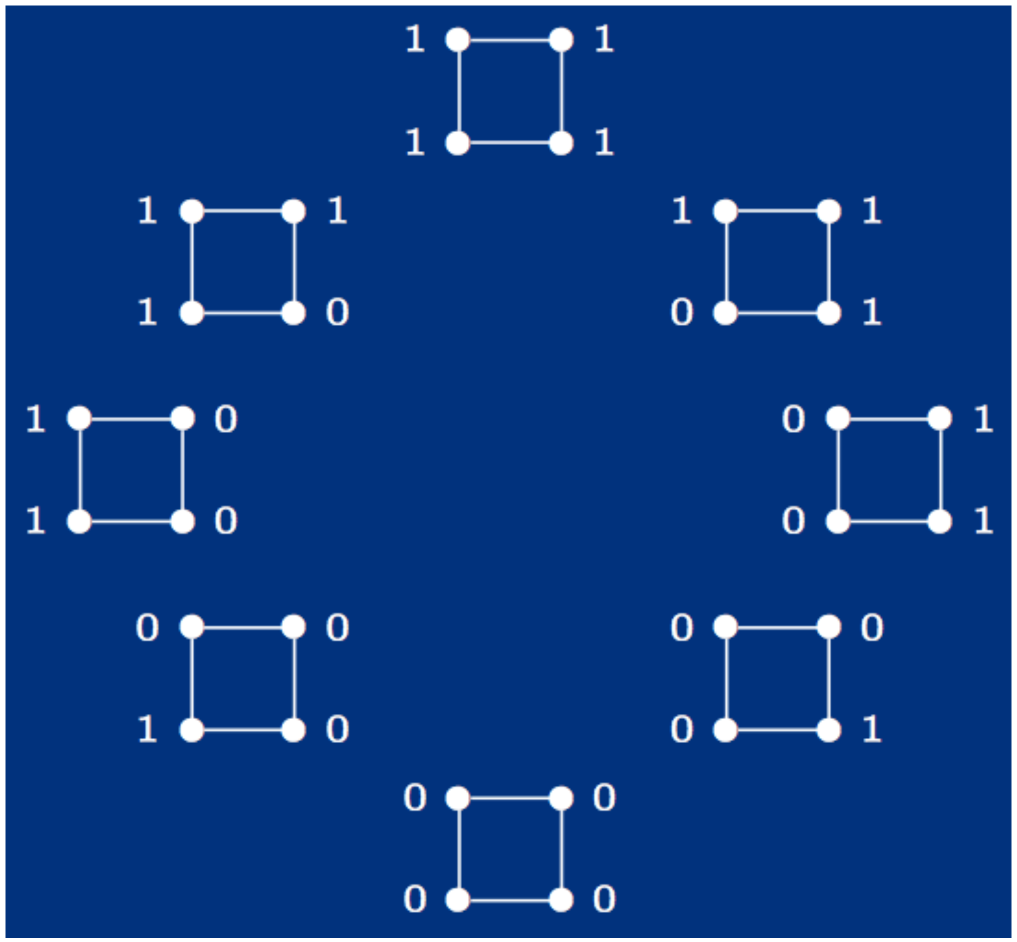


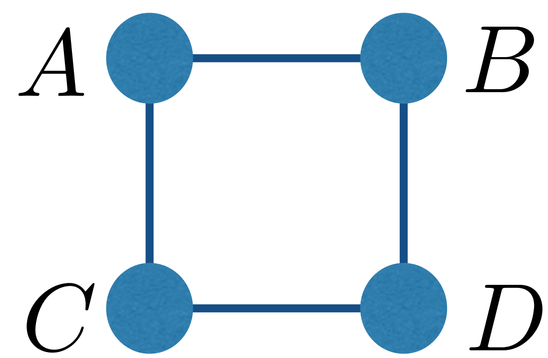


- Is p a Gibbs distribution?

Global Markov but not Factorizing

- Consider 4 binary random variables A, B, C, D, and the following distribution:





- Is p a Gibbs distribution?
  - Assume $p(x) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C)$

$$0 < Z \cdot p(0,0,0,0) = \psi_{AB}(0,0)\psi_{BD}(0,0)\psi_{DC}(0,0)\psi_{CA}(0,0)$$

so all these factors are strictly positive

  - Trying all 8 positive events implies all factors are strictly positive!

Parameter Estimation

- So far, we have described two families of graphical models, with pros and cons.
- In practice, given some dataset, how to choose which one? Which parameters?
- We assume data is sampled from an underlying (unknown) distribution p^* , associated to some network model $\mathcal{M}^* = (G^*, \theta^*)$

Parameter Estimation

- So far, we have described two families of graphical models, with pros and cons.
- In practice, given some dataset, how to choose which one? Which parameters?
- We assume data is sampled from an underlying (unknown) distribution p^* , associated to some network model $\mathcal{M}^* = (G^*, \theta^*)$
- Samples $\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \sim p^* \text{ iid.}$
- In order to "search" for $\mathcal{M}^*$, we parametrize the search within a family of graphical models
  - We can learn both model parameters for a fixed graph structure,
  - or both structure and parameters.

Task-driven inference

- Depending on the task, we might want to perform different kinds of estimation.
    1. Density Estimation: we are interested in the joint distribution, which can be subsequently used to perform any inference query.
    2. Prediction: we are only interested in a specific set of conditional distribution, e.g classification, or output prediction.
    3. Structural discovery: We are interested in the graph itself (not so much the parameters), e.g. determining dependencies between genes.

- (1) is typically harder than (2). (3) is typically harder than (2) and (1).

Parameter Estimation

- Let us focus on (1) first. $\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \sim p^*$ iid.
- Suppose $p^* = p_{\theta^*}$ for some θ^*.
- Two main approaches for parameter estimation:

Parameter Estimation

- Let us focus on (1) first. $\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \sim p^*$ iid.
- Suppose $p^* = p_{\theta^*}$ for some θ^*.
- Two main approaches for parameter estimation:
  - Maximum Likelihood Estimation:

$$E(\theta) = \log p(\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \mid \theta) = \sum_{l \leq L} \log p(\mathbf{X}^l \mid \theta)$$

$$\hat{\theta}_{MLE} = \arg\max_{\theta} E(\theta)$$

Parameter Estimation

- Let us focus on (1) first. $\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \sim p^*$ iid.
- Suppose $p^* = p_{\theta^*}$ for some θ^*.
- Two main approaches for parameter estimation:
  - Maximum Likelihood Estimation:

$$E(\theta) = \log p(\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \mid \theta) = \sum_{l \leq L} \log p(\mathbf{X}^l \mid \theta)$$

$$\hat{\theta}_{MLE} = \arg\max_{\theta} E(\theta)$$

  - Under appropriate assumptions, $\hat{\theta}_{MLE}$ is
    - consistent (as sample size grows, $\hat{\theta}_{MLE} \to \theta^*$ (in probability)
    - asymptotically efficient (no other consistent estimator has lower asymptotic mean-squared error).
  - However, in general this estimation is computationally intractable.

Parameter Estimation

- Let us focus on (1) first. $\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \sim p^*$ iid.
- Suppose $p^* = p_{\theta^*}$ for some θ^*.
- Two main approaches for parameter estimation:
  - Method of Moments:

    Consider measurable functions $g_1, \ldots, g_S$.

    (e.g. $g_i(\mathbf{x}) = x_{i_1} x_{i_2}$)

For each θ, we have $\mu_s(\theta) = \mathbb{E}_{X \sim p_\theta}(g_s(\mathbf{X})) \quad s = 1 \ldots S$

For appropriate choice of moments/functions, system is invertible:

$$\theta = F(\mu)$$

Parameter Estimation

- Let us focus on (1) first. $\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \sim p^*$ iid.
- Suppose $p^* = p_{\theta^*}$ for some θ^*.
- Two main approaches for parameter estimation:
  - Method of Moments:

Consider measurable functions $g_1, \ldots, g_S$.

(e.g. $g_i(\mathbf{x}) = x_{i_1} x_{i_2}$)

For each θ, we have $\mu_s(\theta) = \mathbb{E}_{X \sim p_\theta}(g_s(\mathbf{X})) \quad s = 1 \ldots S$

For appropriate choice of moments/functions, system is invertible:

$$\theta = F(\mu)$$

We estimate μ by replacing expectations with empirical averages:

$$\hat{\mu}_s = \frac{1}{L} \sum_{l \leq L} g_s(X^l) \quad s = 1 \ldots S$$

And we plug-in the estimator for θ: $\hat{\theta}_{MM} = F(\hat{\mu})$

MLE in Bayesian Networks

- Let us illustrate ML estimation on BN, assuming we know the Bayesian structure G .

$$p(x_1, \ldots, x_n; \theta) = \prod_{i=1}^{n} p(x_i \mid x_{Pa(i)}; \theta)$$

- Given iid samples $\{X^1, \ldots, X^L\}$, its log-likelihood is

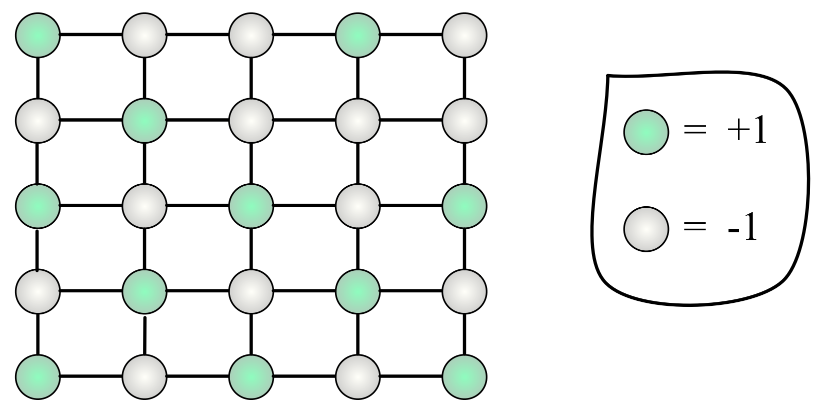
$$E(\theta) = \sum_{l \leq L} \sum_{i \leq n} \log p(X_i^l \mid X_{Pa(i)}^l; \theta)$$

$$= \sum_{i \leq n} \sum_{l \leq L} \log p(X_i^l \mid X_{Pa(i)}^l; \theta_i) \ .$$

  – so the estimation is separable across different factors, breaking the curse of dimensionality.

- Q: How about Markov Random Fields?

Parameter Estimation in MRFs



$$p(x_1, \cdots, x_n) = \frac{1}{Z} \exp\left(\sum_{i<j} w_{i,j} x_i x_j - \sum_i u_i x_i \right)$$

- In a MRF, we also have a factorization into local potentials…

$$p(x_1, \ldots, x_n; \theta) = \frac{1}{Z(\theta)} \prod_{C \in \mathcal{C}} \psi_C(x_C; \theta) \ .$$

- … but the partition function entangles the estimation!

$$\sum_{l \leq L} \log p(X^l; \theta) = \sum_{l \leq L} \left(\sum_{C \in \mathcal{C}} \log \psi(X_C^l; \theta) - \log Z(\theta) \right) \ .$$

Inference in a Graphical Model

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \ , \mathcal{C} = \text{cliques of } G$$

- What does *inference* mean?

Inference in a Graphical Model

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \ , \mathcal{C} = \text{cliques of } G$$

- What does *inference* mean?

- In general, the ability to compute marginal (or equivalently conditional) probabilities:

$$p(x_S) = \sum_{i \notin S} \sum_{x_i} p(x_1, \ldots, x_N) \ .$$

Inference in a Graphical Model

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \ , \mathcal{C} = \text{cliques of } G$$

- What does *inference* mean?

- In general, the ability to compute marginal (or equivalently conditional) probabilities:

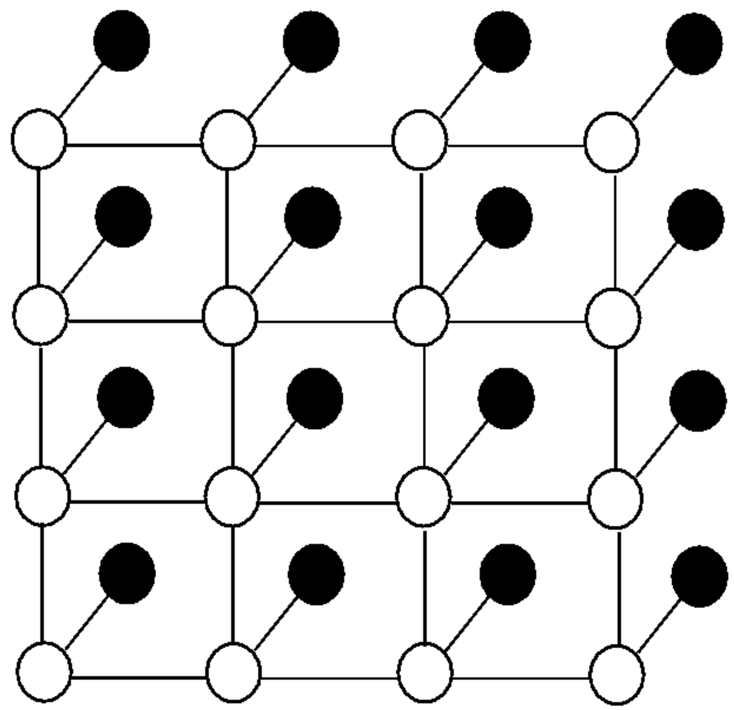
$$p(x_S) = \sum_{i \notin S} \sum_{x_i} p(x_1, \ldots, x_N) \ .$$

- This is an intractable problem for general graphs.
  - Technically, it is "#P-complete" (if a poly-time algorithm existed, then P=NP).
- Approximate inference?

Belief Propagation

- For simplicity (without loss of generality), we consider a pair-wise MRF setting:

$$p(x, y) = \frac{1}{Z} \prod_{(i,j)} \psi_{ij}(x_i, x_j) \prod_i \phi_i(x_i, y_i) \ .$$



y=observed (black)
x=hidden (white)

- Goal: compute $p(x \mid y)$

Belief Propagation

- We need to find a "consensus" amongst the hidden variables to commonly explain observations.
- Intuition of BP algorithm: consensus is reached after repeated "conversation" between local variables, until they agree.

"It looks like we have a consensus."

Belief Propagation

- We need to find a “consensus” amongst the hidden variables to commonly explain observations.
- Intuition of BP algorithm: consensus is reached after repeated “conversation” between local variables, until they agree.

- How to mathematically specify such “conversation” and consensus?

BP for pairwise MRF

- The marginal distribution wrt x becomes

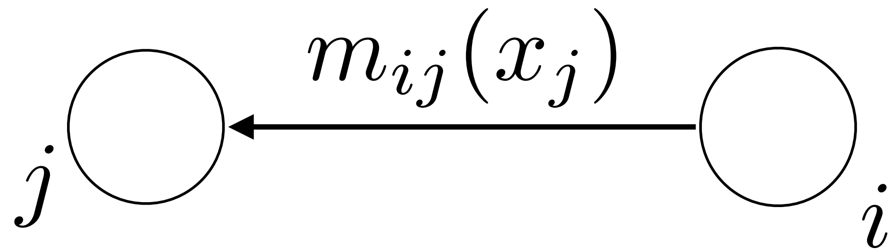
$$p(x|y) = \frac{1}{Z} \prod_{(i,j)} \psi_{ij}(x_i, x_j) \prod_i \tilde{\phi}_i(x_i; y) \ .$$

BP for pairwise MRF

- The marginal distribution wrt x becomes

$$p(x|y) = \frac{1}{Z} \prod_{(i,j)} \psi_{ij}(x_i, x_j) \prod_i \tilde{\phi}_i(x_i; y) \ .$$

- We introduce the *messages* $m_{ij}(x_j)$:



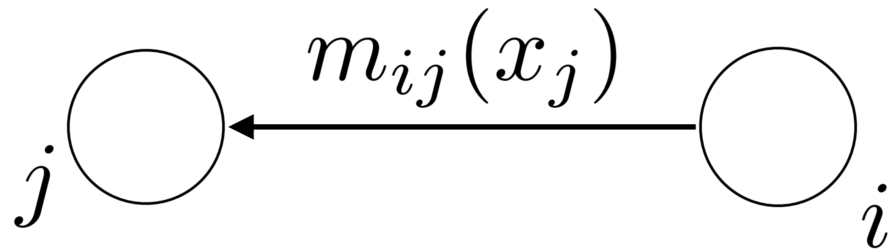
$$m_{ij}(x_j) \ \propto \ \text{how likely node } i \text{ thinks node } j \text{ is in state } x_j.$$

BP for pairwise MRF

- The marginal distribution wrt x becomes

$$p(x|y) = \frac{1}{Z} \prod_{(i,j)} \psi_{ij}(x_i, x_j) \prod_i \tilde{\phi}_i(x_i; y) \ .$$

- We introduce the *messages* $m_{ij}(x_j)$:



$$m_{ij}(x_j) \ \propto \ \ \text{how likely node } i \text{ thinks node } j \text{ is in state } x_j.$$

- *Belief* at node j aggregates incoming messages and unary potential:

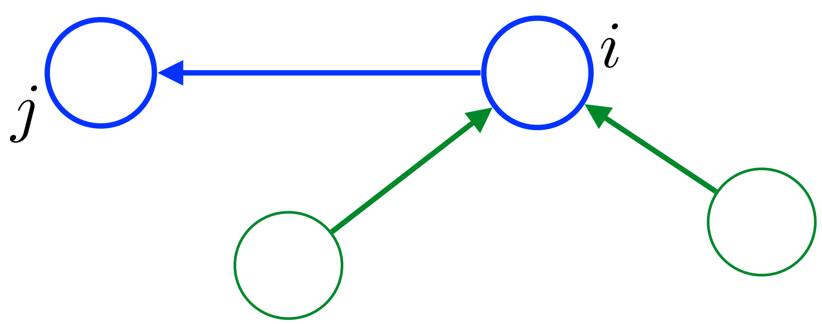
$$b_j(x_j) = \frac{1}{Z_j} \tilde{\phi}_j(x_j; y) \prod_{i \in N(j)} m_{ij}(x_j) \ .$$

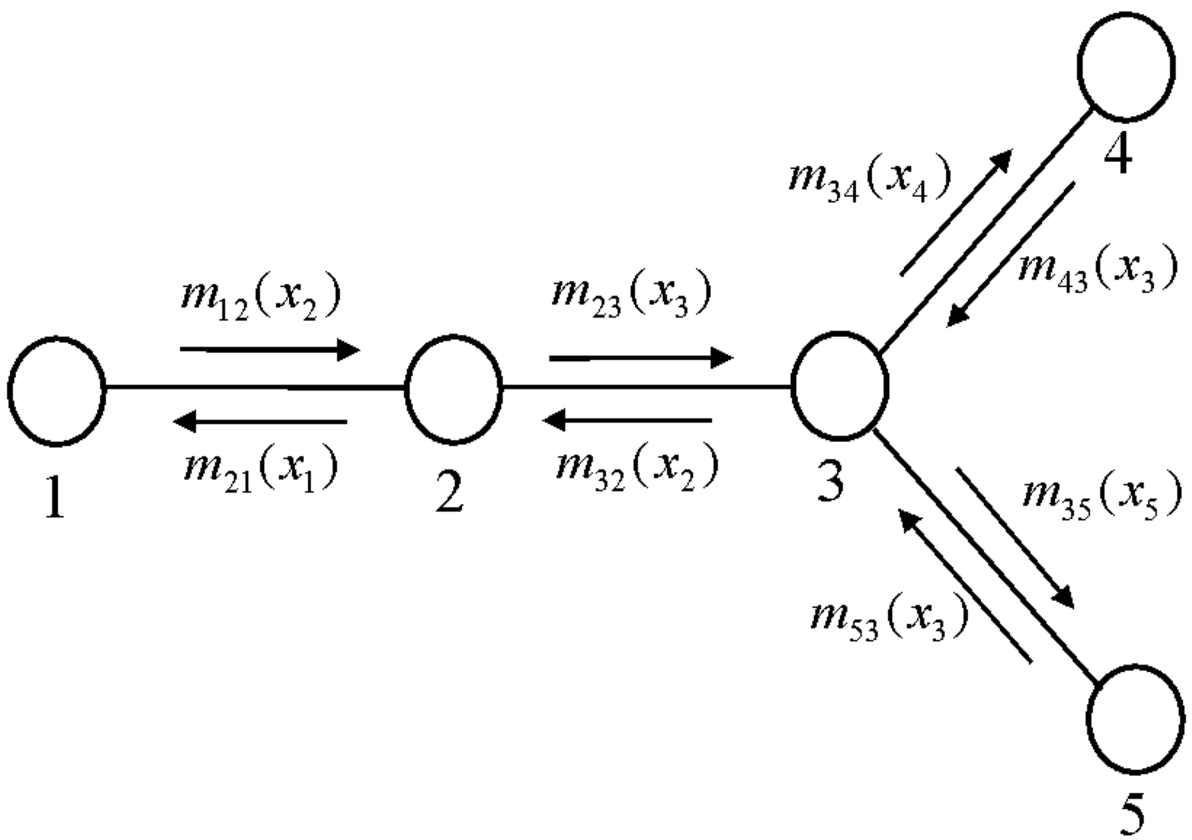
$N(j)$: Neighbors of node j.

BP for pair-wise MRF

- How are messages computed/updated?

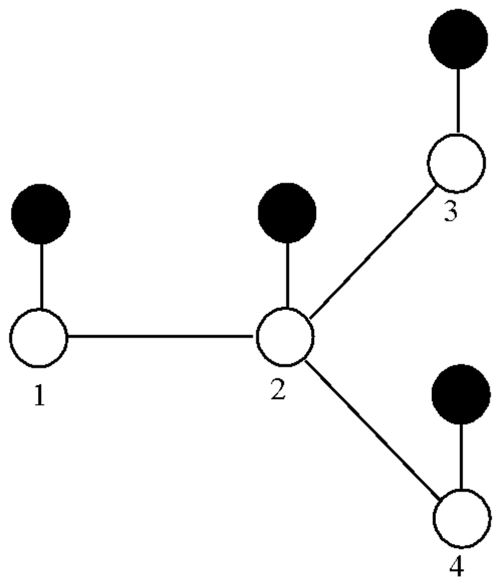
$$m_{ij}(x_j) \leftarrow \sum_{x_i} \left(\tilde{\phi}_i(x_i; y) \psi_{ij}(x_i, x_j) \prod_{k \in N(i) \backslash j} m_{ki}(x_i) \right) .$$





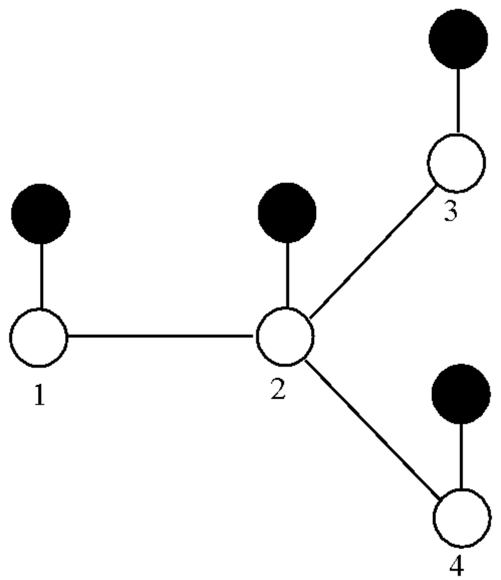
Example: BP with no cycles

- Consider this pair-wise MRF:



Example: BP with no cycles

- Consider this pair-wise MRF:

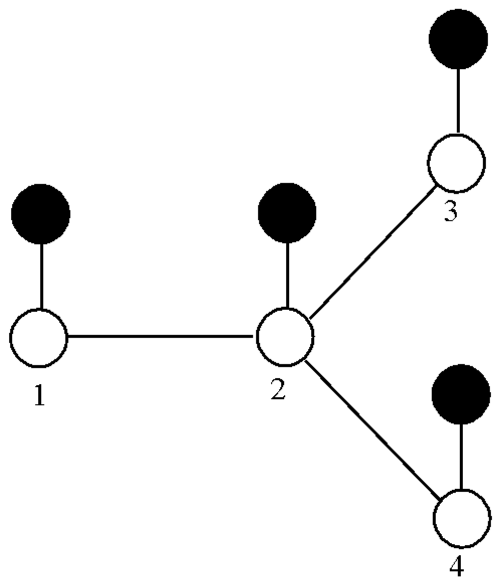


- Belief at node 1:

$$b_1(x_1) = \frac{1}{Z_1}\tilde{\phi}_1(x_1; y)m_{21}(x_1) \ ,$$

Example: BP with no cycles

- Consider this pair-wise MRF:



- Belief at node 1:

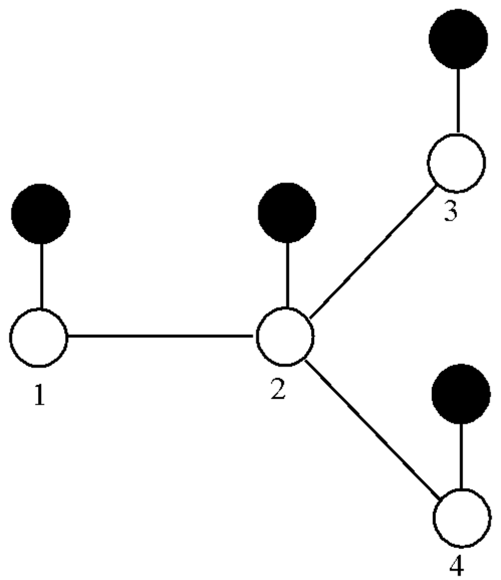
$$b_1(x_1) = \frac{1}{Z_1}\tilde{\phi}_1(x_1;y)m_{21}(x_1) \ ,$$

- Message-update rule for $m_{21}(x_1)$:

$$b_1(x_1) = \frac{1}{Z_1}\tilde{\phi}_1(x_1;y)\sum_{x_2}\psi_{12}(x_1,x_2)\tilde{\phi}_2(x_2;y)m_{32}(x_2)m_{42}(x_2) \ .$$

Example: BP with no cycles

- Consider this pair-wise MRF:



- Belief at node 1:

$$b_1(x_1) = \frac{1}{Z_1}\tilde{\phi}_1(x_1;y)m_{21}(x_1) \ ,$$

- Message-update rule for $m_{21}(x_1)$:

$$b_1(x_1) = \frac{1}{Z_1}\tilde{\phi}_1(x_1;y)\sum_{x_2}\psi_{12}(x_1,x_2)\tilde{\phi}_2(x_2;y)m_{32}(x_2)m_{42}(x_2) \ .$$

- Substituting $m_{32}, \ m_{42}$ yields

$$b_1(x_1) = \frac{1}{Z_1}\tilde{\phi}_1(x_1;y)\sum_{x_2}\tilde{\phi}_2(x_2;y)\psi_{12}(x_1,x_2)\sum_{x_3}\tilde{\phi}_3(x_3;y)\psi_{23}(x_2,x_3)\sum_{x_4}\tilde{\phi}_4(x_4;y)\psi_{24}(x_2,x_4) \ .$$

Example: BP with no cycles

- Q: What is

$$b_1(x_1) = \frac{1}{Z_1}\tilde{\phi}_1(x_1;y)\sum_{x_2}\tilde{\phi}_2(x_2;y)\psi_{12}(x_1,x_2)\sum_{x_3}\tilde{\phi}_3(x_3;y)\psi_{23}(x_2,x_3)\sum_{x_4}\tilde{\phi}_4(x_4;y)\psi_{24}(x_2,x_4)\ .$$

Example: BP with no cycles

- Q: What is

$$b_1(x_1) = \frac{1}{Z_1}\tilde{\phi}_1(x_1;y)\sum_{x_2}\tilde{\phi}_2(x_2;y)\psi_{12}(x_1,x_2)\sum_{x_3}\tilde{\phi}_3(x_3;y)\psi_{23}(x_2,x_3)\sum_{x_4}\tilde{\phi}_4(x_4;y)\psi_{24}(x_2,x_4)\ .$$

- It is the marginal probability of node 1:

$$b_1(x_1) = \frac{1}{Z_1}\sum_{x_2,x_3,x_4} p(x|y)$$

BP on simply-connected graphs

- This example illustrates the power of BP: expressing a global computation (marginalization) as a chain of local computations (messages).
- In this example, BP is exact. Only one message computation per node is sufficient.

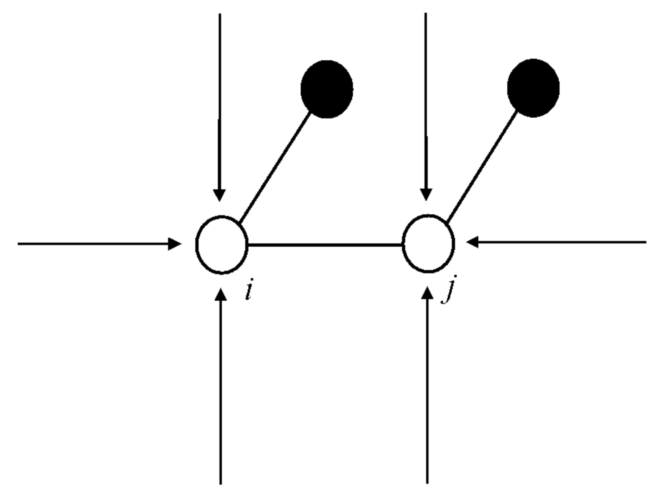
BP on simply-connected graphs

- This example illustrates the power of BP: expressing a global computation (marginalization) as a chain of local computations (messages).
- In this example, BP is exact. Only one message computation per node is sufficient.
- What happens in presence of loops?
- Let $$p_{ij}(x_i, x_j) := \sum_{z : z_i = x_i, z_j = x_j} p(z)$$

  denote the pairwise joint distribution of two neighboring sites.
- We can derive a similar message-passing algorithm for the pair-wise distribution.

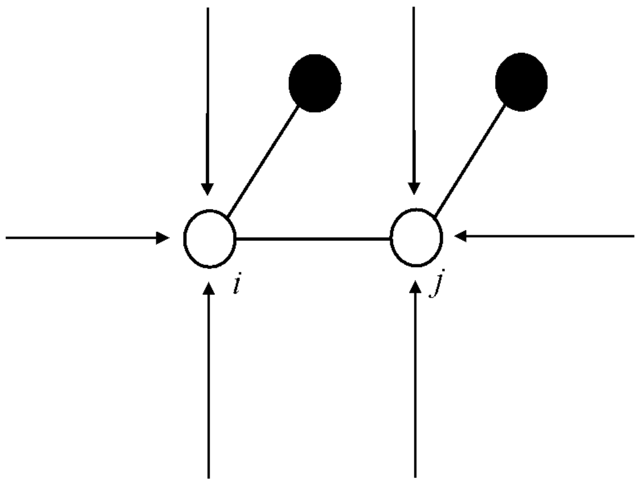
BP on simply-connected graphs

$$b_{ij}(x_i, x_j) = \frac{1}{Z_{ij}} \phi_i(x_i) \phi_j(x_j) \psi_{ij}(x_i, x_j) \prod_{k \in N(i) \setminus j} m_{ki}(x_i) \prod_{l \in N(j) \setminus i} m_{lj}(x_j) \ .$$



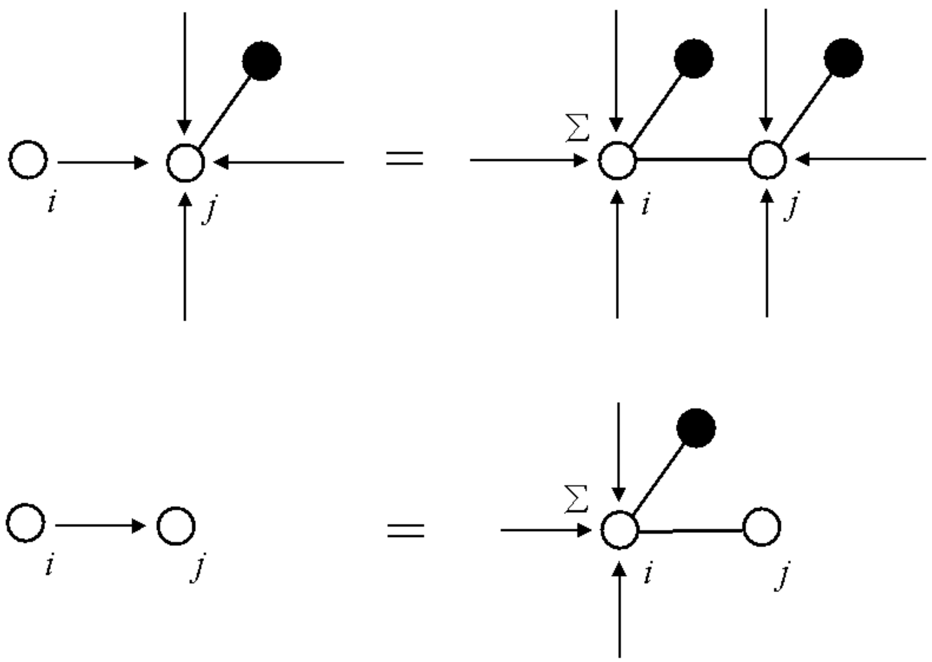
BP on simply-connected graphs

$$b_{ij}(x_i, x_j) = \frac{1}{Z_{ij}} \phi_i(x_i)\phi_j(x_j)\psi_{ij}(x_i, x_j) \prod_{k \in N(i)\backslash j} m_{ki}(x_i) \prod_{l \in N(j)\backslash i} m_{lj}(x_j) \ .$$



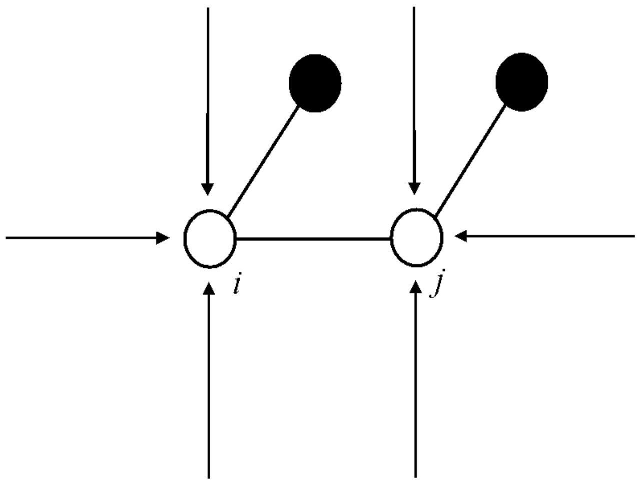
- We verify that

$$b_i(x_i) = \sum_{x_j} b_{ij}(x_i, x_j) \ .$$



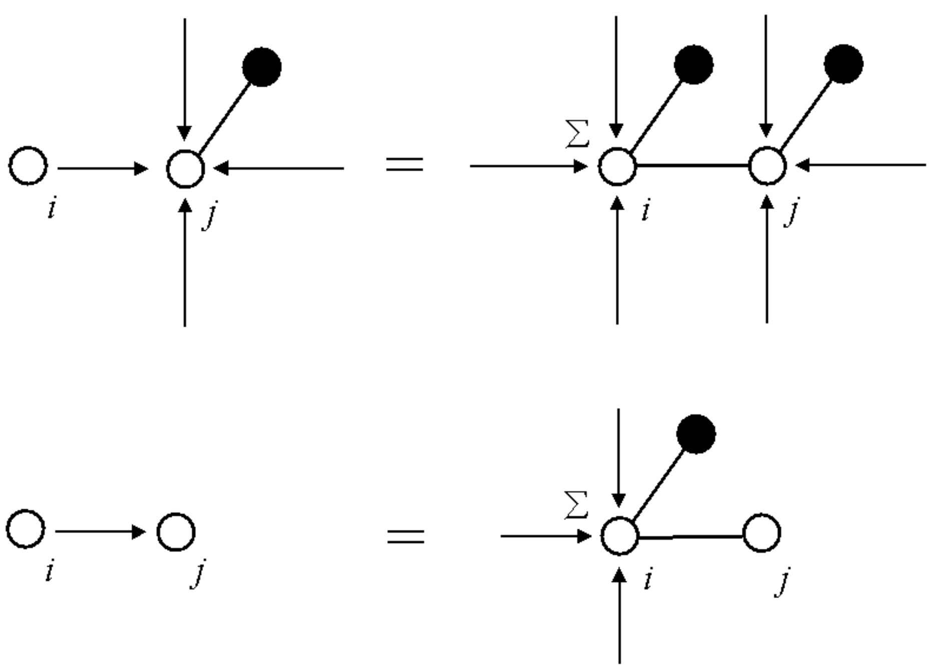
BP on simply-connected graphs

$$b_{ij}(x_i, x_j) = \frac{1}{Z_{ij}} \phi_i(x_i)\phi_j(x_j)\psi_{ij}(x_i, x_j) \prod_{k \in N(i)\backslash j} m_{ki}(x_i) \prod_{l \in N(j)\backslash i} m_{lj}(x_j) \ .$$



- We verify that

$$b_i(x_i) = \sum_{x_j} b_{ij}(x_i, x_j) \ .$$



- Thus

$$\forall\ i, j\ ,\ \sum_{x_i, x_j} b_{ij}(x_i, x_j) = \sum_{x_i} b_i(x_i) = 1 \ .$$

BP on general graphs

- The rules of computing messages do not rely on any topology of the graph.
- What happens if we apply it nonetheless?

BP on general graphs

- The rules of computing messages do not rely on any topology of the graph.
- What happens if we apply it nonetheless?

- For that, we initialize messages with prior distributions $m_{ij} \sim p_j^0$, and update them using

$$m_{ij}^{(n+1)}(x_j) \leftarrow \sum_{x_i} \left(\tilde{\phi}_i(x_i; y)\psi_{ij}(x_i, x_j) \prod_{k \in N(i) \backslash j} m_{ki}^{(n)}(x_i) \right) .$$

- Does it work?

BP on general graphs

- The rules of computing messages do not rely on any topology of the graph.
- What happens if we apply it nonetheless?

- For that, we initialize messages with prior distributions $m_{ij} \sim p_j^0$, and update them using

$$m_{ij}^{(n+1)}(x_j) \leftarrow \sum_{x_i} \left(\tilde{\phi}_i(x_i; y) \psi_{ij}(x_i, x_j) \prod_{k \in N(i) \setminus j} m_{ki}^{(n)}(x_i) \right) .$$

- Does it work?
  - In theory, no. One can build counter-examples where BP does not converge to the correct solution [Pearl, '88].
  - In practice, often it does work well: *Loopy BP*. Why?

BP and Free Energy

- Let $p(x)$ be the joint distribution defined by our pairwise MRF. Consider another joint distribution $q(x)$ defined over the same domain.

BP and Free Energy

- Let $p(x)$ be the joint distribution defined by our pairwise MRF. Consider another joint distribution $q(x)$ defined over the same domain.
- Assuming positive densities, we define a divergence

$$D_{KL}(q \mid\mid p) = \sum_x q(x) \log \frac{q(x)}{p(x)}$$

  - Kullback-Lieblier is not a distance (not symmetric and no triangle ineq.).
  - but non-negative:

BP and Free Energy

- Let $p(x)$ be the joint distribution defined by our pairwise MRF. Consider another joint distribution $q(x)$ defined over the same domain.
- Assuming positive densities, we define a divergence

$$D_{KL}(q \mid\mid p) = \sum_x q(x) \log \frac{q(x)}{p(x)}$$

  - Kullback-Lieblier is not a distance (not symmetric and no triangle ineq.).
  - but non-negative:

$$\begin{aligned} D_{KL}(q \mid\mid p) &= \mathbb{E}_{x\sim q} \log \frac{q}{p}(x) \\ &= -\mathbb{E}_{x\sim q} \log \frac{p}{q}(x) \\ &\geq -\log \mathbb{E}_{x\sim q} \frac{p}{q}(x) \\ &= 0 \ . \end{aligned}$$

BP and Free Energy

- If we write $p(x)$ as a Gibbs distribution with energy $E(x)$

$$p(x) = \frac{1}{Z} e^{-E(x)}$$

the Kullback-Liebler divergence becomes

$$D_{KL}(q||p) = \sum_x q(x) E(x) + \sum_x q(x) \log q(x) + \log Z \ (\ \geq 0 \ ) \ .$$

BP and Free Energy

- If we write $p(x)$ as a Gibbs distribution with energy $E(x)$

$$p(x) = \frac{1}{Z} e^{-E(x)}$$

the Kullback-Liebler divergence becomes

$$D_{KL}(q||p) = \sum_x q(x)E(x) + \sum_x q(x) \log q(x) + \log Z \quad (\;\geq 0 \;) \; .$$

- Zero divergence when

$$\sum_x q(x)E(x) + \sum_x q(x) \log q(x) := \underset{\text{avg.energy}}{U(q)} - \underset{\text{entropy}}{S(q)}$$

reaches *free energy value* $F = -\log Z$.

$$G(q) = U(q) - S(q)\text{: Gibbs free energy}$$

Mean-Field Free Energy

- Consider an approximation $q(x)$ with separable form:

$$q(x) = \prod_i q_i(x_i)$$

  - It is called *mean-field.* it does not explicitly model pairwise interactions.

Mean-Field Free Energy

- Consider an approximation $q(x)$ with separable form:

$$q(x) = \prod_i q_i(x_i)$$

  - It is called *mean-field* because it does not explicitly model pairwise interactions.
  - What is the Gibbs free energy of this model when $E(x)$ is a pair-wise MRF?

$$E(x) = -\sum_{i,j} \log \psi_{ij}(x_i, x_j) - \sum_i \log \phi_i(x_i) \ .$$

Mean-Field Free Energy

- Consider an approximation $q(x)$ with separable form:

$$q(x) = \prod_i q_i(x_i)$$

  - It is called *mean-field* because it does not explicitly model pairwise interactions.
  - What is the Gibbs free energy of this model when $E(x)$ is a pair-wise MRF?

$$E(x) = -\sum_{i,j} \log \psi_{ij}(x_i, x_j) - \sum_i \log \phi_i(x_i) \ .$$

  - Mean-field average Energy:

$$U(q) = -\sum_{(ij)} \sum_{x_i, x_j} q_i(x_i) q_j(x_j) \log \psi_{ij}(x_i, x_j) - \sum_i \sum_{x_i} q_i(x_i) \log \phi_i(x_i) \ .$$

$$S(q) = -\sum_i \sum_{x_i} q_i(x_i) \log q_i(x_i) \ .$$

Mean Field Free Energy

- Mean-field approximation: Minimize Gibbs Free Energy $q(x)$.
- *Variational Inference* (later in course) exploits such mean-field approximations over specific parametric families.
- The mean field model corresponds to one-node beliefs

$$q_i(x_i) \; \leftrightarrow \; b_i(x_i)$$

Mean Field Free Energy

- Mean-field approximation: Minimize Gibbs Free Energy $q(x)$.
- *Variational Inference* (later in course) exploits such mean-field approximations over specific parametric families.
- The mean field model corresponds to one-node beliefs

$$q_i(x_i) \leftrightarrow b_i(x_i)$$

- What about a two-node belief model?

Bethe Free Energy

- Let us construct a mean-field approximation that contains unary and pair-wise beliefs: b_i, b_{ij}

$$\forall\ i, j\ \ , \quad \sum_{x_i} b_i(x_i) = \sum_{x_i, x_j} b_{ij}(x_i, x_j) = 1\ .$$

$$\forall\ i, j\ \ , \quad \sum_{x_j} b_{ij}(x_i, x_j) = b_i(x_i)\ .$$

Bethe Free Energy

- Let us construct a mean-field approximation that contains unary and pair-wise beliefs: b_i, b_{ij}

$$\forall\ i,j\ ,\quad \sum_{x_i} b_i(x_i) = \sum_{x_i,x_j} b_{ij}(x_i,x_j) = 1\ .$$

$$\forall\ i,j\ ,\quad \sum_{x_j} b_{ij}(x_i,x_j) = b_i(x_i)\ .$$

- Under this approximation, the average energy is

$$U = -\sum_{ij}\sum_{x_i,x_j} b_{ij}(x_i,x_j)\log\psi_{ij}(x_i,x_j) - \sum_{i}\sum_{x_i} b_i(x_i)\log\phi_i(x_i)\ .$$

- Important observation: since $p(x)$ is a pair-wise MRF, its average energy has the previous form, and is exact (reaches global minima of free energy).

Bethe Free Energy

- The Entropy of a pairwise MRF does not have closed-form expression for general graphs, but for simply connected graphs we have

$$b(x) = \frac{\prod_{(ij)} b_{ij}(x_i, x_j)}{\prod_i b_i(x_i)^{d_i - 1}} \ .$$

d_i: degree of node i

Bethe Free Energy

- The Entropy of a pairwise MRF does not have closed-form expression for general graphs, but for simply connected graphs we have

$$b(x) = \frac{\prod_{(ij)} b_{ij}(x_i, x_j)}{\prod_i b_i(x_i)^{d_i - 1}} \; .$$

d_i: degree of node i

- It follows that

$$H_{\text{Bethe}} = -\sum_{(ij)} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \log b_{ij}(x_i, x_j) + \sum_i (d_i - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i) \; .$$

Bethe Free Energy

- The Entropy of a pairwise MRF does not have closed-form expression for general graphs, but for simply connected graphs we have

$$b(x) = \frac{\prod_{(ij)} b_{ij}(x_i, x_j)}{\prod_i b_i(x_i)^{d_i - 1}} \; .$$

d_i: degree of node i

- It follows that

$$H_{\text{Bethe}} = -\sum_{(ij)} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \log b_{ij}(x_i, x_j) + \sum_i (d_i - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i) \; .$$

- Thus minimizer of Bethe free energy $G_{\text{Bethe}} = U - H_{\text{Bethe}}$ contains the true Gibbs distribution $p(x)$ (recall

$$D_{KL}(q||p) = 0 \Leftrightarrow q = p \; .$$

Bethe Free Energy

- Bethe free energy: $G_{\text{Bethe}} = U - H_{\text{Bethe}}$

$$U = -\sum_{ij}\sum_{x_i,x_j} b_{ij}(x_i,x_j)\log\psi_{ij}(x_i,x_j) - \sum_i\sum_{x_i} b_i(x_i)\log\phi_i(x_i)\ .$$

$$H_{\text{Bethe}} = -\sum_{(ij)}\sum_{x_i,x_j} b_{ij}(x_i,x_j)\log b_{ij}(x_i,x_j) + \sum_i (d_i - 1)\sum_{x_i} b_i(x_i)\log b_i(x_i)\ .$$

- On simply connected graphs, BP beliefs are global minima of the Bethe free energy.

Bethe Free Energy

- Bethe free energy: $G_{\text{Bethe}} = U - H_{\text{Bethe}}$

$$U = -\sum_{ij} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \log \psi_{ij}(x_i, x_j) - \sum_i \sum_{x_i} b_i(x_i) \log \phi_i(x_i) \ .$$

$$H_{\text{Bethe}} = -\sum_{(ij)} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \log b_{ij}(x_i, x_j) + \sum_i (d_i - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i) \ .$$

- On simply connected graphs, BP beliefs are global minima of the Bethe free energy.

- On general graphs, the Bethe Free Energy does not satisfy

$$G_{\text{Bethe}} \geq -\log Z$$

Bethe Free Energy

- Bethe free energy: $G_{\text{Bethe}} = U - H_{\text{Bethe}}$

$$U = -\sum_{ij}\sum_{x_i,x_j} b_{ij}(x_i,x_j)\log\psi_{ij}(x_i,x_j) - \sum_i\sum_{x_i} b_i(x_i)\log\phi_i(x_i) \ .$$

$$H_{\text{Bethe}} = -\sum_{(ij)}\sum_{x_i,x_j} b_{ij}(x_i,x_j)\log b_{ij}(x_i,x_j) + \sum_i (d_i - 1)\sum_{x_i} b_i(x_i)\log b_i(x_i) \ .$$

- On simply connected graphs, BP beliefs are global minima of the Bethe free energy.

- On general graphs, the Bethe Free Energy does not satisfy

$$G_{\text{Bethe}} \geq -\log Z$$

- However, they provide a powerful characterization of BP solutions:

A set of beliefs gives BP a fixed point in any graph G if and only if they are stationary points of the Bethe free energy.

Bethe Free Energy

- We construct a Lagrangian $\mathcal{L}(b)$ corresponding to the constraints

$$\forall\ i, j, x_i\ ,\ b_i(x_i) = \sum_{x_j} b_{ij}(x_i, x_j) \rightarrow\ \lambda_{ij}(x_i)$$

$$\forall\ i, j\ ,\ \sum_{x_i} \sum_{x_j} b_{ij}(x_i, x_j) = 1 \rightarrow\ \gamma_{ij}$$

$$\forall\ i,\ \sum_{x_i} b_i(x_i) = 1 \rightarrow\ \gamma_i$$

Bethe Free Energy

- We construct a Lagrangian $\mathcal{L}(b)$ corresponding to the constraints

$$\forall\ i, j, x_i\ ,\ b_i(x_i) = \sum_{x_j} b_{ij}(x_i, x_j) \rightarrow\ \lambda_{ij}(x_i)$$

$$\forall\ i, j\ ,\ \sum_{x_i}\sum_{x_j} b_{ij}(x_i, x_j) = 1 \rightarrow\ \gamma_{ij}$$

$$\forall\ i,\ \sum_{x_i} b_i(x_i) = 1 \rightarrow\ \gamma_i$$

- From $\frac{\partial \mathcal{L}(b)}{\partial b_{ij}(x_i, x_j)} = 0$ $\frac{\partial \mathcal{L}(b)}{\partial b_i(x_i)} = 0$, stationary points of BFE satisfy

$$\log b_{ij}(x_i, x_j) = \log \psi_{ij}(x_i, x_j) + \log \phi_i(x_i) + \log \phi_j(x_j) + \lambda_{ij}(x_j) + \lambda_{ji}(x_i) + \gamma_{ij} - 1$$

$$(d_i - 1)(\log b_i(x_i) + 1) = -(1 - d_i) \log \phi_i(x_i) + \sum_{j \in N(i)} \lambda_{ji}(x_i) + \gamma_i$$

Bethe Free Energy and BP

$$\log b_{ij}(x_i, x_j) = \log \psi_{ij}(x_i, x_j) + \log \phi_i(x_i) + \log \phi_j(x_j) + \lambda_{ij}(x_j) + \lambda_{ji}(x_i) + \gamma_{ij} - 1$$

$$(d_i - 1)(\log b_i(x_i) + 1) = -(1 - d_i) \log \phi_i(x_i) + \sum_{j \in N(i)} \lambda_{ji}(x_i) + \gamma_i$$

- Now, if we suppose messages/beliefs that are fixed point of BP, we define $\lambda_{ij}(x_j) = \log \prod_{k \in N(j) \setminus i} m_{kj}(x_j)$

Bethe Free Energy and BP

$$\log b_{ij}(x_i, x_j) = \log \psi_{ij}(x_i, x_j) + \log \phi_i(x_i) + \log \phi_j(x_j) + \lambda_{ij}(x_j) + \lambda_{ji}(x_i) + \gamma_{ij} - 1$$

$$(d_i - 1)(\log b_i(x_i) + 1) = -(1 - d_i) \log \phi_i(x_i) + \sum_{j \in N(i)} \lambda_{ji}(x_i) + \gamma_i$$

- Now, if we suppose messages/beliefs that are fixed point of BP, we define $\lambda_{ij}(x_j) = \log \prod_{k \in N(j) \setminus i} m_{kj}(x_j)$
- These multipliers satisfy the optimality KKT conditions of Lagrange multipliers, so

$$\text{Lagrange multipliers } \lambda_{ij}(x_j) \text{ of Bethe Free energy}$$

$$\updownarrow$$

$$\text{Messages } m_{ij}(x_j) \text{ of BP algorithm}$$

- This is a first hint of a major tool: characterize inference as solutions of optimization problems: **variational inference.**

Max-Product

- We have described an algorithm to estimate marginal (and conditional) distributions.
- How about inference tasks of the form $\arg\max_x p(x \mid y)$?
  - I.e. Maximum-a-posteriori inference.

Max-Product

- We have described an algorithm to estimate marginal (and conditional) distributions.
- How about inference tasks of the form $\arg\max_x p(x \mid y)$?
  - I.e. Maximum-a-posteriori inference.

- A simple variant is the *max-product algorithm,* used to estimate the state configuration with maximum probability.
- Marginalization:

$$m_{ij}^{(n+1)}(x_j) \leftarrow \sum_{x_i} \left(\tilde{\phi}_i(x_i; y) \psi_{ij}(x_i, x_j) \prod_{k \in N(i) \setminus j} m_{ki}^{(n)}(x_i) \right) .$$

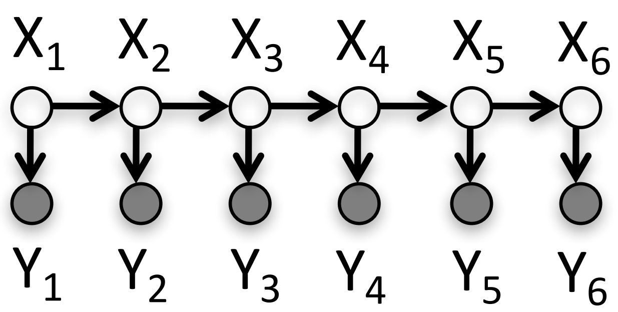
- Maximization:

$$m_{ij}^{(n+1)}(x_j) \leftarrow \max_{x_i} \left(\tilde{\phi}_i(x_i; y) \psi_{ij}(x_i, x_j) \prod_{k \in N(i) \setminus j} m_{ki}^{(n)}(x_i) \right) .$$

Marginal inference in HMMs

- "Filtering" problem is to do marginal inference to find:

$$\Pr(x_n \mid y_1, \ldots, y_n)$$



- How does one **compute** this?

- Applying rule of conditional probability, we have:

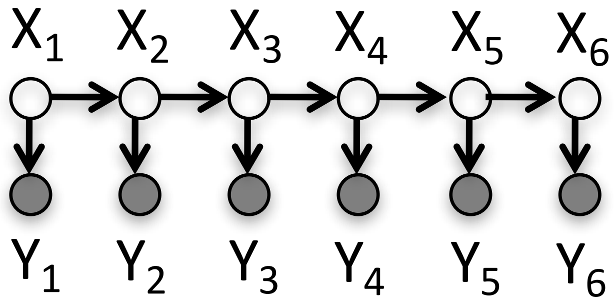
$$\Pr(x_n \mid y_1, \ldots, y_n) = \frac{\Pr(x_n, y_1, \ldots, y_n)}{\Pr(y_1, \ldots, y_n)}$$

- Naively, would seem to require k^{n-1} summations,

$$\Pr(x_n, y_1, \ldots, y_n) = \sum_{x_1, \ldots, x_{n-1}} \Pr(x_1, \ldots, x_n, y_1, \ldots, y_n)$$

Is there a more efficient algorithm?

Marginal inference in HMMs:



- Use **dynamic programming**

$\Pr(A=a) = \sum_b \Pr(B=b, A=a)$

$$\Pr(x_n, y_1, \ldots, y_n) = \sum_{x_{n-1}} \Pr(x_{n-1}, x_n, y_1, \ldots, y_n)$$

$\Pr(\vec{A}=\vec{a}, \vec{B}=\vec{b}) = \Pr(\vec{A}=\vec{a}) \Pr(\vec{B}=\vec{b} \mid \vec{A}=\vec{a})$

$$= \sum_{x_{n-1}} \Pr(x_{n-1}, y_1, \ldots, y_{n-1}) \Pr(x_n, y_n \mid x_{n-1}, y_1, \ldots, y_{n-1})$$

Conditional independence in HMMs

$$= \sum_{x_{n-1}} \Pr(x_{n-1}, y_1, \ldots, y_{n-1}) \Pr(x_n, y_n \mid x_{n-1})$$

$\Pr(A=a, B=b) = \Pr(A=a) \Pr(B=b \mid A=a)$

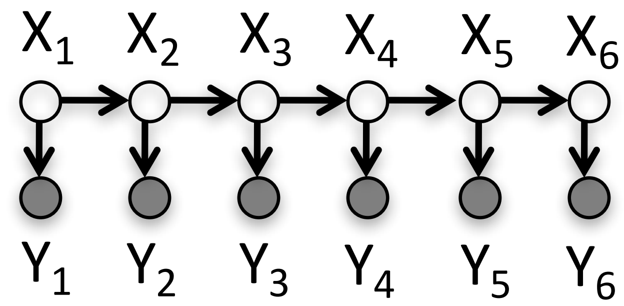
$$= \sum_{x_{n-1}} \Pr(x_{n-1}, y_1, \ldots, y_{n-1}) \Pr(x_n \mid x_{n-1}) \Pr(y_n \mid x_n, x_{n-1})$$

Conditional independence in HMMs

$$= \sum_{x_{n-1}} \Pr(x_{n-1}, y_1, \ldots, y_{n-1}) \Pr(x_n \mid x_{n-1}) \Pr(y_n \mid x_n)$$

- For n=1, initialize $\Pr(x_1, y_1) = \Pr(x_1) \Pr(y_1 \mid x_1)$
- Total running time is $O(nk^2)$ – linear time! Easy to do **filtering**

Marginal Inference in MRF



- This is a simply connected graph:
- Thus we can apply the BP algorithm:

$$\Pr(x_n \;,\; y) = b_n(x_n)$$

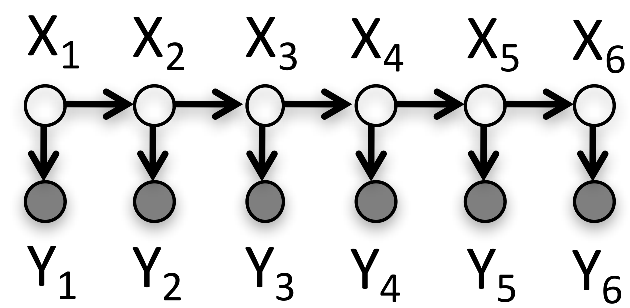
$$b_n(x_n) = \frac{1}{Z_n} \Pr(y_n \mid x_n) m_{n-1,n}(x_n) \;.$$

$$m_{n-1,n}(x_n) = \sum_{x_{n-1}} \boxed{\Pr(y_{n-1} \mid x_{n-1})} \boxed{\Pr(x_n \mid x_{n-1})} m_{n-2,n-1}(x_{n-1}) \;.$$

$\Pr(y_{n-1} \mid x_{n-1})$ → $\phi_{n-1}(x_{n-1}, y_{n-1})$

$\Pr(x_n \mid x_{n-1})$ → $\psi_{n,n-1}(x_n, x_{n-1})$

MAP inference in HMMs:



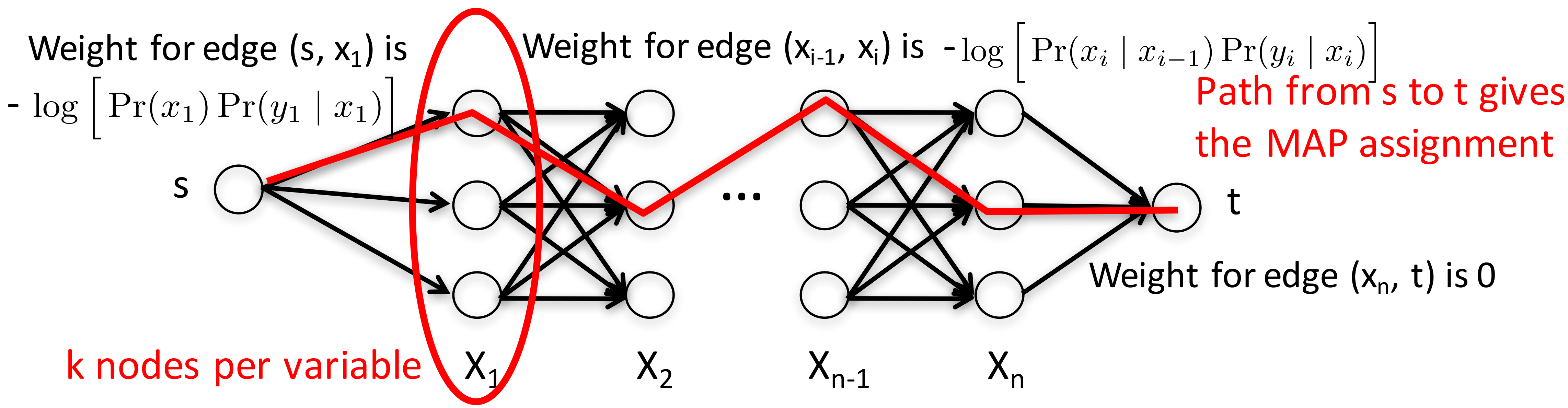
- MAP inference in HMMs can be solved in linear time!

$$\arg\max_{\mathbf{x}} \Pr(x_1, \ldots x_n \mid y_1, \ldots, y_n) = \arg\max_{\mathbf{x}} \Pr(x_1, \ldots x_n, y_1, \ldots, y_n)$$

$$= \arg\max_{\mathbf{x}} \log \Pr(x_1, \ldots x_n, y_1, \ldots, y_n)$$

$$= \arg\max_{\mathbf{x}} \quad \log \Big[\Pr(x_1) \Pr(y_1 \mid x_1) \Big] + \sum_{i=2}^{n} \log \Big[\Pr(x_i \mid x_{i-1}) \Pr(y_i \mid x_i) \Big]$$

- Formulate as a shortest paths problem



Called the Viterbi algorithm

Monte-Carlo Estimation

- BP is an instance of optimization-based inference.
- Let's focus on marginal inference:

$$p(x_i) = \sum_{j \neq i} \sum_{x_j} p(x_1, \ldots, x_n) \ .$$

- This object can be written as an expectation:

$$p(x_i) = \mathbb{E}_{X \sim p} f_{i,x_i}(X) \ , \ f_{i,x_i}(X) = \mathbf{1}(X_i = x_i) \ .$$

- Thus, another route to approximate inference is by replacing this expectation with iid samples:

$$x^1, \ldots, x^M \sim p(X) \text{ iid}$$

$$\hat{p}(x_i) = \frac{1}{M} \sum_{m=1}^{M} f_{i,x_i}(x^m) \ .$$

Monte-Carlo Estimation

- Thus, provided we can (efficiently) sample from the model, we can estimate any quantity that depends smoothly on the density.
- What is the quality of such estimate?

- Bias?

$$\mathbb{E}_{x^1 \dots x^M \sim p} \left[\hat{p}(x_i) \right] = \frac{1}{M} \sum_{m=1}^{M} \mathbb{E}_{x^m \sim p} f_{i,x_i}(x^m) \ . = \mathbb{E} f_i(x) = p(x_i)$$

- Variance?
  - Law of large numbers: $\hat{p}(x_i) \overset{a.s.}{\to} p(x_i) \ , \ (m \to \infty) \ .$
  - CLT:Under mild assumptions, $\sqrt{m}(\hat{p}(x_i) - p(x_i)) \ \overset{d}{\to} \ \mathcal{N}(0,1) \ .$

Monte-Carlo Estimation

- But, how do we sample from a graphical model?
  - If it is a BN, we saw in the first lecture that it lends itself to sampling by following topological order.
  - But how about undirected graphical models?

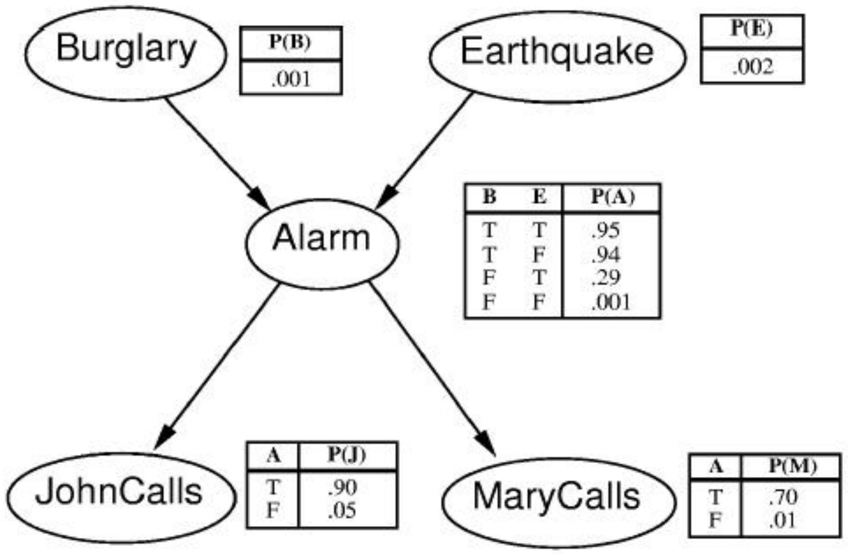
Gibbs Sampling

- Gibbs Sampling is an iterative algorithm that produces samples from undirected models.

- Suppose the model contains variables $x_1 \ldots x_n$
- Initialize starting values (e.g from uniform distribution)
- Do until (convergence):
  - Pick an ordering of the variables
  - For each x_i,
    - ❖ Sample $p(x_i \mid X_j = x_j)\, ,\, j \neq i\ .$
    - ❖ update x_i

- Recall that we only need to condition on the Markov Blanket.

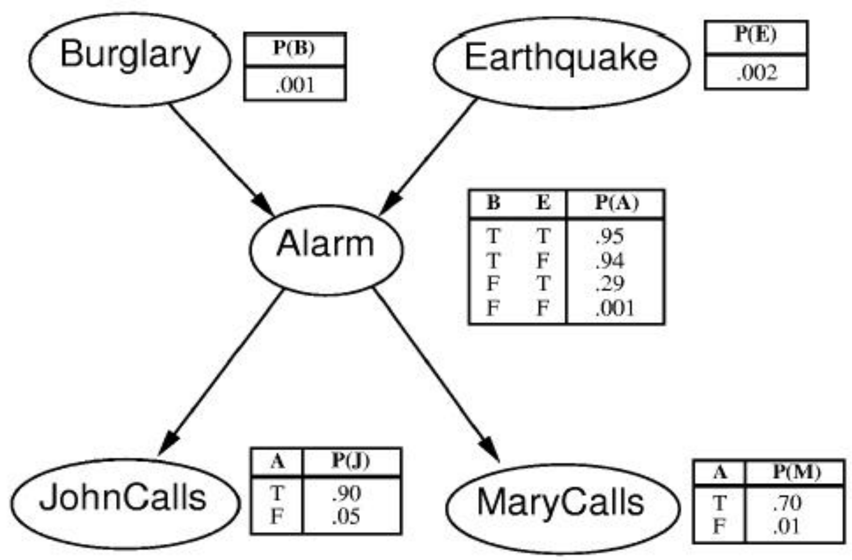
Gibbs Sampling: An Example



t	B	E	A	J	M
0	F	F	F	F	F
1					
2					
3					
4					

- Consider the alarm network
  - Assume we sample variables in the order B,E,A,J,M
  - Initialize all variables at t = 0 to False

Gibbs Sampling: An Example



t	B	E	A	J	M
0	F	F	F	F	F
1	F				
2					
3					
4					

- Sampling P(B|A,E) at t = 1: Using Bayes Rule,

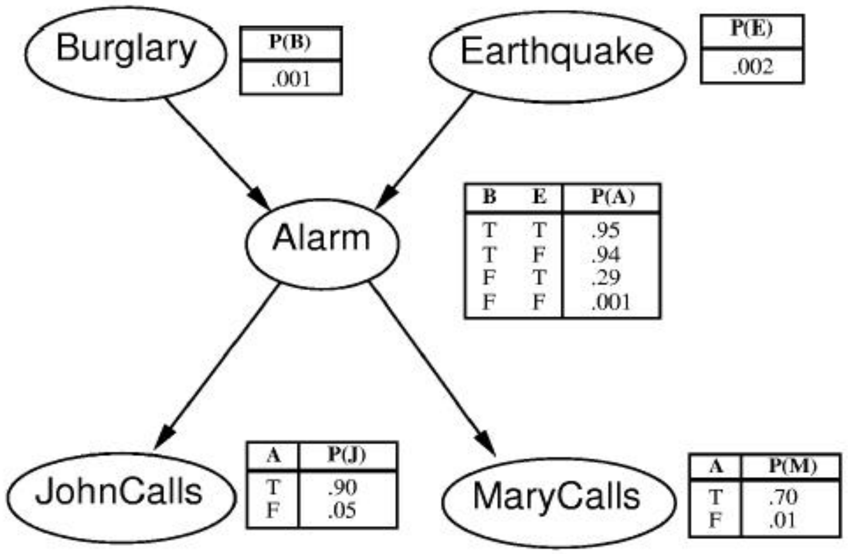
$$P(B \mid A,E) \propto P(A \mid B,E)P(B)$$

- A=false, E=false, so we compute:

$$P(B=T \mid A=F, E=F) \propto (0.06)(0.01) = 0.0006$$

$$P(B=F \mid A=F, E=F) \propto (0.999)(0.999) = 0.9980$$

Gibbs Sampling: An Example



t	B	E	A	J	M
0	F	F	F	F	F
1	F	T			
2					
3					
4					

- Sampling P(E|A,B): Using Bayes Rule,

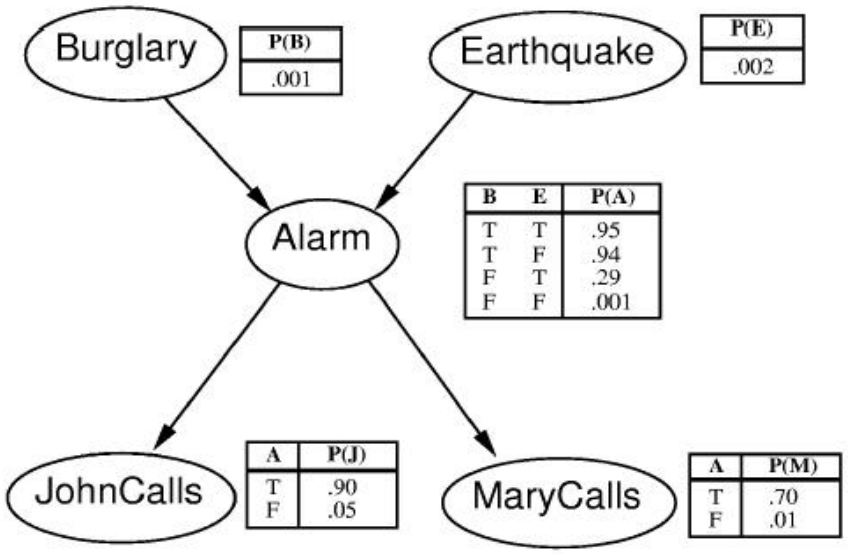
$$P(E \mid A, B) \propto P(A \mid B, E)P(E)$$

- (A,B) = (F,F), so we compute the following,

$$P(E = T \mid A = F, B = F) \propto (0.71)(0.02) = 0.0142$$

$$P(E = F \mid A = F, B = F) \propto (0.999)(0.998) = 0.9970$$

Gibbs Sampling: An Example



t	B	E	A	J	M
0	F	F	F	F	F
1	F	T	F		
2					
3					
4					

- Sampling P(A|B,E,J,M): Using Bayes Rule,

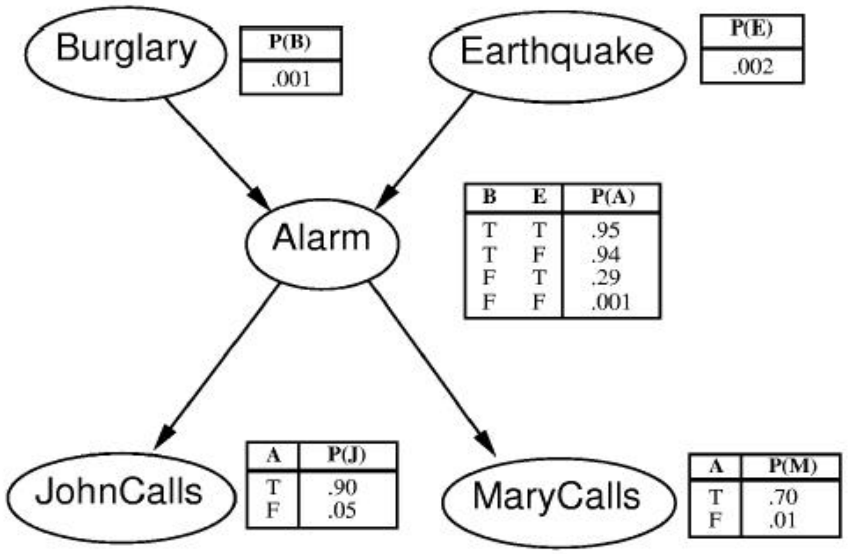
$$P(A \mid B,E,J,M) \propto P(J \mid A)P(M \mid A)P(A \mid B,E)$$

- (B,E,J,M) = (F,T,F,F), so we compute:

$$P(A=T \mid B=F,E=T,J=F,M=F) \propto (0.1)(0.3)(0.29) = 0.0087$$

$$P(A=F \mid B=F,E=T,J=F,M=F) \propto (0.95)(0.99)(0.71) = 0.6678$$

Gibbs Sampling: An Example



t	B	E	A	J	M
0	F	F	F	F	F
1	F	T	F	T	
2					
3					
4					

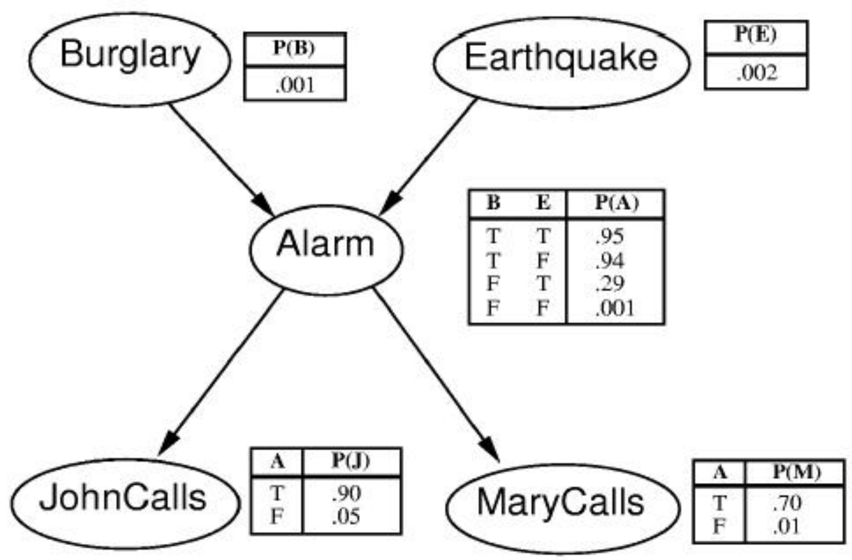
- Sampling P(J|A): No need to apply Bayes Rule

- A = F, so we compute the following, and sample

$$P(J = T \mid A = F) \propto 0.05$$

$$P(J = F \mid A = F) \propto 0.95$$

Gibbs Sampling: An Example



t	B	E	A	J	M
0	F	F	F	F	F
1	F	T	F	T	F
2					
3					
4					

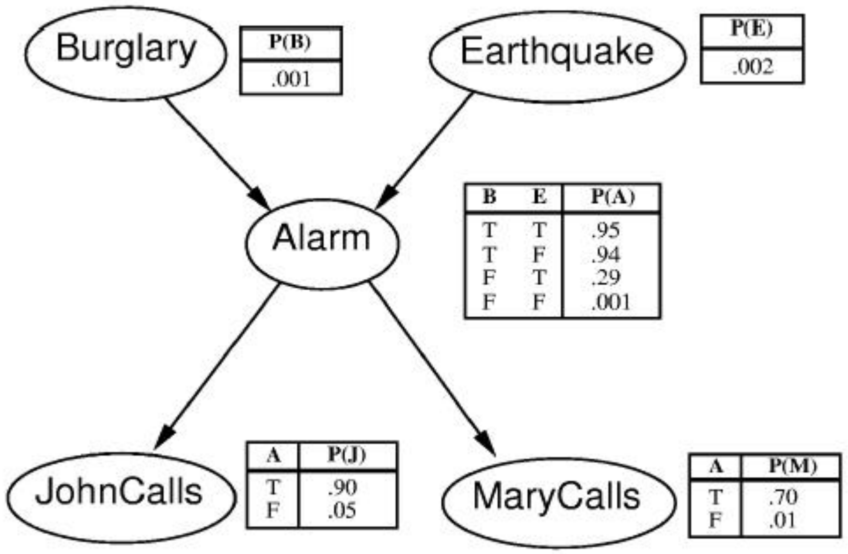
- Sampling P(M|A): No need to apply Bayes Rule

- A = F, so we compute the following, and sample

$$P(M = T \mid A = F) \propto 0.01$$

$$P(M = F \mid A = F) \propto 0.99$$

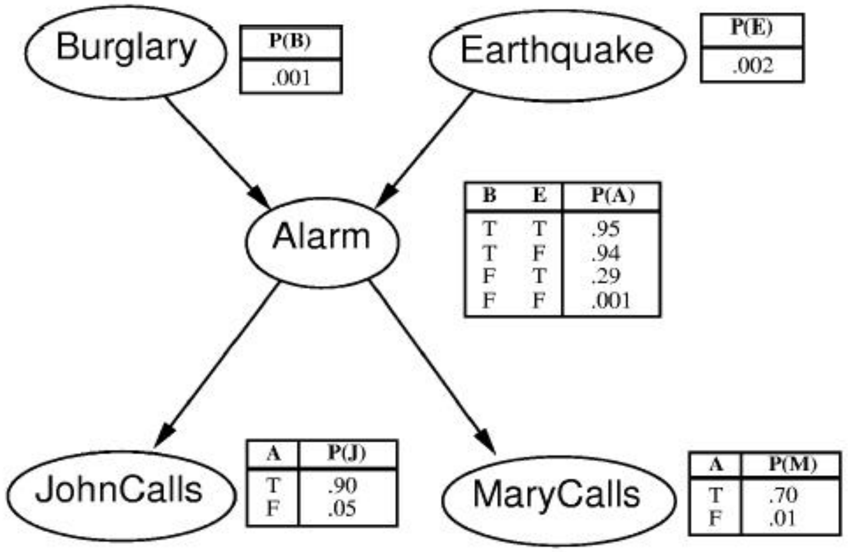
Gibbs Sampling: An Example



t	B	E	A	J	M
0	F	F	F	F	F
1	F	T	F	T	F
2	F	T	T	T	T
3					
4					

- Now t = 2, and we repeat the procedure to sample new values of B,E,A,J,M …

Gibbs Sampling: An Example



t	B	E	A	J	M
0	F	F	F	F	F
1	F	T	F	T	F
2	F	T	T	T	T
3	T	F	T	F	T
4	T	F	T	F	F

- Now t = 2, and we repeat the procedure to sample new values of B,E,A,J,M …
- And similarly for t = 3, 4, etc.

Gibbs Sampling and Markov Chains

- This algorithm is an instance of a broad family of tools: MCMC
- We will study in future lecture the main properties and uses of general MCMC methods.