

# Probabilistic Graphical Models

Weinan Zhang

Shanghai Jiao Tong University

<http://wnzhang.net>

<http://wnzhang.net/teaching/cs420/index.html>

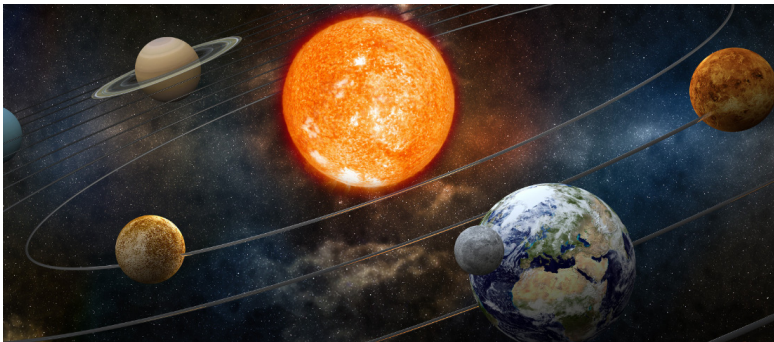
# Content of This Lecture

- Introduction
- Bayes Networks (Directed Graphs)
- Markov Networks (Undirected Graphs)
- Inferences in Graphical Models

# Review: Data Science

- Physics

- **Goal**: discover the underlying principle of the world



- **Solution**: build the model of the world from observations

$$F = G \frac{m_1 m_2}{r^2}$$

- Data Science

- **Goal**: discover the underlying principle of the data



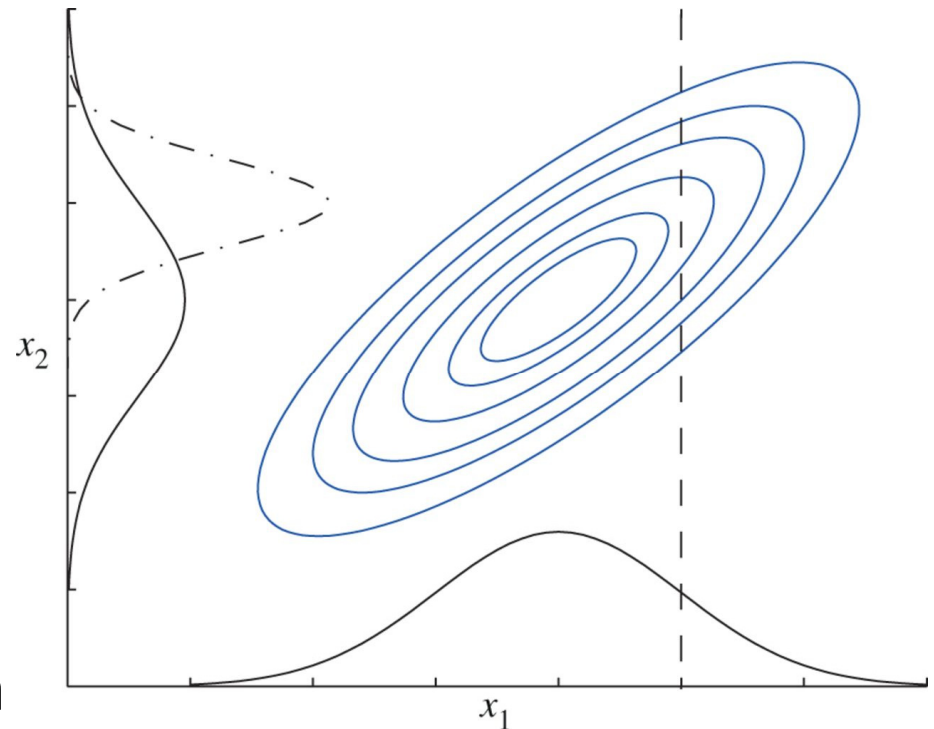
- **Solution**: build the model of the data from observations

$$p(x) = \frac{e^{f(x)}}{\sum_{x'} e^{f(x')}}$$

# Data Science

- Mathematically
  - Find joint data distribution  $p(x)$
  - Then the conditional distribution  $p(x_2|x_1)$
- E.g., Gaussian distribution
  - Multivariate

$$p(x) = \frac{e^{-(x-\mu)^\top \Sigma^{-1}(x-\mu)}}{\sqrt{|2\pi\Sigma|}}$$



- Univariate

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

# A Simple Example in User Behavior Modeling

| Interest | Gender | Age | BBC Sports | PubMed | Bloomberg Business | Spotify |
|----------|--------|-----|------------|--------|--------------------|---------|
| Finance  | Male   | 29  | Yes        | No     | Yes                | No      |
| Sports   | Male   | 21  | Yes        | No     | No                 | Yes     |
| Medicine | Female | 32  | No         | Yes    | No                 | No      |
| Music    | Female | 25  | No         | No     | No                 | Yes     |
| Medicine | Male   | 40  | Yes        | Yes    | Yes                | No      |

- Joint data distribution

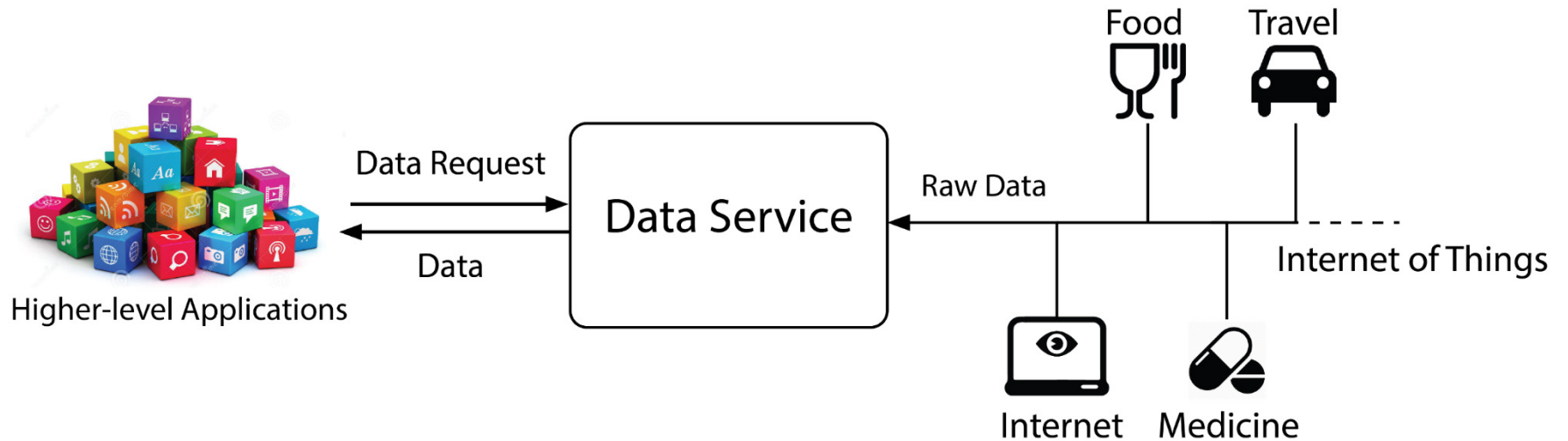
$p(\text{Interest}=\text{Finance}, \text{Gender}=\text{Male}, \text{Age}=29, \text{Browsing}=\text{BBC Sports}, \text{Bloomberg Business})$

- Conditional data distribution

$p(\text{Interest}=\text{Finance} \mid \text{Browsing}=\text{BBC Sports}, \text{Bloomberg Business})$

$p(\text{Gender}=\text{Male} \mid \text{Browsing}=\text{BBC Sports}, \text{Bloomberg Business})$

# Data Technology



# Key Problem of Data Science

- How to build the data model?
- Specifically, how to model the joint data distribution  $p(x)$  ?
  - For example, the data of temperature and people's cloth

| Temperature | Cloth | Probability |
|-------------|-------|-------------|
| Hot         | Shirt | 48%         |
| Hot         | Coat  | 12%         |
| Cold        | Shirt | 8%          |
| Cold        | Coat  | 32%         |

# Data Probabilistic Modeling

| Temperature | Cloth | Probability |
|-------------|-------|-------------|
| Hot         | Shirt | 48%         |
| Hot         | Coat  | 12%         |
| Cold        | Shirt | 8%          |
| Cold        | Coat  | 32%         |

- From the table, we can directly build a joint distribution model

$P(\text{temperature}=\text{hot}, \text{cloth}=\text{shirt}) = 48\%$

$P(\text{temperature}=\text{hot}, \text{cloth}=\text{coat}) = 12\%$

$P(\text{temperature}=\text{cold}, \text{cloth}=\text{shirt}) = 8\%$

$P(\text{temperature}=\text{cold}, \text{cloth}=\text{coat}) = 32\%$

- to estimate and maintain  $2 \times 2 = 4$  probabilities



# Data Probabilistic Modeling

- What if we have a high dimensional data

| Temperature | Cloth | Gender | Weekday  | Probability |
|-------------|-------|--------|----------|-------------|
| Hot         | Shirt | Male   | Monday   | 2.4%        |
| Hot         | Coat  | Female | Friday   | 1.2%        |
| Cold        | Shirt | Female | Sunday   | 3.8%        |
| Cold        | Coat  | Male   | Thursday | 3.1%        |

...

- Directly build a joint distribution model to estimate and maintain  $2 \times 2 \times 2 \times 7 = 56$  probabilities
  - Exponential complexity
- We should find a better way to model the data distribution

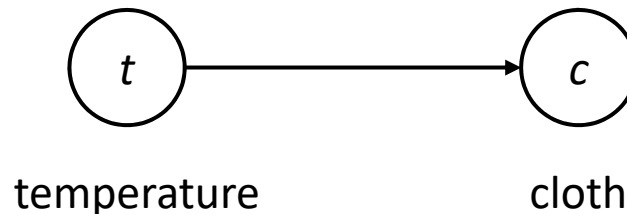
# Domain Knowledge

| Temperature | Cloth | Probability |
|-------------|-------|-------------|
| Hot         | Shirt | 48%         |
| Hot         | Coat  | 12%         |
| Cold        | Shirt | 8%          |
| Cold        | Coat  | 32%         |

- Build data dependency with domain knowledge
  - People choose clothes according to the temperature
  - Thus the cloth variable depends on the temperature variable

$$p(t, c) = p(t)p(c|t)$$

$P(\text{temperature}=\text{hot}) = 60\%$   
 $P(\text{temperature}=\text{cold}) = 40\%$

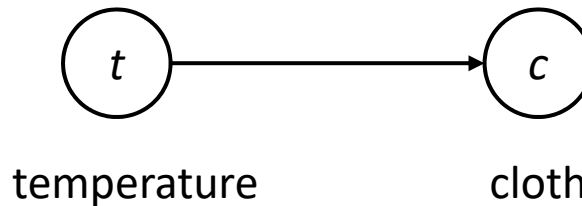


$P(\text{cloth}=\text{shirt} | \text{temperature}=\text{hot}) = 80\%$   
 $P(\text{cloth}=\text{coat} | \text{temperature}=\text{hot}) = 20\%$   
 $P(\text{cloth}=\text{shirt} | \text{temperature}=\text{cold}) = 20\%$   
 $P(\text{cloth}=\text{coat} | \text{temperature}=\text{cold}) = 80\%$

# Graphical Model

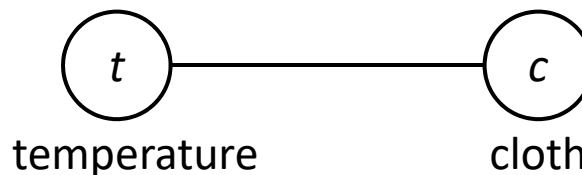
- Graphical model is a methodology to formulate such data dependency from any domain knowledge
  - Bayesian network (directed graphs)

$$p(t, c) = p(t)p(c|t)$$



- Markov network (undirected graphs)

$$p(t, c) = \frac{e^{\phi(t, c)}}{\sum_{t', c'} e^{\phi(t', c')}} e^{\phi(t, c)}$$



# Content of This Lecture

- Introduction
- Bayes Networks (Directed Graphs)
- Markov Networks (Undirected Graphs)
- Inferences in Graphical Models

# A Simple Bayesian Network

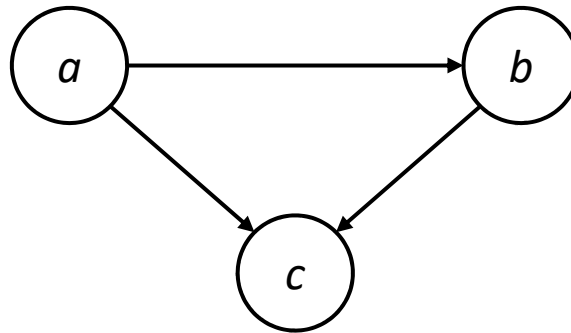
- Consider an arbitrary joint distribution  $p(a, b, c)$
- One may apply the product rule of probability

$$p(a, b, c) = p(c|a, b)p(a, b)$$

Symmetrical  
w.r.t.  $a, b$  and  $c$

$$= p(c|a, b)p(b|a)p(a)$$

Asymmetrical  
w.r.t.  $a, b$  and  $c$



- One of the powerful aspects of graphical models is that a specific graph can make probabilistic statements for a broad class of distributions
- We say this graph is **fully connected** if there is a link between every pair of nodes

# A More Complex Bayesian Network

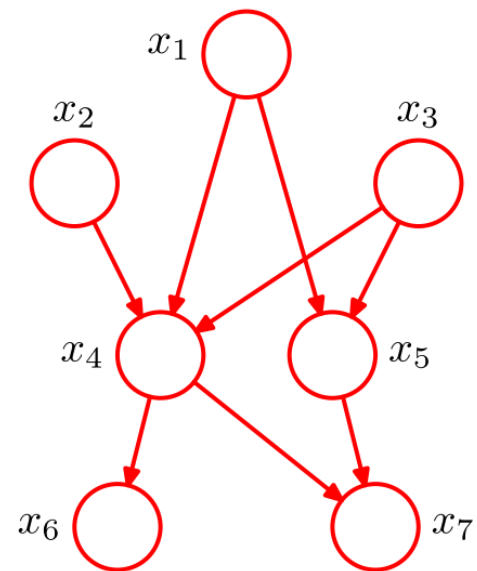
- A 7-dimensional data distribution

$$\begin{aligned} p(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = \\ p(x_1)p(x_2)p(x_3) \\ p(x_4|x_1, x_2, x_3)p(x_5|x_1, x_3) \\ p(x_6|x_4)p(x_7|x_4, x_5) \end{aligned}$$

- For a graph with  $K$  nodes, the joint distribution is

$$p(\mathbf{x}) = \prod_{k=1}^K p(x_k | \text{pa}_k)$$

↑  
Parent nodes of  $x_k$



- An important restriction: directed acyclic graphs (DAGs)

# An ML Example

- For the training data  $D = \{(x_i, t_i)\}$
- We build a linear prediction model with observation Gaussian noise

$$p(\mathbf{t}, \mathbf{w}) = p(\mathbf{w}) \prod_{i=1}^N p(t_i | \mathbf{w})$$

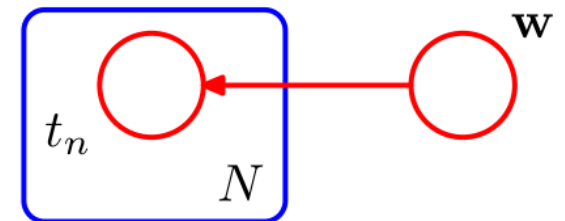
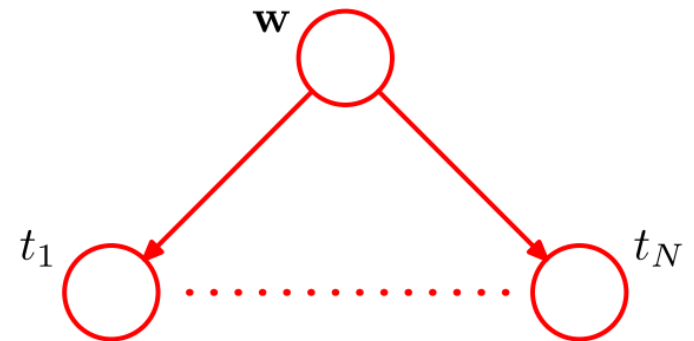
↑  
Prior distribution

- More explicitly

$$p(\mathbf{w} | \alpha) = \mathcal{N}(\mathbf{w} | \mathbf{0}, \alpha)$$

$$p(t_i | x_i, \mathbf{w}, \sigma^2) = \mathcal{N}(t_i | \mathbf{w}^\top x_i, \sigma^2)$$

$$p(\mathbf{t}, \mathbf{w} | \mathbf{x}, \alpha, \sigma^2) = p(\mathbf{w} | \alpha) \prod_{i=1}^N p(t_i | x_i, \mathbf{w}, \sigma^2)$$




An alternative, more compact, representation of the graph

# An ML Example

- For the training data  $D = \{(x_i, t_i)\}$
- We build a linear prediction model with observation Gaussian noise

$$p(\mathbf{t}, \mathbf{w}) = p(\mathbf{w}) \prod_{i=1}^N p(t_i | \mathbf{w})$$

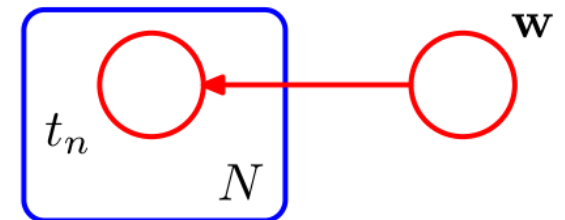
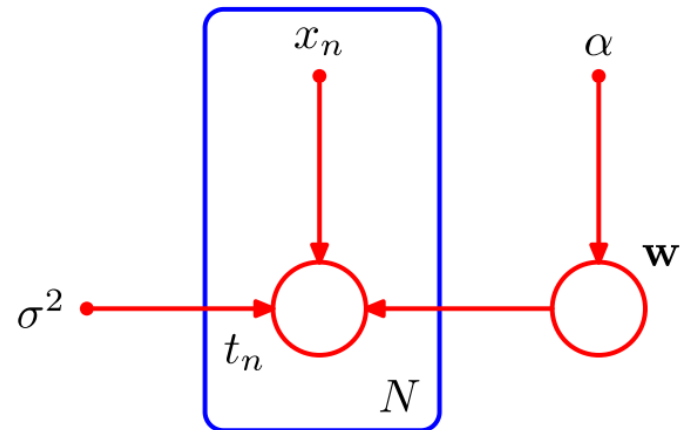
  
 Prior distribution

- More explicitly

$$p(\mathbf{w} | \alpha) = \mathcal{N}(\mathbf{w} | \mathbf{0}, \alpha)$$

$$p(t_i | x_i, \mathbf{w}, \sigma^2) = \mathcal{N}(t_i | \mathbf{w}^\top x_i, \sigma^2)$$

$$p(\mathbf{t}, \mathbf{w} | \mathbf{x}, \alpha, \sigma^2) = p(\mathbf{w} | \alpha) \prod_{i=1}^N p(t_i | x_i, \mathbf{w}, \sigma^2)$$



An alternative, more compact, representation of the graph



# Posterior Distribution

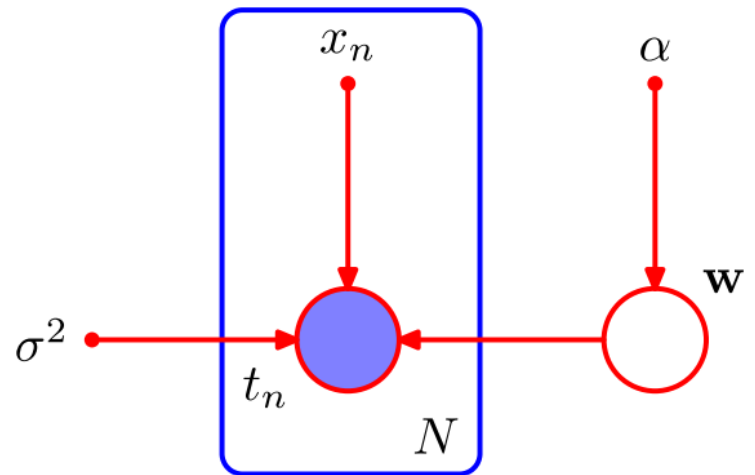
- With  $\{t_n\}$  observed, we can evaluate the posterior distribution of coefficients  $\mathbf{w}$

$$p(\mathbf{w}|\mathbf{t}) = \frac{p(\mathbf{w})p(\mathbf{t}|\mathbf{w})}{p(\mathbf{t})}$$

Posterior distribution

$$\propto p(\mathbf{w}) \prod_{i=1}^N p(t_i|\mathbf{w})$$

Prior distribution      Data likelihood



# Maximum A Posteriori Estimation

- Maximum A Posteriori (MAP) estimation of the model coefficients  $\mathbf{w}$

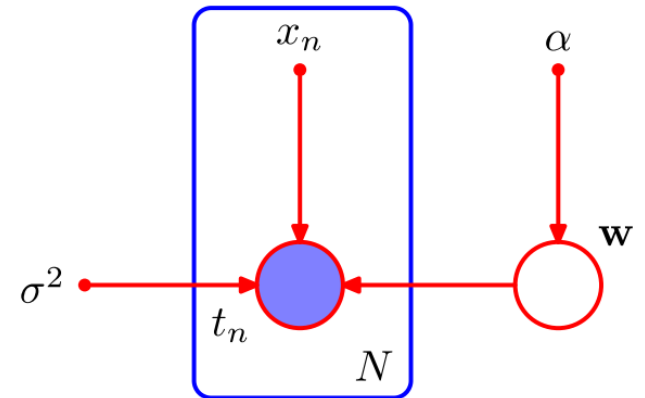
$$\max_{\mathbf{w}} p(\mathbf{w}|\mathbf{t}) = \max_{\mathbf{w}} p(\mathbf{w}, \mathbf{t}) = \max_{\mathbf{w}} p(\mathbf{w})p(\mathbf{t}|\mathbf{w})$$

$$\begin{aligned} p(\mathbf{w})p(\mathbf{t}|\mathbf{w}) &= p(\mathbf{w}|\alpha) \prod_{i=1}^N p(t_i|x_i, \mathbf{w}, \sigma^2) \\ &= \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha) \prod_{i=1}^N \mathcal{N}(t_i|\mathbf{w}^\top \mathbf{x}, \sigma^2) \end{aligned}$$

$$= \frac{1}{\sqrt{(2\pi\alpha)^d}} \exp\left(-\frac{\mathbf{w}^\top \mathbf{w}}{2\alpha}\right) \prod_{i=1}^N \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(t_i - \mathbf{w}^\top \mathbf{x})^2}{2\sigma^2}\right)$$

$$\log p(\mathbf{w})p(\mathbf{t}|\mathbf{w}) = -\frac{\mathbf{w}^\top \mathbf{w}}{2\alpha} - \sum_{i=1}^N \frac{(t_i - \mathbf{w}^\top \mathbf{x})^2}{2\sigma^2} + \text{const}$$

Equivalent to  $\min_{\mathbf{w}} \frac{\sigma^2}{\alpha} \|\mathbf{w}\|^2 + \sum_{i=1}^N (t_i - \mathbf{w}^\top \mathbf{x})^2$  i.e., ridge regression with square loss



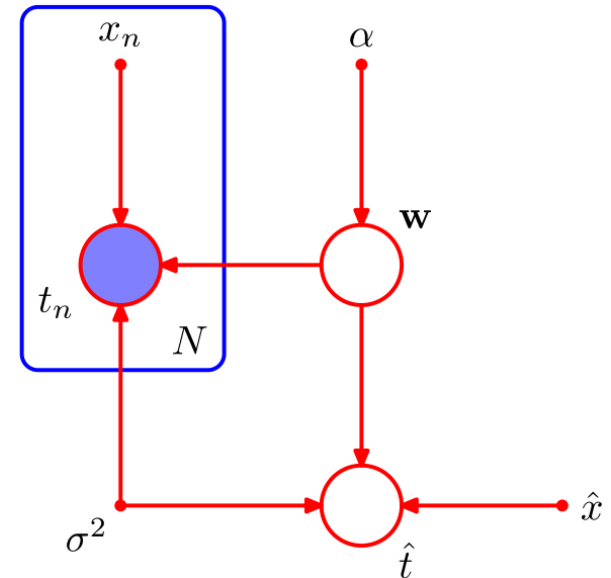
# Prediction on New Instance

- Given a new input value  $\hat{x}$ , predict the corresponding probability distribution for its label  $\hat{t}$ 
  - Joint distribution of random variables

$$p(\hat{t}, \mathbf{t}, \mathbf{w} | \hat{x}, \mathbf{x}, \alpha, \sigma^2) = \left[ \prod_{i=1}^N p(t_i | x_i, \mathbf{w}, \sigma^2) \right] p(\mathbf{w} | \alpha) p(\hat{t} | \hat{x}, \mathbf{w}, \sigma^2)$$

- Marginalize out the coefficients  $\mathbf{w}$

$$\begin{aligned} p(\hat{t} | \hat{x}, \mathbf{x}, \mathbf{t}, \alpha, \sigma^2) &= \frac{p(\hat{t}, \mathbf{t} | \hat{x}, \mathbf{x}, \alpha, \sigma^2)}{p(\mathbf{t})} \\ &\propto p(\hat{t}, \mathbf{t} | \hat{x}, \mathbf{x}, \alpha, \sigma^2) \\ &= \int p(\hat{t}, \mathbf{t}, \mathbf{w} | \hat{x}, \mathbf{x}, \alpha, \sigma^2) d\mathbf{w} \end{aligned}$$



# Conditional Independence

- Consider three variables  $a$ ,  $b$ , and  $c$
- Suppose that the conditional distribution of  $a$ , given  $b$  and  $c$ , is such that it does not depend on the value of  $b$

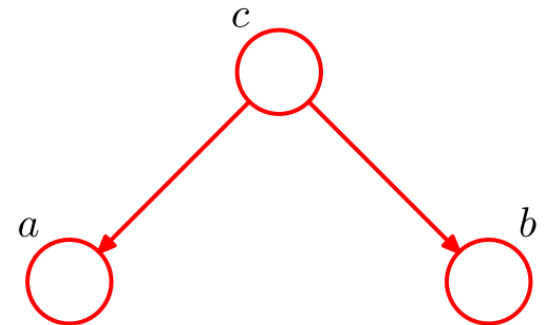
$$p(a|b, c) = p(a|c)$$

- We say that  $a$  is conditionally independent of  $b$  given  $c$
- A slightly different presentation

$$\begin{aligned} p(a, b|c) &= p(a|b, c)p(b|c) \\ &= p(a|c)p(b|c) \end{aligned}$$

- A notation for conditional independence

$$a \perp\!\!\!\perp b \mid c$$



# Conditional Independence in Graph

- Conditional independence properties of the joint distribution can be read directly from the graph
- Example 1: tail-to-tail

- With  $c$  unobserved

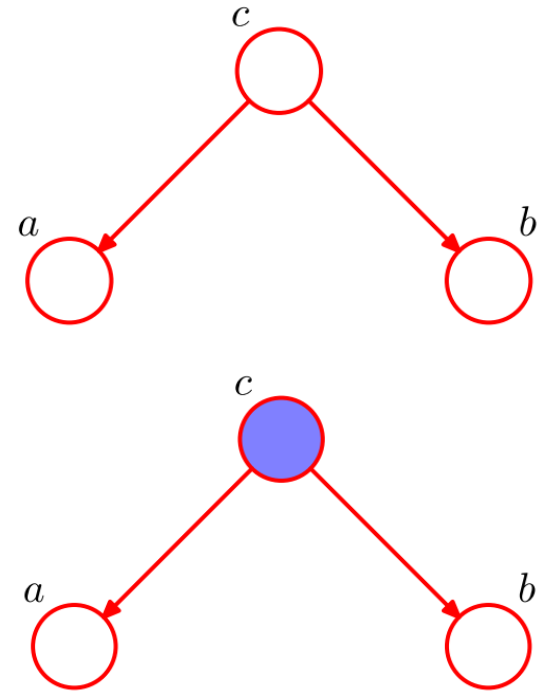
$$p(a, b, c) = p(a|c)p(b|c)p(c)$$

Not conditional independence  $a \not\perp b \mid \emptyset$

- With  $c$  observed

$$p(a, b|c) = p(a|c)p(b|c)$$

Conditional independence  $a \perp b \mid c$

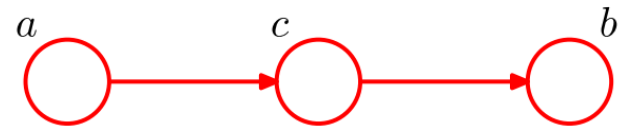


# Conditional Independence in Graph

- Example 2: head-to-tail
  - With  $c$  unobserved

$$p(a, b, c) = p(a)p(c|a)p(b|c)$$

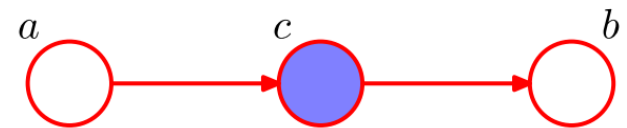
Not conditional independence  $a \not\perp\!\!\!\perp b \mid \emptyset$



- With  $c$  observed

$$\begin{aligned} p(a, b|c) &= \frac{p(a, b, c)}{p(c)} \\ &= \frac{p(a)p(c|a)p(b|c)}{p(c)} \\ &= p(a|c)p(b|c) \end{aligned}$$

Conditional independence  $a \perp\!\!\!\perp b \mid c$



# Conditional Independence in Graph

- Example 3: head-to-head

- With  $c$  unobserved

$$p(a, b, c) = p(c|a, b)p(a)p(b)$$

Marginalize both sides over  $c$

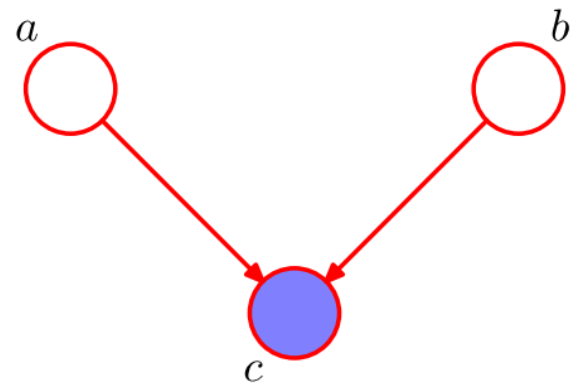
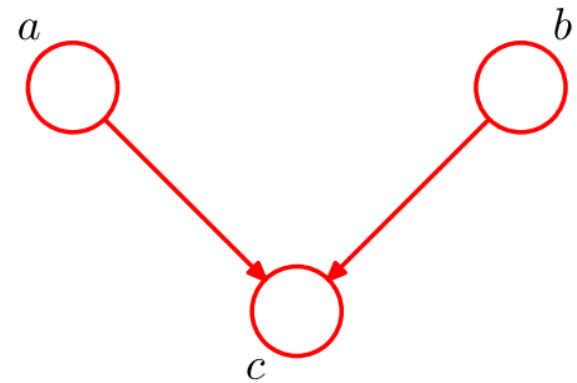
$$p(a, b) = p(a)p(b)$$

Conditional independence  $a \perp\!\!\!\perp b \mid \emptyset$

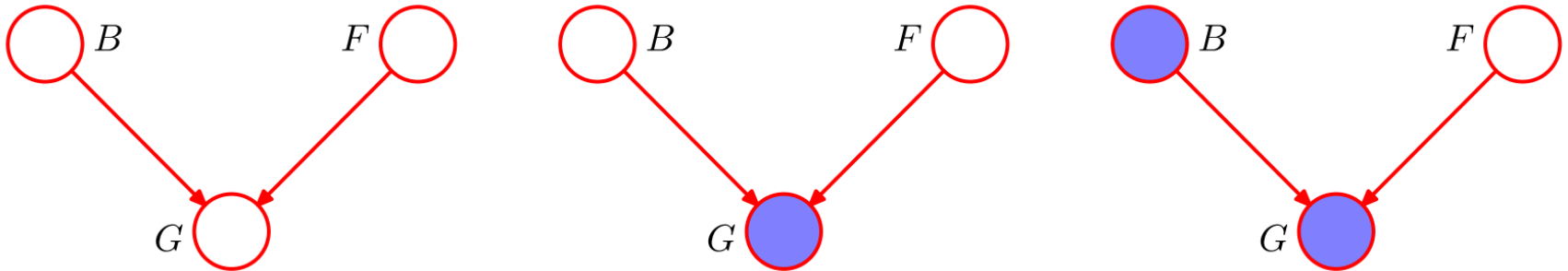
- With  $c$  observed

$$\begin{aligned} p(a, b|c) &= \frac{p(a, b, c)}{p(c)} \\ &= \frac{p(a)p(b)p(c|a, b)}{p(c)} \end{aligned}$$

Not conditional independence  $a \not\perp\!\!\!\perp b \mid c$



# Understanding head-to-head Case



- Variables

- $B$ : battery state, either charged ( $B=1$ ) or flat ( $B=0$ )
- $F$ : fuel tank state, either full of fuel ( $F=1$ ) or empty ( $F=0$ )
- $G$ : electric fuel gauge, either full ( $G=1$ ) or empty ( $G=0$ )



- (Conditional) probabilities

$$p(B = 1) = 0.9$$

$$p(F = 1) = 0.9$$

$$p(G = 1 | B = 1, F = 1) = 0.8$$

$$p(G = 1 | B = 1, F = 0) = 0.2$$

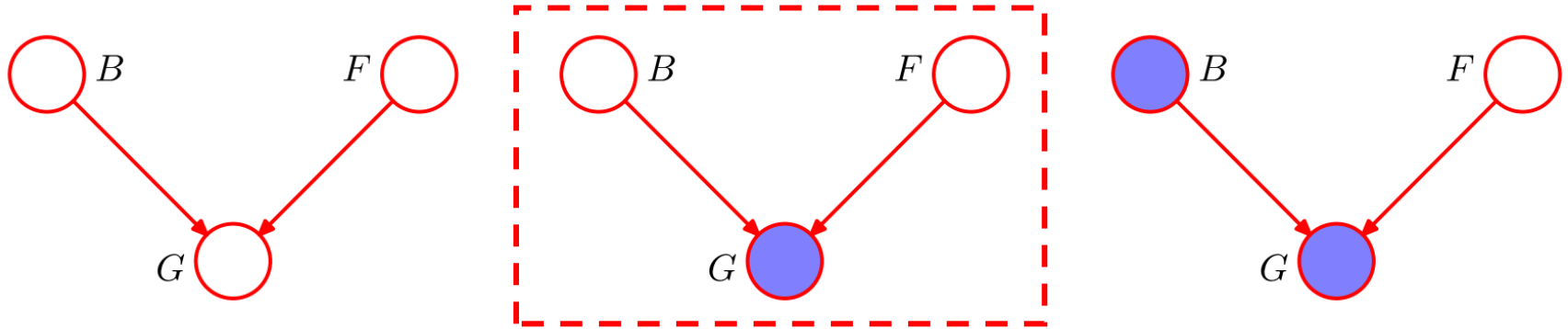
$$p(G = 1 | B = 0, F = 1) = 0.2$$

$$p(G = 1 | B = 0, F = 0) = 0.1$$

All remaining probabilities are determined by the requirement that probabilities sum to one



# Understanding head-to-head Case



- If we observe the fuel gauge reads empty, i.e.,  $G=0$

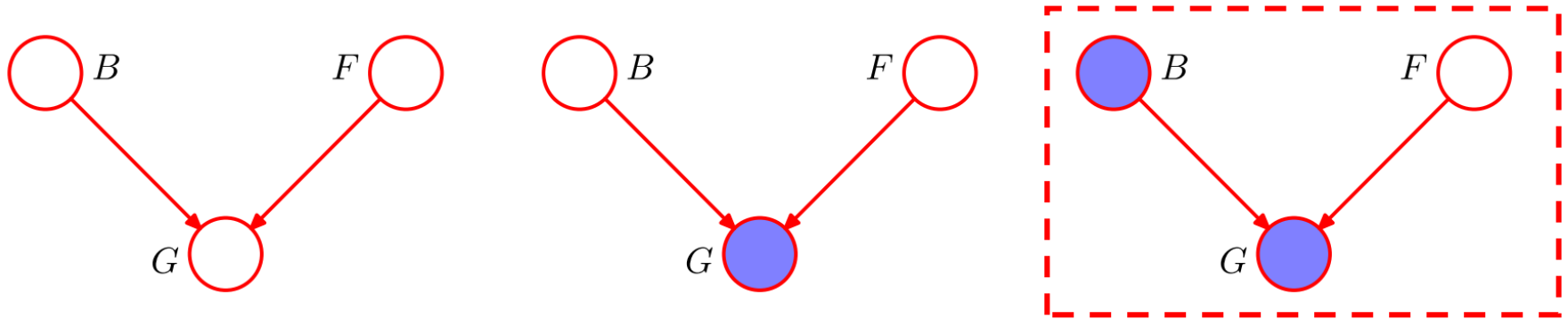
$$p(G = 0) = \sum_{B \in \{0,1\}} \sum_{F \in \{0,1\}} p(G = 0|B, F)p(B)p(F) = 0.315$$

$$p(G = 0|F = 0) = \sum_{B \in \{0,1\}} p(G = 0|B, F = 0)p(B) = 0.81$$

$$p(F = 0|G = 0) = \frac{p(G = 0|F = 0)p(F = 0)}{p(G = 0)} \simeq 0.257 > p(F = 0) = 0.1$$

Thus observing that the gauge reads empty makes it more likely that the tank is indeed empty

# Understanding head-to-head Case



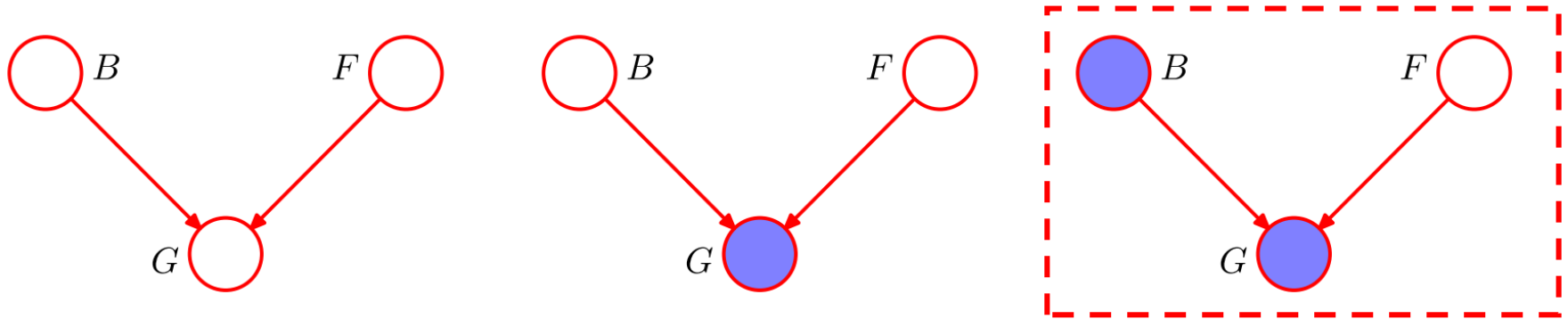
- If we observe the fuel gauge reads empty, i.e.,  $G=0$

$$p(F = 0|G = 0) = \frac{p(G = 0|F = 0)p(F = 0)}{p(G = 0)} \simeq 0.257 > p(F = 0) = 0.1$$

$$p(F = 0|G = 0, B = 0) = \frac{p(G = 0|B = 0, F = 0)p(F = 0)}{\sum_{F \in \{0,1\}} p(G = 0|B = 0, F)p(F)} \simeq 0.111 > p(F = 0) = 0.1$$

- The probability that the tank is empty has decreased (from 0.257 to 0.111) as a result of the observation of the state of the battery
- **Explaining away:** the battery is flat explains away the observation that the fuel gauge reads empty

# Understanding head-to-head Case



- If we observe the fuel gauge reads empty, i.e.,  $G=0$

$$p(F = 0|G = 0) = \frac{p(G = 0|F = 0)p(F = 0)}{p(G = 0)} \simeq 0.257 > p(F = 0) = 0.1$$

$$p(F = 0|G = 0, B = 0) = \frac{p(G = 0|B = 0, F = 0)p(F = 0)}{\sum_{F \in \{0,1\}} p(G = 0|B = 0, F)p(F)} \simeq 0.111 > p(F = 0) = 0.1$$

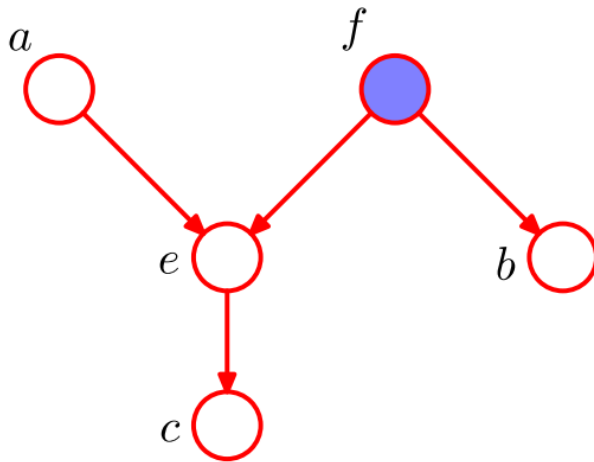
- Note that the probability  $p(F=0|G=0,B=0) \sim 0.111$  is greater than the prior probability  $p(F=0)=0.1$  because the observation that the fuel gauge reads zero still provides some evidence in favor of an empty fuel tank.

# D-separation

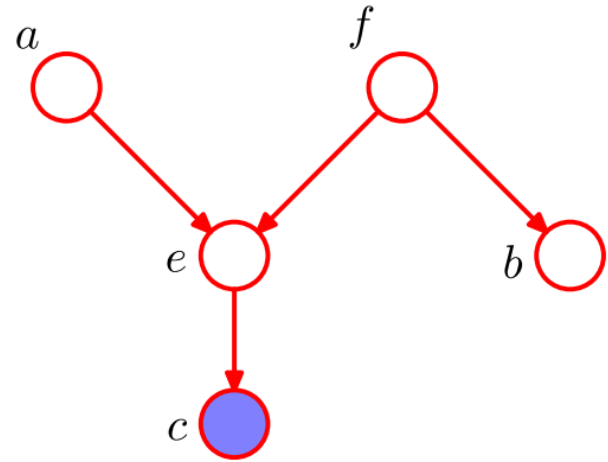
- Consider a general directed graph in which  $A$ ,  $B$ , and  $C$  are arbitrary nonintersecting sets of nodes.
- Any such path is said to be blocked if it includes a node such that either
  - a) the arrows on the path meet either **head-to-tail** or **tail-to-tail** at the node, and the node is in the set  $C$ , or
  - b) the arrows meet **head-to-head** at the node, and neither the node, nor any of its descendants, is in the set  $C$ .
- If all paths are blocked, then  $A$  is said to be d-separated from  $B$  by  $C$ , and the joint distribution over all of the variables in the graph will satisfy

$$A \perp\!\!\!\perp B \mid C$$

# D-separation Illustration



$$a \perp\!\!\!\perp b \mid f$$

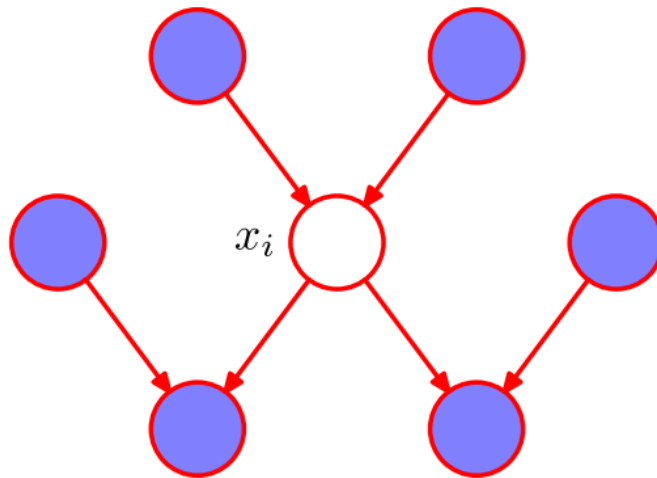


$$a \not\perp\!\!\!\perp b \mid c$$

- $A, B, C$  satisfy  $A \perp\!\!\!\perp B \mid C$  if
  - a) the arrows on the path meet either **head-to-tail** or **tail-to-tail** at the node, and the node is in the set  $C$ , or
  - b) the arrows meet **head-to-head** at the node, and neither the node, nor any of its descendants, is in the set  $C$ .

# Markov Blanket in Bayesian Network

- The Markov blanket of a node  $x_i$  comprises the set of **parents**, **children** and **co-parents of the children**.
- It has the property that the conditional distribution of  $x_i$ , conditioned on all the remaining variables in the graph, is dependent only on the variables in the Markov blanket.

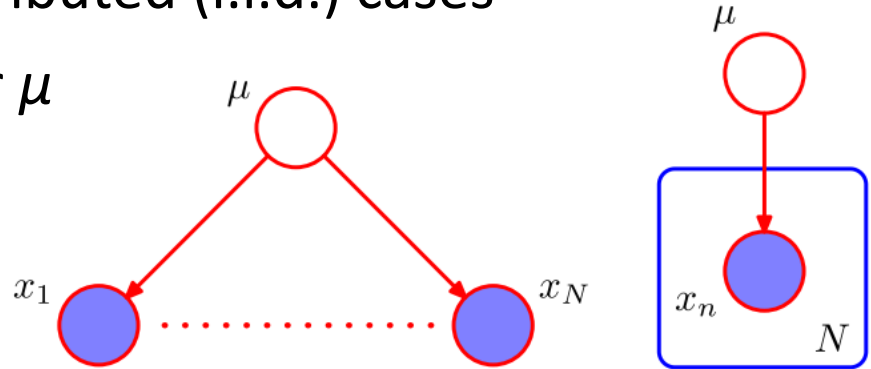


# i.i.d. Cases

- Independent identically distributed (i.i.d.) cases
- Goal: given  $\mathbf{x}$  observed, infer  $\mu$

$$p(\mu|D) \propto p(\mu)p(D|\mu)$$

$$= p(\mu) \prod_{i=1}^N p(x_i|\mu)$$



- If we integrate over  $\mu$ , the observations are in general non-independent

$$p(D) = \int p(D|\mu)p(\mu)d\mu \neq \prod_{i=1}^N p(x_i)$$

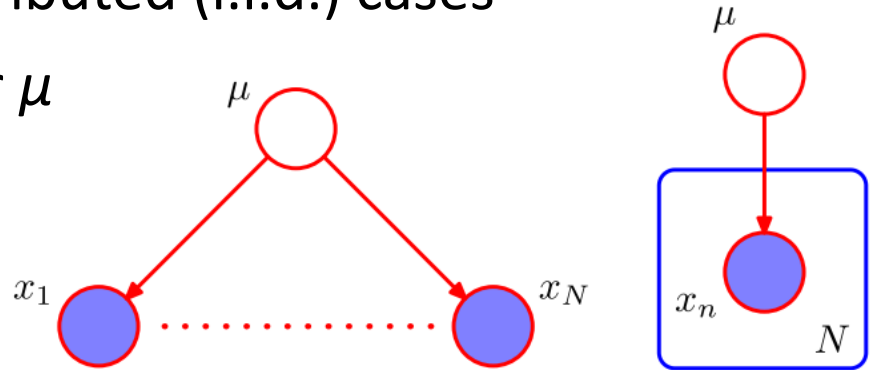
- We may say these data instances are jointly distributed.

# i.i.d. Cases

- Independent identically distributed (i.i.d.) cases
- Goal: given  $\mathbf{x}$  observed, infer  $\mu$

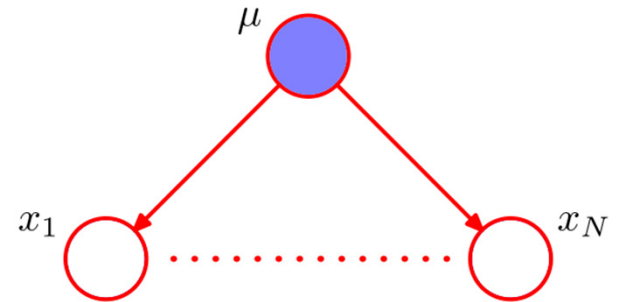
$$p(\mu|D) \propto p(\mu)p(D|\mu)$$

$$= p(\mu) \prod_{i=1}^N p(x_i|\mu)$$



- If we condition on  $\mu$  and consider the joint distribution of the observations

- A unique path from  $x_i$  to  $x_j$
- The path is tail-to-tail w.r.t.  $\mu$
- Thus the path is blocked given  $\mu$  observed



- Data instances are independent conditioned on the model



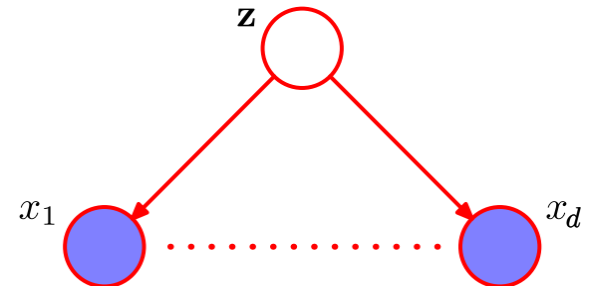
# Naive Bayes Classification Model

- $K$ -class classification
- The classes  $\mathbf{z}$  are represented in 1-of- $K$  encoding vector
  - Multinomial prior  $p(\mathbf{z}|\boldsymbol{\mu})$
  - $\mu_k$  is the prior probability of class  $C_k$
- Each data instance (e.g. a piece of text) is represented by a  $d$ -dimensional vector  $\mathbf{x}$  (each dimension as a word)
  - The generation of  $\mathbf{x}$  conditioned on  $\mathbf{z}$  is  $p(\mathbf{x}|\mathbf{z})$
  - The principle of naive Bayes is the conditional independence of  $x_j$ 's

$$p(\mathbf{x}|\mathbf{z}) = \prod_{j=1}^d p(x_j|\mathbf{z})$$

- Class label inference

$$p(\mathbf{z}|\mathbf{x}) \propto p(\mathbf{x}|\mathbf{z})p(\mathbf{z}|\boldsymbol{\mu})$$



# Multinomial Naive Bayes

- Each class  $y$  is modeled as a histogram of words
  - $y=y(\mathbf{z})$  is the index of 1 in  $\mathbf{z}$

$$\theta_y = (\theta_{y1}, \theta_{y2}, \dots, \theta_{yn})$$

- The parameter  $\vartheta_y$  is estimated as

$$\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha d}$$

- $N_{yi}$  is the count of word  $i$  appears in any instance of class  $y$  in the training set
- $N_y$  is the total count of all words for class  $y$

$$p(\mathbf{z}|\mathbf{x}) \propto p(\mathbf{z}|\boldsymbol{\mu})p(\mathbf{x}|\mathbf{z}) = p(\mathbf{z}|\boldsymbol{\mu}) \prod_{i=1}^d \theta_{y(\mathbf{z})i}$$

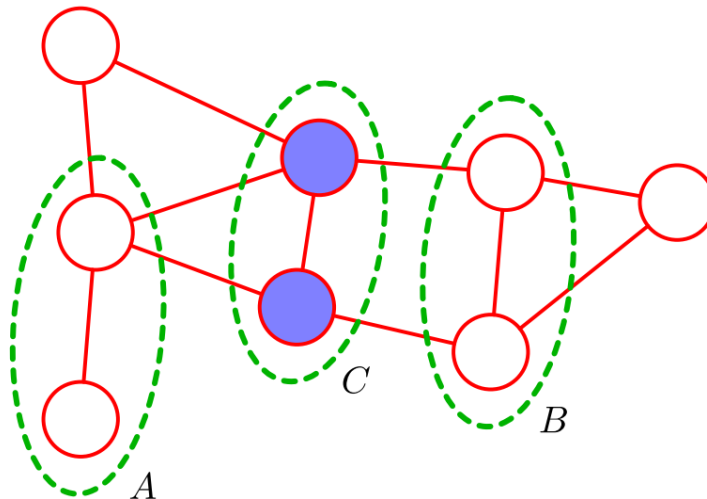
# Content of This Lecture

- Introduction
- Bayes Networks (Directed Graphs)
- Markov Networks (Undirected Graphs)
- Inferences in Graphical Models

# Markov Random Fields

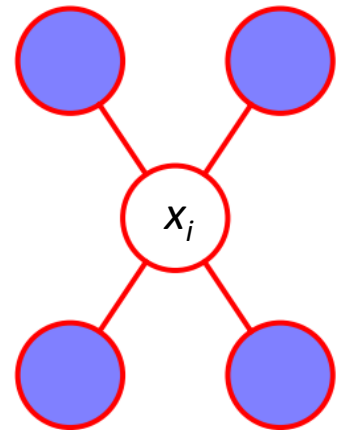
- Undirected network, also called Markov network
- Compared to Bayes Network, it is more straightforward to ascertain the conditional independence in Markov network:
  - If all paths linking any nodes in A and B is blocked by the nodes in C, then

$$A \perp\!\!\!\perp B \mid C$$



# Markov Blanket in Markov Network

- For an undirected graph, the Markov blanket of a node  $x_i$  consists of the set of neighboring nodes.
- It has the property that the conditional distribution of  $x_i$ , conditioned on all the remaining variables in the graph, is dependent only on the variables in the Markov blanket.



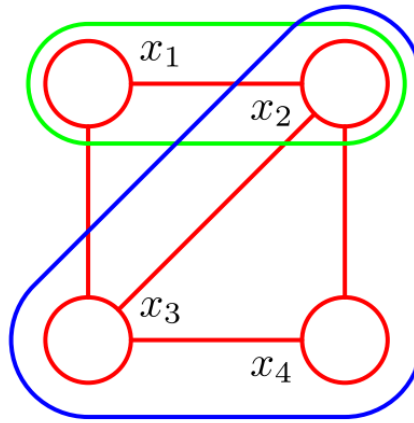
# Conditional Independence in Markov Network

- Consider two nodes  $x_i$  and  $x_j$  that are not connected by a link, then these variables must be conditionally independent given all other nodes in the graph

$$p(x_i, x_j | \mathbf{x} \setminus \{i, j\}) = p(x_i | \mathbf{x} \setminus \{i, j\}) p(x_j | \mathbf{x} \setminus \{i, j\})$$

- The factorization of the joint distribution must therefore be such that  $x_i$  and  $x_j$  do not appear in the same factor

# An Example of Cliques in Markov Networks



- **Clique:** a subset of the nodes in a graph in which the nodes are fully connected
- A Markov network of four nodes  $\{x_1, x_2, x_3, x_4\}$ 
  - 5 two-node cliques  $\{x_1, x_2\}, \{x_2, x_3\}, \{x_3, x_4\}, \{x_2, x_4\}, \{x_1, x_3\}$
  - 2 maximal cliques  $\{x_1, x_2, x_3\}, \{x_2, x_3, x_4\}$
- Note that  $\{x_1, x_2, x_3, x_4\}$  is not a clique

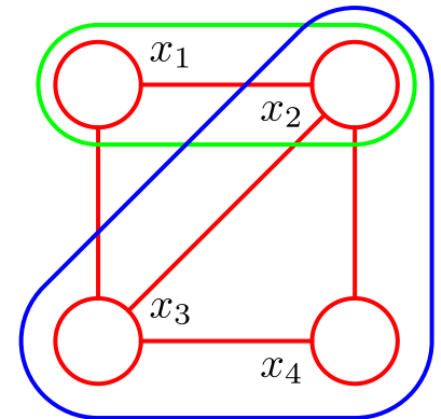
# Joint Distribution Decomposition

- Define the factors in the decomposition of the joint distribution to be functions of the variables in the cliques
- Let  $C$  denote a clique and the set of variables in it as  $\mathbf{x}_C$

$$p(\mathbf{x}) = \frac{1}{Z} \prod_C \psi_C(\mathbf{x}_C)$$

Potential function

$$p(\mathbf{x}) = \frac{1}{Z} \psi_{\{2,3,4\}}(x_2, x_3, x_4) \psi_{\{1,2,3\}}(x_1, x_2, x_3)$$



- The quantity  $Z$ , also called the partition function, is a normalization factor

$$Z = \sum_{\mathbf{x}} \prod_C \psi_C(\mathbf{x}_C)$$



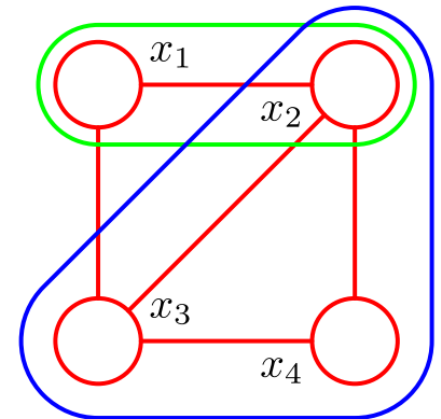
# Joint Distribution Decomposition

- Define the factors in the decomposition of the joint distribution to be functions of the variables in the cliques
- Let  $C$  denote a clique and the set of variables in it as  $\mathbf{x}_C$

$$p(\mathbf{x}) = \frac{1}{Z} \prod_C \psi_C(\mathbf{x}_C)$$

↑  
Potential function

- Potential function satisfies  $\psi_C(\mathbf{x}_C) \geq 0$  to ensure the probability is non-negative
- Potential functions can be defined with domain knowledge



# Energy Function for Potential

- If we define the potential function to be strictly positive, i.e.,

$$\psi_C(\mathbf{x}_C) > 0$$

- It is convenient to express the potential functions as exponentials

$$\psi_C(\mathbf{x}_C) = \exp\{-E(\mathbf{x}_C)\}$$

- $E(\mathbf{x}_C)$  is called an energy function
- With such an exponential representation, the distribution  $p(\mathbf{x})$  is called Boltzmann distribution

$$p(\mathbf{x}) = \frac{1}{Z} \prod_C \psi_C(\mathbf{x}_C) = \frac{1}{Z} \exp\left\{-\sum_C E(\mathbf{x}_C)\right\}$$

# Boltzmann Distribution

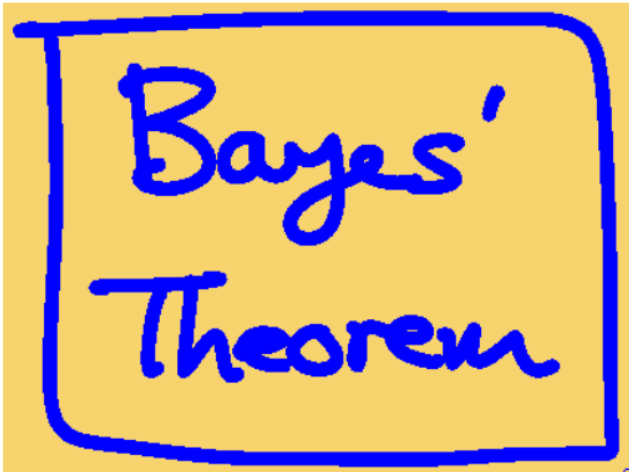
- Boltzmann distribution is a probability distribution, probability measure, or frequency distribution of particles in a system over various possible states

$$p(s) = \frac{e^{-E(s)/kT}}{\sum_{s'} e^{-E(s')/kT}}$$

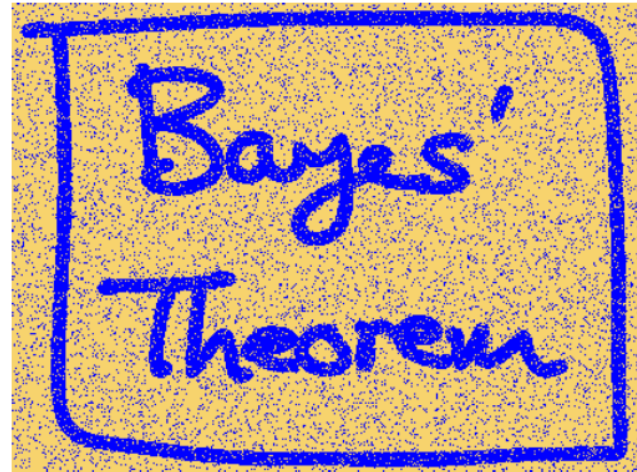
- $s$  denotes a particular state
  - $E(s)$  is the state energy
  - $k = 1.381 \times 10^{-23}$  J/K is Boltzmann constant
  - $T$  is thermodynamic temperature
- Low-energy state is more stable, i.e., with higher probability

# MRF Application Example: Image Denoising

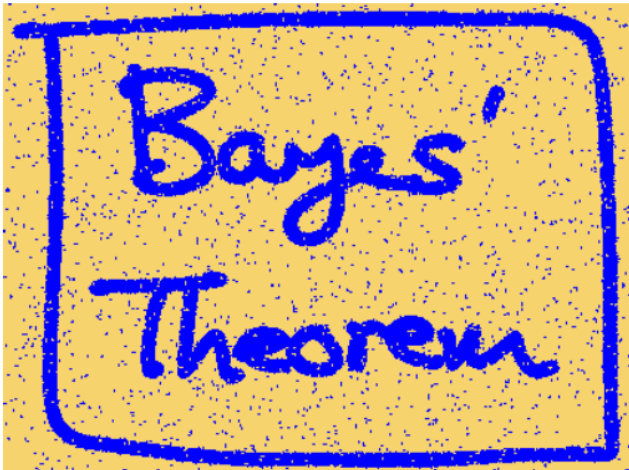
Original  
Image



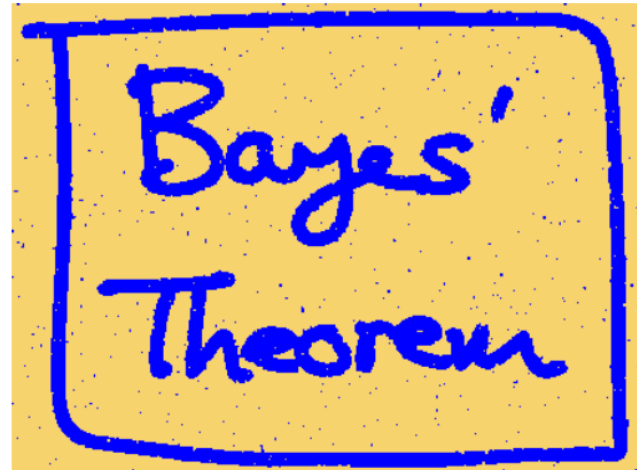
Corrupted  
Image



Denoised  
By ICM



Denoised  
By Graph-Cut



# MRF Application Example: Image Denoising

- Observed noisy image is described by an array of binary pixel values

$$y_i \in \{-1, +1\}, \quad i = 1, \dots, d \quad \text{runs over all pixels}$$

- Suppose the ground-truth noise-free image

$$x_i \in \{-1, +1\}, \quad i = 1, \dots, d$$

- Noise generation: randomly flipping the sign of pixels with some small probability, e.g., 10%

- Model assumptions

- There is a strong correlation between  $x_i$  and  $y_i$
- There is a strong correlation between neighboring pixels  $x_i$  and  $x_j$

# MRF for Image Denoising

- Model assumptions

- There is a strong correlation between  $x_i$  and  $y_i$
- There is a strong correlation between neighboring pixels  $x_i$  and  $x_j$

- Model

- For the cliques  $\{x_i, y_i\}$

$$E(\{x_i, y_i\}) = -\eta x_i y_i$$

- For the cliques  $\{x_i, x_j\}$

$$E(\{x_i, x_j\}) = -\beta x_i x_j$$

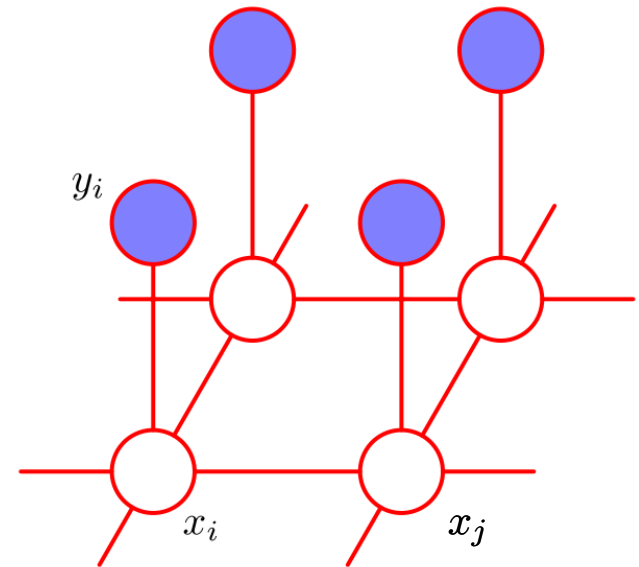
- Moreover, for each  $\{x_i\}$

$$E(\{x_i\}) = h x_i$$

- Complete energy function

$$E(\mathbf{x}, \mathbf{y}) = h \sum_i x_i - \beta \sum_{i,j} x_i x_j - \eta \sum_i x_i y_i$$

$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp\{-E(\mathbf{x}, \mathbf{y})\}$$



# Solution: Iterated Conditional Modes (ICM)

- Objective

$$\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) = \max_{\mathbf{x}} p(\mathbf{x}, \mathbf{y}) \quad \text{Given } \mathbf{y} \text{ observed}$$

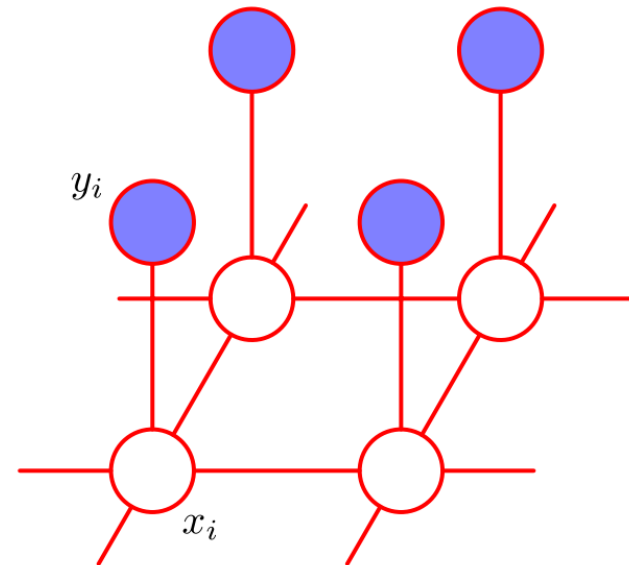
- Idea: coordinate-wise gradient ascent

- For each node  $x_j$ , check which one of  $x_j=+1$  or  $-1$  leads to lower  $E(\mathbf{x}, \mathbf{y})$
- Implementation  $\beta=1.0$ ,  $\eta=2.1$  and  $h=0$

- Energy function

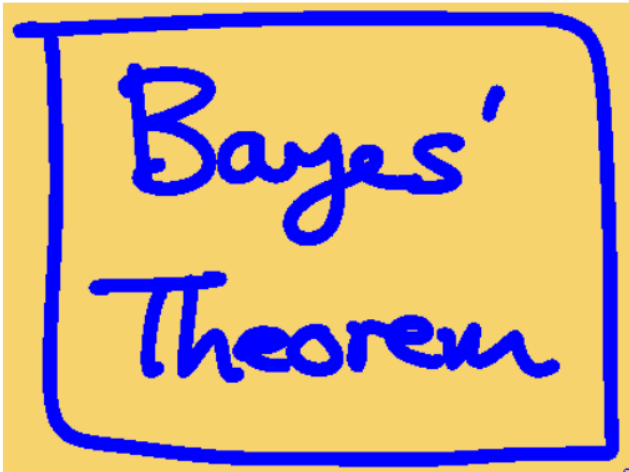
$$E(\mathbf{x}, \mathbf{y}) = h \sum_i x_i - \beta \sum_{i,j} x_i x_j - \eta \sum_i x_i y_i$$

$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp\{-E(\mathbf{x}, \mathbf{y})\}$$

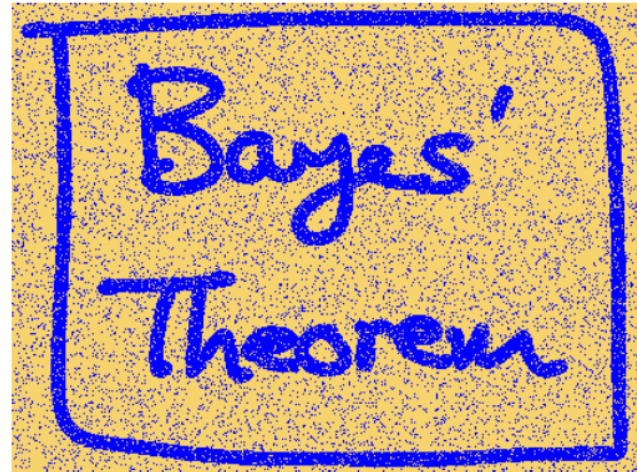


# MRF Application Example: Image Denoising

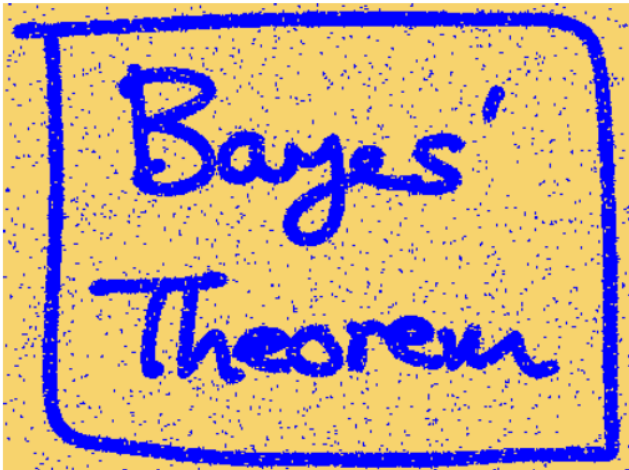
Original  
Image



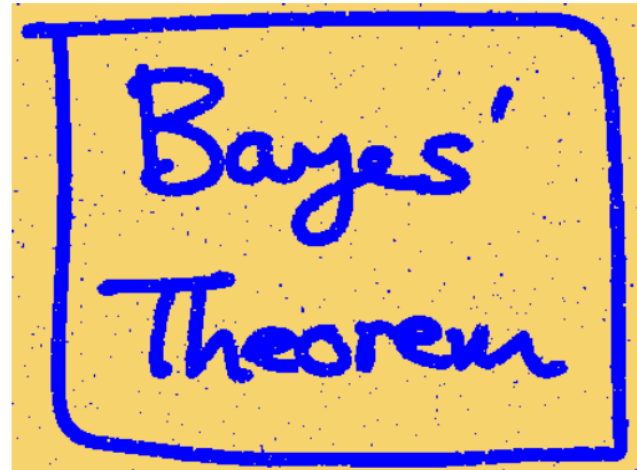
Corrupted  
Image



Denoised  
By ICM

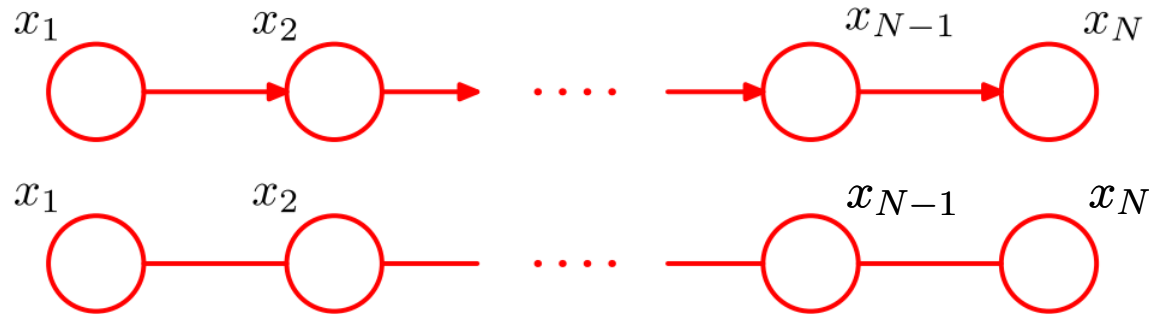


Denoised  
By Graph-Cut





# Directed Graphs vs. Undirected Graphs



- Convert a directed graph to an undirected graph

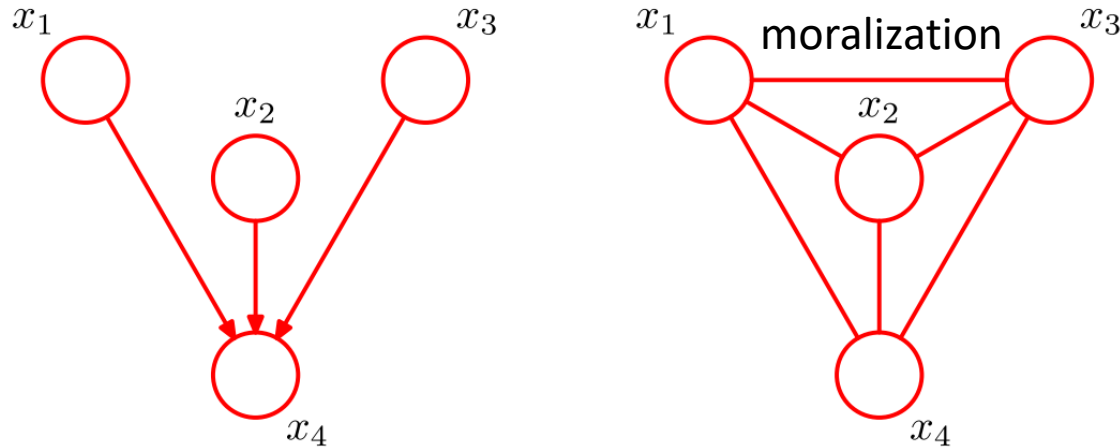
- Directed graphical model

$$p(\mathbf{x}) = p(x_1)p(x_2|x_1)p(x_3|x_2) \cdots p(x_N|x_{N-1})$$

- Undirected graphical model

$$p(\mathbf{x}) = \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \cdots \psi_{N-1,N}(x_{N-1}, x_N)$$

# Directed Graphs vs. Undirected Graphs



- Convert a directed graph to an undirected graph
  - Directed graphical model

$$p(\mathbf{x}) = p(x_1)p(x_2)p(x_3)p(x_4|x_1, x_2, x_3)$$

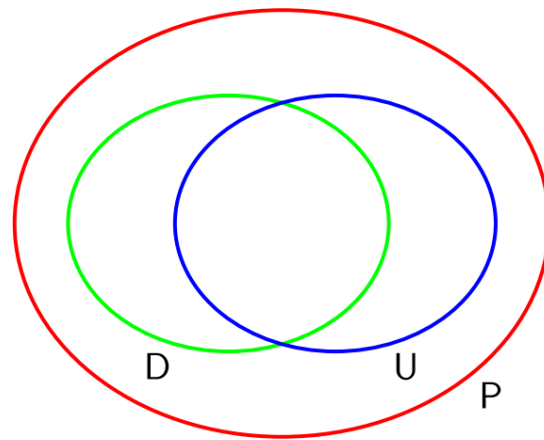
- Undirected graphical model

$$p(\mathbf{x}) = \frac{1}{Z} \psi_{1,2,3,4}(x_1, x_2, x_3, x_4)$$

Moralization: marrying the parents

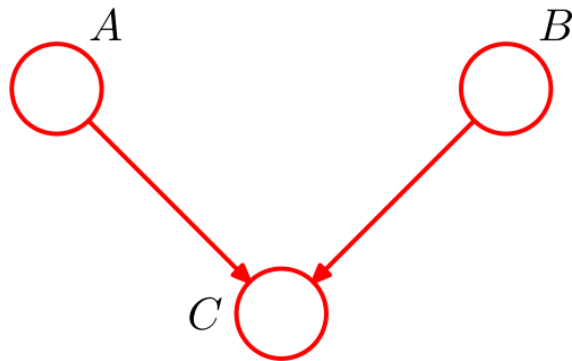
# Directed Graphs vs. Undirected Graphs

- Although each directed graph can be converted into an undirected graph
  - One brute-force solution is to use a fully connected undirected graph
- Directed and undirected graphs can express different conditional independence properties



- P: all possible distributions
- D/U: distributions that can be represented by directed/undirected graphs

# Directed Graphs vs. Undirected Graphs

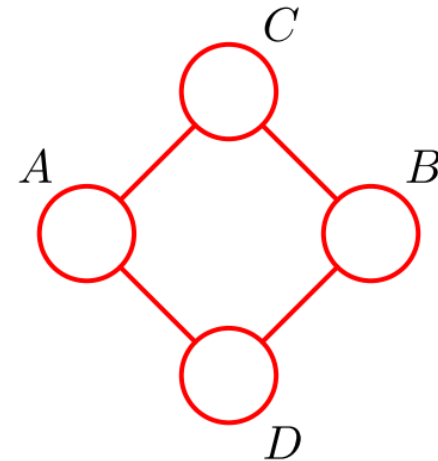


A directed graph whose conditional independence properties cannot be expressed using an undirected graph over the same three variables

- Directed graph

$$A \perp\!\!\!\perp B \mid \emptyset$$

$$A \not\perp\!\!\!\perp B \mid C$$



An undirected graph whose conditional independence properties cannot be expressed in terms of a directed graph over the same variables

- Undirected graph

$$A \not\perp\!\!\!\perp B \mid \emptyset, C \perp\!\!\!\perp D \mid A \cup B$$

$$A \perp\!\!\!\perp B \mid C \cup D$$

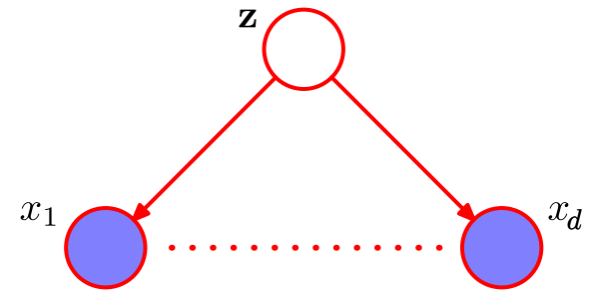
# Content of This Lecture

- Introduction
- Bayes Networks (Directed Graphs)
- Markov Networks (Undirected Graphs)
- Inferences in Graphical Models

# Variable Inference and Parameter Estimation

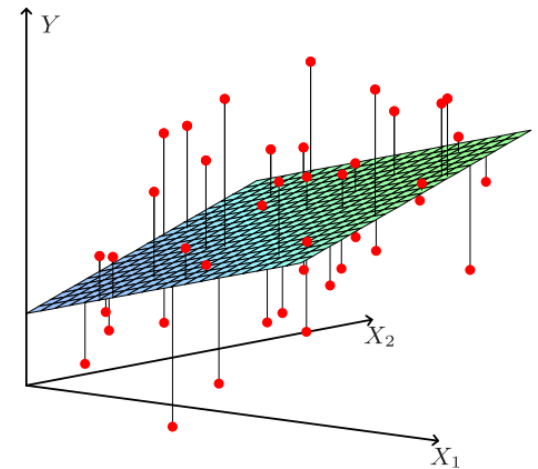
- Random variable inference
  - Infer the posterior distribution of random variables given their prior and the observed data

$$p(\mathbf{z}|\mathbf{x}) \propto p(\mathbf{z}|\boldsymbol{\mu})p(\mathbf{x}|\mathbf{z})$$

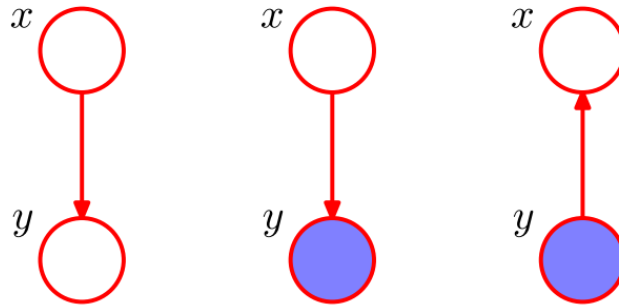


- Parameter estimation
  - Find the optimal parameter value for an objective, e.g., minimum loss or maximum likelihood

$$\hat{\theta} = \arg \min_{\theta} \mathcal{L}(D; \theta)$$



# A Basic Case for Inference



- Joint distribution of two random variables  $x$  and  $y$

$$p(x, y) = p(x)p(y|x)$$

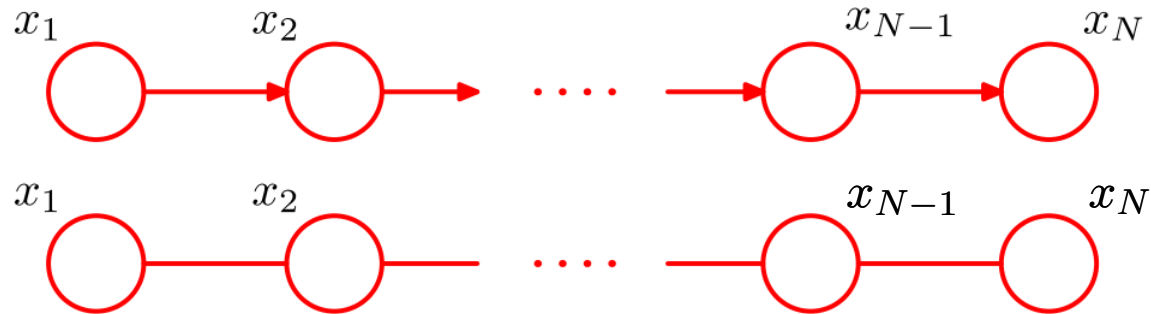
- The marginal distribution of  $y$

$$p(y) = \sum_{x'} p(x')p(y|x')$$

- The inverse conditional distribution

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)}$$

# Inference on a Chain



- Joint distribution

$$p(\mathbf{x}) = \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \cdots \psi_{N-1,N}(x_{N-1}, x_N)$$

- Discrete variable setting
  - $N$  nodes represent discrete variables each having  $K$  states
  - Each potential function  $\psi_{n-1,n}(x_{n-1}, x_n)$  comprises a  $K \times K$  table
  - Thus the joint distribution has  $(N-1)K^2$  parameters



# Calculate the Marginal Distribution

- Inference problem of finding the marginal distribution  $p(x_n)$

$$p(x_n) = \sum_{x_1} \cdots \sum_{x_{n-1}} \sum_{x_{n+1}} \cdots \sum_{x_N} p(\mathbf{x})$$

- A brute-force solution
  - Sum up  $K^{N-1}$  values, introducing exponential complexity  $O(K^{N-1})$
- An efficient dynamic programming solution
  - Exploit the conditional independence properties

$$p(\mathbf{x}) = \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \cdots \psi_{N-1,N}(x_{N-1}, x_N)$$



general sum-product algorithm  $ab + ac = a(b + c)$

# DP for Calculating Marginal Distribution



$$p(\mathbf{x}) = \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \cdots \psi_{N-1,N}(x_{N-1}, x_N)$$

- Conditional independence

- The potential  $\psi_{N-1,N}(x_{N-1}, x_N)$  is the only one that depends on  $x_N$

$$\begin{aligned} p(x_n) &= \sum_{x_1} \cdots \sum_{x_{n-1}} \sum_{x_{n+1}} \cdots \sum_{x_N} p(\mathbf{x}) \\ &= \frac{1}{Z} \sum_{x_1} \cdots \sum_{x_{n-1}} \sum_{x_{n+1}} \cdots \sum_{x_N} \psi_{1,2}(x_1, x_2) \cdots \psi_{N-1,N}(x_{N-1}, x_N) \\ &= \frac{1}{Z} \sum_{x_1} \cdots \sum_{x_{n-1}} \sum_{x_{n+1}} \cdots \sum_{x_{N-1}} \psi_{1,2}(x_1, x_2) \cdots \psi_{N-2,N-1}(x_{N-2}, x_{N-1}) \sum_{x_N} \psi_{N-1,N}(x_{N-1}, x_N) \end{aligned}$$

general sum-product algorithm  $ab + ac = a(b + c)$

# DP for Calculating Marginal Distribution



$$p(\mathbf{x}) = \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \cdots \psi_{N-1,N}(x_{N-1}, x_N)$$

- Conditional independence
  - The potential  $\psi_{1,2}(x_1, x_2)$  is the only one that depends on  $x_1$

$$\begin{aligned} p(x_n) &= \sum_{x_1} \cdots \sum_{x_{n-1}} \sum_{x_{n+1}} \cdots \sum_{x_N} p(\mathbf{x}) \\ &= \frac{1}{Z} \sum_{x_n} \cdots \sum_{x_{n+1}} \sum_{x_{n-1}} \cdots \sum_{x_1} \psi_{N-1,N}(x_{N-1}, x_N) \cdots \psi_{1,2}(x_1, x_2) \\ &= \frac{1}{Z} \sum_{x_n} \cdots \sum_{x_{n+1}} \sum_{x_{n-1}} \cdots \sum_{x_2} \psi_{N-1,N}(x_{N-1}, x_N) \cdots \psi_{1,2}(x_1, x_2) \sum_{x_1} \psi_{1,2}(x_1, x_2) \end{aligned}$$

# DP for Calculating Marginal Distribution



$$p(\mathbf{x}) = \frac{1}{Z} \psi_{1,2}(x_1, x_2) \psi_{2,3}(x_2, x_3) \cdots \psi_{N-1,N}(x_{N-1}, x_N)$$

- Conditional independence

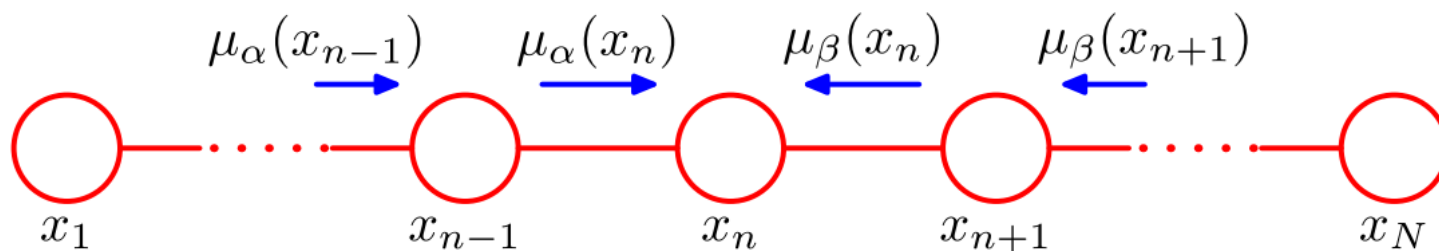
- The potential  $\psi_{1,2}(x_1, x_2)$  is the only one that depends on  $x_1$

$$p(x_n) = \frac{1}{Z} \underbrace{\left[ \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \cdots \left[ \sum_{x_2} \psi_{2,3}(x_2, x_3) \left[ \sum_{x_1} \psi_{1,2}(x_1, x_2) \right] \right] \cdots \right]}_{\mu_\alpha(x_n)} \underbrace{\left[ \sum_{x_{n+1}} \psi_{n,n+1}(x_n, x_{n+1}) \cdots \left[ \sum_{x_N} \psi_{N-1,N}(x_{N-1}, x_N) \right] \cdots \right]}_{\mu_\beta(x_n)}$$

Complexity  $O(NK^2)$

# Interpretation: Message Passing

- Passing of local messages around on the graph

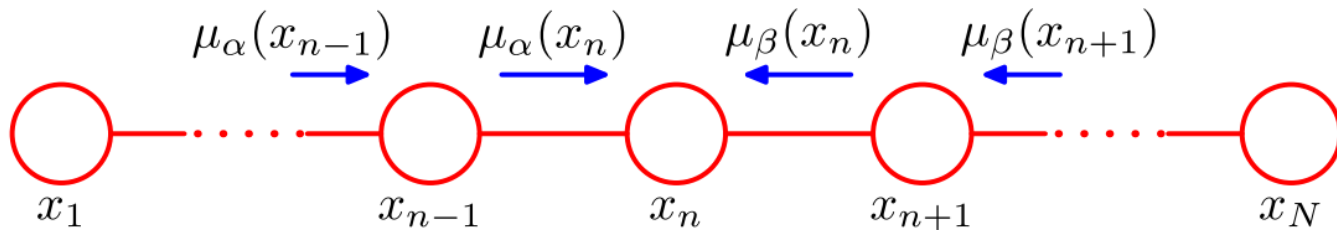


$$p(x_n) = \frac{1}{Z} \underbrace{\left[ \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \cdots \left[ \sum_{x_2} \psi_{2,3}(x_2, x_3) \left[ \sum_{x_1} \psi_{1,2}(x_1, x_2) \right] \right] \cdots \right]}_{\mu_\alpha(x_n)} \underbrace{\left[ \sum_{x_{n+1}} \psi_{n,n+1}(x_n, x_{n+1}) \cdots \left[ \sum_{x_N} \psi_{N-1,N}(x_{N-1}, x_N) \right] \cdots \right]}_{\mu_\beta(x_n)}$$

$$p(x_n) = \frac{1}{Z} \mu_\alpha(x_n) \mu_\beta(x_n)$$

# Interpretation: Message Passing

- Passing of local messages around on the graph



$$p(x_n) = \frac{1}{Z} \mu_\alpha(x_n) \mu_\beta(x_n)$$

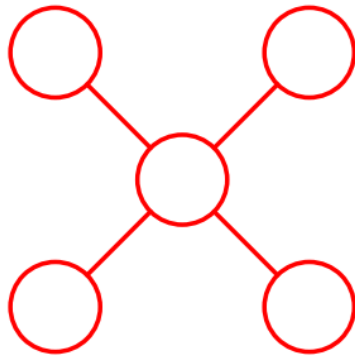
- Message passes recursively

$$\begin{aligned} \mu_\alpha(x_n) &= \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \left[ \sum_{x_{n-2}} \psi_{n-2,n-1}(x_{n-2}, x_{n-1}) \sum_{x_{n-3}} \cdots \right] \\ &= \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \mu_\alpha(x_{n-1}) \end{aligned}$$

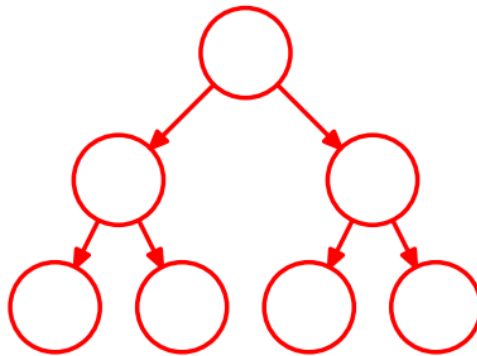
- With the start

$$\mu_\alpha(x_2) = \sum_{x_1} \psi_{1,2}(x_1, x_2)$$

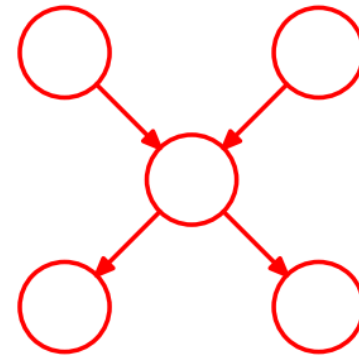
# Tree Graphical Models



Undirected tree



Directed tree



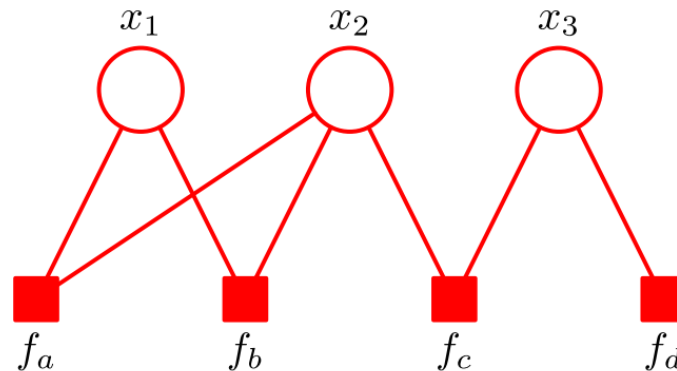
Directed polytree

- **Undirected graph tree:** graph in which there is one, and only one, path between any pair of nodes
- **Directed graph tree:** there is a single node, called the root, which has no parents, and all other nodes have one parent
  - Thus the moralization step will not add any links
- **Polytree:** nodes in a directed graph that have more than one parent, but there is still only one path (ignoring the direction of the arrows) between any two nodes
- Before introducing inference algorithm, let's discuss a general form: factor graph

# Factor Graphs

- Observations: Both directed and undirected graphs allow a global function of several variables to be expressed as a product of factors over subsets of those variables
- Factor graphs make this decomposition explicit by introducing additional nodes for the factors

$$p(\mathbf{x}) = \prod_s f_s(\mathbf{x}_s)$$

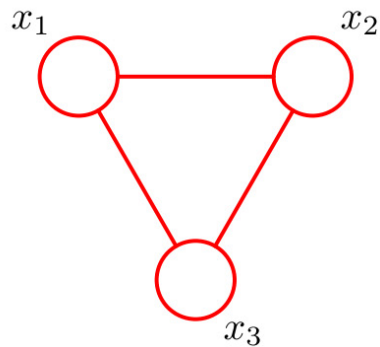


Factor graphs are said to be bipartite



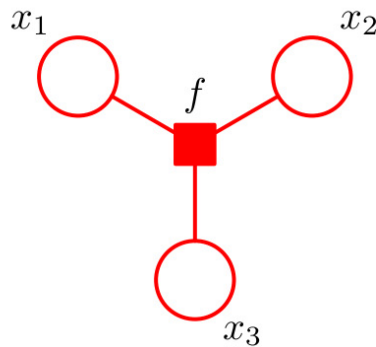
# Factor Graphs

- Undirected graphs to factor graphs



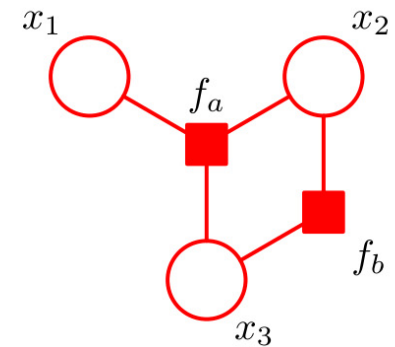
An undirected graph with a single clique potential

$$\psi(x_1, x_2, x_3)$$



A factor graph representing the same distribution with factor

$$f(x_1, x_2, x_3) = \psi(x_1, x_2, x_3)$$

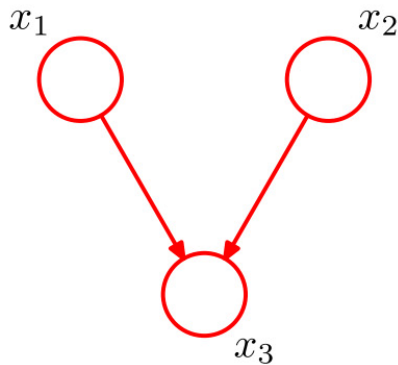


Another factor graph representing the same distribution

$$\begin{aligned} f_a(x_1, x_2, x_3) f_b(x_1, x_2) \\ = \psi(x_1, x_2, x_3) \end{aligned}$$

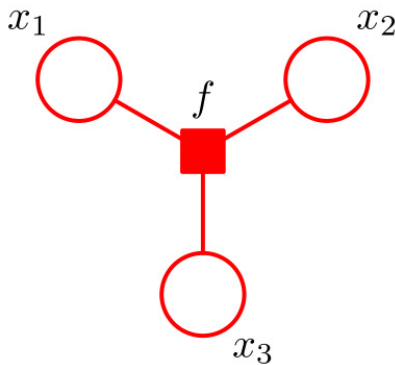
# Factor Graphs

- Directed graphs to factor graphs



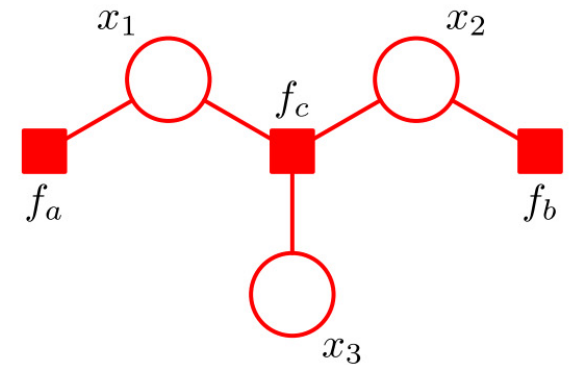
A directed graph  
with factorization

$$p(x_1)p(x_2)p(x_3|x_1, x_2)$$



A factor graph  
representing the  
same distribution  
with factor

$$f(x_1, x_2, x_3) = p(x_1)p(x_2) \\ p(x_3|x_1, x_2)$$



Another factor graph  
representing the  
same distribution

$$f_a(x_1) = p(x_1) \\ f_b(x_2) = p(x_2) \\ f_c(x_1, x_2, x_3) = p(x_3|x_1, x_2)$$

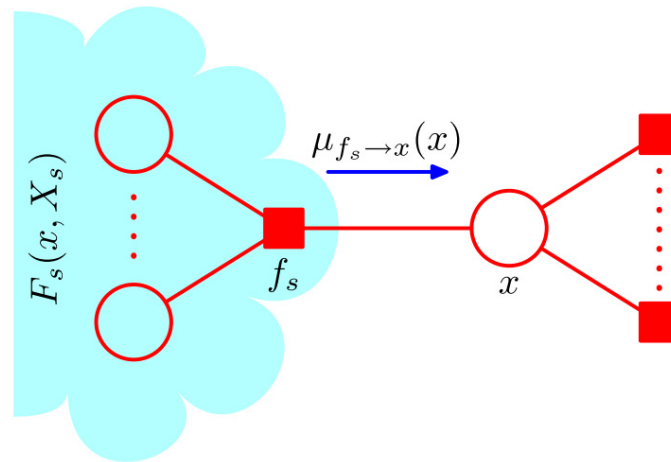
# Inference on a Tree: Sum-Product

- Consider the marginal of a particular variable  $x$  on the factor graph tree

$$p(x) = \sum_{\mathbf{x} \setminus x} p(\mathbf{x})$$

$$p(x) = \prod_{s \in \text{ne}(x)} \sum_{X_s} F_s(x, X_s)$$

- $\text{ne}(x)$ : set of neighbor factors of  $x$
- $X_s$ : set of all variables in the subtree connected to the variable node  $x$  via the factor node
- $F_s(x, X_s)$ : the product of all the factors in the group associated with factor  $f_s$

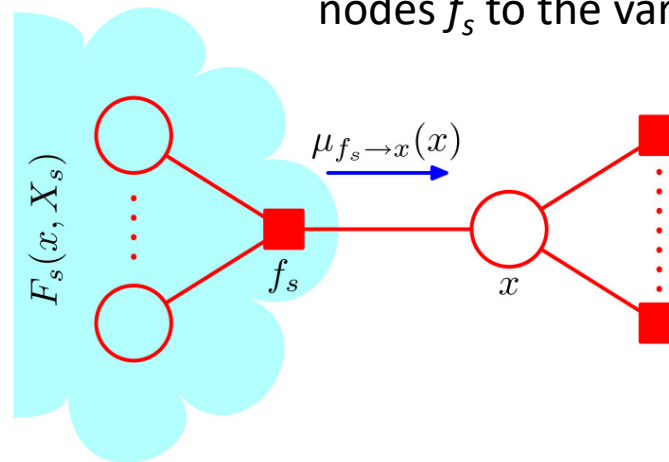


# Message Passing

- Consider the marginal of a particular variable  $x$  on the factor graph tree

$$p(x) = \prod_{s \in \text{ne}(x)} \left[ \sum_{X_s} F_s(x, X_s) \right]$$
$$\equiv \prod_{s \in \text{ne}(x)} \mu_{f_s \rightarrow x}(x)$$

messages from the factor  
nodes  $f_s$  to the variable node  $x$

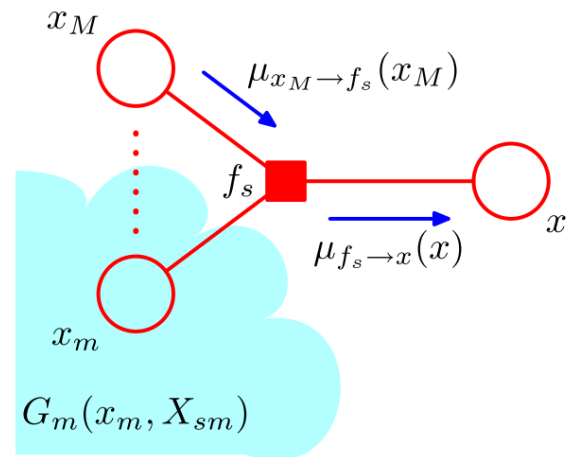
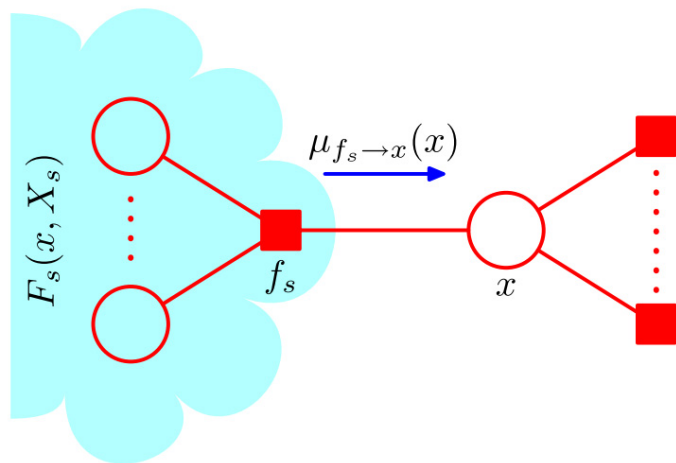


# Message Passing Iteration

- Denote  $\{x, x_1, \dots, x_M\}$  as the set of variables on which the factor  $f_s$  depends, then

$$F_s(x, X_s) = f_s(x, x_1, \dots, x_M) G_1(x_1, X_{s1}) \cdots G_M(x_M, X_{sM})$$

$$\begin{aligned} \mu_{f_s \rightarrow x}(x) &= \sum_{x_1} \cdots \sum_{x_M} f_s(x, x_1, \dots, x_M) \prod_{m \in \text{ne}(f_s) \setminus x} \left[ \sum_{X_{sm}} G_m(x_m, X_{sm}) \right] \\ &= \sum_{x_1} \cdots \sum_{x_M} f_s(x, x_1, \dots, x_M) \prod_{m \in \text{ne}(f_s) \setminus x} \mu_{x_m \rightarrow f_s}(x_m) \end{aligned}$$

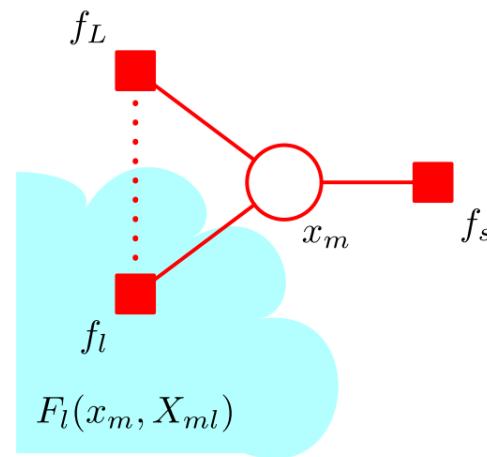
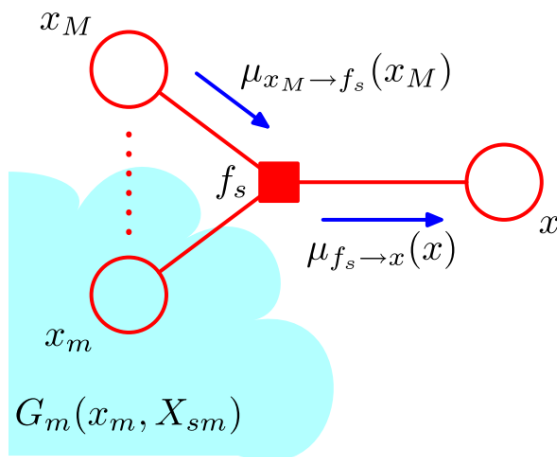


# Message Passing Iteration

- Denote  $\{x, x_1, \dots, x_M\}$  as the set of variables on which the factor  $f_s$  depends, then

$$F_s(x, X_s) = f_s(x, x_1, \dots, x_M) G_1(x_1, X_{s1}) \cdots G_M(x_M, X_{sM})$$

$$G_m(x_m, X_{sm}) = \prod_{l \in \text{ne}(x_m) \setminus f_s} F_l(x_m, X_{ml})$$



# Two Types of Messages

- Messages from factor nodes to variable nodes

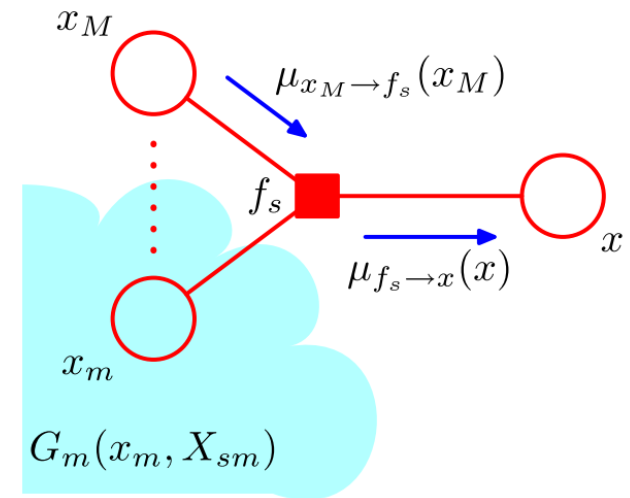
$$\mu_{f_s \rightarrow x}(x) = \sum_{X_s} F_s(x, X_s)$$

$$F_s(x, X_s) = f_s(x, x_1, \dots, x_M) G_1(x_1, X_{s1}) \cdots G_M(x_M, X_{sM})$$

- Messages from variable nodes to factor nodes

$$\mu_{x_m \rightarrow f_s}(x_m) = \sum_{X_{sm}} G_m(x_m, X_{sm})$$

$$G_m(x_m, X_{sm}) = \prod_{l \in \text{ne}(x_m) \setminus f_s} F_l(x_m, X_{ml})$$

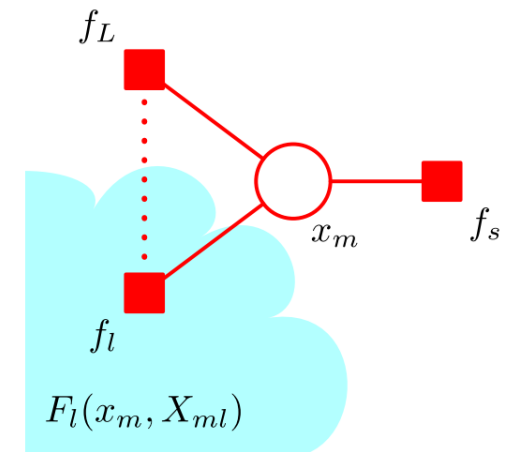
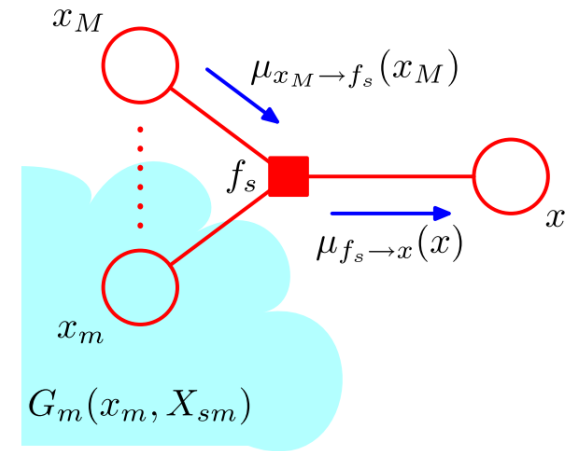


# Two Types of Messages

- Relationships of two types of messages

$$\begin{aligned}\mu_{x_m \rightarrow f_s}(x_m) &= \sum_{X_{sm}} G_m(x_m, X_{sm}) \\ &= \sum_{X_{sm}} \prod_{l \in \text{ne}(x_m) \setminus f_s} F_l(x_m, X_{ml})\end{aligned}$$

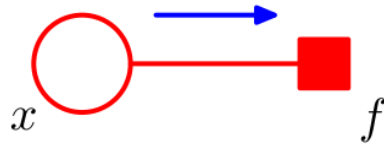
$$\begin{aligned}(\text{Tree structure}) \quad &= \prod_{l \in \text{ne}(x_m) \setminus f_s} \left[ \sum_{X_{ml}} F_l(x_m, X_{ml}) \right] \\ &= \prod_{l \in \text{ne}(x_m) \setminus f_s} \mu_{f_l \rightarrow x_m}(x_m)\end{aligned}$$



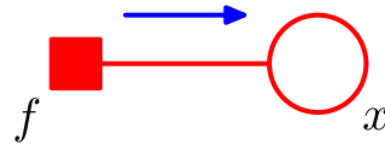


# Start of Recursion

$$\mu_{x \rightarrow f}(x) = 1$$



$$\mu_{f \rightarrow x}(x) = f(x)$$



- Messages from variable nodes to factor nodes

$$\mu_{x_m \rightarrow f_s}(x_m) = \sum_{X_{sm}} G_m(x_m, X_{sm})$$

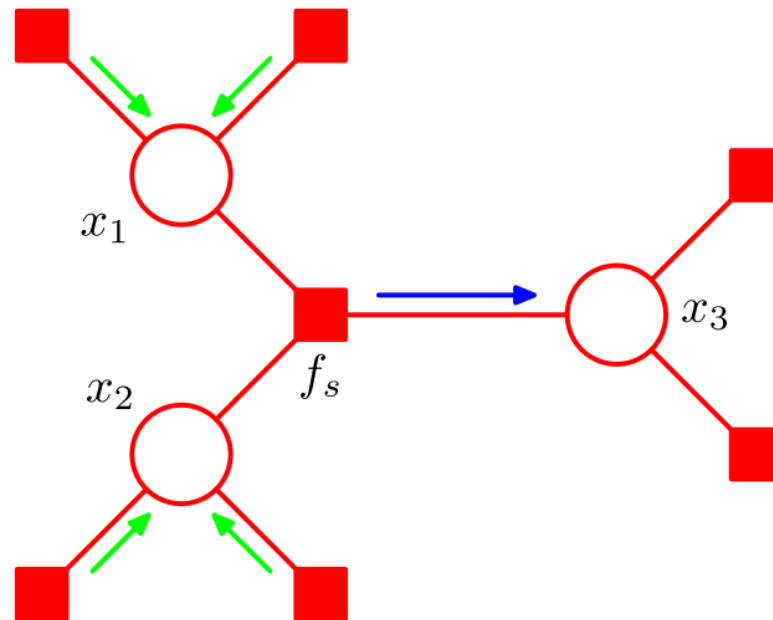
$$G_m(x_m, X_{sm}) = \prod_{l \in \text{ne}(x_m) \setminus f_s} F_l(x_m, X_{ml})$$

- Messages from factor nodes to variable nodes

$$\mu_{f_s \rightarrow x}(x) = \sum_{X_s} F_s(x, X_s)$$

$$F_s(x, X_s) = f_s(x, x_1, \dots, x_M) G_1(x_1, X_{s1}) \cdots G_M(x_M, X_{sM})$$

# Marginal of Variables of a Factor



$$p(\mathbf{x}_s) = f_s(\mathbf{x}_s) \prod_{i \in \text{ne}(f_s)} \mu_{x_i \rightarrow f_s}(x_i)$$

# An Example for Practice

- Unnormalized joint distribution

$$\tilde{p}(\mathbf{x}) = f_a(x_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4)$$

- Designate node  $x_3$  as the root, messages

$$\mu_{x_1 \rightarrow f_a}(x_1) = 1$$

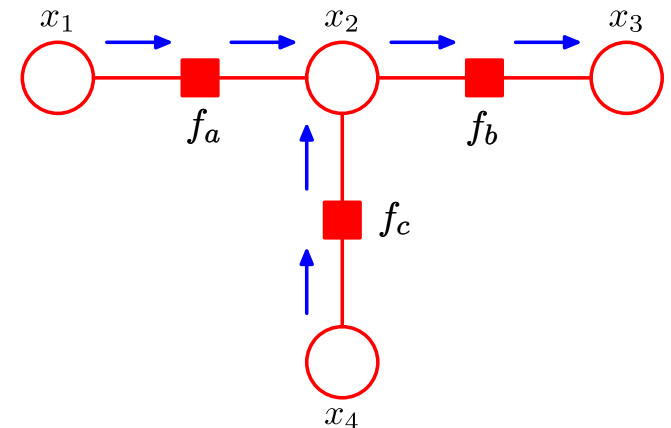
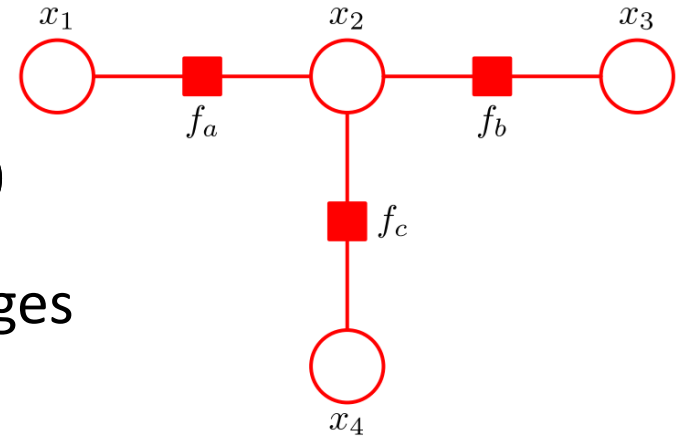
$$\mu_{f_a \rightarrow x_2}(x_2) = \sum_{x_1} f_a(x_1, x_2)$$

$$\mu_{x_4 \rightarrow f_c}(x_4) = 1$$

$$\mu_{f_c \rightarrow x_2}(x_2) = \sum_{x_4} f_c(x_2, x_4)$$

$$\mu_{x_2 \rightarrow f_b}(x_2) = \mu_{f_a \rightarrow x_2}(x_2) \mu_{f_c \rightarrow x_2}(x_2)$$

$$\mu_{f_b \rightarrow x_3}(x_3) = \sum_{x_2} f_b(x_2, x_3) \mu_{x_2 \rightarrow f_b}$$



# An Example for Practice

- Unnormalized joint distribution

$$\tilde{p}(\mathbf{x}) = f_a(x_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4)$$

- Messages from the root node out to the leaf nodes

$$\mu_{x_3 \rightarrow f_b}(x_3) = 1$$

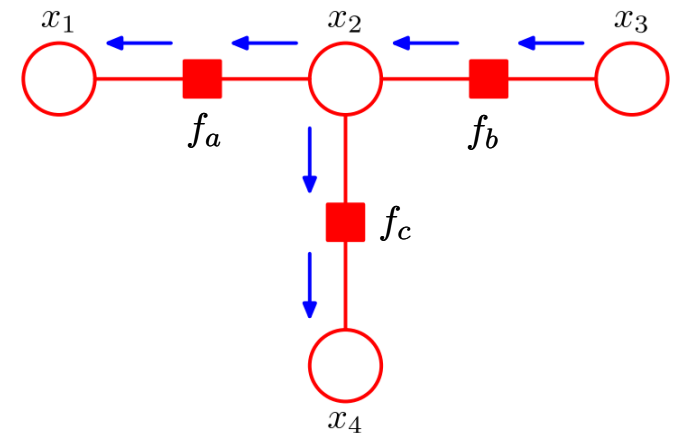
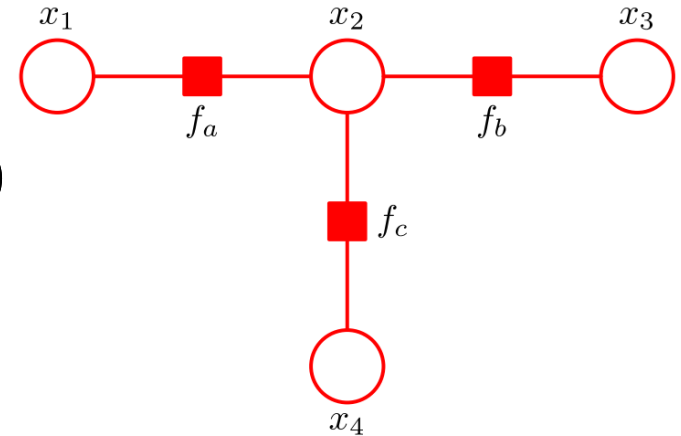
$$\mu_{f_b \rightarrow x_2}(x_2) = \sum_{x_3} f_b(x_2, x_3)$$

$$\mu_{x_2 \rightarrow f_a}(x_2) = \mu_{f_b \rightarrow x_2}(x_2) \mu_{f_c \rightarrow x_2}(x_2)$$

$$\mu_{f_a \rightarrow x_1}(x_2) = \sum_{x_2} f_a(x_1, x_2) \mu_{x_2 \rightarrow f_a}(x_2)$$

$$\mu_{x_2 \rightarrow f_c}(x_2) = \mu_{f_a \rightarrow x_2}(x_2) \mu_{f_b \rightarrow x_2}(x_2)$$

$$\mu_{f_c \rightarrow x_4}(x_4) = \sum_{x_2} f_c(x_2, x_4) \mu_{x_2 \rightarrow f_c}(x_2)$$



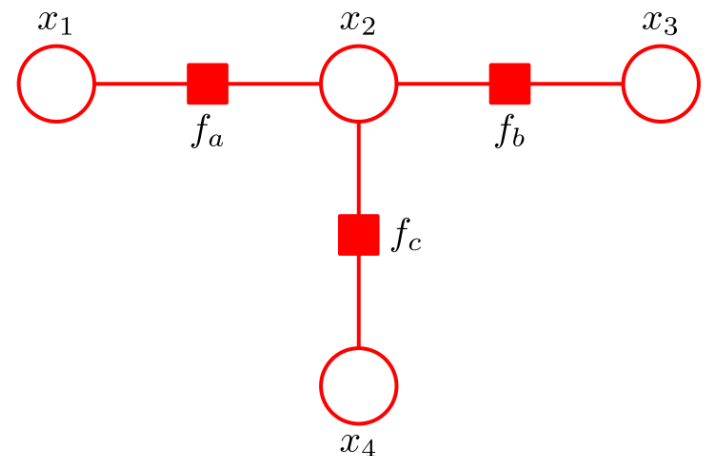
# An Example for Practice

- Verify the marginal  $p(x_2)$

$$\begin{aligned}\tilde{p}(x_2) &= \mu_{f_a \rightarrow x_2}(x_2) \mu_{f_b \rightarrow x_2}(x_2) \mu_{f_c \rightarrow x_2}(x_2) \\ &= \left[ \sum_{x_1} f_a(x_1, x_2) \right] \left[ \sum_{x_3} f_b(x_2, x_3) \right] \left[ \sum_{x_4} f_c(x_2, x_4) \right] \\ &= \sum_{x_1} \sum_{x_3} \sum_{x_4} f_a(x_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4) \\ &= \sum_{x_1} \sum_{x_3} \sum_{x_4} \tilde{p}(\mathbf{x})\end{aligned}$$

Consistent with

$$p(x) = \sum_{\mathbf{x} \setminus x} p(\mathbf{x})$$



# Conditioned on Observed Variables

- Suppose we partition  $\mathbf{x}$  into

- hidden variables  $\mathbf{h}$
- observed variables  $\mathbf{v} = \hat{\mathbf{v}}$

- For the calculation  $p(h|\mathbf{v} = \hat{\mathbf{v}}) = \sum_{\mathbf{x} \setminus h} p(\mathbf{x})$

- We just need to update  $p(\mathbf{x})$  as

$$p(\mathbf{x}) \leftarrow p(\mathbf{x}) \prod_i I(v_i = \hat{v}_i)$$

- The sum-product algorithm is efficient