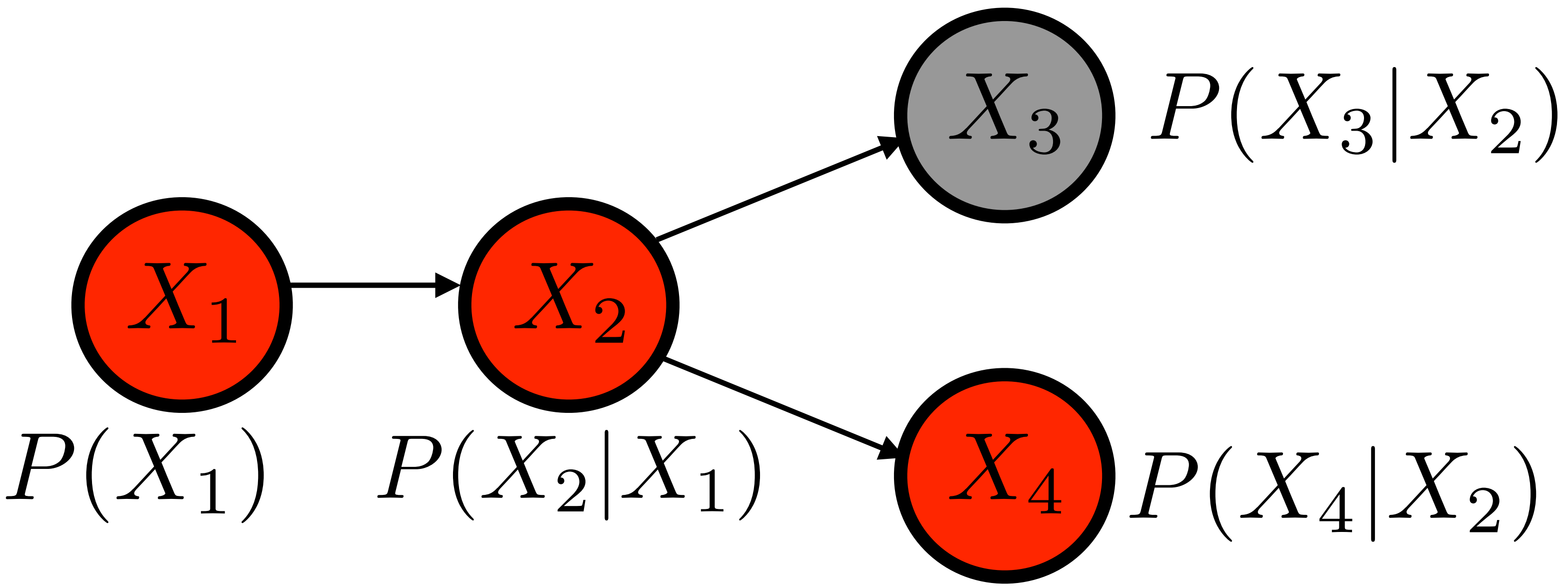
Machine Learning for Data Science (CS4786)
Lecture 23

Graphical Models

Course Webpage :
http://www.cs.cornell.edu/Courses/cs4786/2016sp/

Message Passing Example



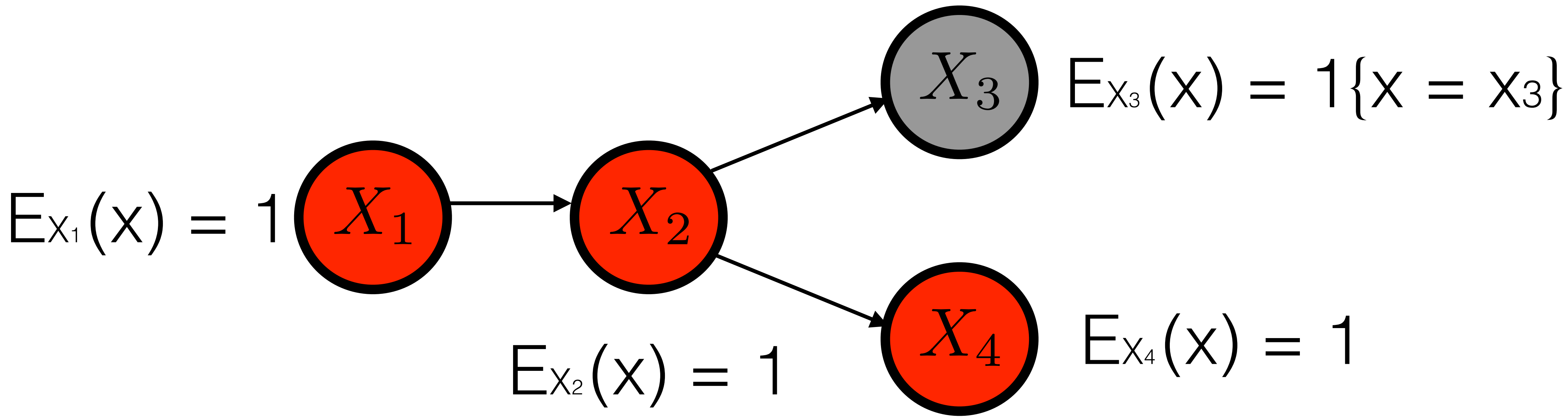
Message to Parent Xj

$$\sum_{x,\text{all parents but X}_\text{j}} E_{X_i}(x) P(X_i = x|\text{Parent}(X_i) = u)(\text{product of all messages but one from X}_\text{j})$$

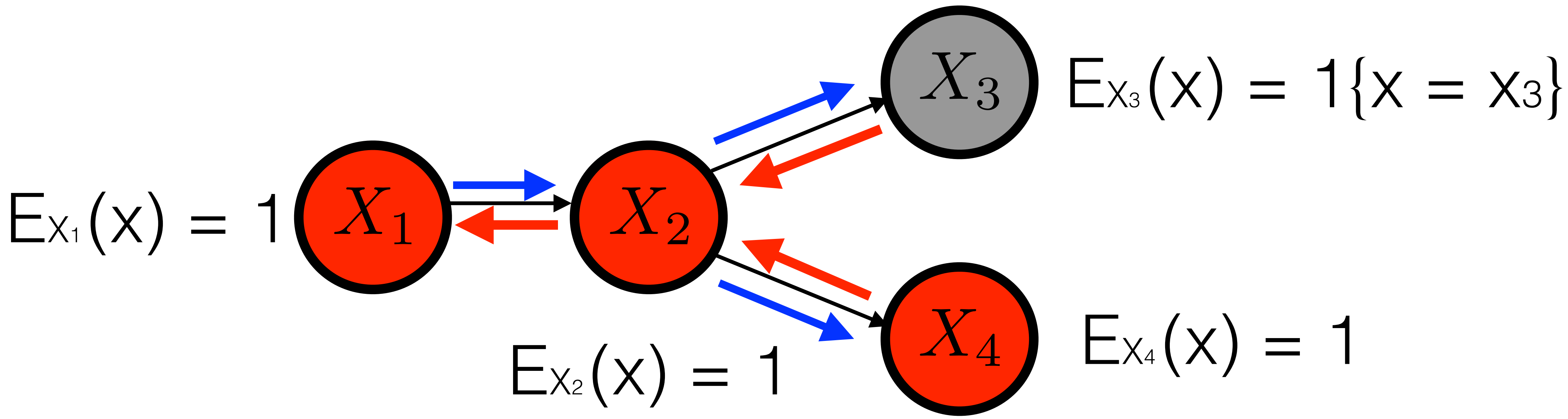
Message to child Xj

$$\sum_{\text{all parents}} E_{X_i}(x) P(X_i = x|\text{Parent}(X_i) = u)(\text{product of all messages but one from X}_\text{j})$$

Message Passing Example

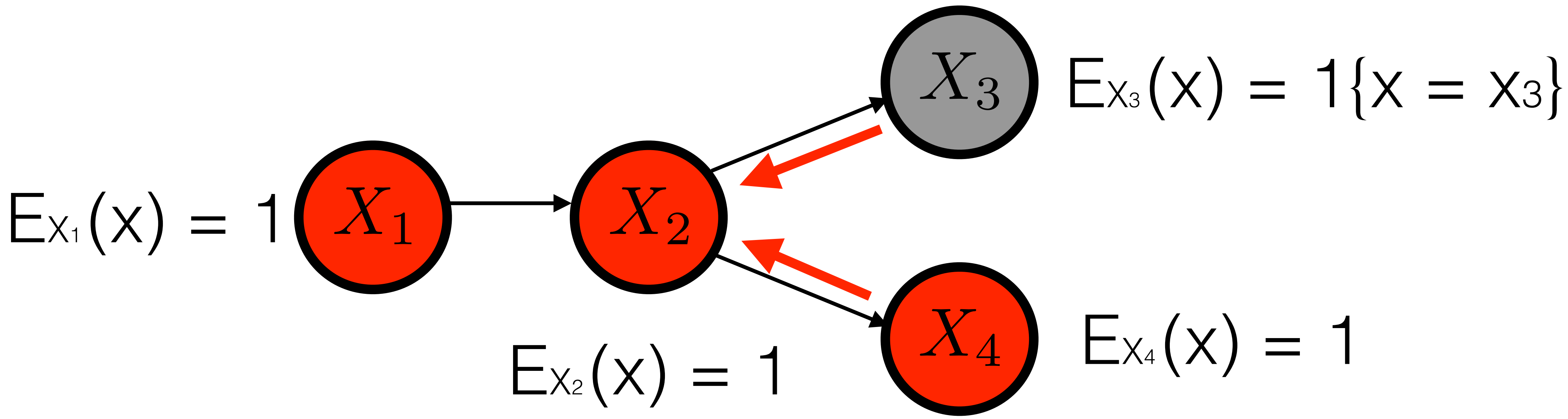


Message Passing Example



Round 0 : All messages are 1's

Message Passing Example



Message to Parent Xj

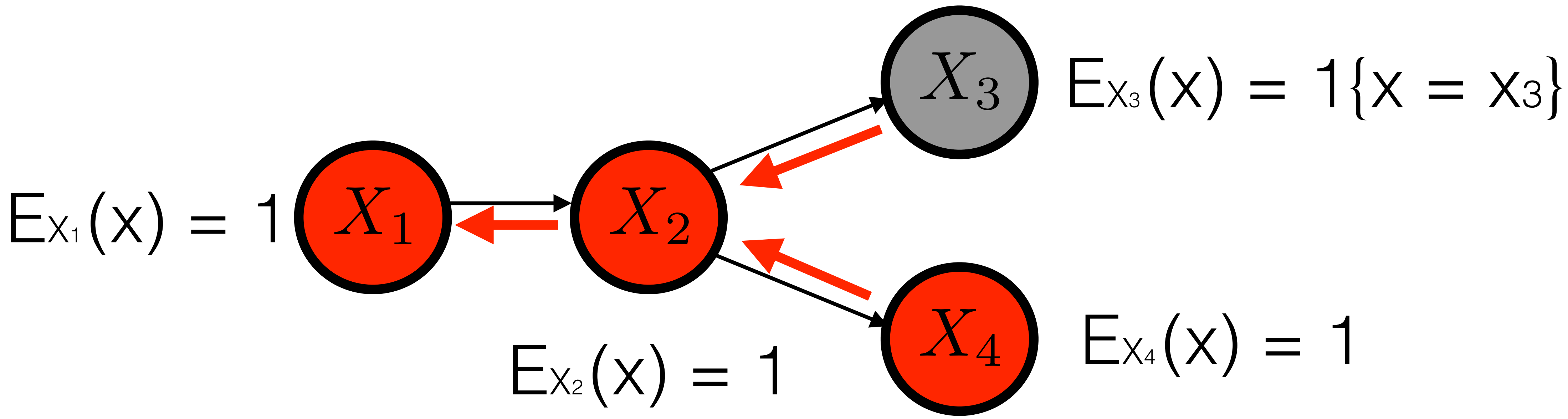
$$\sum_{x,\text{all parents but } \mathrm{X_j}} E_{X_i}(x) P(X_i = x | \text{Parent}(X_i) = u)(\text{product of all messages but one from } \mathrm{X_j})$$

Round 1 : Leaves have exactly one neighbor

$$m_{3\to2}(u_2) = P(X_3 = x_3 | X_2 = u_2)$$

$$m_{4\to2}(u_2) = \sum_x P(X_3 = x | X_2 = u_2) = 1$$

Message Passing Example

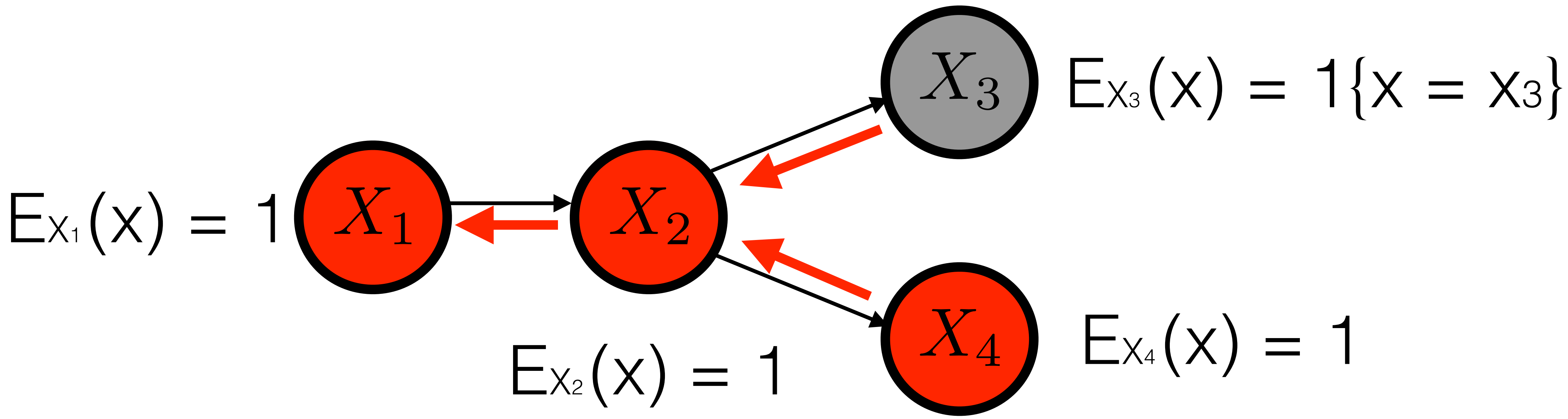


Message to Parent Xj

$$\sum_{x,\text{all parents but } X_j} E_{X_i}(x) P(X_i = x | \text{Parent}(X_i) = u)(\text{product of all messages but one from } X_j)$$

$$\begin{aligned} m_{2\to1}(u_1) &= \sum_x P(X_2 = x | X_1 = u_1)\left(m_{3\to2}(x) \times m_{2\to2}(x)\right) \\ &= \sum_x P(X_2 = x | X_1 = u_1) P(X_3 = x_3 | X_2 = x) = P(X_3 = x_3 | X_1 = u_1) \end{aligned}$$

Message Passing Example



Message to Parent Xj

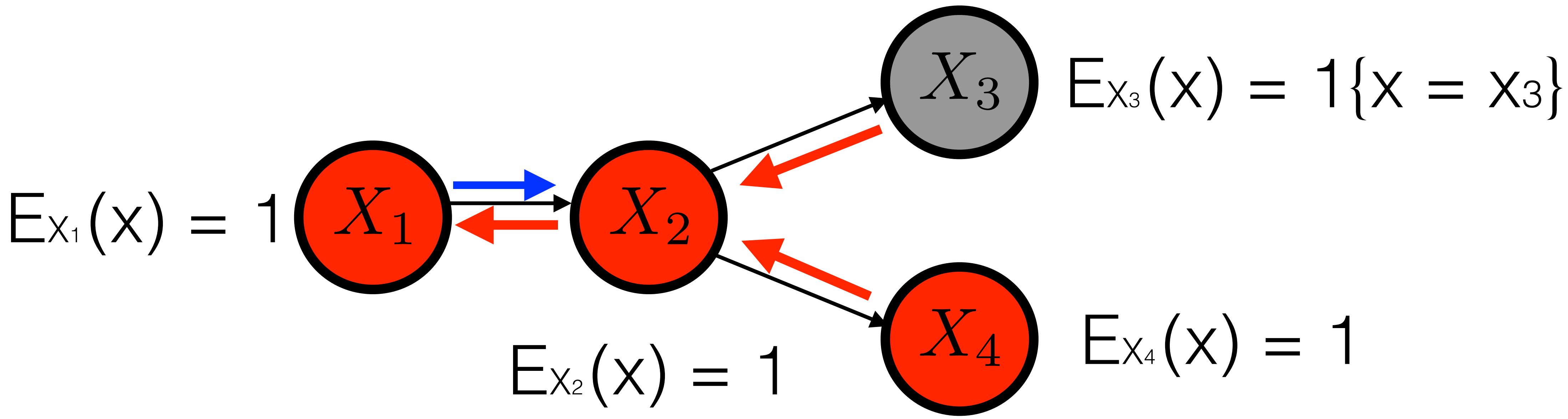
$$\sum_{x,\text{all parents but X}_\text{j}} E_{X_i}(x) P(X_i = x|\text{Parent}(X_i) = u)(\text{product of all messages but one from X}_\text{j})$$

Round 2 :

$$m_{2\to1}(u_1) = \sum_x P(X_2 = x|X_1 = u_1)\left(m_{3\to2}(x) \times m_{2\to2}(x)\right)$$

$$= \sum_x P(X_2 = x|X_1 = u_1)P(X_3 = x_3|X_2 = x) = P(X_3 = x_3|X_1 = u_1)$$

Message Passing Example



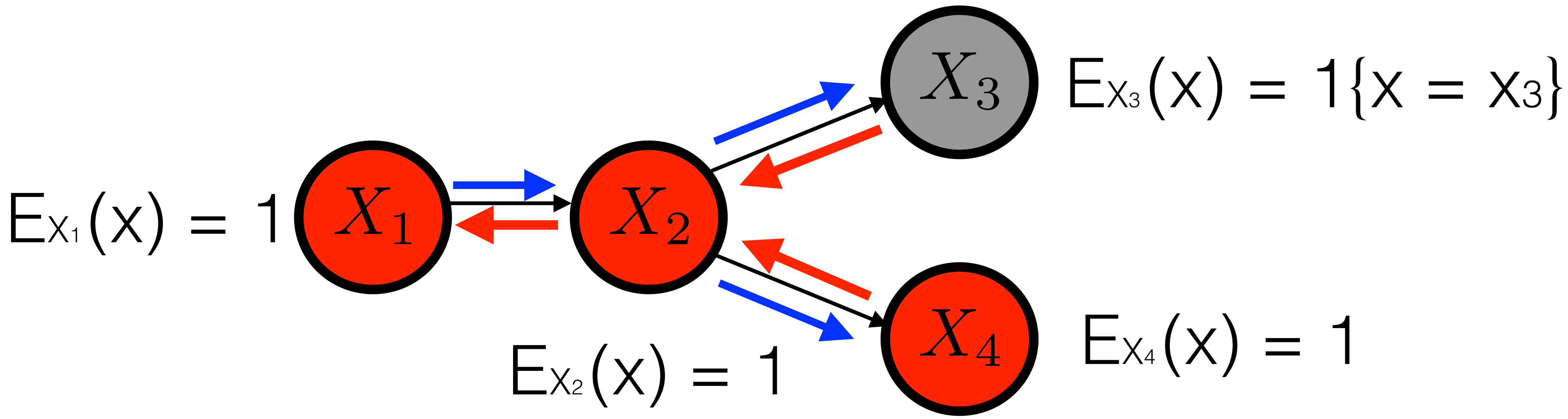
Message to child Xj

$$\sum_{\text{all parents}} E_{X_i}(x) P(X_i = x | \text{Parent}(X_i) = u) (\text{product of all messages but one from } \mathrm{X_j})$$

Round 3 :

$$m_{1\to 2}(u_1) = P(X_1 = u_1)$$

Message Passing Example



Round 3 :

$$m_{2\to3}(u_2) = \sum_{x_1} P(X_2 = u_2|X_1 = x_1)\left(m_{1\to2}(x_1) \times m_{2\to4}(u_2)\right)$$

$$= \sum_{x_1} P(X_2 = u_2|X_1 = x_1)P(X_1 = x_1) = P(X_2 = u_2)$$

$$m_{2\to4}(u_2) = \sum_{x_1} P(X_2 = u_2|X_1 = x_1)\left(m_{1\to2}(x_1) \times m_{2\to3}(u_2)\right)$$

$$= \sum_{x_1} P(X_2 = u_2|X_1 = x_1)P(X_1 = x_1)P(X_3 = x_3|X_2 = u_2)$$

$$= P(X_2 = u_2, X_3 = x_3)$$

Belief Propagation

For any node X_i

- Incoming message to node from children:

$$\lambda(x) = E_{X_i}(x) \prod_{j \in \text{children}(X_i)} \lambda_{X_j}(x)$$

- Incoming message from Parents:

$$\pi(x) = \sum_u P(X_i = x | \text{Parent}(X_i) = u) \prod_{k \in \text{Parent}(X_i)} \pi_{X_i}(u_k)$$

- Outgoing message to Parent X_j:

$$\lambda_{X_i}(u_i) \propto \sum_x \lambda(x) \sum_{u \setminus u_i} P(X_i = x | \text{Parent}(X_i) = u) \prod_{k \neq i} \pi_{X_i}(u_k)$$

- Outgoing message to child X_j:

$$\pi_{X_j}(x) \propto \pi(x) E_{X_i}(x) \prod_{k \neq j} \lambda_{X_j}(x)$$

Parameter Estimation (Learning)

- What are the parameters for a Baysian Network?
    - The conditional probability distributions/tables/density functions

Parameter Estimation (Learning)

- MLE: n independent samples $(X_1^1, \ldots, X_N^1), \ldots, (X_1^n, \ldots, X_N^n)$ where each $(X_1^t, \ldots, X_N^t)$ is drawn from the Bayesian network

$$\arg\max_\theta \sum_{t=1}^{n} \log(P_\theta(X_1^t, \ldots, X_N^t))$$

$$= \arg\max_\theta \sum_{t=1}^{n} \sum_{i=1}^{N} \log(P_\theta(X_i^t | \text{Parent}(X_i^t)))$$

If θ_i is the parameter only involving $P_\theta(X_i^t | \text{Parent}(X_i^t))$ then

$$\theta_i^{MLE} = \arg\max_{\theta_i} \sum_{t=1}^{n} \log(P_{\theta_i}(X_i^t | \text{Parent}(X_i^t)))$$

Parameter Estimation (Learning)

- Simple case of finite outcomes

$$\theta_i^{MLE} = \text{empirical conditional probability table}$$

Parameter Estimation: Latent Variables

- EM Algorithm: Initialize parameters randomly
- For $j = 1$ to convergence
  - E-step: For each of the Latent variable X_i, perform inference to compute

$$Q^{(j)}(\text{Latent variables}) = P_{\theta^{(j-1)}}(\text{Latent variables}|\text{Observation})$$

  - M-step:

$$\theta^{(j)} = \arg\max_{\theta} \sum_{\text{Latent variables}} Q^{(j)}(\text{Latent variables}) \sum_{t=1}^{n} \log P_{\theta}(X_1^t, \ldots, X_N^t)$$

which can be simplified to:

$$\theta_i^{(j)} = \arg\max_{\theta_i} \sum_{\text{Latent}} Q^{(j)}(\text{Latent}) \sum_{t=1}^{n} \log P_{\theta_i}(X_i^t|\text{Parent}(X_i^t))$$

Parameter Estimation: Latent Variables

M-step for simple case of finite outcomes

$\theta_i^{(j)}$ = empirical conditional probability table weighted by $Q^{(j)}$

For HMM this is called the Baum Welch algorithm

Inference is Computationally Hard!

- Belief propagation is exact on trees
- For general graphs, belief propagation need not work
- Inference for general graphs can be computationally hard

Can we perform inference approximately?

Inference Via Sampling

1. Sample from the generative model
2. Calculate empirical marginals
3. Might require many samples to be accurate

Markov Networks

- Not all distributions can be represented by Bayesian networks
- We also have undirected graphical models.
- Undirected graph $G = (V, E)$ and a set of RV's $X_1, \ldots, X_N$ form a markov network if
  - Any two non adjacent variables are conditionally independent given all other variables
  - Given its neighbors a variable is conditionally independent of all other variables
  - Any two sets of variables are conditionally independent given a separating set