

732A96/TDDE15 Advanced Machine Learning

Graphical Models

Jose M. Peña
IDA, Linköping University, Sweden

Lecture 2: Probabilistic Inference

Contents

- ▶ Probabilistic Inference for BNs
 - ▶ Naive Solution
 - ▶ Variable Elimination
- ▶ Probabilistic Inference for MNs
- ▶ Most Probable Configuration

Literature

- ▶ Main source
 - ▶ Koller, D. and Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009. Chapter 9.2-9.4.
- ▶ Additional source
 - ▶ Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer, 2006. Chapter 8.

Probabilistic Inference for BNs: Naive Solution

- What is the state of a random variable X_k if a random variable X_i is observed to be in the state x_i ?

$$p(x_k | x_i) = \frac{p(x_k, x_i)}{p(x_i)} = \frac{\sum_{x \setminus \{x_i, x_k\}} p(x)}{\sum_{x \setminus x_i} p(x)}$$

- For instance, $p(d) = \sum_{a,b,c} p(a, b, c, d) = \sum_{a,b,c} p(a)p(b|a)p(c|b)p(d|c)$ for the DAG $A \rightarrow B \rightarrow C \rightarrow D$.

$$\begin{array}{l} P(a^1) \quad P(b^1 | a^1) \quad P(c^1 | b^1) \quad P(d^1 | c^1) \\ + P(a^2) \quad P(b^1 | a^2) \quad P(c^1 | b^1) \quad P(d^1 | c^1) \\ + P(a^1) \quad P(b^2 | a^1) \quad P(c^1 | b^2) \quad P(d^1 | c^1) \\ + P(a^2) \quad P(b^2 | a^2) \quad P(c^1 | b^2) \quad P(d^1 | c^1) \\ + P(a^1) \quad P(b^1 | a^1) \quad P(c^2 | b^1) \quad P(d^1 | c^2) \\ + P(a^2) \quad P(b^1 | a^2) \quad P(c^2 | b^1) \quad P(d^1 | c^2) \\ + P(a^1) \quad P(b^2 | a^1) \quad P(c^2 | b^2) \quad P(d^1 | c^2) \\ + P(a^2) \quad P(b^2 | a^2) \quad P(c^2 | b^2) \quad P(d^1 | c^2) \end{array}$$

$$\begin{array}{l} P(a^1) \quad P(b^1 | a^1) \quad P(c^1 | b^1) \quad P(d^2 | c^1) \\ + P(a^2) \quad P(b^1 | a^2) \quad P(c^1 | b^1) \quad P(d^2 | c^1) \\ + P(a^1) \quad P(b^2 | a^1) \quad P(c^1 | b^2) \quad P(d^2 | c^1) \\ + P(a^2) \quad P(b^2 | a^2) \quad P(c^1 | b^2) \quad P(d^2 | c^1) \\ + P(a^1) \quad P(b^1 | a^1) \quad P(c^2 | b^1) \quad P(d^2 | c^2) \\ + P(a^2) \quad P(b^1 | a^2) \quad P(c^2 | b^1) \quad P(d^2 | c^2) \\ + P(a^1) \quad P(b^2 | a^1) \quad P(c^2 | b^2) \quad P(d^2 | c^2) \\ + P(a^2) \quad P(b^2 | a^2) \quad P(c^2 | b^2) \quad P(d^2 | c^2) \end{array}$$

Figure 9.2 Computing $P(D)$ by summing over the joint distribution for a chain $A \rightarrow B \rightarrow C \rightarrow D$; all of the variables are binary valued.

Probabilistic Inference for BNs: Naive Solution

$$\begin{array}{cccc}
 P(a^1) & P(b^1 | a^1) & P(c^1 | b^1) & P(d^1 | c^1) \\
 + & P(a^2) & P(b^1 | a^2) & P(c^1 | b^1) & P(d^1 | c^1) \\
 + & P(a^1) & P(b^2 | a^1) & P(c^1 | b^2) & P(d^1 | c^1) \\
 + & P(a^2) & P(b^2 | a^2) & P(c^1 | b^2) & P(d^1 | c^1) \\
 + & P(a^1) & P(b^1 | a^1) & P(c^2 | b^1) & P(d^1 | c^2) \\
 + & P(a^2) & P(b^1 | a^2) & P(c^2 | b^1) & P(d^1 | c^2) \\
 + & P(a^1) & P(b^2 | a^1) & P(c^2 | b^2) & P(d^1 | c^2) \\
 + & P(a^2) & P(b^2 | a^2) & P(c^2 | b^2) & P(d^1 | c^2)
 \end{array}$$

$$\begin{array}{cccc}
 P(a^1) & P(b^1 | a^1) & P(c^1 | b^1) & P(d^2 | c^1) \\
 + & P(a^2) & P(b^1 | a^2) & P(c^1 | b^1) & P(d^2 | c^1) \\
 + & P(a^1) & P(b^2 | a^1) & P(c^1 | b^2) & P(d^2 | c^1) \\
 + & P(a^2) & P(b^2 | a^2) & P(c^1 | b^2) & P(d^2 | c^1) \\
 + & P(a^1) & P(b^1 | a^1) & P(c^2 | b^1) & P(d^2 | c^2) \\
 + & P(a^2) & P(b^1 | a^2) & P(c^2 | b^1) & P(d^2 | c^2) \\
 + & P(a^1) & P(b^2 | a^1) & P(c^2 | b^2) & P(d^2 | c^2) \\
 + & P(a^2) & P(b^2 | a^2) & P(c^2 | b^2) & P(d^2 | c^2)
 \end{array}$$

Figure 9.2 Computing $P(D)$ by summing over the joint distribution for a chain $A \rightarrow B \rightarrow C \rightarrow D$; all of the variables are binary valued.

$$\begin{array}{ccc}
 (P(a^1)P(b^1 | a^1) + P(a^2)P(b^1 | a^2)) & P(c^1 | b^1) & P(d^1 | c^1) \\
 + & (P(a^1)P(b^2 | a^1) + P(a^2)P(b^2 | a^2)) & P(c^1 | b^2) & P(d^1 | c^1) \\
 + & (P(a^1)P(b^1 | a^1) + P(a^2)P(b^1 | a^2)) & P(c^2 | b^1) & P(d^1 | c^2) \\
 + & (P(a^1)P(b^2 | a^1) + P(a^2)P(b^2 | a^2)) & P(c^2 | b^2) & P(d^1 | c^2)
 \end{array}$$

$$\begin{array}{ccc}
 (P(a^1)P(b^1 | a^1) + P(a^2)P(b^1 | a^2)) & P(c^1 | b^1) & P(d^2 | c^1) \\
 + & (P(a^1)P(b^2 | a^1) + P(a^2)P(b^2 | a^2)) & P(c^1 | b^2) & P(d^2 | c^1) \\
 + & (P(a^1)P(b^1 | a^1) + P(a^2)P(b^1 | a^2)) & P(c^2 | b^1) & P(d^2 | c^2) \\
 + & (P(a^1)P(b^2 | a^1) + P(a^2)P(b^2 | a^2)) & P(c^2 | b^2) & P(d^2 | c^2)
 \end{array}$$

Figure 9.3 The first transformation on the sum of figure 9.2

Probabilistic Inference for BNs: Naive Solution

$$\begin{array}{rcl}
 & (P(a^1)P(b^1 \mid a^1) + P(a^2)P(b^1 \mid a^2)) & P(c^1 \mid b^1) \quad P(d^1 \mid c^1) \\
 + & (P(a^1)P(b^2 \mid a^1) + P(a^2)P(b^2 \mid a^2)) & P(c^1 \mid b^2) \quad P(d^1 \mid c^1) \\
 + & (P(a^1)P(b^1 \mid a^1) + P(a^2)P(b^1 \mid a^2)) & P(c^2 \mid b^1) \quad P(d^1 \mid c^2) \\
 + & (P(a^1)P(b^2 \mid a^1) + P(a^2)P(b^2 \mid a^2)) & P(c^2 \mid b^2) \quad P(d^1 \mid c^2) \\
 \\
 & (P(a^1)P(b^1 \mid a^1) + P(a^2)P(b^1 \mid a^2)) & P(c^1 \mid b^1) \quad P(d^2 \mid c^1) \\
 + & (P(a^1)P(b^2 \mid a^1) + P(a^2)P(b^2 \mid a^2)) & P(c^1 \mid b^2) \quad P(d^2 \mid c^1) \\
 + & (P(a^1)P(b^1 \mid a^1) + P(a^2)P(b^1 \mid a^2)) & P(c^2 \mid b^1) \quad P(d^2 \mid c^2) \\
 + & (P(a^1)P(b^2 \mid a^1) + P(a^2)P(b^2 \mid a^2)) & P(c^2 \mid b^2) \quad P(d^2 \mid c^2)
 \end{array}$$

Figure 9.3 The first transformation on the sum of figure 9.2

$$\begin{array}{rcl}
 & \tau_1(b^1) & P(c^1 \mid b^1) \quad P(d^1 \mid c^1) \\
 + & \tau_1(b^2) & P(c^1 \mid b^2) \quad P(d^1 \mid c^1) \\
 + & \tau_1(b^1) & P(c^2 \mid b^1) \quad P(d^1 \mid c^2) \\
 + & \tau_1(b^2) & P(c^2 \mid b^2) \quad P(d^1 \mid c^2) \\
 \\
 & \tau_1(b^1) & P(c^1 \mid b^1) \quad P(d^2 \mid c^1) \\
 + & \tau_1(b^2) & P(c^1 \mid b^2) \quad P(d^2 \mid c^1) \\
 + & \tau_1(b^1) & P(c^2 \mid b^1) \quad P(d^2 \mid c^2) \\
 + & \tau_1(b^2) & P(c^2 \mid b^2) \quad P(d^2 \mid c^2)
 \end{array}$$

Figure 9.4 The second transformation on the sum of figure 9.2

Probabilistic Inference for BNs: Naive Solution

$$\begin{array}{lll}
 & \tau_1(b^1) & P(c^1 | b^1) & P(d^1 | c^1) \\
 + & \tau_1(b^2) & P(c^1 | b^2) & P(d^1 | c^1) \\
 + & \tau_1(b^1) & P(c^2 | b^1) & P(d^1 | c^2) \\
 + & \tau_1(b^2) & P(c^2 | b^2) & P(d^1 | c^2) \\
 \\
 & \tau_1(b^1) & P(c^1 | b^1) & P(d^2 | c^1) \\
 + & \tau_1(b^2) & P(c^1 | b^2) & P(d^2 | c^1) \\
 + & \tau_1(b^1) & P(c^2 | b^1) & P(d^2 | c^2) \\
 + & \tau_1(b^2) & P(c^2 | b^2) & P(d^2 | c^2)
 \end{array}$$

Figure 9.4 The second transformation on the sum of figure 9.2

$$\begin{array}{ll}
 (\tau_1(b^1)P(c^1 | b^1) + \tau_1(b^2)P(c^1 | b^2)) & P(d^1 | c^1) \\
 + (\tau_1(b^1)P(c^2 | b^1) + \tau_1(b^2)P(c^2 | b^2)) & P(d^1 | c^2) \\
 \\
 (\tau_1(b^1)P(c^1 | b^1) + \tau_1(b^2)P(c^1 | b^2)) & P(d^2 | c^1) \\
 + (\tau_1(b^1)P(c^2 | b^1) + \tau_1(b^2)P(c^2 | b^2)) & P(d^2 | c^2)
 \end{array}$$

Figure 9.5 The third transformation on the sum of figure 9.2

Probabilistic Inference for BNs: Naive Solution

$$\begin{aligned} & (\tau_1(b^1)P(c^1 | b^1) + \tau_1(b^2)P(c^1 | b^2)) \quad P(d^1 | c^1) \\ + & (\tau_1(b^1)P(c^2 | b^1) + \tau_1(b^2)P(c^2 | b^2)) \quad P(d^1 | c^2) \\ \\ & (\tau_1(b^1)P(c^1 | b^1) + \tau_1(b^2)P(c^1 | b^2)) \quad P(d^2 | c^1) \\ + & (\tau_1(b^1)P(c^2 | b^1) + \tau_1(b^2)P(c^2 | b^2)) \quad P(d^2 | c^2) \end{aligned}$$

Figure 9.5 The third transformation on the sum of figure 9.2

$$\begin{aligned} & \tau_2(c^1) \quad P(d^1 | c^1) \\ + & \tau_2(c^2) \quad P(d^1 | c^2) \\ \\ & \tau_2(c^1) \quad P(d^2 | c^1) \\ + & \tau_2(c^2) \quad P(d^2 | c^2) \end{aligned}$$

Figure 9.6 The fourth transformation on the sum of figure 9.2

- Then, instead of $p(d) = \sum_{a,b,c} p(a)p(b|a)p(c|b)p(d|c)$, we can do

$$\begin{aligned} p(d) &= \sum_c p(d|c)\tau_2(c) \\ &= \sum_c p(d|c) \sum_b p(c|b)\tau_1(b) \\ &= \sum_c p(d|c) \sum_b p(c|b) \sum_a p(a)p(b|a) \end{aligned}$$

- Do we gain anything ? **Yes**, the former case implies 62 operations (multiplications and additions) and the latter only 18.

Probabilistic Inference for BNs: Variable Elimination

- Let us define a factor $\phi(U)$ as a function $\phi: \text{Values}(U) \rightarrow \mathbb{R}$. Let us also define $\text{Scope}(\phi) = U$.

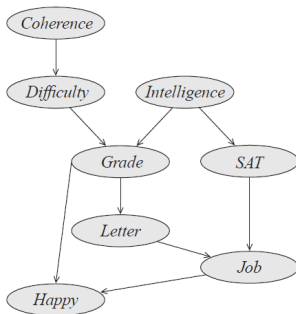


Figure 9.8 The Extended-Student Bayesian network

$$\begin{aligned} P(C, D, I, G, S, L, J, H) &= P(C)P(D | C)P(I)P(G | I, D)P(S | I) \\ &\quad P(L | G)P(J | L, S)P(H | G, J) \\ &= \phi_C(C)\phi_D(D, C)\phi_I(I)\phi_G(G, I, D)\phi_S(S, I) \\ &\quad \phi_L(L, G)\phi_J(J, L, S)\phi_H(H, G, J). \end{aligned}$$

Probabilistic Inference for BNs: Variable Elimination

- ▶ The following algorithm returns $p(y)$ where $Y = X \setminus Z$.

Algorithm 9.1 Sum-product variable elimination algorithm

Procedure Sum-Product-VE (

Φ , // Set of factors

Z , // Set of variables to be eliminated

\prec // Ordering on Z

)

1 Let Z_1, \dots, Z_k be an ordering of Z such that

2 $Z_i \prec Z_j$ if and only if $i < j$

3 **for** $i = 1, \dots, k$

4 $\Phi \leftarrow \text{Sum-Product-Eliminate-Var}(\Phi, Z_i)$

5 $\phi^* \leftarrow \prod_{\phi \in \Phi} \phi$

6 **return** ϕ^*

Procedure Sum-Product-Eliminate-Var (

Φ , // Set of factors

Z // Variable to be eliminated

)

1 $\Phi' \leftarrow \{\phi \in \Phi : Z \in \text{Scope}[\phi]\}$

2 $\Phi'' \leftarrow \Phi - \Phi'$

3 $\psi \leftarrow \prod_{\phi \in \Phi'} \phi$

4 $\tau \leftarrow \sum_Z \psi$

5 **return** $\Phi'' \cup \{\tau\}$

Probabilistic Inference for BNs: Variable Elimination

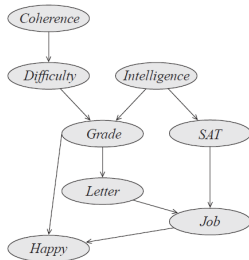


Figure 9.8 The Extended-Student Bayesian network

$$\begin{aligned}
 P(C, D, I, G, S, L, J, H) &= P(C)P(D | C)P(I)P(G | I, D)P(S | I) \\
 &\quad P(L | G)P(J | L, S)P(H | G, J) \\
 &= \phi_C(C)\phi_D(D, C)\phi_I(I)\phi_G(G, I, D)\phi_S(S, I) \\
 &\quad \phi_L(L, G)\phi_J(J, L, S)\phi_H(H, G, J).
 \end{aligned}$$

Step	Variable eliminated	Factors used	Variables involved	New factor
1	C	$\phi_C(C), \phi_D(D, C)$	C, D	$\tau_1(D)$
2	D	$\phi_G(G, I, D), \tau_1(D)$	G, I, D	$\tau_2(G, I)$
3	I	$\phi_I(I), \phi_S(S, I), \tau_2(G, I)$	G, S, I	$\tau_3(G, S)$
4	H	$\phi_H(H, G, J)$	H, G, J	$\tau_4(G, J)$
5	G	$\tau_4(G, J), \tau_3(G, S), \phi_L(L, G)$	G, J, L, S	$\tau_5(J, L, S)$
6	S	$\tau_5(J, L, S), \phi_J(J, L, S)$	J, L, S	$\tau_6(J, L)$
7	L	$\tau_6(J, L)$	J, L	$\tau_7(J)$

Table 9.1 A run of variable elimination for the query $P(J)$

Probabilistic Inference for BNs: Variable Elimination

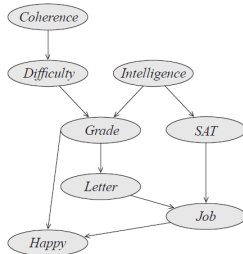


Figure 9.8 The Extended-Student Bayesian network

$$\begin{aligned}
 P(C, D, I, G, S, L, J, H) &= P(C)P(D | C)P(I)P(G | I, D)P(S | I) \\
 &\quad P(L | G)P(J | L, S)P(H | G, J) \\
 &= \phi_C(C)\phi_D(D, C)\phi_I(I)\phi_G(G, I, D)\phi_S(S, I) \\
 &\quad \phi_L(L, G)\phi_J(J, L, S)\phi_H(H, G, J).
 \end{aligned}$$

Step	Variable eliminated	Factors used	Variables involved	New factor
1	G	$\phi_G(G, I, D), \phi_L(L, G), \phi_H(H, G, J)$	G, I, D, L, J, H	$\tau_1(I, D, L, J, H)$
2	I	$\phi_I(I), \phi_S(S, I), \tau_1(I, D, L, S, J, H)$	S, I, D, L, J, H	$\tau_2(D, L, S, J, H)$
3	S	$\phi_J(J, L, S), \tau_2(D, L, S, J, H)$	D, L, S, J, H	$\tau_3(D, L, J, H)$
4	L	$\tau_3(D, L, J, H)$	D, L, J, H	$\tau_4(D, J, H)$
5	H	$\tau_4(D, J, H)$	D, J, H	$\tau_5(D, J)$
6	C	$\phi_C(C), \phi_D(D, C)$	D, J, C	$\tau_6(D)$
7	D	$\tau_5(D, J), \tau_6(D)$	D, J	$\tau_7(J)$

Table 9.2 A different run of variable elimination for the query $P(J)$

Probabilistic Inference for BNs: Variable Elimination

- What is the state of a random variable X_k if a random variable X_i is observed to be in the state x_i ?

$$p(x_k|x_i) = \frac{p(x_k, x_i)}{p(x_i)} = \frac{\sum_{x \setminus \{x_i, x_k\}} p(x)}{\sum_{x \setminus x_i} p(x)}$$

- E.g., $p(d|a) = \frac{\sum_c p(d|c) \sum_b p(d|b)p(a)p(b|a)}{\sum_{a,c} p(d|c) \sum_b p(d|b)p(a)p(b|a)}$ for $A \rightarrow B \rightarrow C \rightarrow D$.
- The following algorithm returns $p(y|e) = \frac{p(y,e)}{p(e)} = \frac{\phi^*}{\alpha}$ where $Y \subseteq X \setminus E$.
- Given a factor $\phi(U)$, let us define the reduced factor $\phi[E=e](Y)$ as a factor with scope $Y = U \setminus E$ such that $\phi[E=e](y) = \phi(y, z)$ where $Z = U \cap E$.

Algorithm 9.2 Using Sum-Product-VE for computing conditional probabilities

Procedure Cond-Prob-VE (

\mathcal{K} , // A network over \mathcal{X}

Y , // Set of query variables

$E = e$ // Evidence

)

- 1 $\Phi \leftarrow$ Factors parameterizing \mathcal{K}
- 2 Replace each $\phi \in \Phi$ by $\phi[E=e]$
- 3 Select an elimination ordering \prec
- 4 $Z \leftarrow \mathcal{X} - Y - E$
- 5 $\phi^* \leftarrow$ Sum-Product-VE(Φ, \prec, Z)
- 6 $\alpha \leftarrow \sum_{y \in \text{Val}(Y)} \phi^*(y)$
- 7 **return** α, ϕ^*

Probabilistic Inference for BNs: Variable Elimination

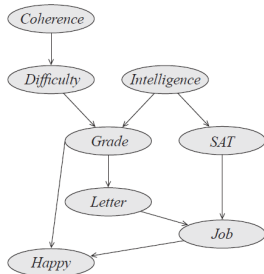


Figure 9.8 The Extended-Student Bayesian network

$$\begin{aligned}
 P(C, D, I, G, S, L, J, H) &= P(C)P(D | C)P(I)P(G | I, D)P(S | I) \\
 &\quad P(L | G)P(J | L, S)P(H | G, J) \\
 &= \phi_C(C)\phi_D(D, C)\phi_I(I)\phi_G(G, I, D)\phi_S(S, I) \\
 &\quad \phi_L(L, G)\phi_J(J, L, S)\phi_H(H, G, J).
 \end{aligned}$$

Step	Variable eliminated	Factors used	Variables involved	New factor
1'	C	$\phi_C(C), \phi_D(D, C)$	C, D	$\tau_1'(D)$
2'	D	$\phi_G[I = i^1](G, D), \phi_I[I = i^1](), \tau_1'(D)$	G, D	$\tau_2'(G)$
5'	G	$\tau_2'(G), \phi_L(L, G), \phi_H[H = h^0](G, J)$	G, L, J	$\tau_5'(L, J)$
6'	S	$\phi_S[I = i^1](S), \phi_J(J, L, S)$	J, L, S	$\tau_6'(J, L)$
7'	L	$\tau_6'(J, L), \tau_5'(J, L)$	J, L	$\tau_7'(J)$

Table 9.3 A run of sum-product variable elimination for $P(J, i^1, h^0)$

Probabilistic Inference for BNs: Variable Elimination

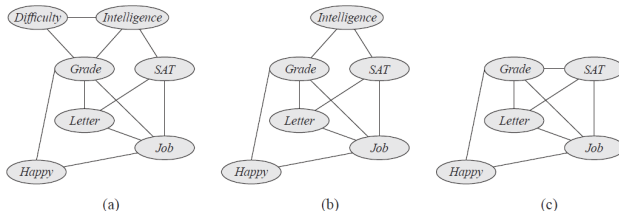


Figure 9.10 Variable elimination as graph transformation in the Student example, using the elimination order of table 9.1: (a) after eliminating D ; (b) after eliminating I ; (c) after eliminating S .

- ▶ VE can be seen as a graph transformation.
- ▶ Given a set of factors Φ such that $X = \cup_{\phi \in \Phi} \text{Scope}(\phi)$, let us define H_Φ as the undirected graph over X that contains an edge $X_i - X_j$ if and only if $X_i, X_j \in \text{Scope}(\phi)$ for some $\phi \in \Phi$.
- ▶ When the variable Z is eliminated, H_Φ changes as follows:
 - ▶ The edge $X_i - X_j$ is added to H_Φ for all $X_i - Z - X_j$ in H_Φ . If this edge was not already in H_Φ , then it is called a fill edge.
 - ▶ The variable Z is removed from H_Φ .

Probabilistic Inference for BNs: Variable Elimination

- ▶ Note that every clique (i.e., maximal complete sets of nodes) in the sequence of graphs H_Φ is the scope of some factor in the VE process.
- ▶ Note also that the size of the largest clique is indicative of the **computational cost** of the VE process.
- ▶ Finally, note that the cliques in the sequence of graphs H_Φ depends on the **elimination ordering**.
- ▶ Unfortunately, finding the optimal elimination ordering is NP-hard. So, we must resort to heuristics.

Algorithm 9.4 Greedy search for constructing an elimination ordering

```
Procedure Greedy-Ordering (  
     $\mathcal{H}$     // An undirected graph over  $\mathcal{X}$  ,  
     $s$     // An evaluation metric  
)  
1  Initialize all nodes in  $\mathcal{X}$  as unmarked  
2  for  $k = 1 \dots |\mathcal{X}|$   
3      Select an unmarked variable  $X \in \mathcal{X}$  that minimizes  $s(\mathcal{H}, X)$   
4       $\pi(X) \leftarrow k$   
5      Introduce edges in  $\mathcal{H}$  between all neighbors of  $X$   
6      Mark  $X$   
7  return  $\pi$ 
```

where the evaluation metric $s(\mathcal{H}, X)$ may be

- ▶ the number of neighbors X has in \mathcal{H} ,
- ▶ the product of the domain cardinality of the neighbors of X in \mathcal{H} , or
- ▶ the number of fill edges to add to \mathcal{H} when eliminating X .

Probabilistic Inference for MNs

- ▶ The VE algorithm can also be used for probabilistic inference in MNs. Simply,
 - ▶ initialize the set of factors Φ to the MN's clique potentials $\{\varphi(k)\}$,
 - ▶ run the VE algorithm, and
 - ▶ normalize the returned unnormalized probability distribution by dividing with the MN's normalization constant Z .

Most Probable Configuration

- ▶ The following algorithm returns $\arg \max_x p(x)$.

Algorithm 13.1 Variable elimination algorithm for MAP. The algorithm can be used both in its max-product form, as shown, or in its max-sum form, replacing factor product with factor addition.

```
Procedure Max-Product-VE (  
     $\Phi$ , // Set of factors over  $X$   
     $\prec$  // Ordering on  $X$   
)  
1  Let  $X_1, \dots, X_k$  be an ordering of  $X$  such that  
2      $X_i \prec X_j$  iff  $i < j$   
3     for  $i = 1, \dots, k$   
4          $(\Phi, \phi_{X_i}) \leftarrow \text{Max-Product-Eliminate-Var}(\Phi, X_i)$   
5      $x^* \leftarrow \text{Traceback-MAP}(\{\phi_{X_i} : i = 1, \dots, k\})$   
6     return  $x^*, \Phi$  //  $\Phi$  contains the probability of the MAP  
  
Procedure Max-Product-Eliminate-Var (  
     $\Phi$ , // Set of factors  
     $Z$  // Variable to be eliminated  
)  
1   $\Phi' \leftarrow \{\phi \in \Phi : Z \in \text{Scope}[\phi]\}$   
2   $\Phi'' \leftarrow \Phi - \Phi'$   
3   $\psi \leftarrow \prod_{\phi \in \Phi'} \phi$   
4   $\tau \leftarrow \max_Z \psi$   
5  return  $(\Phi'' \cup \{\tau\}, \psi)$   
  
Procedure Traceback-MAP (  
     $\{\phi_{X_i} : i = 1, \dots, k\}$   
)  
1  for  $i = k, \dots, 1$   
2       $u_i \leftarrow (x_{i+1}^*, \dots, x_k^*) \langle \text{Scope}[\phi_{X_i}] - \{X_i\} \rangle$   
3      // The maximizing assignment to the variables eliminated after  
         $X_i$   
4       $x_i^* \leftarrow \arg \max_{x_i} \phi_{X_i}(x_i, u_i)$   
5      //  $x_i^*$  is chosen so as to maximize the corresponding entry in  
        the factor, relative to the previous choices  $u_i$   
6  return  $x^*$ 
```

Beyond Variable Elimination

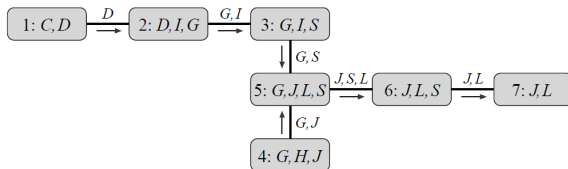


Figure 10.1 Cluster tree for the VE execution in table 9.1

- ▶ The execution of VE defines a cluster tree, a graphical flowchart of the factor-manipulation process:
 - ▶ Each node C_i in the tree is called a cluster and it contains the variables in $Scope(\psi_i)$.
 - ▶ The tree has an edge annotated with an arrow from C_i to C_j if the factor (a.k.a. message) τ_i is used in the computation of τ_j .
- ▶ Now, consider passing messages towards the cluster C_6 in order to compute $p(s)$. Notice the **reusing** of previously computed messages.
- ▶ Therefore, a clique tree is a data structure to perform **repeated** probabilistic inference efficiently.
- ▶ The process of building the cluster tree from a given DAG is known as compilation. You will learn more about cluster trees in the lab.

Contents

- ▶ Probabilistic Inference for BNs
 - ▶ Naive Solution
 - ▶ Variable Elimination
- ▶ Probabilistic Inference for MNs
- ▶ Most Probable Configuration

Thank you