Variable Elimination

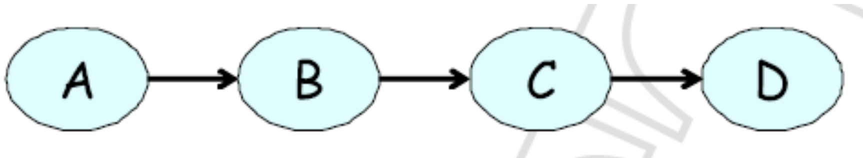
Dhruv Batra,
10-708 Recitation
10/16/2008

Overview

- Variable Elimination
  - Example on chain networks
  - Intuition
  - Tools for VE, factor product, marginalization
  - Implementation hints

Chain Networks

- BN:



- Goal: Need all marginals P(X)

Chain Networks

- Naïve solution

$$
\begin{aligned}
& P(a^1) \; P(b^1 \mid a^1) \; P(c^1 \mid b^1) \; P(d^1 \mid c^1) \\
+ \; & P(a^2) \; P(b^1 \mid a^2) \; P(c^1 \mid b^1) \; P(d^1 \mid c^1) \\
+ \; & P(a^1) \; P(b^2 \mid a^1) \; P(c^1 \mid b^2) \; P(d^1 \mid c^1) \\
+ \; & P(a^2) \; P(b^2 \mid a^2) \; P(c^1 \mid b^2) \; P(d^1 \mid c^1) \\
+ \; & P(a^1) \; P(b^1 \mid a^1) \; P(c^2 \mid b^1) \; P(d^1 \mid c^2) \\
+ \; & P(a^2) \; P(b^1 \mid a^2) \; P(c^2 \mid b^1) \; P(d^1 \mid c^2) \\
+ \; & P(a^1) \; P(b^2 \mid a^1) \; P(c^2 \mid b^2) \; P(d^1 \mid c^2) \\
+ \; & P(a^2) \; P(b^2 \mid a^2) \; P(c^2 \mid b^2) \; P(d^1 \mid c^2)
\end{aligned}
$$

$$
\begin{aligned}
& P(a^1) \; P(b^1 \mid a^1) \; P(c^1 \mid b^1) \; P(d^2 \mid c^1) \\
+ \; & P(a^2) \; P(b^1 \mid a^2) \; P(c^1 \mid b^1) \; P(d^2 \mid c^1) \\
+ \; & P(a^1) \; P(b^2 \mid a^1) \; P(c^1 \mid b^2) \; P(d^2 \mid c^1) \\
+ \; & P(a^2) \; P(b^2 \mid a^2) \; P(c^1 \mid b^2) \; P(d^2 \mid c^1) \\
+ \; & P(a^1) \; P(b^1 \mid a^1) \; P(c^2 \mid b^1) \; P(d^2 \mid c^2) \\
+ \; & P(a^2) \; P(b^1 \mid a^2) \; P(c^2 \mid b^1) \; P(d^2 \mid c^2) \\
+ \; & P(a^1) \; P(b^2 \mid a^1) \; P(c^2 \mid b^2) \; P(d^2 \mid c^2) \\
+ \; & P(a^2) \; P(b^2 \mid a^2) \; P(c^2 \mid b^2) \; P(d^2 \mid c^2)
\end{aligned}
$$

Chain Networks

- A little smarter solution: Phased Computation

$$P(B) = \sum_a P(a)P(B \mid a).$$

$$P(C) = \sum_b P(b)P(C \mid b).$$

- General Chains:

$$P(X_{i+1}) = \sum_{x_i} P(X_{i+1} \mid x_i)P(x_i)$$

- Why is this better?

$O(nk^2)$ vs $O(\ k^n\ )$

- What's the intuition?

Chain Networks

- A lot of structure

$\tau_1(b^1)$

$\tau_1(b^2)$

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

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

Chain Networks

- Let's cache

$$\tau_2(c^1)$$

$$\tau_2(c^2)$$

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

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

Chain Networks

- Let's cache again

$$
\begin{array}{rl}
\tau_2(c^1) & P(d^1 \mid c^1) \\
+\ \tau_2(c^2) & P(d^1 \mid c^2)
\end{array}
$$

$$
\begin{array}{rl}
\tau_2(c^1) & P(d^2 \mid c^1) \\
+\ \tau_2(c^2) & P(d^2 \mid c^2)
\end{array}
$$

$$
\begin{array}{rcl}
\tau_2(c^1) & = & \tau_1(b^1)P(c^1 \mid b^1) + \tau_1(b^2)P(c^1 \mid b^2) \\
\tau_2(c^2) & = & \tau_1(b^1)P(c^2 \mid b^1) + \tau_1(b^2)P(c^2 \mid b^2)
\end{array}
$$

Chain Networks

- Intuitions from this process
  - Group common things/terms/factors based on scope
  - Dynamic programming ideas: cache computations
- VE extends/formalizes these intuitions to general graphs
  - but separates the elimination ordering from the process

Another Idea

- A commonly used idea

- Goal $$P(\boldsymbol{Y} \mid \boldsymbol{E} = \boldsymbol{e}) = \frac{P(\boldsymbol{Y}, \boldsymbol{e})}{P(\boldsymbol{e})}$$

$$P(\boldsymbol{y}, \boldsymbol{e}) = \sum_{\boldsymbol{w}} P(\boldsymbol{y}, \boldsymbol{e}, \boldsymbol{w}).$$

$$P(\boldsymbol{e}) = \sum_{\boldsymbol{y}} P(\boldsymbol{y}, \boldsymbol{e})$$

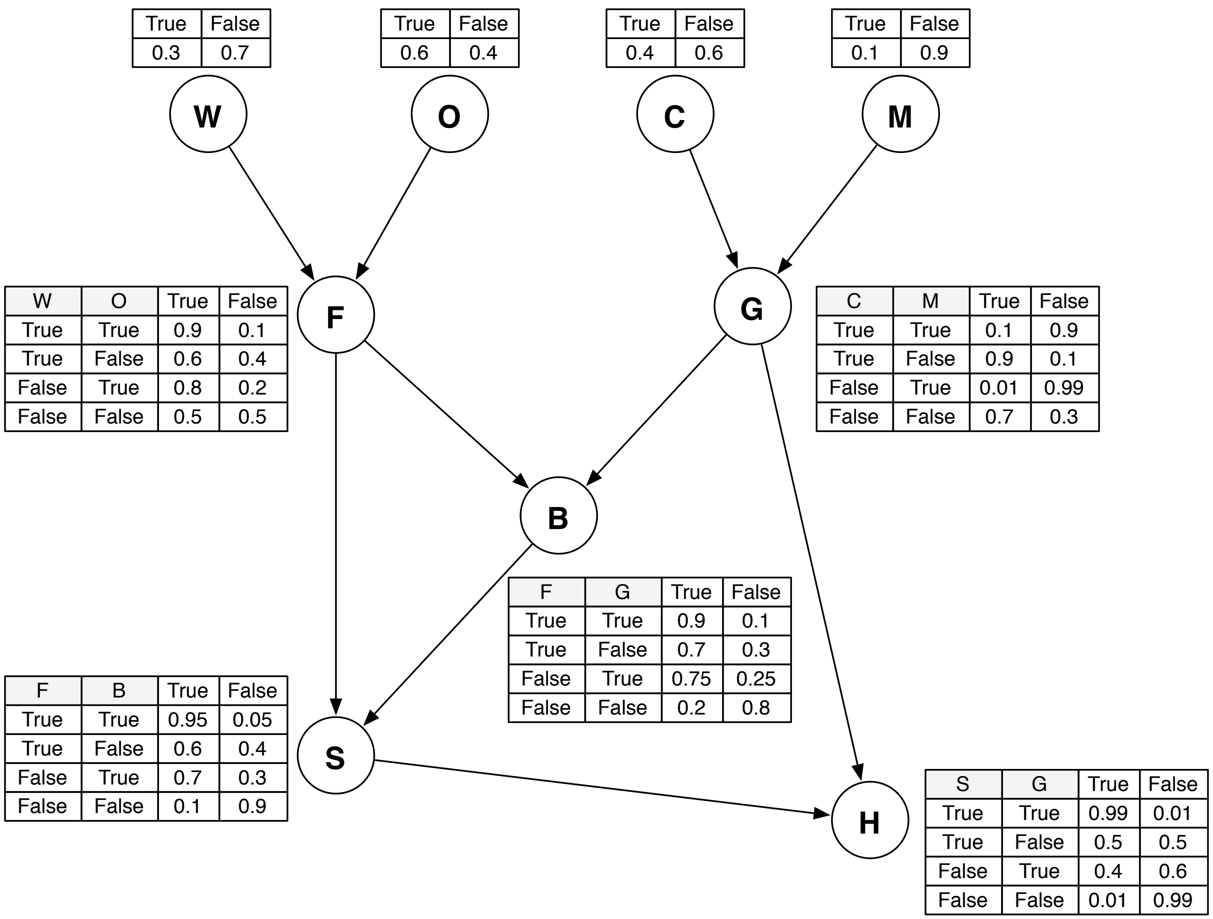
- Can forget about denominator; just renormalize when done

Example

- Let’s do an example for general graphs

$$P(S = T | C = T) = ?$$

$$P(F = T | G = T)$$



W:

True	False
0.3	0.7

O:

True	False
0.6	0.4

C:

True	False
0.4	0.6

M:

True	False
0.1	0.9

F:

W	O	True	False
True	True	0.9	0.1
True	False	0.6	0.4
False	True	0.8	0.2
False	False	0.5	0.5

G:

C	M	True	False
True	True	0.1	0.9
True	False	0.9	0.1
False	True	0.01	0.99
False	False	0.7	0.3

B:

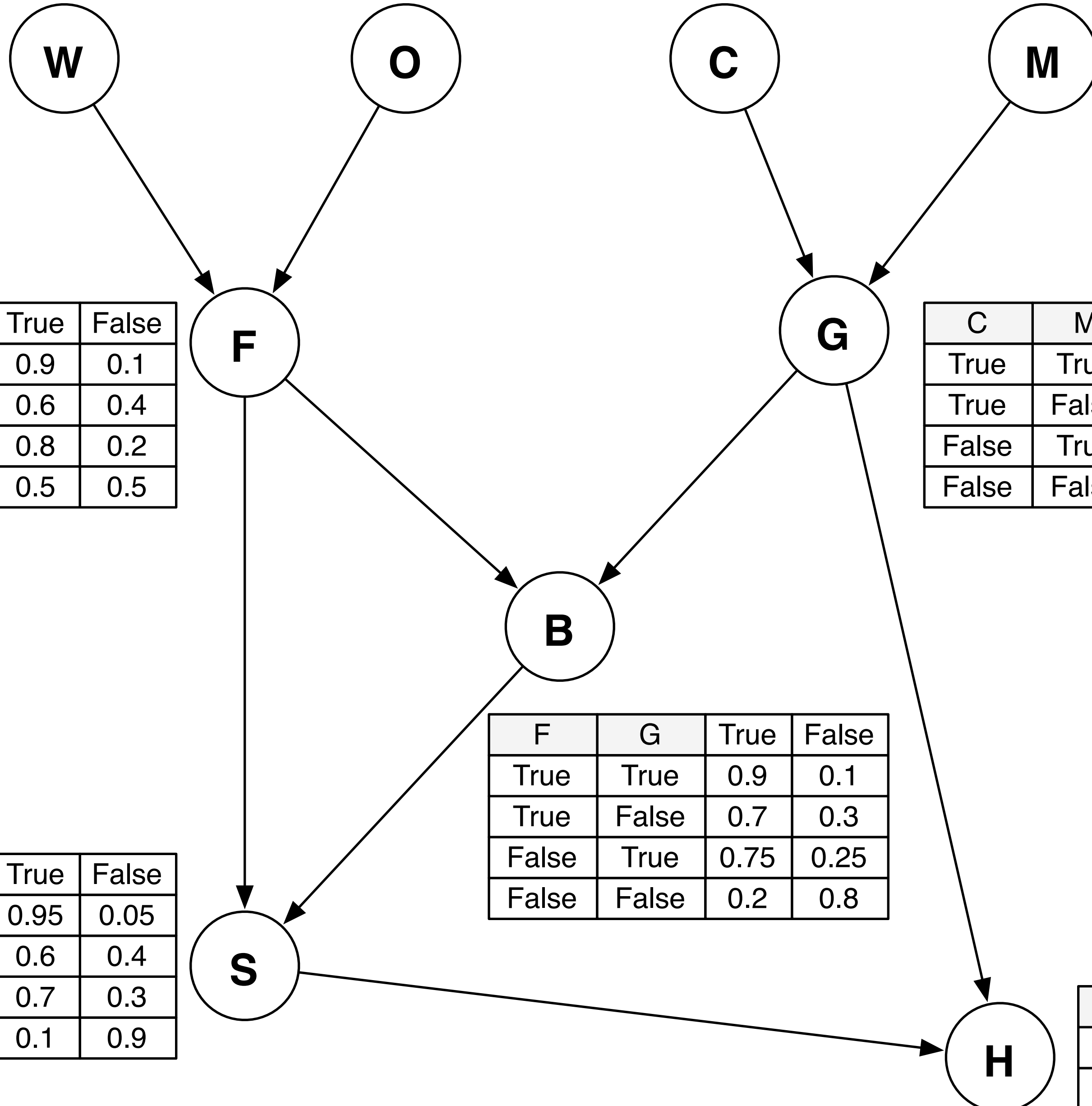
F	G	True	False
True	True	0.9	0.1
True	False	0.7	0.3
False	True	0.75	0.25
False	False	0.2	0.8

S:

F	B	True	False
True	True	0.95	0.05
True	False	0.6	0.4
False	True	0.7	0.3
False	False	0.1	0.9

H:

S	G	True	False
True	True	0.99	0.01
True	False	0.5	0.5
False	True	0.4	0.6
False	False	0.01	0.99



W:

True	False
0.3	0.7

O:

True	False
0.6	0.4

C:

True	False
0.4	0.6

M:

True	False
0.1	0.9

F:

W	O	True	False
True	True	0.9	0.1
True	False	0.6	0.4
False	True	0.8	0.2
False	False	0.5	0.5

G:

C	M	True	False
True	True	0.1	0.9
True	False	0.9	0.1
False	True	0.01	0.99
False	False	0.7	0.3

B:

F	G	True	False
True	True	0.9	0.1
True	False	0.7	0.3
False	True	0.75	0.25
False	False	0.2	0.8

S:

F	B	True	False
True	True	0.95	0.05
True	False	0.6	0.4
False	True	0.7	0.3
False	False	0.1	0.9

H:

S	G	True	False
True	True	0.99	0.01
True	False	0.5	0.5
False	True	0.4	0.6
False	False	0.01	0.99

Implementing VE

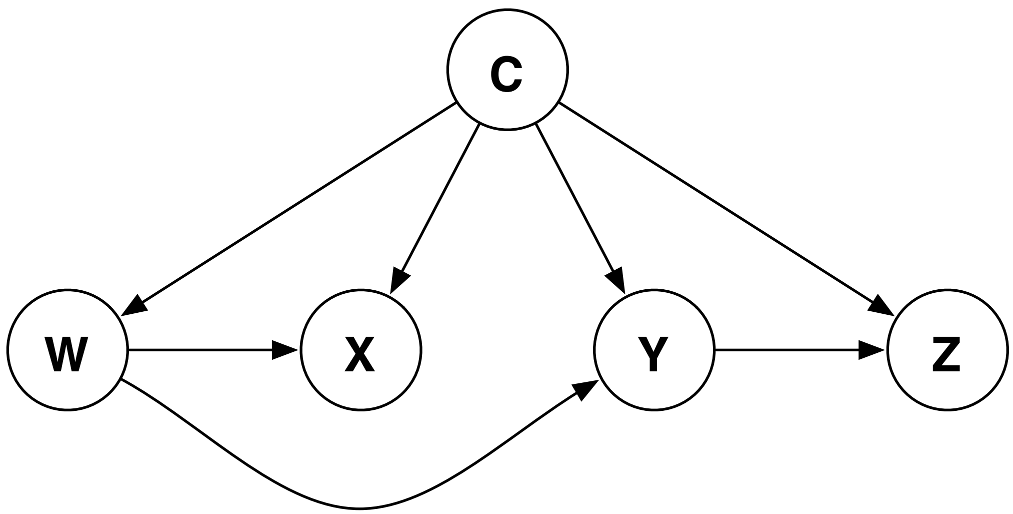
- What do you need to implement VE?
  - Reuse some code from HW2

Tools for VE

- Factors

$$\pi_1[A, B] \; : \; Val(A, B) \mapsto I\!R^+$$

- Special kind of factors: CPTs



Operations on Factors

- Factor Product
  - Consider two factors

    $\phi_1(\boldsymbol{X}, \boldsymbol{Y})$ and $\phi_2(\boldsymbol{Y}, \boldsymbol{Z})$
  - Define factor product

    $\psi : Val(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}) \mapsto I\!R$
  - such that

    $\psi(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}) = \phi_1(\boldsymbol{X}, \boldsymbol{Y}) \cdot \phi_2(\boldsymbol{Y}, \boldsymbol{Z})$

Operations on Factors

- Factor Product

a^1	b^1	0.5
a^1	b^2	0.8
a^2	b^1	0.1
a^2	b^2	0
a^3	b^1	0.3
a^3	b^2	0.9

b^1	c^1	0.5
b^1	c^2	0.7
b^2	c^1	0.1
b^2	c^2	0.2

a^1	b^1	c^1	$0.5 \cdot 0.5$
a^1	b^1	c^2	$0.5 \cdot 0.7$
a^1	b^2	c^1	$0.8 \cdot 0.1$
a^1	b^2	c^2	$0.8 \cdot 0.2$
a^2	b^1	c^1	$0.1 \cdot 0.5$
a^2	b^1	c^2	$0.1 \cdot 0.7$
a^2	b^2	c^1	$0 \cdot 0.1$
a^2	b^2	c^2	$0 \cdot 0.2$
a^3	b^1	c^1	$0.3 \cdot 0.5$
a^3	b^1	c^2	$0.3 \cdot 0.7$
a^3	b^2	c^1	$0.9 \cdot 0.1$
a^3	b^2	c^2	$0.9 \cdot 0.2$

Operations on Factors

- Factor Marginalization
  - Consider a factor
    $\phi(\boldsymbol{X}, Y)$
  - Define factor marginal
    $\psi(\boldsymbol{X}) : Val(\boldsymbol{X}) \mapsto I\!R.$
  - such that
    $$\psi(\boldsymbol{X}) = \sum_{Y} \phi(\boldsymbol{X}, Y).$$

Operations on Factors

- Factor Marginalization

a^1	b^1	c^1	0.25
a^1	b^1	c^2	0.35
a^1	b^2	c^1	0.08
a^1	b^2	c^2	0.16
a^2	b^1	c^1	0.05
a^2	b^1	c^2	0.07
a^2	b^2	c^1	0
a^2	b^2	c^2	0
a^3	b^1	c^1	0.15
a^3	b^1	c^2	0.21
a^3	b^2	c^1	0.09
a^3	b^2	c^2	0.18

a^1	c^1	0.33
a^1	c^2	0.51
a^2	c^1	0.05
a^2	c^2	0.07
a^3	c^1	0.24
a^3	c^2	0.39

Factors

- Are factors always distributions?
  - Obviously not
- Are factors produced in VE always distributions?
  - Yes, always conditional distributions
  - In SOME graph, not necessarily the original graph
  - HW3, prob 2. Hint: read 8.3.1.3

Implementing VE

- What do you need to implement VE?
  - Reuse some code from HW2

- Representation
  - BN as an array of factors
  - table_factor.m
  - assignment.m

- VE
  - multipy_factors.m
  - marginalize_factor.m
  - min_fill.m

Variable elimination algorithm

- Given a BN and a query P(X|e) / P(X,e)
- Instantiate evidence **e**
- Prune non-active vars for {X,**e**} ← optional
- Choose an ordering on variables, e.g., $X_1, \ldots, X_n$
- Initial *factors* $\{f_1,\ldots,f_n\}$: $f_i = P(X_i|\mathbf{Pa}_{X_i})$ (CPT for X_i)
- For i = 1 to n, If $X_i \notin \{X,\mathbf{E}\}$
  - Collect factors $f_1,\ldots,f_k$ that include X_i
  - Generate a new factor by eliminating X_i from these factors

$$g = \sum_{X_i} \prod_{j=1}^{k} f_j$$

  - Variable X_i has been eliminated!
- Normalize P(X,**e**) to obtain P(X|**e**)

Questions?