Reasoning Under Uncertainty: Variable Elimination

CPSC 322 – Uncertainty 6

Textbook §6.4.1

March 28, 2011

Announcements (1)

- Assignment 4 due in one week
  - Can only use 2 late days
    - So we can give out solutions to study for the final exam

- Final exam in two weeks: Monday, April 11
  - 3:30 – 6pm in DMP 310
  - Same format as midterm (60% short questions)
    - List of short questions is on WebCT up to uncertainty
  - Emphasis on material after midterm
  - How to study?
    - Practice exercises, assignments, short questions, lecture notes, book, …
    - Use office hours (extra office hours next week)

Announcements (2)

- Teaching Evaluations are online
  - You should have gotten an email about them

- Your feedback is important!
  - I use it to assess and improve my teaching
  - The department as a whole uses it to shape the curriculum
  - Teaching evaluation results are important for instructors
    - Appointment, reappointment, tenure, promotion and merit

  - Evaluations close at 11PM on December 10th, 2011
    - Before exams, but
      instructors can’t see results until grades are submitted

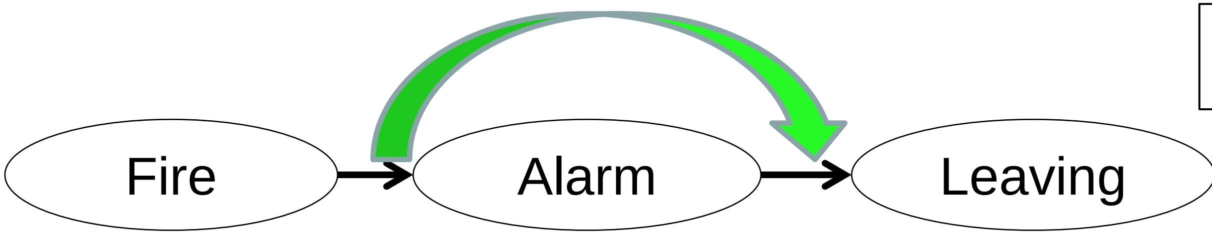
Lecture Overview

- Entailed Independencies: Recap and Examples

- Inference in General Bayesian Networks
  - Factors:
    - Assigning Variables
    - Summing out Variables
    - Multiplication of Factors
  - The variable elimination algorithm

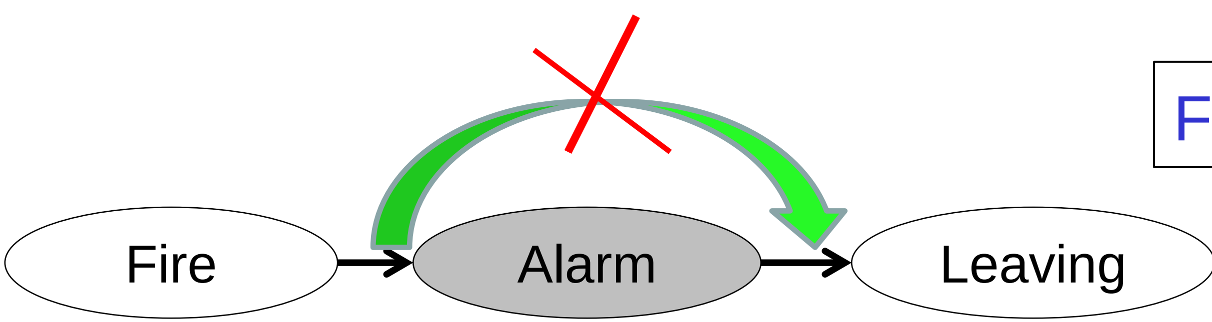
Recap: Information flow through chain structure

- Unobserved node in a chain lets information pass



Fire $\not\!\perp\!\!\!\perp$ Leaving

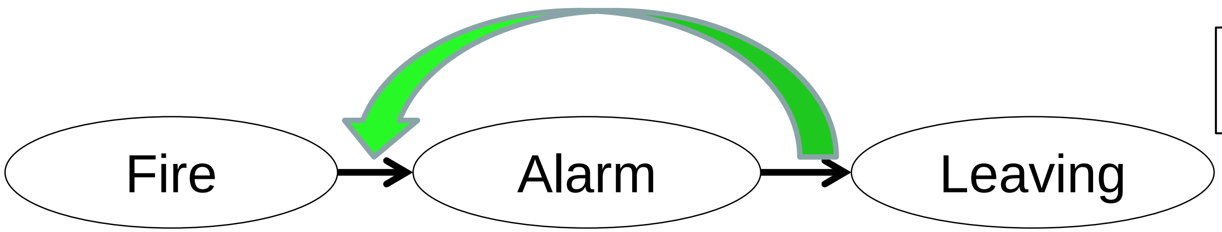
- Observed node in a chain blocks information



Fire $\perp\!\!\!\perp$ Leaving | Alarm

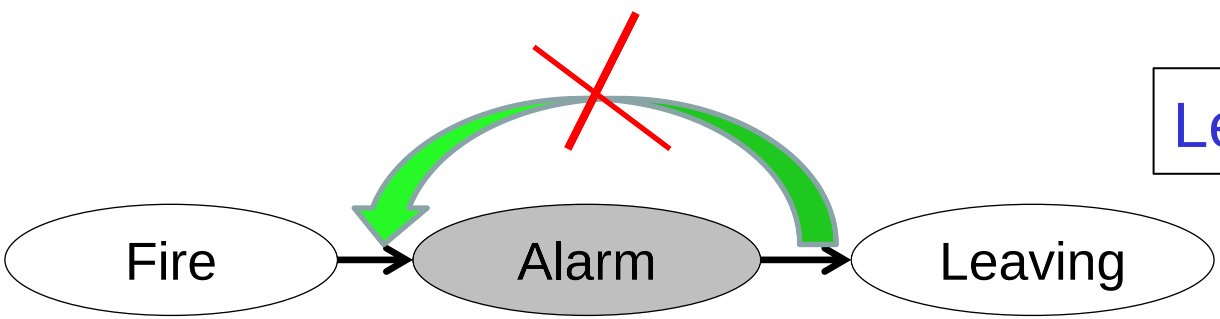
Recap: Information flow through chain structure

- Information flow is symmetric ($X \perp\!\!\!\perp Y \mid Z$ and $Y \perp\!\!\!\perp X \mid Z$ are identical)
  - Unobserved node in a chain lets information pass (both ways)



Leaving $\not\perp\!\!\!\perp$ Fire

  - Observed node in a chain blocks information (both ways)

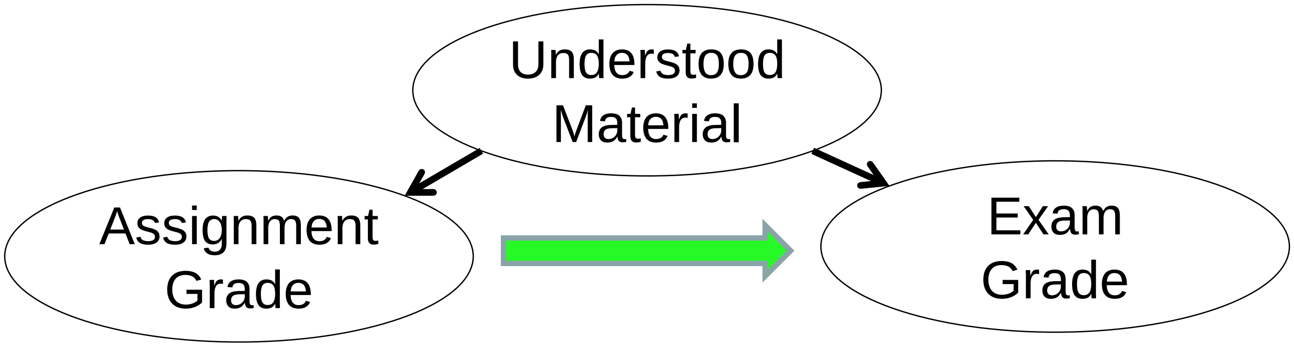


Leaving $\perp\!\!\!\perp$ Fire | Alarm

Recap: Information flow through common parent

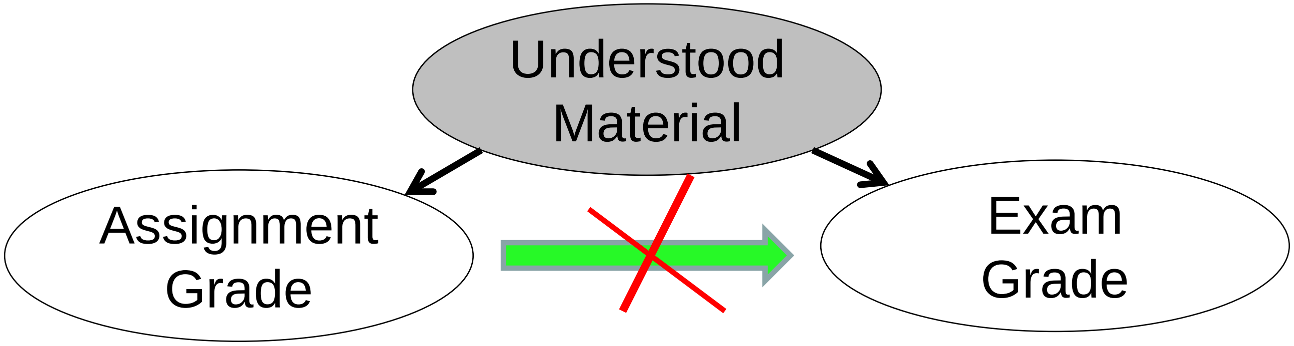
- Unobserved common parent lets information pass

AssignmentGrade $\not\perp\!\!\!\perp$ ExamGrade



- Observed common parent blocks information

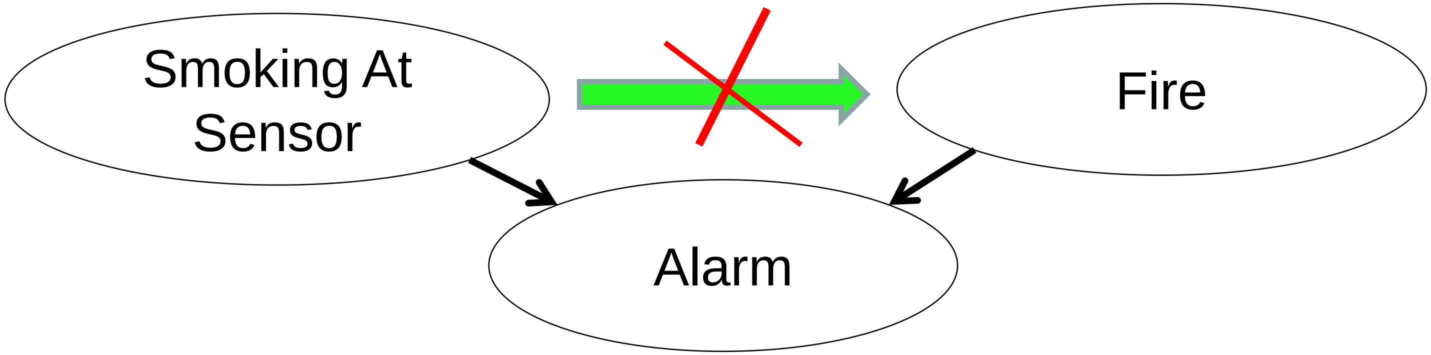
AssignmentGrade $\perp\!\!\!\perp$ ExamGrade | UnderstoodMaterial



Recap: Information flow through common child

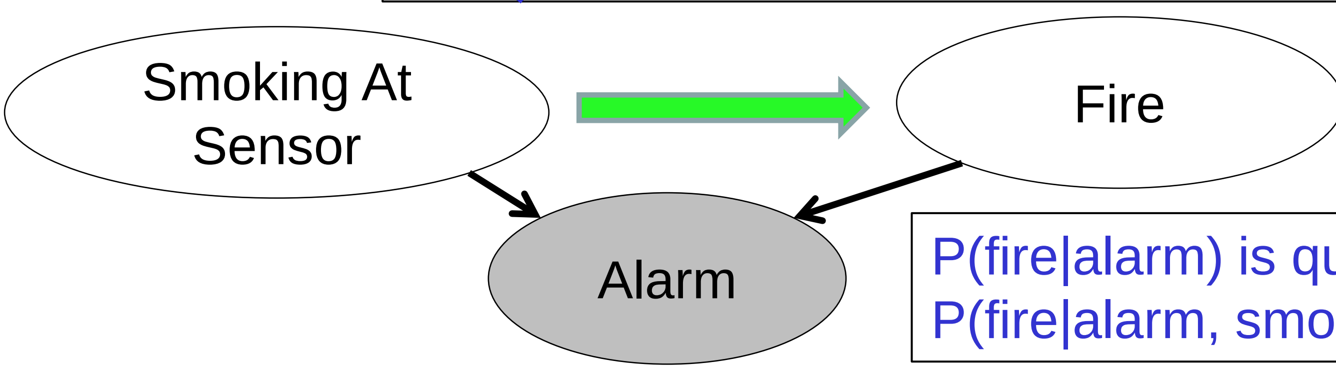
- Unobserved common child blocks information

$$\text{SmokingAtSensor} \perp\!\!\!\perp \text{Fire}$$



- Observed common child lets information pass: explaining away

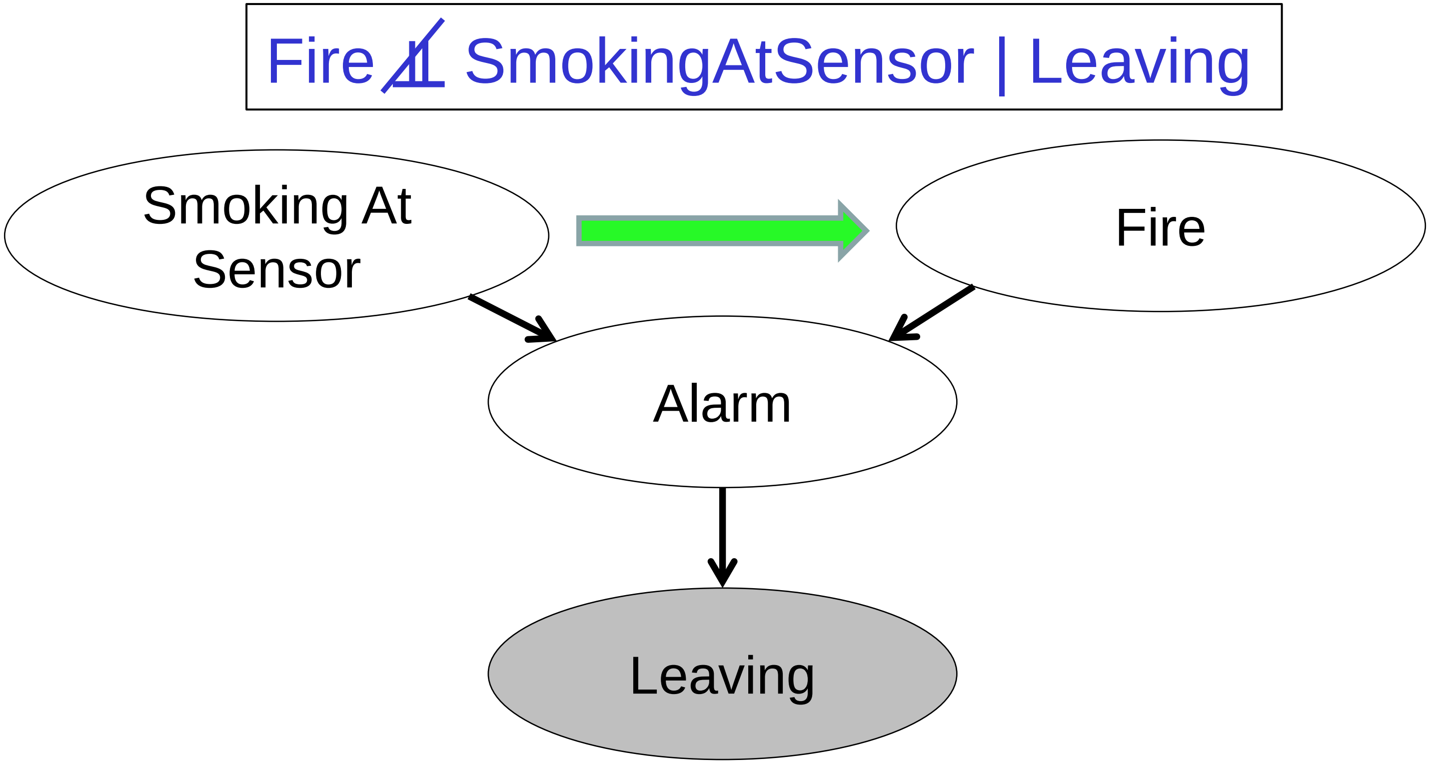
$$\text{Fire} \not\perp\!\!\!\perp \text{SmokingAtSensor} \mid \text{Alarm}$$



P(fire|alarm) is quite high
P(fire|alarm, smokingAtSensor) is low

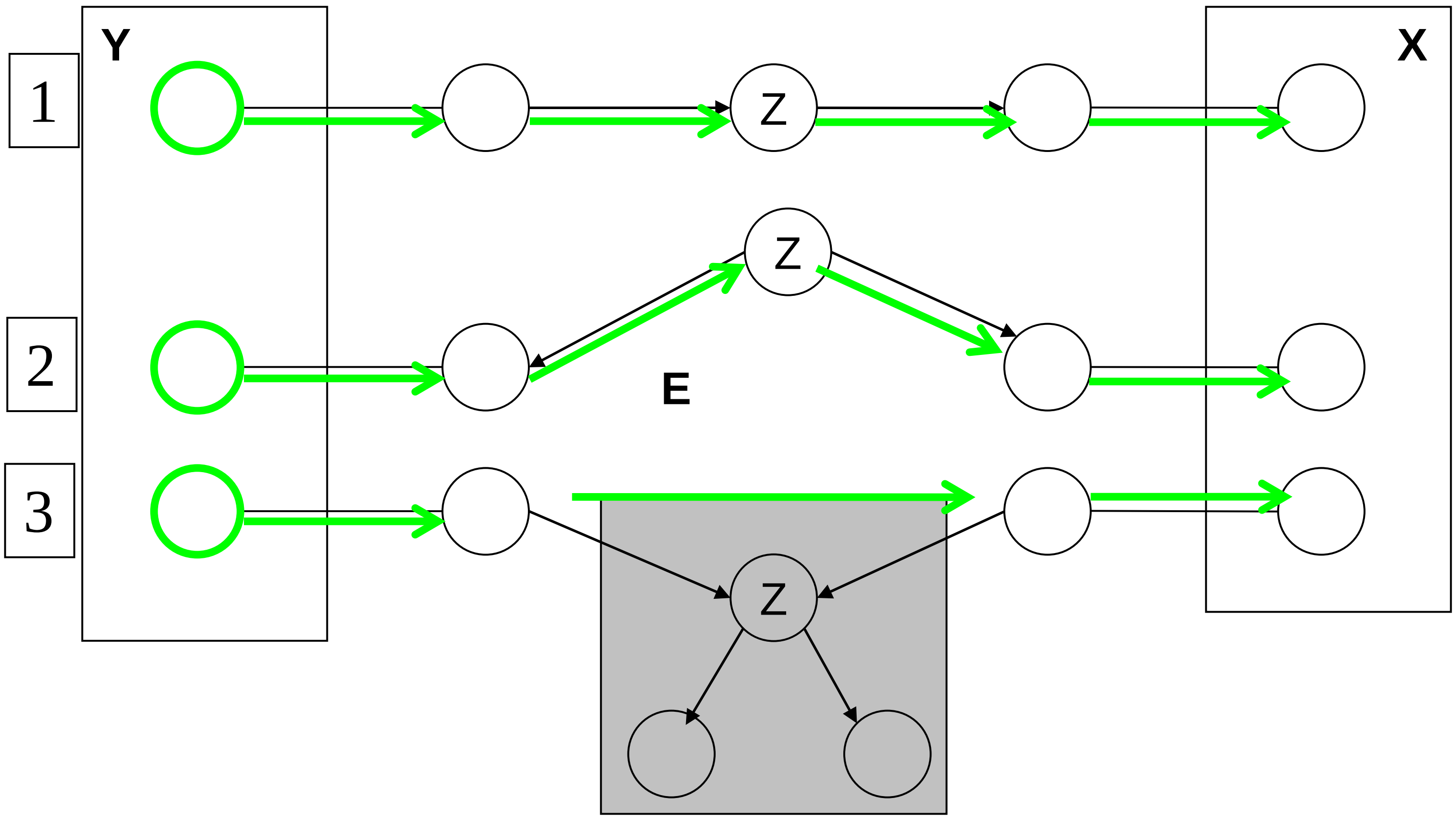
Recap: Information flow through common child

- Exception: unobserved common child lets information pass if one of its descendants is observed
  - This is just as if the child itself was observed
    - E.g., Leaving could be a deterministic function of Alarm, so observing Leaving means you know Alarm as well



Summary: (Conditional) Dependencies

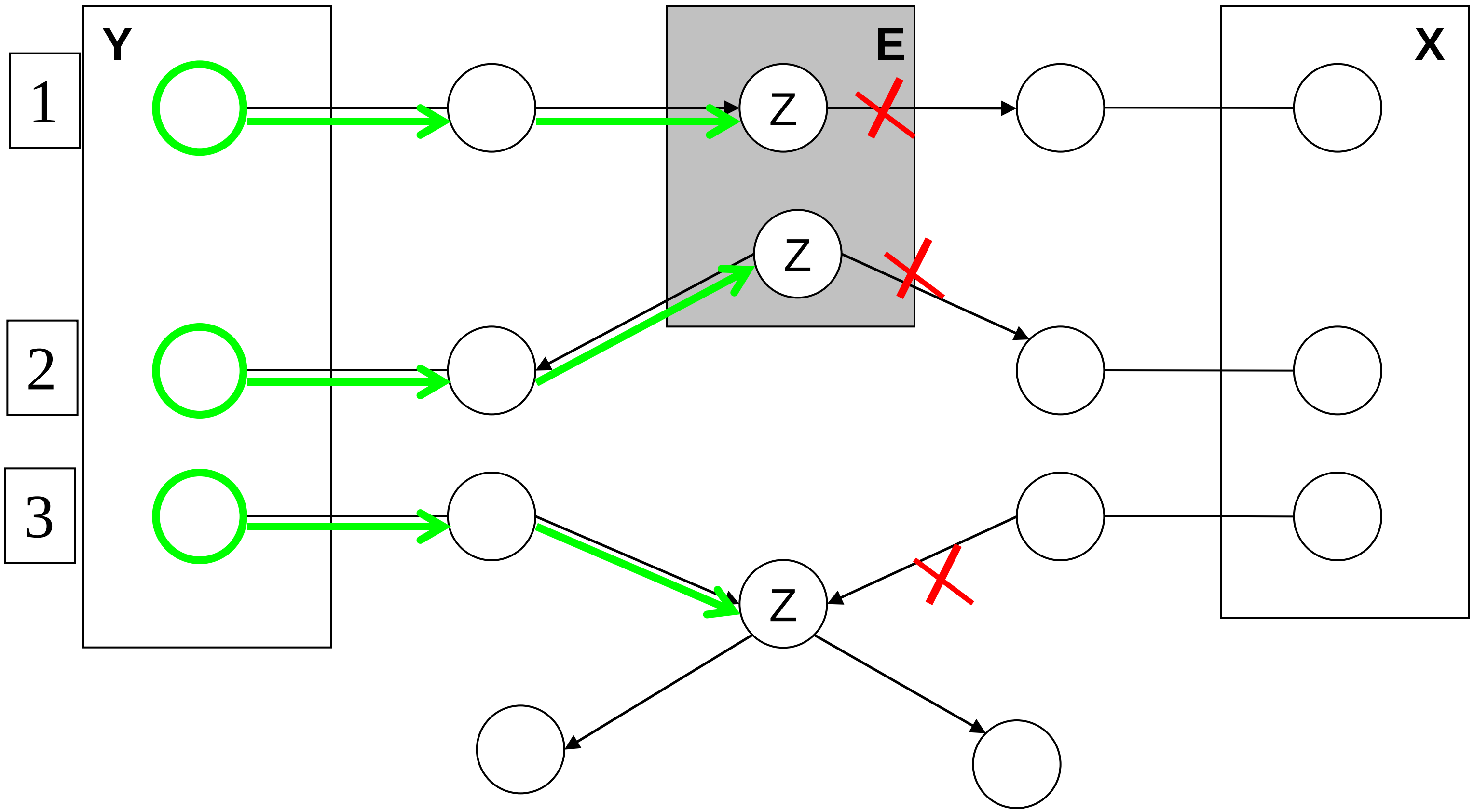
- In these cases, X and Y are (conditionally) dependent



- In 3, X and Y become dependent as soon as there is evidence on Z or on *any of its descendants.*

Summary: (Conditional) Independencies

- Blocking paths for probability propagation. Three ways in which a path between Y to X (or vice versa) can be blocked, given evidence E



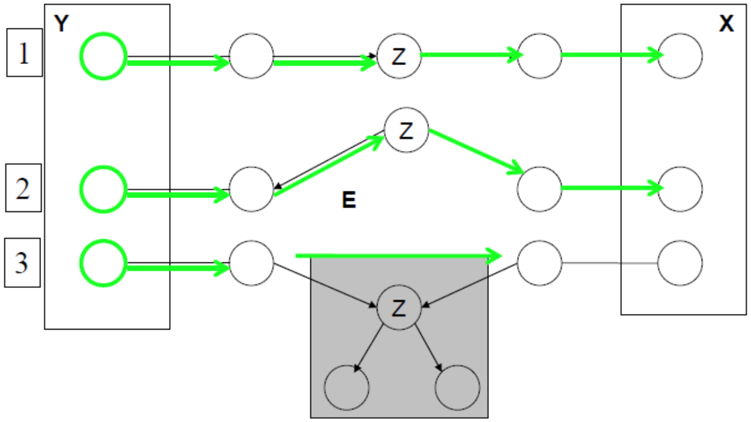
Training your understanding of conditional independencies in AIspace

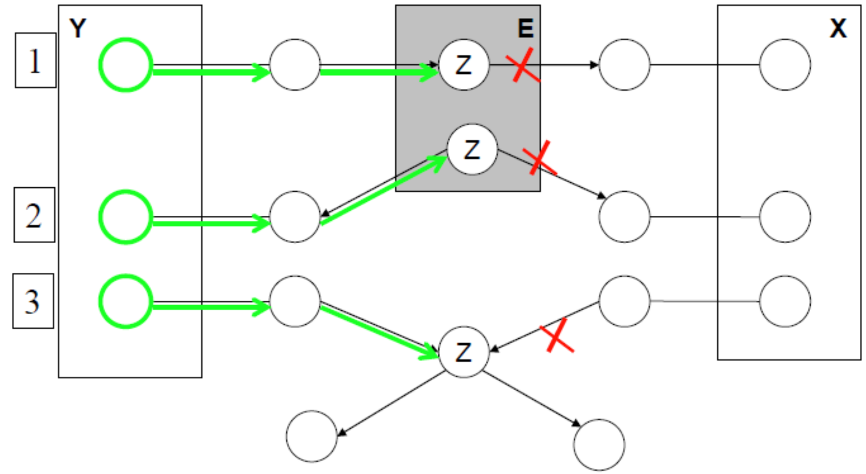
- These concepts take practice to get used to

- Use the AIspace applet for Belief and Decision networks (http://aispace.org/bayes/)
  - Load the “conditional independence quiz” network (or any other one)
  - Go in “Solve” mode and select “Independence Quiz”

- You can take an unbounded number of quizzes:
  - It generates questions, you answer, and then get the right answer
  - It also allows you to ask arbitrary queries

Conditional Independencies in a BN



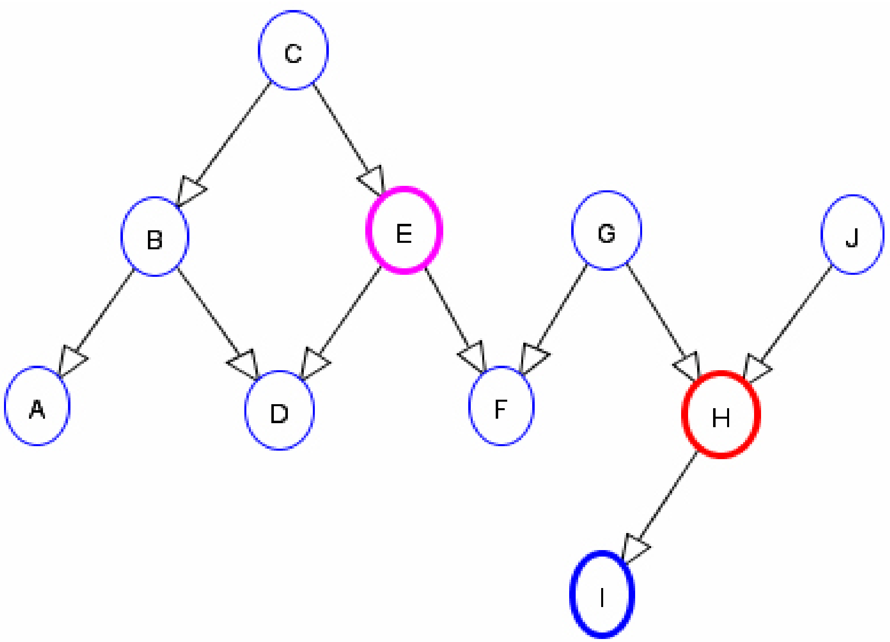


Is **H** conditionally independent of **E** given **I**?

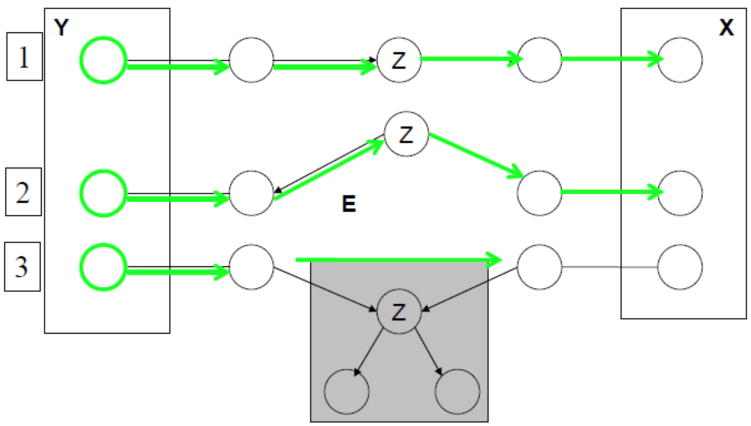
I.e., **H** ⫫ **E** | **I** ?

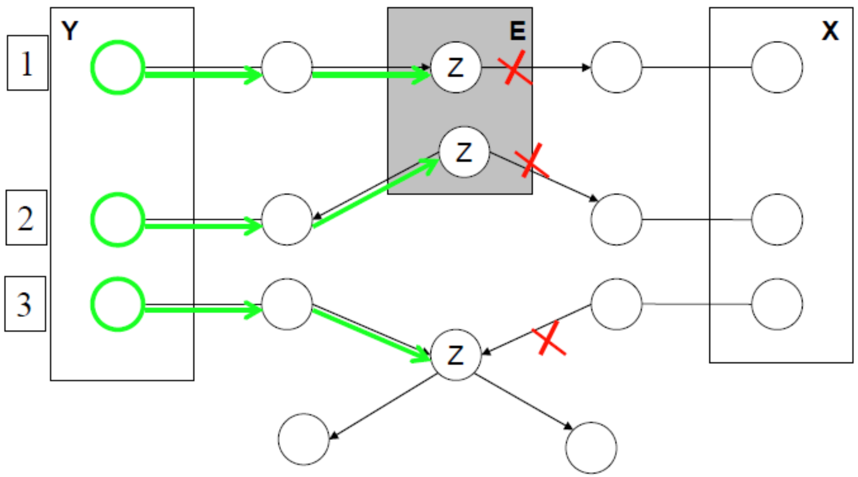
Yes

No



Conditional Independencies in a BN

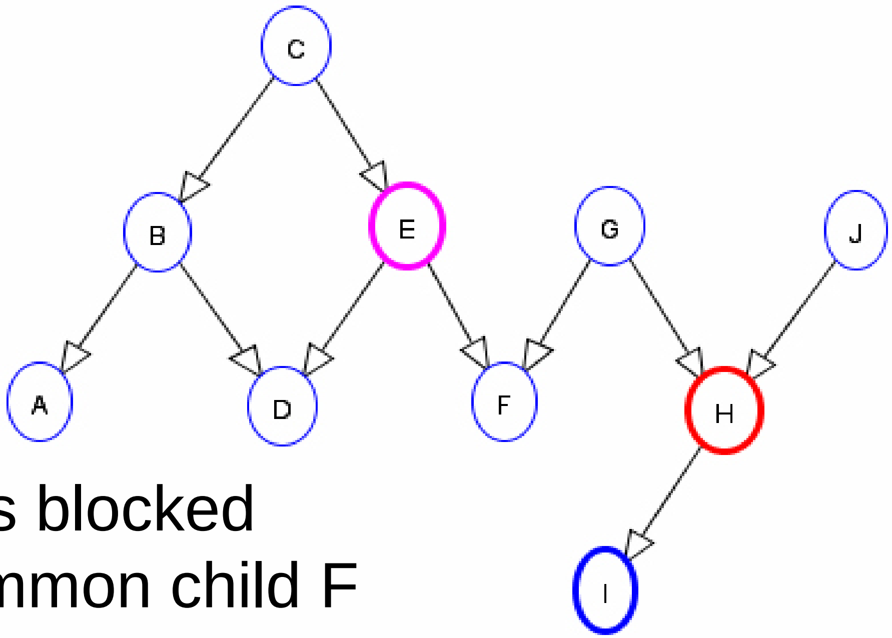




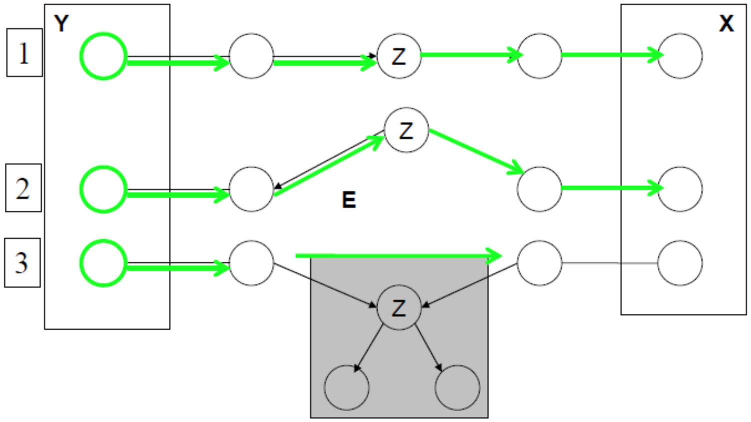
Is **H** conditionally independent of **E** given **I**?

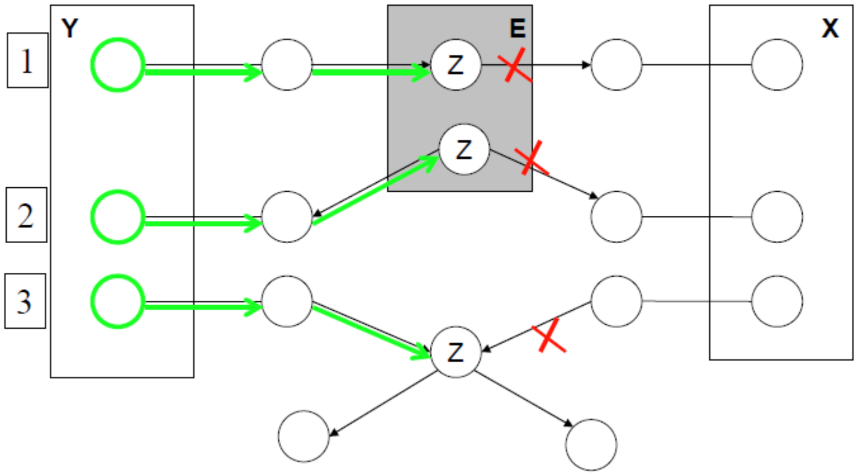
I.e., **H** ⫫ **E** | **I** ?

Yes! Information flow is blocked by the unobserved common child F



Conditional Independencies in a BN

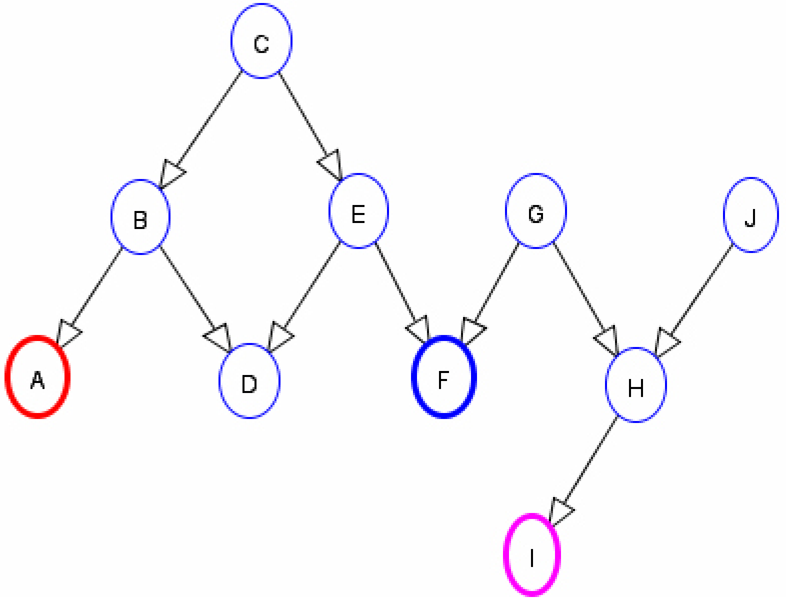




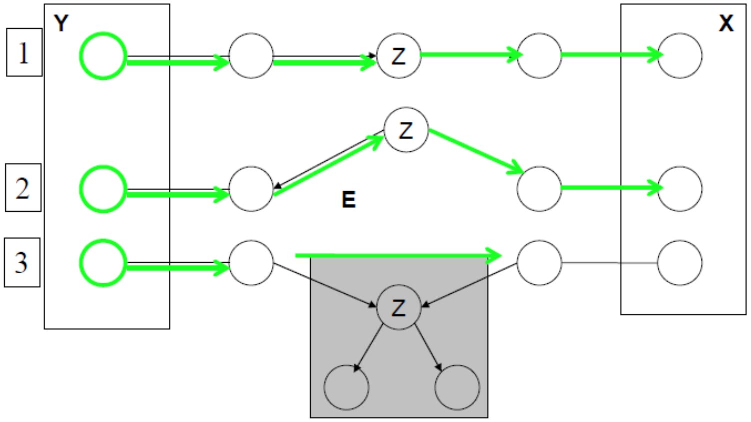
Is **A** conditionally independent of **I** given **F**?

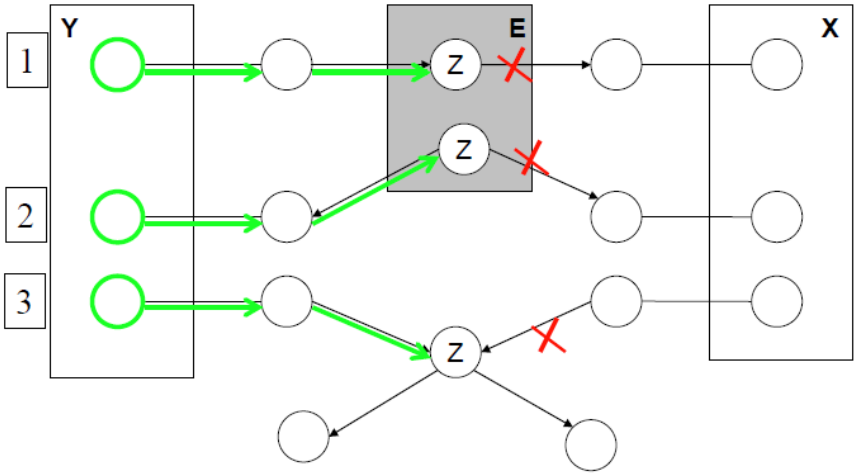
I.e., **A** ⫫ **I** | **F** ?

Yes No



Conditional Independencies in a BN

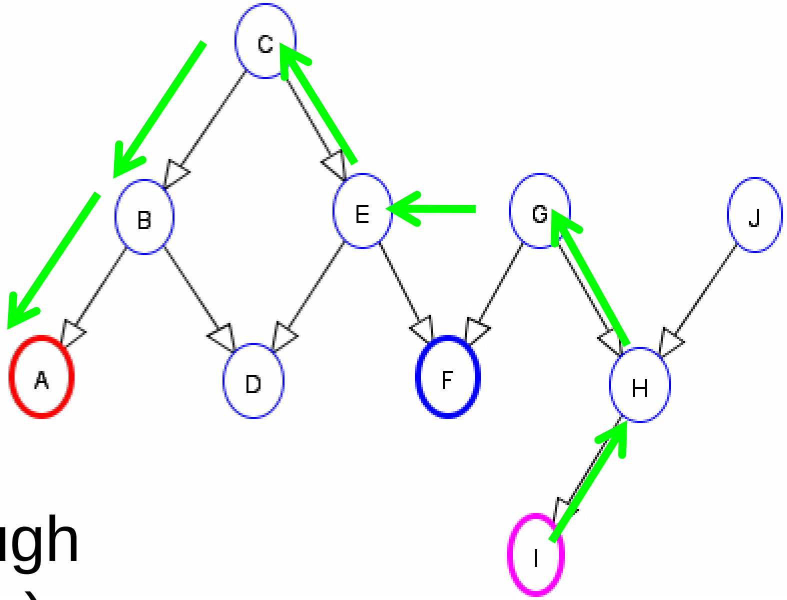




Is **A** conditionally independent of **I** given **F**?

I.e., **A** ⫫ **I** | **F** ?

No. Information can pass through (all it takes is one possible path)



Lecture Overview

- Entailed Independencies: Recap and Examples
- Inference in General Bayesian Networks
  - Factors:
    - Assigning Variables
    - Summing out Variables
    - Multiplication of Factors
  - The variable elimination algorithm

Factors

- A factor is a function from a tuple of random variables to the real numbers $\mathbb{R}$
- We write a factor on variables $X_1, \ldots, X_j$ as $f(X_1, \ldots, X_j)$

- P(Z|X,Y) is a factor f (X,Y,Z)
  - Factors do not have to sum to one
  - P(Z|X,Y) is a set of probability distributions: one for each combination of values of X and Y

- P(Z=f|X,Y) is a factor f(X,Y)

$f(X, Y)_{Z=f}$

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
f	t	t	0.4
f	t	f	0.6
f	f	t	0.3
f	f	f	0.7

Operation 1: assigning a variable

- We can make new factors out of an existing factor

- Our first operation:
  we can assign some or all of the variables of a factor.

f(X,Y,Z):

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
~~f~~	~~t~~	~~t~~	~~0.4~~
~~f~~	~~t~~	~~f~~	~~0.6~~
~~f~~	~~f~~	~~t~~	~~0.3~~
~~f~~	~~f~~	~~f~~	~~0.7~~

What is the result of assigning X= t ?

$f(X=t,Y,Z) = f(X, Y, Z)_{X=t}$

Y	Z	val
t	t	0.1
t	f	0.9
f	t	0.2
f	f	0.8

Factor of Y,Z

More examples of assignment

f(X,Y,Z):

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
f	t	t	0.4
f	t	f	0.6
f	f	t	0.3
f	f	f	0.7

f(X=t,Y,Z)

Factor of Y,Z

Y	Z	val
t	t	0.1
t	f	0.9
f	t	0.2
f	f	0.8

f(X=t,Y,Z=f):

Y	val
t	0.9
f	0.8

Factor of Y

f(X=t,Y=f,Z=f): 0.8

Number

Operation 2: Summing out a variable

- Our second operation on factors:
  we can marginalize out (or sum out) a variable
  - Exactly as before. Only difference: factors don't sum to 1
  - Marginalizing out a variable X from a factor f($X_1,\dots,X_n$) yields a new factor defined on {$X_1,\dots,X_n$} \ {X}

$$\left(\sum_{X_1} f\right)(X_2,\dots,X_j) = \sum_{x \in dom(X_1)} f(X_1 = x, X_2,\dots,X_j)$$

f_3=

B	A	C	val
t	t	t	0.03
t	t	f	0.07
f	t	t	0.54
f	t	f	0.36
t	f	t	0.06
t	f	f	0.14
f	f	t	0.48
f	f	f	0.32

$(\sum_B f_3)(A,C)$

A	C	val
t	t	0.57
t	f	0.43
f	t	0.54
f	f	0.46

Operation 3: multiplying factors

$f_1(A,B)$:

A	B	Val
t	t	0.1
t	f	0.9
f	t	0.2
f	f	0.8

$f_2(B,C)$:

B	C	Val
t	t	0.3
t	f	0.7
f	t	0.6
f	f	0.4

$f_1(A,B) \times f_2(B,C)$:

A	B	C	val
t	t	t	0.03
t	t	f	0.1x0.7
t	f	t	0.9x0.6
t	f	f	…
f	t	t	
f	t	f	
f	f	t	
f	f	f	

Operation 3: multiplying factors

- The product of factor $f_1(A, B)$ and $f_2(B, C)$, where B is the variable in common, is the factor $(f_1 \times f_2)(A, B, C)$ defined by

$$(f_1 \times f_2)(A, B, C) = f_1(A, B) f_2(B, C)$$

- Note: A, B, and C can be sets of variables
  - The domain of $f_1 \times f_2$ is $A \cup B \cup C$

Lecture Overview

- Entailed Independencies: Recap and Examples

- Inference in General Bayesian Networks
  - Factors:
    - Assigning Variables
    - Summing out Variables
    - Multiplication of Factors
  - The variable elimination algorithm
  - Example trace of variable elimination

General Inference in Bayesian Networks

Given

- A Bayesian Network BN, and
- Observations of a subset of its variables E: E=e
- A subset of its variables Y that is queried

Compute the conditional probability P(Y=y|E=e)

Definition of conditional probability

Marginalization over Y:
$P(E=e) = \Sigma_{y' \in dom(Y)} P(E=e, Y=y')$

$$P(Y=y \mid E=e) = \frac{P(Y=y, E=e)}{P(E=e)} = \frac{P(Y=y, E=e)}{\sum_{y' \in dom(Y)} P(Y=y', E=e)}$$

All we need to compute is the
joint probability of the query variable(s) and the evidence!

Variable Elimination: Intro (1)

- We can express the joint probability as a factor
  - observed: $E_1,\ldots, E_j$; Other variables not involved in the query: $Z_1,\ldots,Z_k$
  - $f(Y, E_1,\ldots, E_j, Z_1,\ldots,Z_k)$
- We can compute $P(Y, E_1=e_1, \ldots, E_j=e_j)$ by
  - Assigning $E_1=e_1, \ldots, E_j=e_j$
  - Marginalizing out variables $Z_1, \ldots, Z_k$, one at a time
    - the order in which we do this is called our elimination ordering

$$P(Y, E_1 = e_1, \ldots, E_j = e_j) = \sum_{Z_k} \cdots \sum_{Z_1} f(Y, E_1, .., E_j, Z_1, .., Z_k)_{E_1=e_1,\ldots,E_j=e_j}$$

- Are we done?
  - No. This would still represent the whole JPD (as a single factor)
  - We need to exploit the compactness of Bayesian networks

Variable Elimination: Intro (2)

- Recall the joint probability distribution of a Bayesian network
  - $P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|X_1,\ldots,X_{i-1})$

    $= \prod_{i=1}^{n} P(X_i|pa(X_i))$

- We will have a factor f_i for each conditional probability:
  - For each variable X_i, there is a factor f_i with domain $\{X_i\} \cup pa(X_i)$:

    $f_i(\{X_i\} \cup pa(X_i)) = P(X_i|pa(X_i))$

$$P(Y, E_1 = e_1, \ldots, E_j = e_j) = \sum_{Z_k} \cdots \sum_{Z_1} f(Y, E_1, .., E_j, Z_1, .., Z_k)_{E_1 = e_1, \ldots, E_j = e_j}$$

$$= \sum_{Z_k} \cdots \sum_{Z_1} \prod_{i=1}^{n} (f_i)_{E_1 = e_1, \ldots, E_j = e_j}$$

Computing sums of products

- Inference in Bayesian networks thus reduces to computing the sums of products
  - Example: it takes 9 multiplications to evaluate the expression ab + ac + ad + aeh + afh + agh.
  - How can this expression be evaluated more efficiently?
    - Factor out the a and then the h giving a(b + c + d + h(e + f + g))
    - This takes only 2 multiplications (same number of additions as above)
- Similarly, how can we compute $\sum_{Z_k} \prod_{i=1}^{n} f_i$ efficiently?
  - Factor out those terms that don't involve Z_k, e.g.:

$$\sum_{Z_k} f_1(Z_k) f_2(Y) f_3(Z_k, Y) f_4(X, Y)$$

$$= f_2(Y) f_4(X, Y) \left(\sum_{Z_k} f_1(Z_k) f_3(Z_k, Y) \right)$$

Summing out a variable efficiently

- To sum out a variable Z_k from a product $f_1 \times \ldots \times f_k$ of factors:
  - Partition the factors into
    - those that don't contain Z_k, say $f_1 \times \ldots \times f_i$
    - those that contain Z_k, say $f_{i+1} \times \ldots \times f_k$
- We know:

$$\sum_{Z_k} f_1 \times \ldots \times f_k = f_1 \times \ldots \times f_i \times \left(\sum_{Z_k} f_{i+1} \times \ldots \times f_k \right)$$

New factor! Let's call it f'

- We thus have $\sum_{Z_k} f_1 \times \ldots \times f_k = f_1 \times \ldots \times f_i \times f'$
- Store f' explicitly, and discard $f_{i+1} \ldots f_k$
- Now we've summed out Z_k

The variable elimination algorithm

To compute P(Y=y|E=e):

1. Construct a factor for each conditional probability
2. Assign the observed variables E to their observed values
3. Decompose the sum
4. Sum out all variables $Z_1, \ldots, Z_k$ not involved in the query
5. Multiply the remaining factors (which only involve Y)
6. Normalize by dividing the resulting factor f(Y) by $\sum_{y \in dom(Y)} f(Y)$

Lecture Overview

- Entailed Independencies: Recap and Examples
- Inference in General Bayesian Networks
  - Factors:
    - Assigning Variables
    - Summing out Variables
    - Multiplication of Factors
  - The variable elimination algorithm
  - Example trace of variable elimination

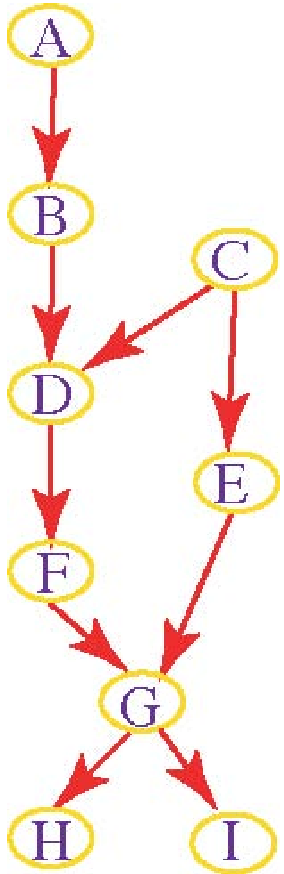
Variable elimination example: compute $P(G|H=h_1)$

Step 1: construct a factor for each cond. probability

$P(G,H) = \sum_{A,B,C,D,E,F,I} P(A,B,C,D,E,F,G,H,I) =$

$= \sum_{A,B,C,D,E,F,I} P(A)P(B|A)P(C)P(D|B,C)P(E|C)P(F|D)P(G|F,E)P(H|G)P(I|G)$

$= \sum_{A,B,C,D,E,F,I} f_0(A)\ f_1(B,A)\ f_2(C)\ f_3(D,B,C)\ f_4(E,C)\ f_5(F, D)\ f_6(G,F,E)\ f_7(H,G)\ f_8(I,G)$

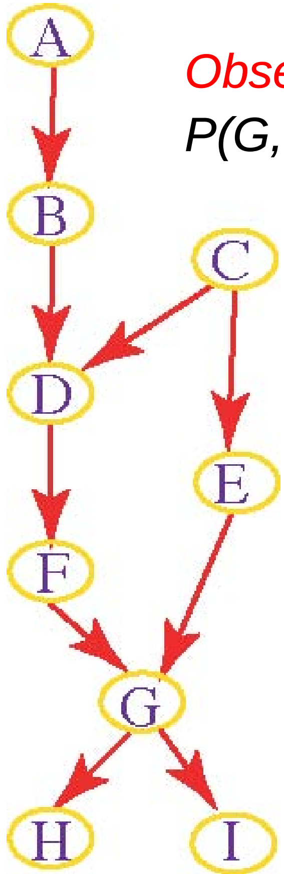


Variable elimination example: compute $P(G|H=h_1)$
Step 2: assign observed variables their observed value

$P(G,H) = \sum_{A,B,C,D,E,F,I} P(A,B,C,D,E,F,G,H,I) =$

$= \sum_{A,B,C,D,E,F,I} P(A)P(B|A)P(C)P(D|B,C)P(E|C)P(F|D)P(G|F,E)P(H|G)P(I|G)$

$= \sum_{A,B,C,D,E,F,I} f_0(A)\ f_1(B,A)\ f_2(C)\ f_3(D,B,C)\ f_4(E,C)\ f_5(F, D)\ f_6(G,F,E)\ f_7(H,G)\ f_8(I,G)$



Observe $H=h_1$:

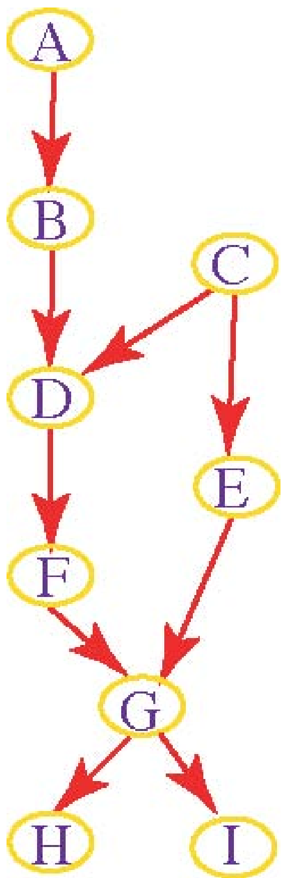
$P(G,H=h_1)=\sum_{A,B,C,D,E,F,I} f_0(A)\ f_1(B,A)\ f_2(C)\ f_3(D,B,C)\ f_4(E,C)$
$f_5(F, D)\ f_6(G,F,E)\ f_9(G)\ f_8(I,G)$

Assigning the variable $H=h_1$: $f_7(H,G)_{H=h1} = f_9(G)$

Variable elimination example: compute $P(G|H=h_1)$
Step 3: decompose sum

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\ f_1(B,A)\ f_2(C)\ f_3(D,B,C)\ f_4(E,C)\ f_5(F, D)\ f_6(G,F,E)\ f_9(G)\ f_8(I,G)$

$= \quad \sum_F \sum_D \quad \sum_B \sum_I \quad \sum_E \quad \sum_C \quad \sum_A$



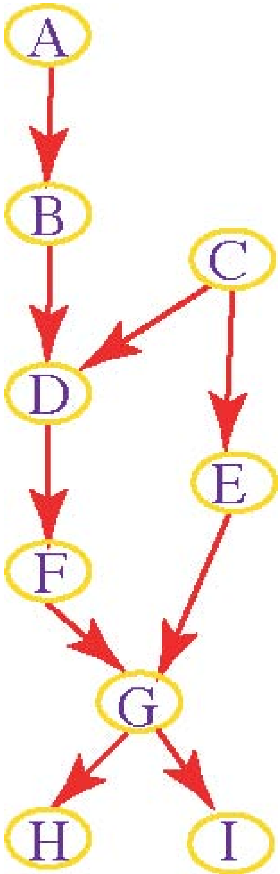
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$

Step 3: decompose sum

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$



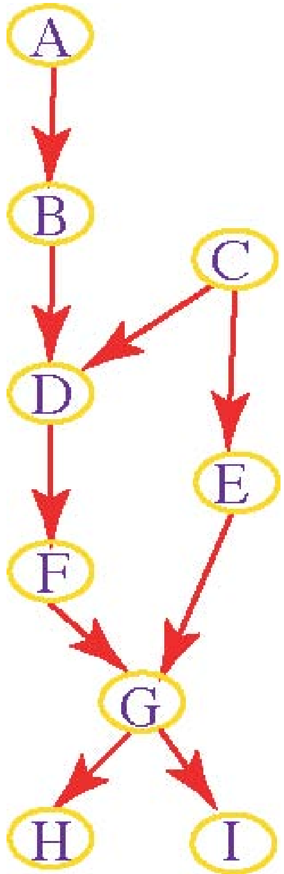
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 4: sum out non- query variables (one at a time)

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



Summing out A: $\sum_A f_0(A)\, f_1(B,A) = f_{10}(B)$
This new factor does not depend on C, E, or I, so we can push it outside of those sums.

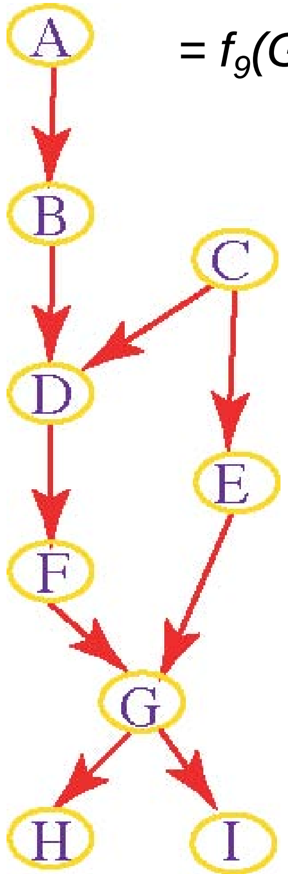
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 4: sum out non- query variables (one at a time)

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(D,B,E)$

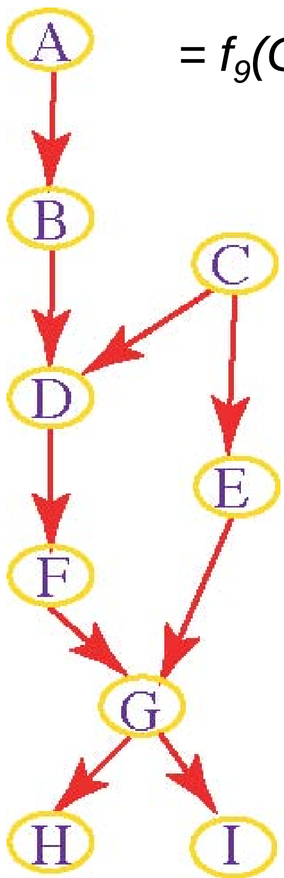
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 4: sum out non- query variables (one at a time)

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(D,B,E)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B) \sum_I f_8(I,G)$

Note the increase in dimensionality: $f_{12}(G,F,D,B)$ is defined over 4 variables

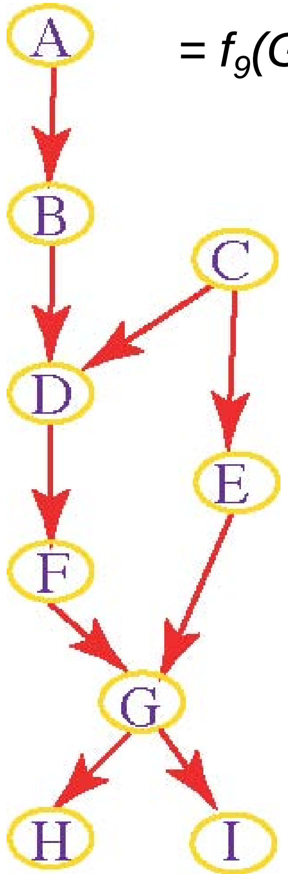
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 4: sum out non- query variables (one at a time)

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(D,B,E)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B) \sum_I f_8(I,G)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B)$

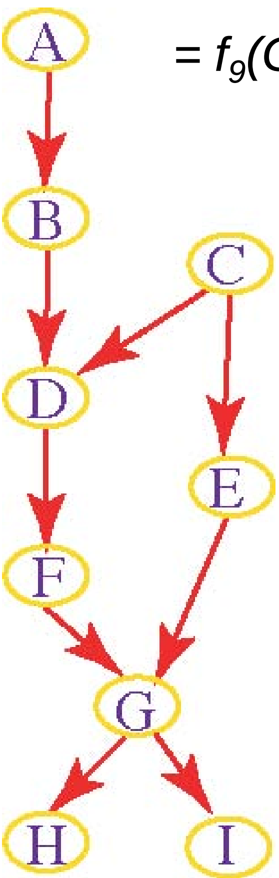
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 4: sum out non- query variables (one at a time)

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(D,B,E)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B) \sum_I f_8(I,G)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D)\, f_{14}(G,F,D)$

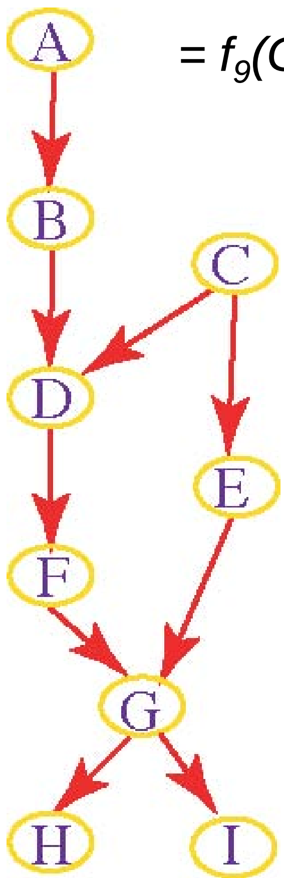
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 4: sum out non- query variables (one at a time)

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(D,B,E)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B) \sum_I f_8(I,G)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D)\, f_{14}(G,F,D)$

$= f_9(G)\, f_{13}(G) \sum_F f_{15}(G,F)$

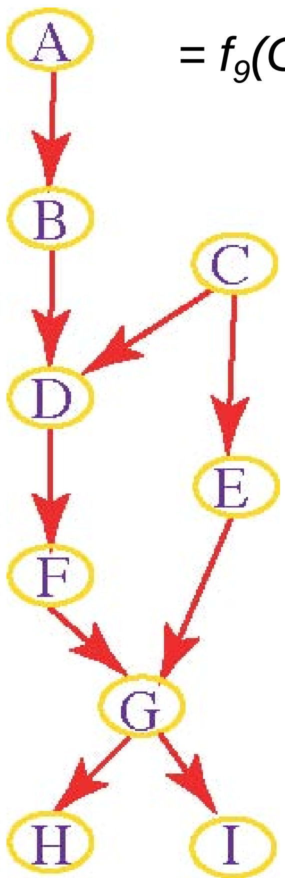
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 4: sum out non- query variables (one at a time)

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(D,B,E)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B) \sum_I f_8(I,G)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D)\, f_{14}(G,F,D)$

$= f_9(G)\, f_{13}(G) \sum_F f_{15}(G,F)$

$= f_9(G)\, f_{13}(G)\, f_{16}(G)$

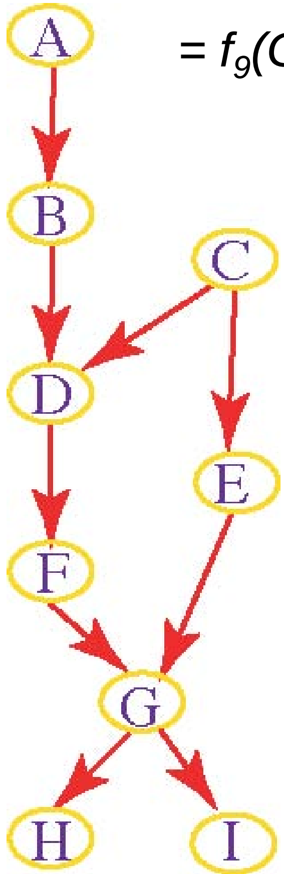
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 5: multiply the remaining factors

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(D,B,E)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B) \sum_I f_8(I,G)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D)\, f_{14}(G,F,D)$

$= f_9(G)\, f_{13}(G) \sum_F f_{15}(G,F)$

$= f_9(G)\, f_{13}(G)\, f_{16}(G)$

$= f_{17}(G)$

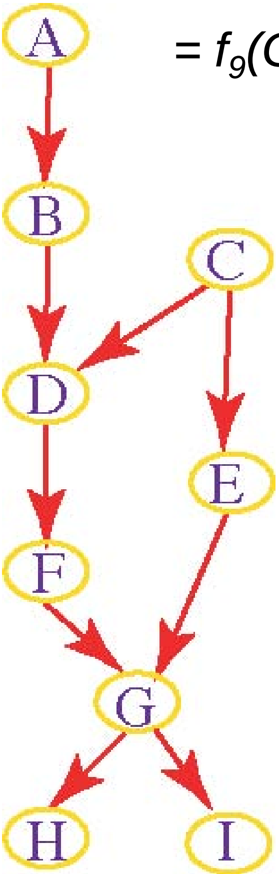
Elimination ordering: A, C, E, I, B, D, F

Variable elimination example: compute $P(G|H=h_1)$
Step 6: normalize

$P(G,H=h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F, D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$



$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(D,B,E)$

$= f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B) \sum_I f_8(I,G)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(G,F,D,B)$

$= f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D)\, f_{14}(G,F,D)$

$= f_9(G)\, f_{13}(G) \sum_F f_{15}(G,F)$

$= f_9(G)\, f_{13}(G)\, f_{16}(G)$

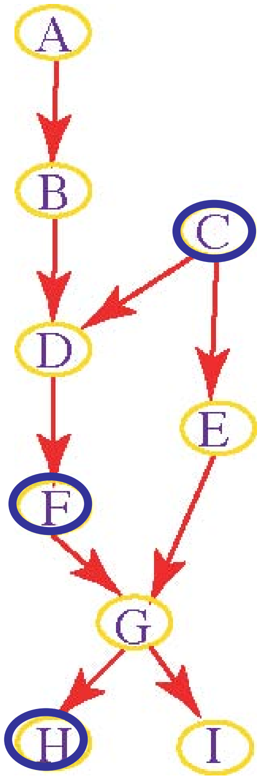
$= f_{17}(G)$

$$P(G=g \mid H=h_1) = \frac{P(G=g, H=h_1)}{P(H=h_1)}$$

$$= \frac{P(G=g, H=h_1)}{\sum_{g' \in dom(G)} P(G=g', H=h_1)} = \frac{f_{17}(g)}{\sum_{g' \in dom(G)} f_{17}(g')}$$

VE and conditional independence

- So far, we haven't use conditional independence!
  - Before running VE, we can prune all variables Z that are conditionally independent of the query Y given evidence E: $Z \perp\!\!\!\perp Y \mid E$

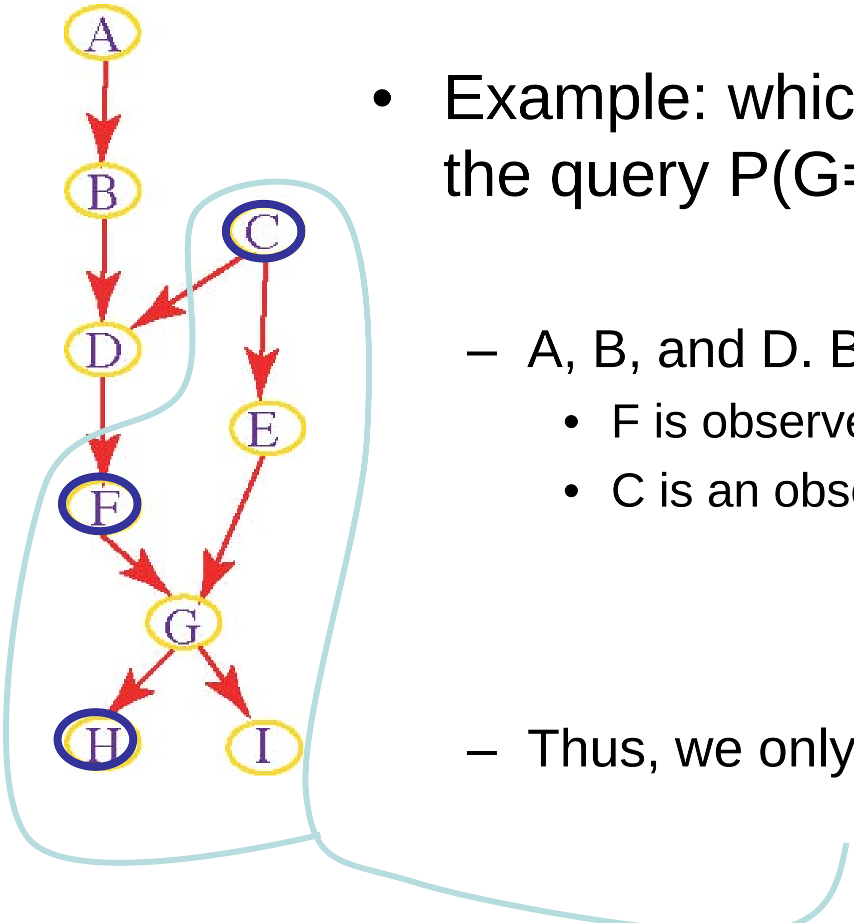


- Example: which variables can we prune for the query $P(G=g \mid C=c_1, F=f_1, H=h_1)$?

A B D E

VE and conditional independence

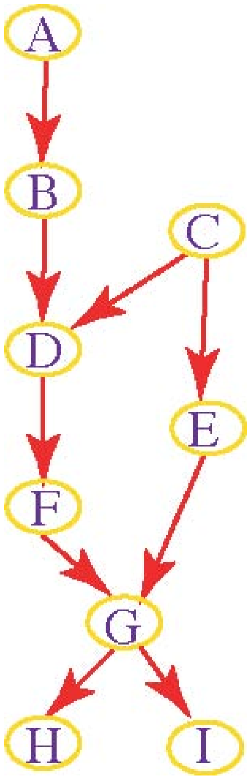
- So far, we haven't use conditional independence!
  - Before running VE, we can prune all variables Z that are conditionally independent of the query Y given evidence E: $Z \perp\!\!\!\perp Y \mid E$



- Example: which variables can we prune for the query $P(G=g \mid C=c_1, F=f_1, H=h_1)$?
  - A, B, and D. Both paths are blocked
    - F is observed node in chain structure
    - C is an observed common parent
  - Thus, we only need to consider this subnetwork

One last trick

- We can also prune unobserved leaf nodes
  - And we can do so recursively



E.g., which nodes can we prune if the query is P(A)?

H   I   G   All nodes other than A

Recursively prune unobserved leaf nodes:
we can prune all nodes other than A !

Learning Goals For Today's Class

- Identify implied (in)dependencies in the network
- Variable elimination
  - Carry out variable elimination by using factor representation and using the factor operations
  - Use techniques to simplify variable elimination

- Practice Exercises
  - Reminder: they are helpful for staying on top of the material, and for studying for the exam
  - Exercise 10 is on independence
  - Exercise 11 is on variable elimination
- Assignment 4 is due in one week
  - You should now be able to solve questions 1, 2, 3, and 5

- Final exam in two weeks: Monday, April 11