Reasoning under uncertainty

- Review of probabilities, independence, and Bayes rule
- Bayes net: concept, construction, inference, and independence
- Variable elimination algorithm

Next: machine learning

Reading: Chap 18, 20, 21

*Slides based on those of Sheila McIlraith

Uncertainty

- In search we viewed actions as being deterministic.
  - executing action A in state S_1 causes transition to state S_2
- Furthermore, there was a fixed initial state S_0.
- So after executing any sequence of actions, we know exactly what state we have arrived at.
- These assumptions are sensible in some domains, but in many domains they are not true.

Uncertainty

- We might not know exactly what state we start off in
  - *e.g.*, we can't see our opponents' cards in a poker game
  - We don't know what a patient's ailment is.
- We might not know all of the effects of an action
  - The action might have a random component, like rolling dice.
  - We might not know all of the long term effects of a drug.
  - An action might fail

Uncertainty

- In such domains we still need to act,
- but we can't act solely on the basis of known true facts.
- We have to "gamble".
- But how do we gamble rationally?

An example

We have to go to the airport. But we don't know for certain what the traffic will be like on the way to the airport. When do we leave?

- If we must arrive at the airport at 9pm on a week night
  - we could "safely" leave for the airport 1 hour before.
  - Some probability of the trip taking longer, but the probability is low.
- If we must arrive at the airport at 4:30pm on Friday
  - we most likely need 1.5 hour or more to get to the airport.

Uncertainty

- To act rationally under uncertainty, we must be able to evaluate how likely certain things are.
- By weighing likelihoods of events (probabilities), we can develop mechanisms for acting rationally under uncertainty.

Probability (over Finite Sets)

- Probability is a function defined over a set of events U, often called the universe of events.
- It assigns a value $Pr(e)$ to each event $e \in U$, in the range [0,1].
- It assigns a value to every set of events F by summing the probabilities of the members of that set: $Pr(F) = \sum_{e \in F} Pr(e)$
- Thus $Pr(U) = 1, Pr(\emptyset) = 0$
- $Pr(A \cup B) = Pr(A) + Pr(B) - Pr(A \cap B)$

Probability in General

Given a set U (universe), a probability function is a function defined over the subsets of U that maps each subset to the real numbers and that satisfies the Axioms of Probability

- $Pr(U) = 1$
- $Pr(A) \in [0, 1]$
- $Pr(A \cup B) = Pr(A) + Pr(B) - Pr(A \cap B)$

Probability over Feature Vectors

- We will work with a universe consisting of a set of vectors of feature values.
- Like CSPs, we have
  - a set of variables $V_1, V_2, \ldots, V_n$
  - a finite domain of values for each variable, Dom[V_1], Dom[V_2],. . . , Dom[V_n].
- The universe of events U is the set of all vectors of values for the variables $\{\langle d_1, d_2, \ldots, d_n\rangle \mid d_i \in Dom[Vi]\}$
- This event space has size $\prod_i |Dom[V_i]|$, *i.e.*, the product of the domain sizes.
- *e.g.*, if $|Dom[V_i]| = 2$, we have 2^n distinct atomic events. (Exponential!)

Probability over Feature Vectors

- Asserting that some subset of variables have particular values allows us to specify a useful collection of subsets of U, *e.g.*
  - $\{V_1 = a\}$ = set of all events where $V_1 = a$
  - $\{V_1 = a, V_3 = d\}$ = set of all events where $V_1 = a$ and $V_3 = d$.
- If we had Pr of every atomic event (full instantiation of the variables) we could compute Pr of any other set, *e.g.*
  $Pr(\{V_1 = a\}) =$
  $\sum_{x_2 \in D[V_2]} \cdots \sum_{x_n \in D[V_n]} Pr(V_1 = a, V_2 = x_2, \ldots, V_n = x_n)$

Problem and solution

Problem

- This is an exponential number of atomic probabilities to specify.
- Requires summing up an exponential number of items.

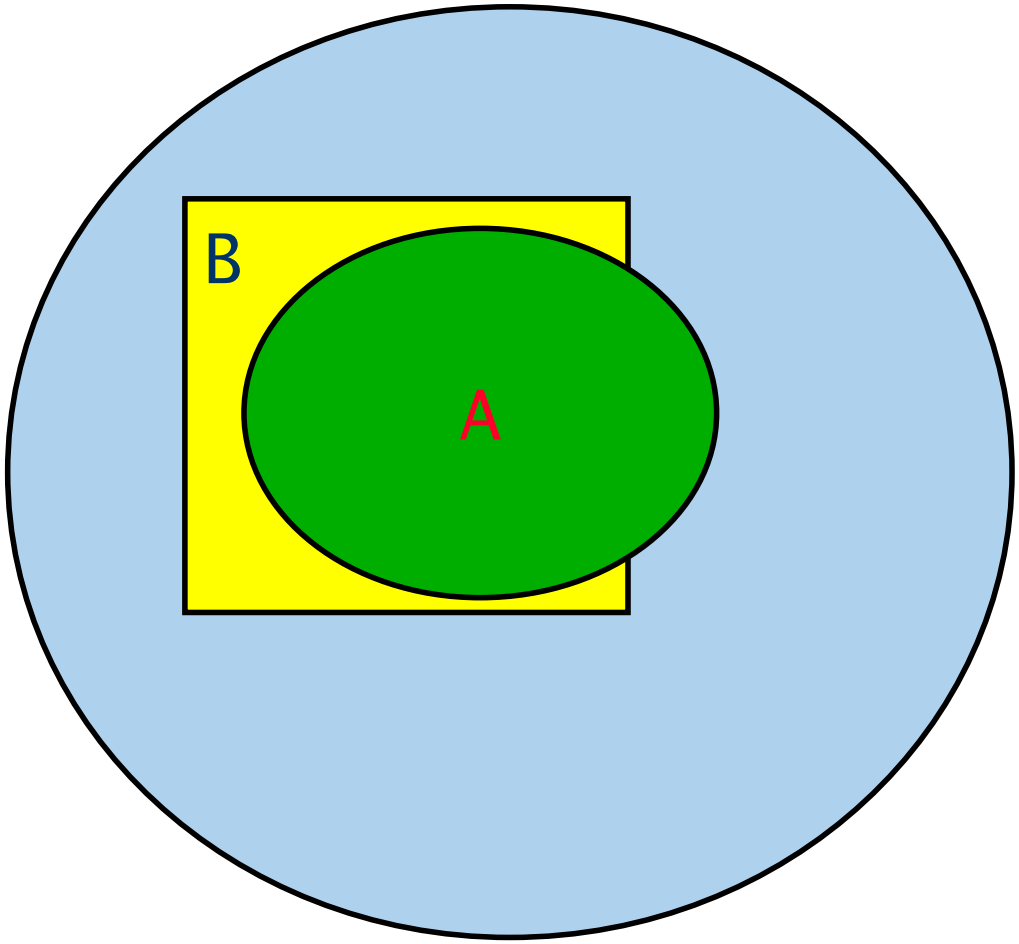
Solution

- Make use of probabilistic independence, especially conditional independence.

Conditional probabilities

- Say that A is a set of events such that $Pr(A) > 0$.
- Then one can define a conditional probability wrt the event A: $Pr(B|A) = Pr(B \cap A)/Pr(A)$
- Conditioning on A, corresponds to restricting one's attention to the events in A.

An example



B covers about 30% of the entire space, but covers over 80% of A.

So $Pr(B) = 0.3$, but $Pr(B|A) = 0.8$

Properties and sets

Any set of events A can be interpreted as a property: the set of events with property A. Hence, we often write

- $A \vee B$ to represent the set of events with either property A or B: the set $A \cup B$
- $A \wedge B$ to represent the set of events with both property A and B: the set $A \cap B$
- $\neg A$ to represent the set of events that do not have property A: the set $U - A$ (*i.e.*, the complement of A wrt the universe of events U)

Summing out rule

- Say that $B_1, B_2, \ldots, B_k$ form a partition of the universe U.
  - $B_i \cap B_j = \emptyset$, $i \neq j$ (mutually exclusive)
  - $B_1 \cup B_2 \cup \ldots \cup B_k = U$ (exhaustive)
- In probabilities:
  - $Pr(B_i \cap B_j) = 0$, $i \neq j$
  - $Pr(B_1 \cup B_2 \cup \ldots \cup B_k) = 1$
- Given any other set of events A, we have that
  $Pr(A) = Pr(A \cap B_1) + \ldots + Pr(A \cap B_k)$
- In conditional probabilities:
  $Pr(A) = Pr(A|B_1)Pr(B_1) + \ldots + Pr(A|B_k)Pr(B_k)$
- Often we know $Pr(A|B_i)$, so we can compute $Pr(A)$ this way.

Independence

- It could be that the density of B on A is identical to its density on the entire set.
  - Density: pick an element at random from the entire set. How likely is it that the picked element is in the set B?
- Alternately, the density of B on A could be much different from its density on the whole space.
- In the first case, $Pr(B|A) = Pr(B)$, we say that B is independent of A.
- In this case, knowing an element belongs to A does not tell us anything more about whether it also belongs to B

Conditional independence

- Say we have already learned that a randomly picked element has property A.
- We want to know whether or not the element has property B:
  - $Pr(B|A)$ expresses the probability of this being true.
- Now we learn that the element also has property C. Does this give us more information about B-ness?
  - $Pr(B|A \cap C)$ expresses the probability of this being true under the additional information.

Conditional independence

- If $Pr(B|A \cap C) = Pr(B|A)$, then we have not gained any additional information from knowing that the element is in C.
- In this case we say that B is conditionally independent of C given A.
- That is, once we know A, additionally knowing C is irrelevant (wrt whether or not B is true).
- Conditional independence is independence in the conditional probability space $Pr(\bullet|A)$.

Computational Impact of Independence

- If A and B are independent, then $Pr(A \cap B) = Pr(A) \cdot Pr(B)$
- If given A, B and C are conditionally independent, then $Pr(B \cap C|A) = Pr(B|A) \cdot Pr(C|A)$

Bayes rule

- Bayes rule is a simple mathematical fact. But it has great implications wrt how probabilities can be reasoned with.
- $Pr(Y|X) = Pr(X|Y)Pr(Y)/Pr(X)$
- *e.g.*, from treating patients with heart disease we might be able to estimate the value of $Pr(high\_Cholesterol|heart\_disease)$
- With Bayes rule, we can turn this around into a predictor for heart disease $Pr(heart\_disease|high\_Cholesterol)$
- With a simple blood test we can determine "high Cholesterol", and use it to help estimate the likelihood of heart disease.

Bayes Rule Example

- Disease $\in \{malaria, cold, flu\}$; Symptom $=$ fever
- Must compute $Pr(Disease|fever)$ to prescribe treatment
- Why not assess this quantity directly?
  - $Pr(mal|fever)$ – is not natural to assess. It does not reflect the underlying "causal mechanism" malaria $\Rightarrow$ fever
  - $Pr(mal|fever)$ – is not "stable": a malaria epidemic changes this quantity (for example)
- So we use Bayes rule:
  $Pr(mal|fever) = Pr(fever|mal)Pr(mal)/Pr(fever)$

Bayes Rule Example

- What about $Pr(fever)$
- Say that malaria, cold and flu are the only possible causes of fever, *i.e.*, $Pr(fever|\neg malaria \wedge \neg cold \wedge \neg flu) = 0$, and they are mutually exclusive.
- Then $Pr(fever) =$ $Pr(malaria \wedge fever) + Pr(cold \wedge fever) + Pr(flu \wedge fever)$
- $Pr(malaria \wedge fever) = Pr(fever|mal)Pr(mal)$
- Similarly, we can obtain $Pr(cold \wedge fever)$ and $Pr(flu \wedge fever)$

Chain rule

$$Pr(A_1 \cap A_2 \cap \ldots \cap A_n) = Pr(A_1|A_2 \cap \ldots \cap A_n) \cdot$$
$$Pr(A_2|A_3 \cap \ldots \cap A_n) \cdot \ldots \cdot Pr(A_{n-1}|A_n) \cdot Pr(A_n)$$

Useful equations

- Conditional probability: $Pr(B|A) = Pr(B \cap A)/Pr(A)$
- Summing out rule:
  Say that $B_1, B_2, \ldots, B_k$ form a partition of U. Then
  $Pr(A) = Pr(A \cap B_1) + \ldots + Pr(A \cap B_k)$
- If A and B are independent, then
  $Pr(A \cap B) = Pr(A) \cdot Pr(B)$
- If given A, B and C are conditionally independent, then
  $Pr(B \cap C|A) = Pr(B|A) \cdot Pr(C|A)$
- Bayes rule: $Pr(Y|X) = Pr(X|Y)Pr(Y)/Pr(X)$
- Chain rule: $Pr(A_1 \cap A_2 \cap \ldots \cap A_n) = Pr(A_1|A_2 \cap \ldots \cap A_n) \cdot$
  $Pr(A_2|A_3 \cap \ldots \cap A_n) \cdot \ldots \cdot Pr(A_{n-1}|A_n) \cdot Pr(A_n)$

Variable Independence

Two variables X and Y are conditionally independent given variable Z if for all $x \in \text{Dom}(X)$, $y \in \text{Dom}(Y)$, $z \in \text{Dom}(Z)$, $X = x$ and $Y = y$ are conditionally independent given $Z = z$, *i.e.*, $Pr(X = x \wedge Y = y | Z = z) =$

$$Pr(X = x | Z = z) \cdot Pr(Y = y | Z = z)$$

Notation/Terminology

- $Pr(X)$ for variable X refers to the (marginal) distribution over X.
- It specifies $Pr(X = d)$ for all $d \in Dom[X]$
- Note $\sum_{d \in Dom[X]} Pr(X = d) = 1$
- Also $Pr(X = d_1 \wedge X = d_2) = 0$, for all $d_1, d_2 \in Dom[X]$ s.t. $d_1 \neq d_2$

Notation/Terminology

- $Pr(X|Y)$ refers to family of conditional distributions over X, one for each $y \in Dom(Y)$.
- For each $d \in Dom[Y]$, $Pr(X|Y = d)$ specifies a distribution over the values of X: $Pr(X = d_1|Y = d)$, $Pr(X = d_2|Y = d), \ldots, Pr(X = d_n|Y = d)$, where $Dom[X] = \{d_1, d_2, \ldots, d_n\}$.
- Distinguish between $Pr(X)$ which is a distribution and $Pr(X = d)$ $(d \in Dom[X])$ – which is a number.
- Think of $Pr(X)$ as a function that accepts any $x \in Dom[X]$ as an argument and returns $Pr(X = x)$.
- Similarly, think of $Pr(X|Y)$ as a function that accepts any $y \in Dom[Y]$ and returns a distribution $Pr(X|Y = y)$.

What does independence buy us?

- Suppose Boolean variables $X_1, X_2, \ldots, X_n$ are mutually independent (*i.e.*, every subset is variable independent of every other subset)
- We can specify full joint distribution (probability function over all vectors of values) using only n parameters (linear) instead of $2^n - 1$ (exponential)
- Simply specify $Pr(X_1 = true), \ldots, Pr(X_n = true)$ (*i.e.*, $Pr(X_i = true)$ for all i)
- We can easily recover probability of any primitive event, *e.g.*
- $Pr(X_1 \neg X_2 X_3 X_4) = Pr(X_1)(1 - Pr(X_2))Pr(X_3)Pr(X_4)$

The Value of Independence

- Complete independence reduces both representation and inference from $O(2^n)$ to $O(n)$!
- Unfortunately, such complete mutual independence is rare.
- Most realistic domains do not exhibit this property.
- Fortunately, most domains do exhibit a fair amount of conditional independence.
- And we can exploit conditional independence for representation and inference as well.
- Bayesian networks do just this

Exploiting Conditional Independence

Consider a story:

- If Craig woke up too early E, Craig probably needs coffee C;
- if C, he's likely angry A.
- If A, there is an increased chance of a burst blood vessel B.
- If B, Craig is quite likely to be hospitalized H.

E - Craig woke too early A - Craig is angry H - Craig hospitalized
C - Craig needs coffee B - Craig burst a blood vessel

Exploiting Conditional Independence

- If you learned any of E, C, A, or B, your assessment of $Pr(H)$ would change.
  - *e.g.*, if any of these are seen to be true, you would increase $Pr(h)$ and decrease $Pr(\neg h)$.
  - So H is not independent of E, or C, or A, or B.
- But if you knew value of B (true or false), learning value of E, C, or A, would not influence $Pr(H)$. Influence these factors have on H is mediated by their influence on B.
  - Craig doesn't get sent to the hospital because he's angry, he gets sent because he's had a burst blood vessel.
  - So H is independent of E, and C, and A, given B

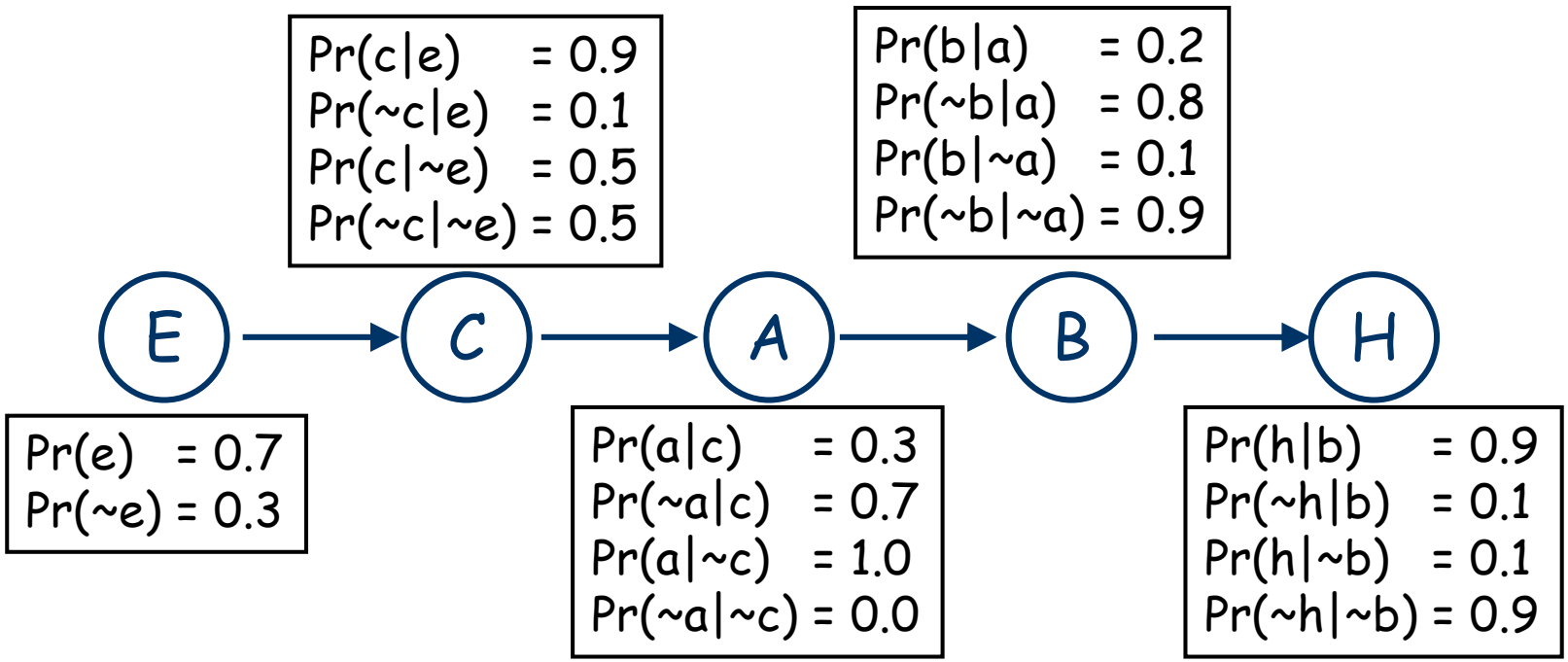
Exploiting Conditional Independence

- Similarly
  - B is independent of E, and C, given A
  - A is independent of E, given C
- This means that:
  - $Pr(H|B, \{A, C, E\}) = Pr(H|B)$
  - $Pr(B|A, \{C, E\}) = Pr(B|A)$
  - $Pr(A|C, \{E\}) = Pr(A|C)$
  - $Pr(C|E)$ and $Pr(E)$ don't simplify

Exploiting Conditional Independence

- By the chain rule (for any instantiation of H,. . . ,E):
  $Pr(H, B, A, C, E) =$
  $Pr(H|B, A, C, E)Pr(B|A, C, E)Pr(A|C, E)Pr(C|E)Pr(E)$
- By our independence assumptions:
  $Pr(H, B, A, C, E) =$
  $Pr(H|B)Pr(B|A)Pr(A|C)Pr(C|E)Pr(E)$
- Thus we can specify the full joint distribution by specifying five local conditional distributions:
  $Pr(H|B); Pr(B|A); Pr(A|C); Pr(C|E)$; and $Pr(E)$

Example quantification



- Note that half of these are "1 minus" the others
- So specifying the full joint requires only 9 parameters, instead of 31 for explicit representation
- Linear in number of variables instead of exponential!
- Linear generally if dependence has a chain structure

Inference is Easy

Want to know P(a)? Use summing out rule:

$$\Pr(a) = \sum_{c_i \in Dom(C)} \Pr(a \mid c_i) \Pr(c_i)$$

$$= \sum_{c_i \in Dom(C)} \Pr(a \mid c_i) \sum_{e_i \in Dom(E)} \Pr(c_i \mid e_i) \Pr(e_i)$$

These are all terms specified in our local distributions!

Inference is Easy

- Computing Pr(a) in more concrete terms:
  - Pr(c) = Pr(c|e)Pr(e) + Pr(c|~e)Pr(~e)
    = 0.9 * 0.7 + 0.5 * 0.3 = 0.78
  - Pr(~c) = Pr(~c|e)Pr(e) + Pr(~c|~e)Pr(~e) = 0.22
    - Pr(~c) = 1 – Pr(c), as well
  - Pr(a) = Pr(a|c)Pr(c) + Pr(a|~c)Pr(~c)
    = 0.3 * 0.78 + 1.0 * 0.22 = 0.454
  - Pr(~a) = 1 – Pr(a) = 0.546

Bayesian Networks: graph + tables

- The structure above is a Bayesian network.
- A BN is a graphical representation of the direct dependencies over a set of variables, together with a set of conditional probability tables (CPTs) quantifying the strength of those influences.
- Bayes nets generalize the above ideas in very interesting ways, leading to effective means of representation and inference under uncertainty.

Bayesian Networks

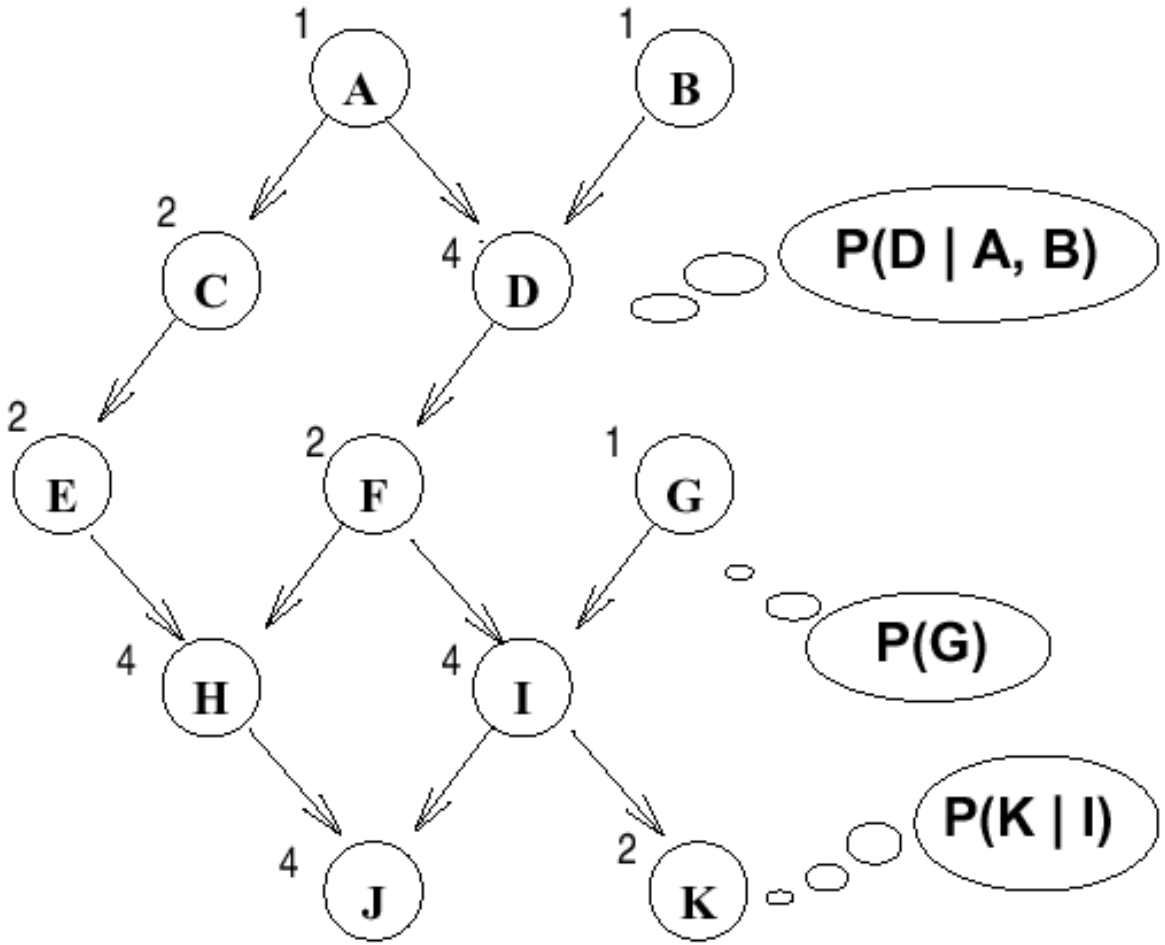
A BN over variables $\{X_1, X_2, \ldots, X_n\}$ consists of:

- a DAG (directed acyclic graph) whose nodes are the variables
- a set of CPTs (conditional probability tables) $Pr(X_i|Par(X_i))$ for each X_i

Key notions

- parents of a node: $Par(X_i)$
- children of a node
- descendents of a node
- ancestors of a node
- family: set of nodes consisting of X_i and its parents

Example (Binary valued Variables)



- A couple CPTS are “shown”
- Explicit joint requires 2^{11} -1 =2047 parmtrs
- BN requires only 27 parmtrs (the number of entries for each CPT is listed)

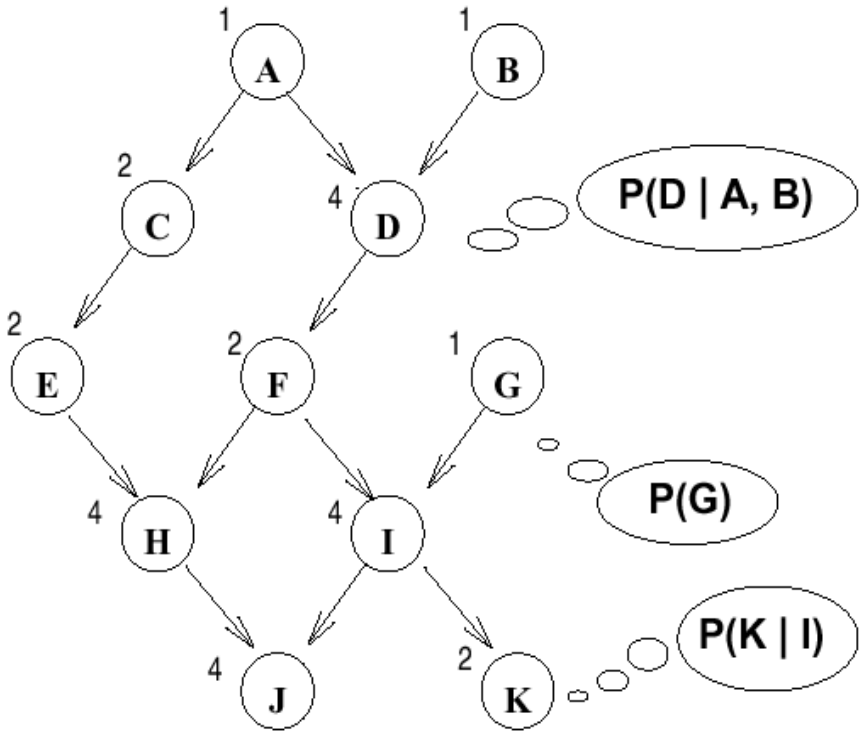
Semantics of Bayes Nets

- A Bayes net specifies that the joint distribution over the variable in the net can be written as the following product decomposition.
  $Pr(X_1, X_2, \ldots, X_n) = Pr(X_n|Par(X_n)) *$
  $Pr(X_{n-1}|Par(X_{n-1})) * \ldots * Pr(X_1|Par(X_1))$
- This equation holds for any set of values $d_1, d_2, \ldots, d_n$ for the variables $X_1, X_2, \ldots, X_n$.
- *e.g.*, We have X_1, X_2, X_3 each with domain $Dom[X_i] = \{a, b, c\}$ and we have $Pr(X_1, X_2, X_3) = P(X_3|X_2)P(X_2)P(X_1)$
- Then $Pr(X_1 = a, X_2 = a, X_3 = a) =$
  $P(X_3 = a|X_2 = a)P(X_2 = a)P(X_1 = a)$

Example (Binary valued Variables)

Pr(A,B,C,D,E,F,G,H,I,J,K) =

Pr(A)
x Pr(B)
x Pr(C|A)
x Pr(D|A,B)
x Pr(E|C)
x Pr(F|D)
x Pr(G)
x Pr(H|E,F)
x Pr(I|F,G)
x Pr(J|H,I)
x Pr(K|I)



Constructing a Bayes Net

It is always possible to construct a Bayes net to represent any distribution over the variables $X_1, X_2, \ldots, X_n$, using any ordering of the variables.

- Step 1. Apply the Chain Rule using any order of variables. (We will see later that you may wish to use causality or some other property to guide the variable ordering.)
  $Pr(X_1, \ldots, X_n) =$
  $Pr(X_n|X_1, \ldots, X_{n-1})Pr(X_{n-1}|X_1, \ldots, X_{n-2}) \ldots Pr(X_1)$
- Step 2. For each X_i go through its conditioning set $X_1, \ldots, X_{i-1}$ and iteratively remove all variables X_j such that X_i is conditionally independent of X_j given the remaining variables. Do this until no more variables can be removed.

Constructing a Bayes Net: Step 3

- Step 2 will yield a product decomposition.
  $Pr(X_n|Par(X_n))Pr(X_{n-1}|Par(X_{n-1}))\ldots Pr(X_1)$
- To create the Bayes Net, create a directed acyclic graph (DAG) such that each variable is a node and the conditioning set $Par(X_i)$ of a variable X_i are X_i's parents in the DAG.

Constructing a Bayes Net: Step 4

- Specify the conditional probability table (CPT) for each family (variable and its parents).
- Typically we represent the CPT as a table mapping each setting of $\{X_i, Par(X_i)\}$ to the numeric probability of X_i taking that particular value given that the variables in $Par(X_i)$ have their specified values.
- If each variable has d different values, we will need a table of size $d^{|\{X_i, Par(X_i)\}|}$.
- *i.e.*, exponential in the size of the parent set.

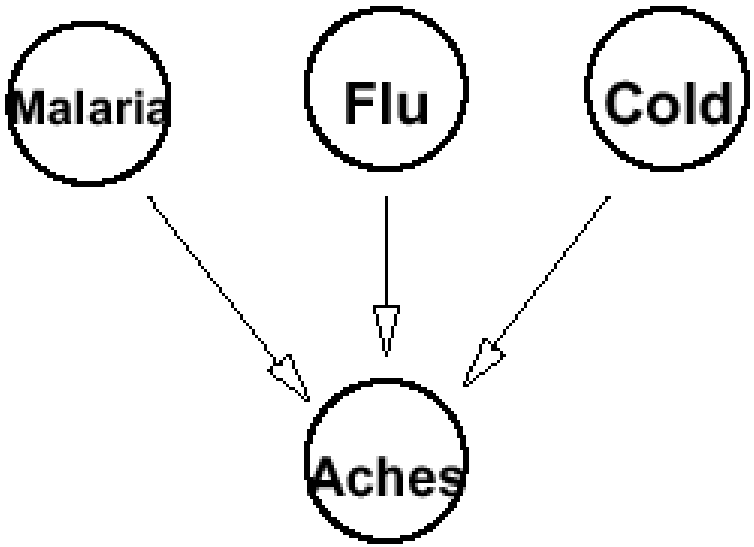
Variable Ordering Matters - Causal Intuitions

- The BN can be constructed using an arbitrary ordering of the variables.
- However, some orderings will yield BNs with very large parent sets. This requires exponential space, and (as we will see later) exponential time to perform inference.
- Empirically, and conceptually, a good way to construct a BN is to use an ordering based on causality. This often yields a more natural and compact BN.

Causal Intuitions

- Malaria, the flu and a cold all cause aches. So use the ordering that causes come before effects: Malaria, Flu, Cold, Aches
  $Pr(M, F, C, A) =$
  $Pr(A|M, F, C)Pr(C|M, F)Pr(F|M)Pr(M)$
- Each of these disease affects the probability of aches, so the first conditional probability does not change.
- It is reasonable to assume that these diseases are independent of each other: having or not having one does not change the probability of having the others.
- So $Pr(C|M, F) = Pr(C)$, $Pr(F|M) = Pr(F)$

Causal intuitions

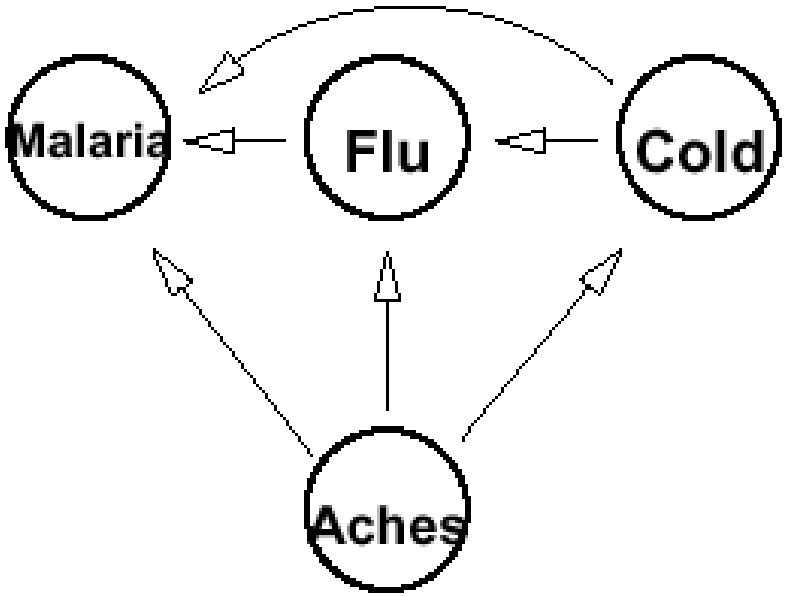


- This yields a fairly simple Bayes net.
- Only need one big CPT, involving the family of "Aches".

Causal Intuitions

- Suppose we build the BN using the opposite ordering: Aches, Cold, Flu, Malaria
  $Pr(A, C, F, M) =$
  $Pr(M|A, C, F)Pr(F|A, C)Pr(C|A)Pr(A)$
- We can't reduce $Pr(M|A, C, F)$.
  - Probability of Malaria is clearly affected by knowing aches.
  - How about knowing aches and cold, or aches and cold and flu?
  - Probability of Malaria is affected by both of these additional pieces of knowledge
  - Knowing Cold and of Flu lowers the probability of Aches indicating Malaria since they "explain away" Aches!
- Similarly, we can't reduce $Pr(F|A, C)$.
- $Pr(C|A) \neq Pr(C)$

Causal intuitions



- Obtain a much more complex Bayes net. In fact, we obtain no savings over explicitly representing the full joint distribution (i.e., representing the probability of every atomic event).

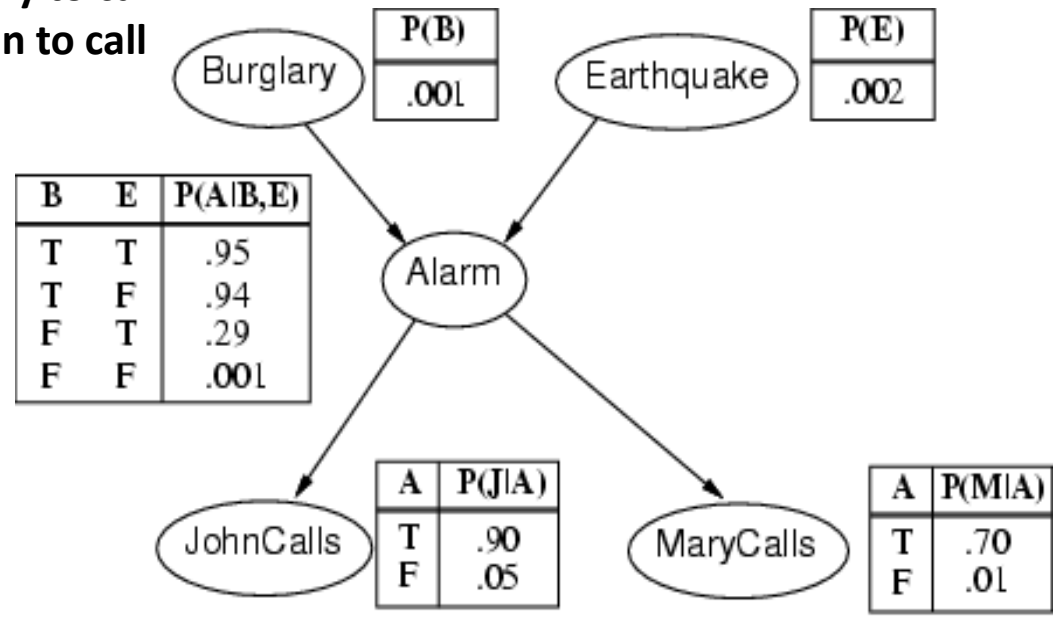
The Classic Burglary Example

- I'm at work, neighbour John calls to say my alarm is ringing, but neighbour Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?
- Variables: Burglary, Earthquake, Alarm, JohnCalls, MaryCalls
- Network topology reflects "causal" knowledge:
  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause Mary to call
  - The alarm can cause John to call

Burglary example

- **A burglary can set the alarm off**
- **An earthquake can set the alarm off**
- **The alarm can cause Mary to call**
- **The alarm can cause John to call**

Note that these tables only provide the probability that Xi is true.
(E.g., Pr(A is true|B,E))
The probability that Xi is false is 1- these values



Number of Parameters: 1 + 1 + 4 + 2 + 2 = **10** (vs. 2^5-1 = **31**)
…and note that these are binary (2) rather than multi-valued variables

Via the chain rule: $Pr(B, E, A, J, M)$
$= Pr(B|E, A, J, M)Pr(E|A, J, M)Pr(A|J, M)Pr(J|M)Pr(M)$
But using the Bayes Net:
$= Pr(B)Pr(E)Pr(A|B, E)Pr(J|A)Pr(M|A)$

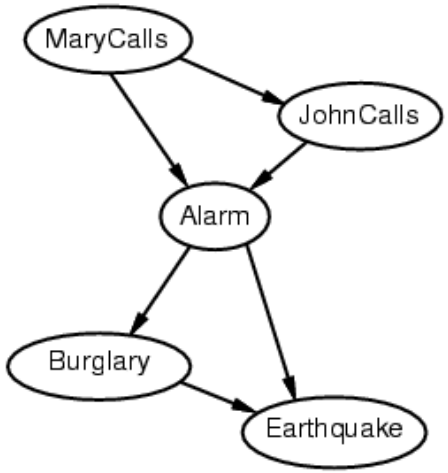
Example of Constructing Bayes Network

- Previously we chose a causal order.
- Now suppose we choose the ordering MaryCalls (M), JohnCalls (J), Alarm (A), Burglary(B), Earthquake(E), *i.e.*, M,J,A,B,E
- These "orderings" are the ordering of the arrows in the Bayes Net DAG, which are the opposite to the ordering of variables in the chain rule, *i.e.*, $Pr(E, B, A, J, M) = Pr(E|B, A, J, M) * Pr(B|A, J, M) * Pr(A|J, M) * Pr(J|M) * Pr(M)$
- Now let's see if we can get rid of the conditioning sets

Example cont'd

Suppose we choose the ordering M, J, A, B, E

Pr(E,B,A,J,M) = Pr(E|B,A,J,M)*Pr(B|A,J,M)*Pr(A|J,M)*Pr(J|M)*Pr(M)



Pr(J | M) = ***P***(*J*)? --- **No**

Pr(A | J, M) = ***Pr***(*A | J*)? --- **No**

Pr(A | J, M) = ***P***(*A*)? --- **No**

Pr(B | A, J, M) = ***Pr***(*B | A*)? --- **Yes**

Pr(B | A, J, M) = ***Pr***(*B*)? --- **No**

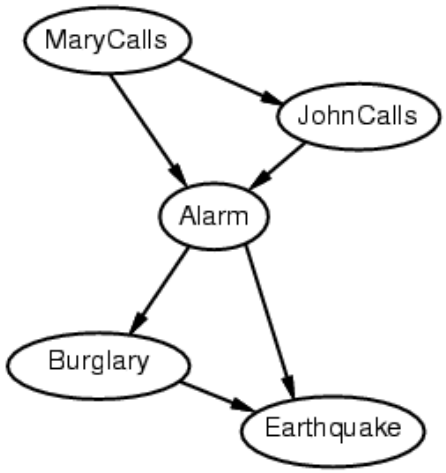
Pr(E|B,A,J,M) = ***Pr***(*E | A*)? --- **No**

Pr(E|B,A,J,M) = ***Pr***(*E | A, B*)? --- **Yes**

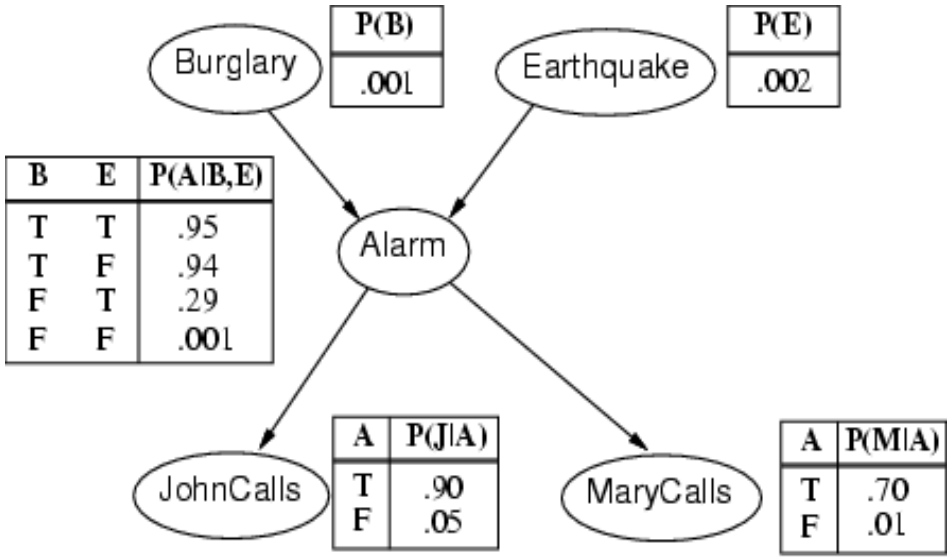
Example cont'd

Deciding conditional independence is hard in the non-causal direction!
Causal models & conditional independence seem hardwired for humans.

Network is **less compact**: 1 + 2 + 4 + 2 + 4 = 13 numbers needed!



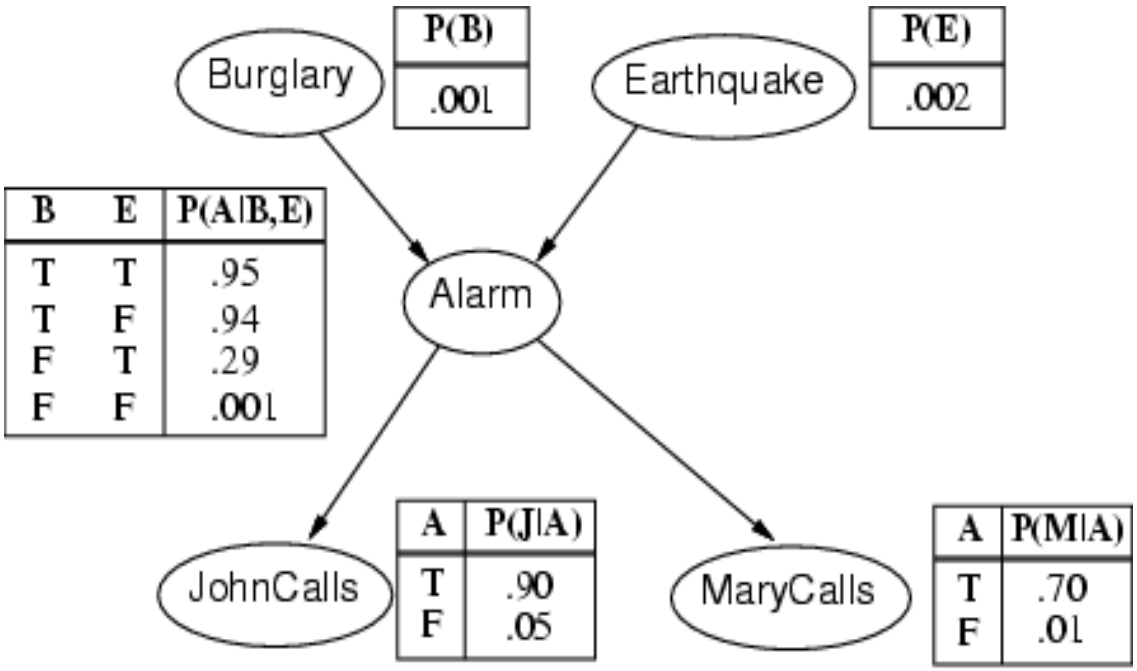
Burglary example



- Why $Pr(J|M) \neq Pr(J)$?
- Why $Pr(E|B, A, J, M) = Pr(E|B, A)$?
- Why $Pr(E|B, A) \neq Pr(E|A)$?

Independence in a Bayes Net

- The structure of the BN means: every X_i is conditionally independent of all of its nondescendants given it parents:
  - $Pr(X_i|S \cup Par(Xi)) = Pr(Xi|Par(Xi))$ for any set S of non-descendents of X_i
- Given Alarm, JohnCalls and Earthquake are independent



More generally

- Many conditional independencies hold in a given BN.
- These independencies are useful in computation, explanation, etc.
- How do we determine if two variables X, Y are independent given a set of variables E?
- Answer: we use a (simple) graphical property

D-separation

- A set of variables E d-separates X and Y if it blocks every undirected path in the BN between X and Y.
- If evidence E d-separates X and Y, then X and Y are conditionally independent given evidence E
- So what is blocking?

Let P be an **undirected path** from X to Y in a BN.

Let **E** be a set of variables.

We say ***E blocks path P*** iff there is some node Z on the path such that:

Case 1: one arc on P ***goes into*** Z and one ***goes out*** of Z, and Z$\in$**E**; or

Case 2: both arcs on P leave Z, and Z$\in$**E**; or

Case 3: both arcs on P enter Z and *neither Z, nor any of its descendants*, are in **E**.

Blocking: Graphical View

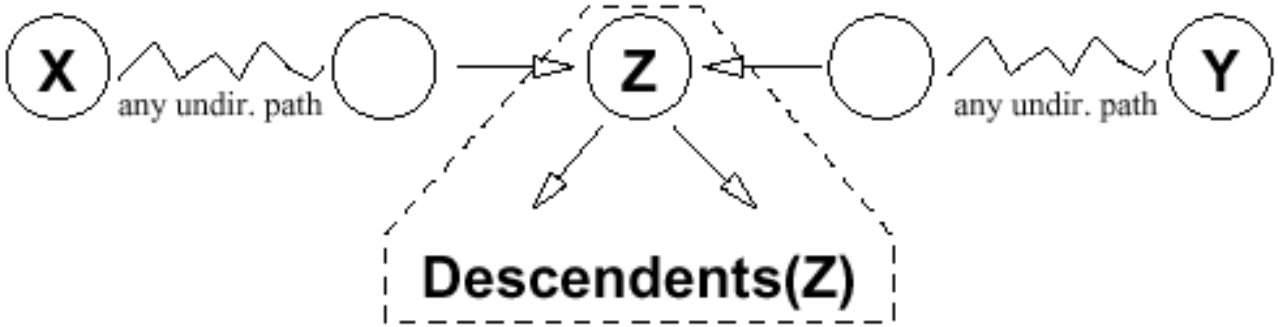
(1)

If Z in evidence, the path between X and Y blocked

(2)

If Z in evidence, the path between X and Y blocked

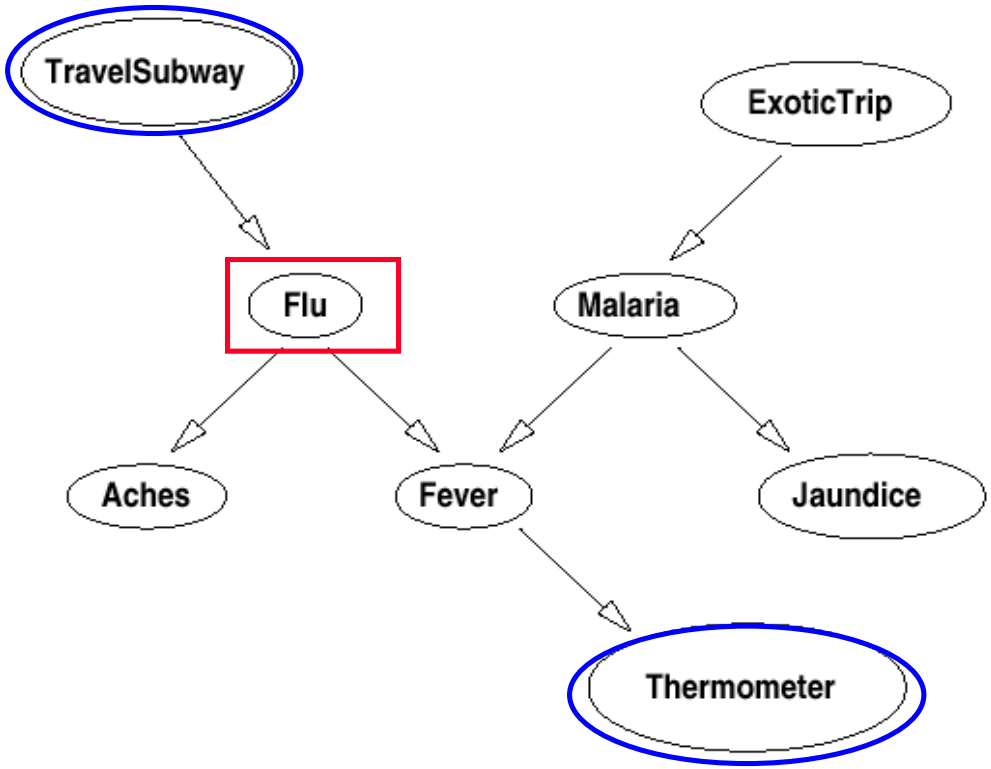
(3)

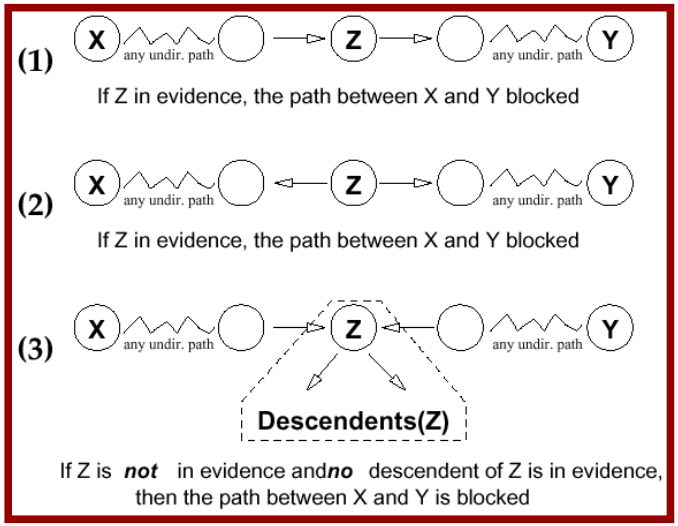


If Z is ***not*** in evidence and***no*** descendent of Z is in evidence, then the path between X and Y is blocked

D-Separation: Intuitions

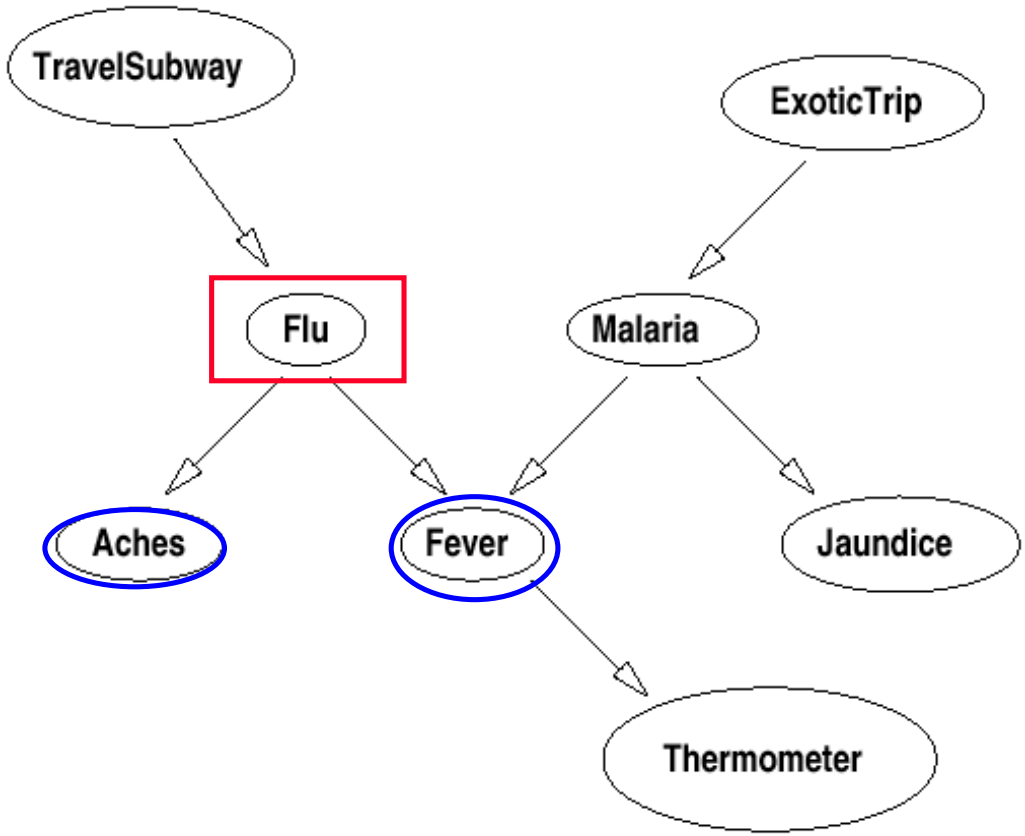
Subway and Thermometer are **dependent**; but are **independent given Flu** (since Flu blocks the only path (1))

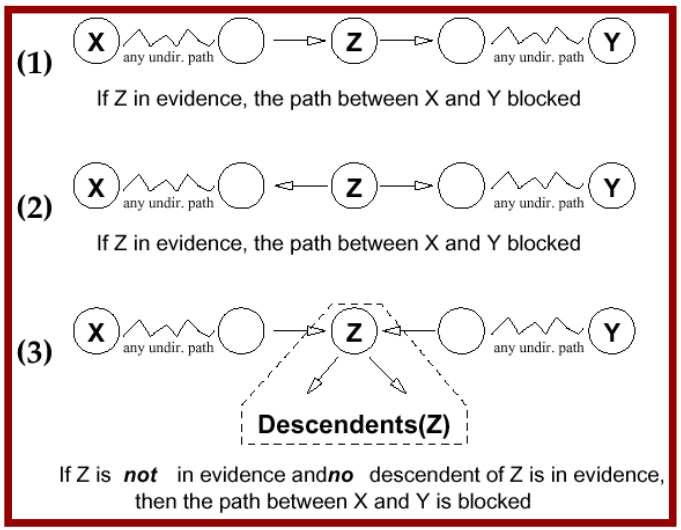




D-Separation: Intuitions

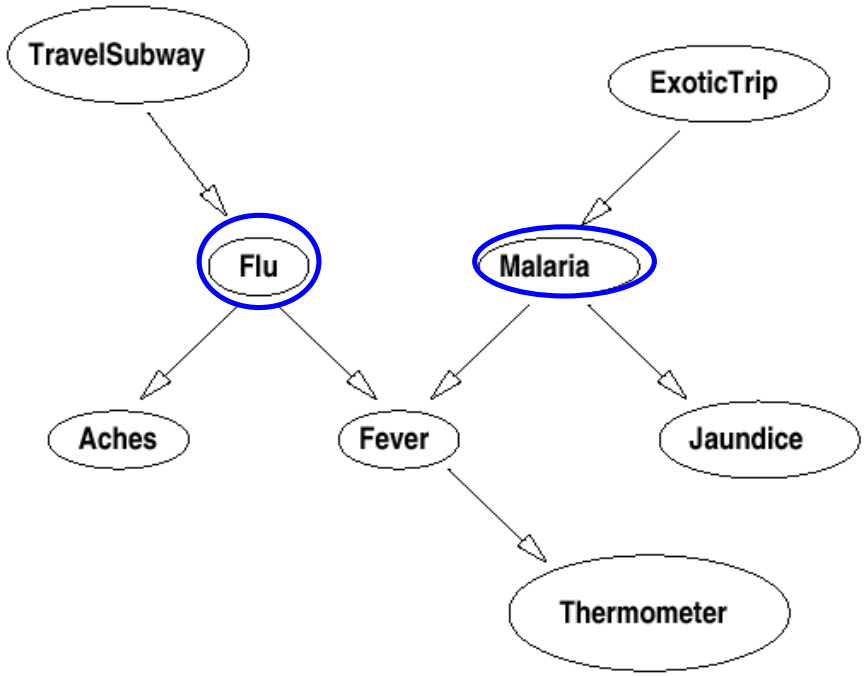
- Aches and Fever are **dependent**; but are **independent given Flu** (since Flu blocks the only path (via (2))).
- Similarly for Aches and Thermometer (dependent, but independent given Flu).

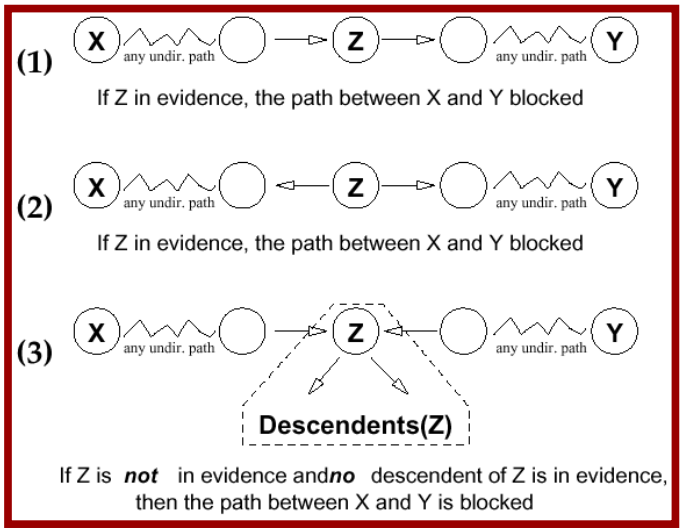




D-Separation: Intuitions

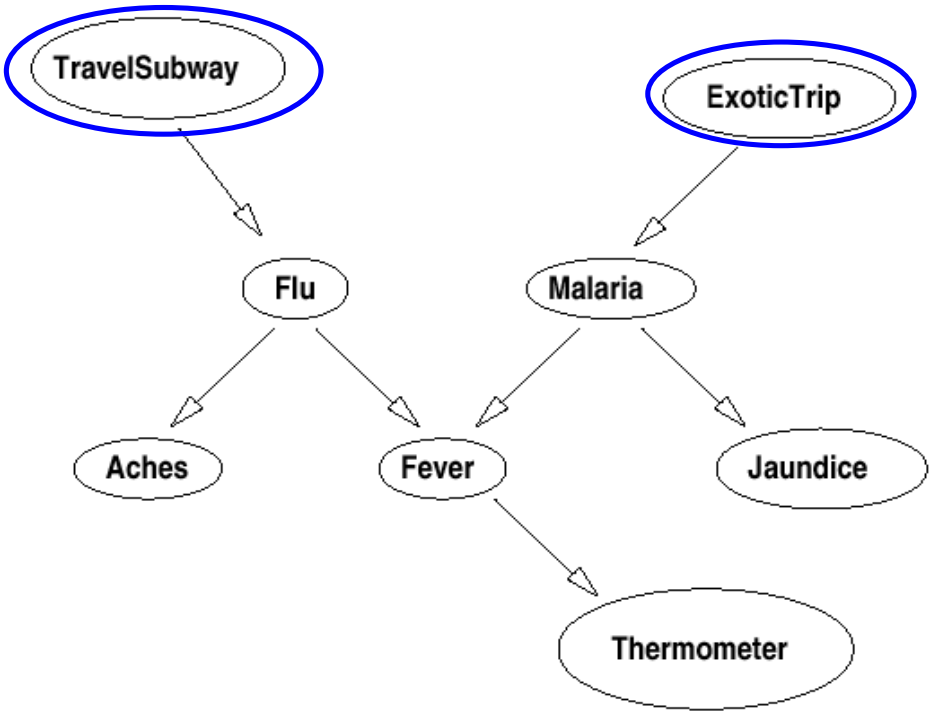
- Flu and Malaria are **independent (given no evidence):** Fever blocks the path, since it is *not in evidence*, nor is its descendant Thermometer (by (3))
- Flu and Malaria are **dependent** given Fever (or given Thermometer): nothing blocks path now. **What's the intuition?** Explaining Away: If you know John has Flu, it's explains the Fever, making Malaria less likely (and vice versa).

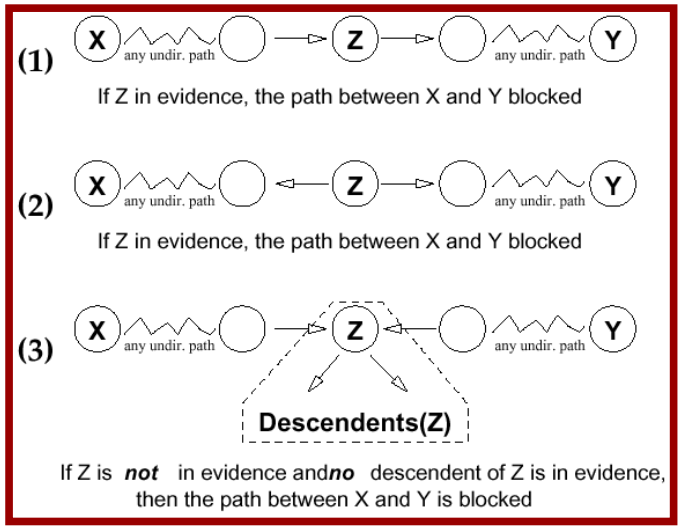




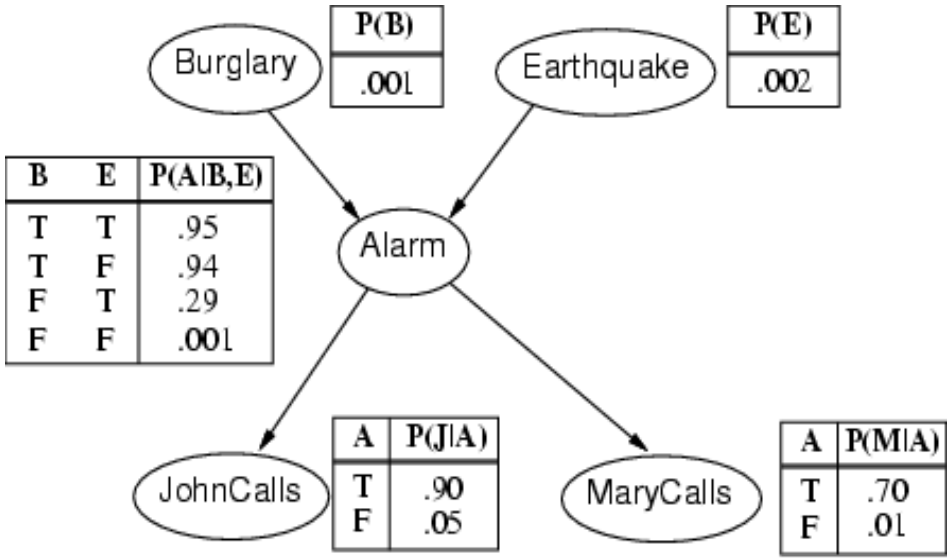
D-Separation: Intuitions

Subway and ExoticTrip are **independent** (by (3));
They are **dependent given Thermometer** ((3) is now violated by Thermometer);
They are **independent given Thermometer and Malaria**. This for exactly the same reasons for Flu/Malaria above.





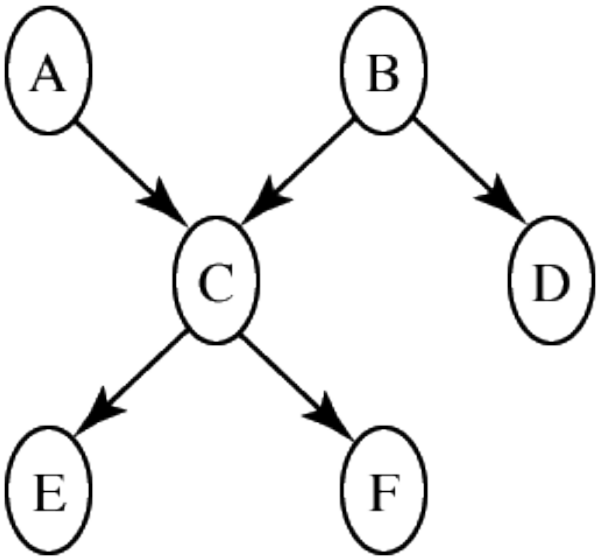
Burglary example



- A and M are dependent given J
- B and M are independent, given A
- J and M are dependent, but independent given A
- B and E are independent
- B and E are dependent, given A, J, or M

Exercise

Compute $P(c|a, b, \neg d, \neg e, \neg f)$. Note that it $\neq P(c|a, b)$



$$
\begin{array}{rclrcl}
P(a) & = & 0.9 & P(d|b) & = & 0.1 \\
P(b) & = & 0.2 & P(d|\neg b) & = & 0.8 \\
P(c|a, b) & = & 0.1 & P(e|c) & = & 0.7 \\
P(c|a, \neg b) & = & 0.8 & P(e|\neg c) & = & 0.2 \\
P(c|\neg a, b) & = & 0.7 & P(f|c) & = & 0.2 \\
P(c|\neg a, \neg b) & = & 0.4 & P(f|\neg c) & = & 0.9
\end{array}
$$

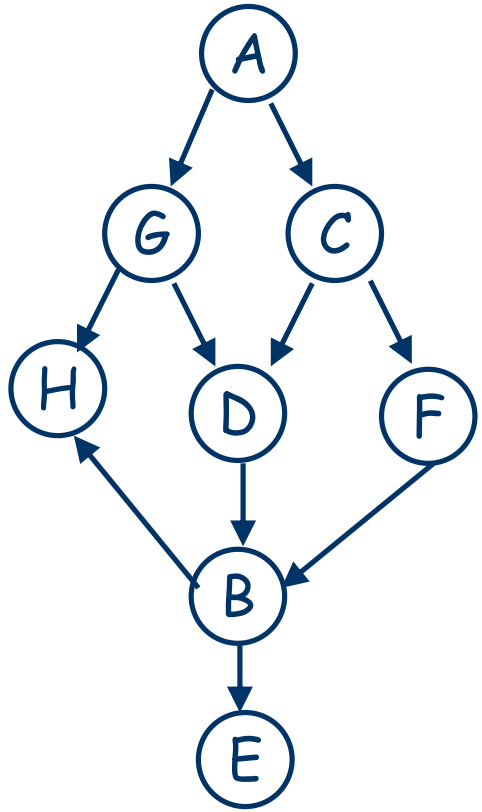
Given A and B, C and E are dependent

D-Separation Example

In the following network,

determine if **A and E are independent** given the evidence:

1. A and E given no evidence? **No**
2. A and E given {C}? **No**
3. A and E given {G,C}? Y
4. A and E given {G,C,H}? Y
5. A and E given {G,F}? **No**
6. A and E given {F,D}? Y
7. A and E given {F,D,H}? **No**
8. A and E given {B}? Y
9. A and E given {H,B}? Y
10. A and E given {G,C,D,H,D,F,B}? Y



Why the answer to 7 is No?

Note: A set of variables E d-separates X and Y if it blocks every undirected path in the BN between X and Y.

The path between A and E via H is not blocked

Inference in Bayes Nets

Given

1) a **Bayes net**

$$\mathbf{Pr(X_1, X_2, \ldots, X_n)}$$
$$\mathbf{= Pr(X_n \mid Par(X_n)) * Pr(X_{n-1} \mid Par(X_{n-1}) * \cdots * Pr(X_1 \mid Par(X_1))}$$

2) some **Evidence**, E

$$\mathbf{E = \{a\ set\ of\ values\ for\ some\ of\ the\ variables\}}$$

We want to

- **compute the new probability <u>distribution</u>**

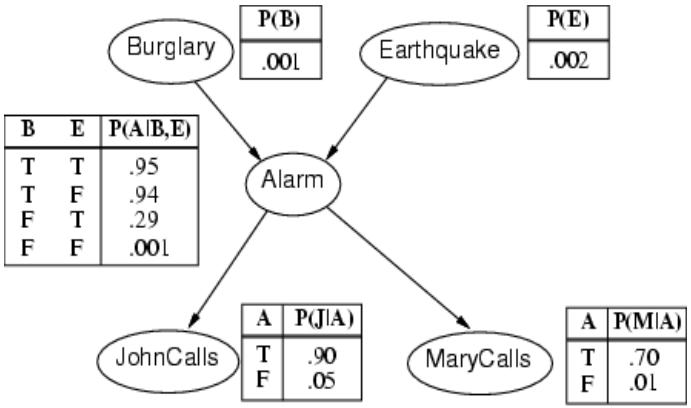
$$\mathbf{Pr(X_k \mid E)}$$

That is, we want to figure out

$$\mathbf{Pr(X_k = d \mid E)\ for\ all\ d \in Dom[X_k]}$$

Burglary example

In the Alarm example*** we have (the compact network):



Recall Burglary(B), Earthquake(E), Alarm(A), MaryCalled (M), JohnCalled(J)

And from our Bayes Net above, we determined:

Pr(B,E,A,M,J) = Pr(E) * Pr(B) * Pr(A|E,B) * Pr(M|A) * P(J|A)

We might want to compute things like:

Pr(B=True| M=true, J=false, E=false)

The probability that there was a burglary, given that Mary called, John didn't call, and there was no earthquake

Other examples

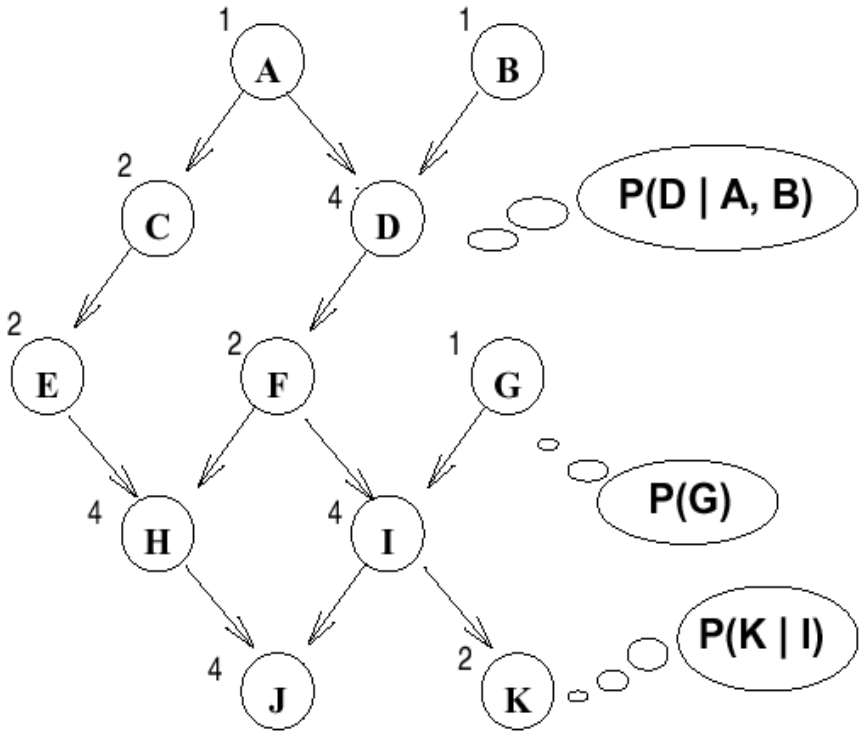
- computing probability of different diseases given symptoms,
- computing probability of hail storms given different metrological evidence

In such cases getting a good estimate of the probability of the unknown event allows us to respond more effectively (gamble rationally)

Example (Binary valued Variables)

Pr(A,B,C,D,E,F,G,H,I,J,K) =

Pr(A)
x Pr(B)
x Pr(C|A)
x Pr(D|A,B)
x Pr(E|C)
x Pr(F|D)
x Pr(G)
x Pr(H|E,F)
x Pr(I|F,G)
x Pr(J|H,I)
x Pr(K|I)



Example (Binary valued Variables)

Given the **Bayes Net** from the previous page:

Pr(A,B,C,D,E,F,G,H,I,J,K) =
Pr(A)*Pr(B)*Pr(C|A)*Pr(D|A,B)*Pr(E|C)*Pr(F|D)*Pr(G)*
Pr(H|E,F)*Pr(I|F,G)*Pr(J|H,I)*Pr(K|I)

And given the following **evidence**

E = {H=true, I=false}, i.e., E= {h,-i} ** NOTATION: h: H is true, -i: I is false

Let's say we **want to know**

Pr(D|h,-i)

---

1) Write as a **sum for each value of D** (i.e., d and –d)

$\sum_{A,B,C,E,F,G,J,K}$ Pr(A,B,C,d,E,F,h,-i,J,K) = Pr(d,h,-i)
$\sum_{A,B,C,E,F,G,J,K}$ Pr(A,B,C,-d,E,F,h,-i,J,K) = Pr(-d,h,-i)

Variable Elimination

2) Now compute $Pr(h, -i)$
$Pr(h, -i) = Pr(d, h, -i) + Pr(-d, h, -i)$

3) Finally, compute $Pr(D|h, -i)$
$Pr(d|h, -i) = Pr(d, h, -i)/Pr(h, -i)$
$Pr(-d|h, -i) = Pr(-d, h, -i)/Pr(h, -i)$

So to compute $Pr(D|h, -i)$, we only need to compute $Pr(d, h, -i)$ and $Pr(-d, h, -i)$, and then normalize to obtain the conditional probabilities we want.

Variable Elimination

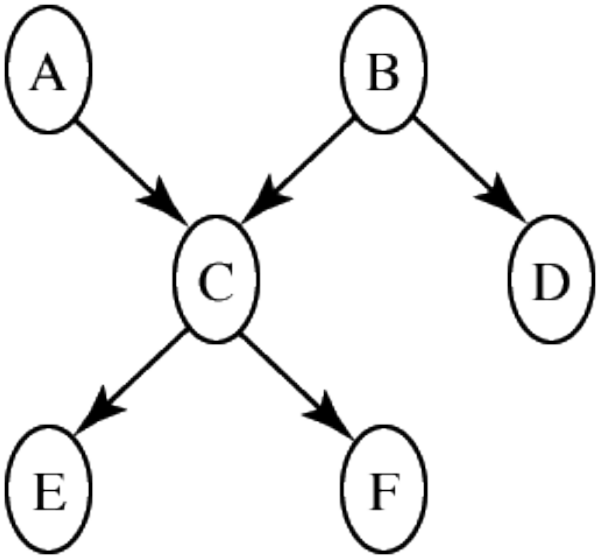
Variable elimination uses

- the product decomposition, and
- the summing out rule

to compute posterior probabilities from the information (CPTs) already in the network.

Exercise

Compute $P(c|a, b, \neg d, \neg e, \neg f)$. Is it $= P(c|a, b)$?



$$
\begin{array}{rclrcl}
P(a) & = & 0.9 & P(d|b) & = & 0.1 \\
P(b) & = & 0.2 & P(d|\neg b) & = & 0.8 \\
P(c|a, b) & = & 0.1 & P(e|c) & = & 0.7 \\
P(c|a, \neg b) & = & 0.8 & P(e|\neg c) & = & 0.2 \\
P(c|\neg a, b) & = & 0.7 & P(f|c) & = & 0.2 \\
P(c|\neg a, \neg b) & = & 0.4 & P(f|\neg c) & = & 0.9
\end{array}
$$

Variable Independence

Two variables X and Y are conditionally independent given variable Z if for all $x \in \text{Dom}(X)$, $y \in \text{Dom}(Y)$, $z \in \text{Dom}(Z)$, $X = x$ and $Y = y$ are conditionally independent given $Z = z$, *i.e.*, $Pr(X = x \wedge Y = y | Z = z) =$

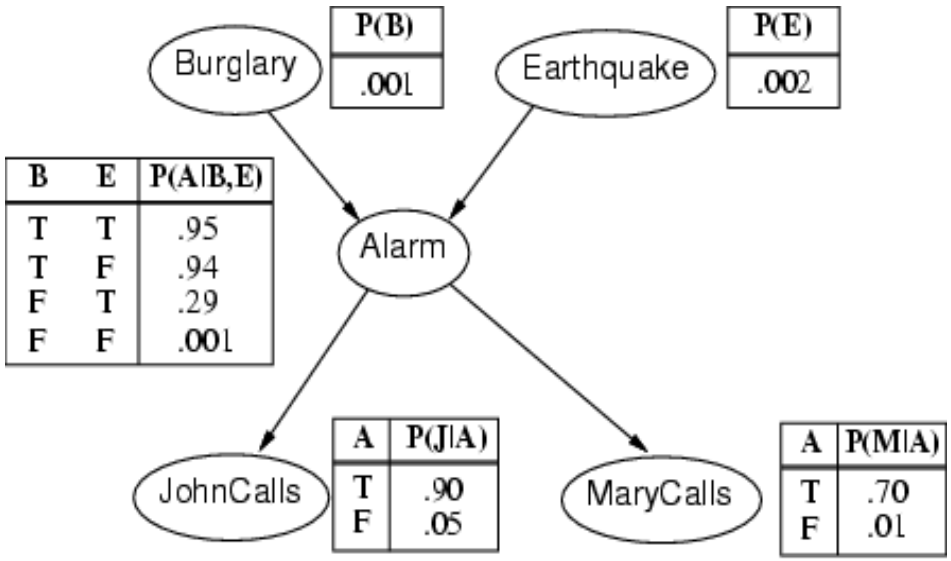
$$Pr(X = x | Z = z) \cdot Pr(Y = y | Z = z)$$

Bayesian Networks

A BN over variables $\{X_1, X_2, \ldots, X_n\}$ consists of:

- a DAG (directed acyclic graph) whose nodes are the variables
- a set of CPTs (conditional probability tables) $Pr(X_i|Par(X_i))$ for each X_i

Inference: Given some evidence E, compute $Pr(X_i|E)$



An example

Let's look a little closer at **how we compute the two sums** in (1)

Consider

$$\Pr(d,h,-i) = \sum_{A,B,C,E,F,G,J,K} \Pr(A,B,C,d,E,F,h,-i,J,K)$$

Use Bayes Net **product decomposition** to rewrite summation:

$$\sum_{A,B,C,E,F,G,J,K} \Pr(A,B,C,d,E,F,h,-i,J,K)$$

$$= \sum_{A,B,C,E,F,G,J,K} \Pr(A)\Pr(B)\Pr(C|A)\Pr(d|A,B)\Pr(E|C)$$
$$\Pr(F|d)\Pr(G)\Pr(h|E,F)\Pr(-i|F,G)\Pr(J|h,-i)$$
$$\Pr(K|-i)$$

An example

1) Move product terms out so they are scoped appropriately

$$= \sum_{A,}\sum_{B,}\sum_{C,}\sum_{E,}\sum_{F,}\sum_{G,}\sum_{J,}\sum_{K} \Pr(A)\Pr(B)\Pr(C|A)\Pr(d|A,B)\Pr(E|C)$$
$$\Pr(F|d)\Pr(G)\Pr(h|E,F)\Pr(\text{-}i|F,G)\Pr(J|h,\text{-}i)$$
$$\Pr(K|\text{-}i)$$

$$= \sum_A \Pr(A) \sum_B \Pr(B) \sum_C \Pr(C|A) \Pr(d|A,B) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \sum_G \Pr(G)\Pr(h|E,F)\Pr(\text{-}i|F,G) \sum_J \Pr(J|h,\text{-}i)$$
$$\sum_K \Pr(K|\text{-}i)$$

$$= \sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(\text{-}i|F,G) \sum_J \Pr(J|h,\text{-}i)$$
$$\sum_K \Pr(K|\text{-}i)$$

An example

Now start computing.

$$\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$$
$$\sum_J \Pr(J|h,-i)$$
$$\sum_K \Pr(K|-i)$$

2) Compute the things that are numbers/constants

$$\sum_K \Pr(K|-i) = \Pr(k|-i) + \Pr(-k|-i) = c_1$$

$$\sum_J \Pr(J|h,-i)\ c_1 = c_1 \sum_J \Pr(J|h,-i)$$
$$= c_1\ (\Pr(j|h,-i) + \Pr(-j|h,-i))$$
$$= c_1 c_2$$

An example

$$\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B)$$
$$\sum_C \Pr(C|A) \sum_E \Pr(E|C) \sum_F \Pr(F|d) \Pr(h|E,F)$$
$$\sum_G \Pr(G) \Pr(-i|F,G) \sum_J \Pr(J|h,-i) \sum_K \Pr(K|-i)$$

$$c_1 c_2 \sum_G \Pr(G) \Pr(-i|F,G)$$
$$= c_1 c_2 (\Pr(g)\Pr(-i|F,g) + \Pr(-g)\Pr(-i|F,-g))$$

!!But $\Pr(-i|F,g)$ depends on the value of F, so this is not a single number.

An example

So, Let's try eliminate in outside->inside order:

$$\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(\text{-}i|F,G)$$
$$\sum_J \Pr(J|h,\text{-}i)$$
$$\sum_K \Pr(K|\text{-}i)$$

=

$$\Pr(a) \sum_B \Pr(B) \Pr(d|a,B) \sum_C \Pr(C|a) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(\text{-}i|F,G)$$
$$\sum_J \Pr(J|h,\text{-}i)$$
$$\sum_K \Pr(K|\text{-}i)$$

+

$$\Pr(\text{-}a) \sum_B \Pr(B) \Pr(d|\text{-}a,B) \sum_C \Pr(C|\text{-}a) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(\text{-}i|F,G)$$
$$\sum_J \Pr(J|h,\text{-}i)$$
$$\sum_K \Pr(K|\text{-}i)$$

An example

=

Pr(a)Pr(b) $\Pr(d|a,b) \sum_C \Pr(C|a) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

+

Pr(a)Pr(-b) $\Pr(d|a,-b) \sum_C \Pr(C|a) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

+

Pr(-a)Pr(b) $\Pr(d|-a,b) \sum_C \Pr(C|-a) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

+

Pr(-a)Pr(-b) $\Pr(d|-a,-b) \sum_C \Pr(C|-a) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

Problem: The size of the sum is doubling as we expand each variable (into -v and v). This approach has exponential complexity. But let's look a bit closer.

An example

$$
\begin{aligned}
&= \\
&\Pr(a)\Pr(b)\ \Pr(d|a,b)\ \textstyle\sum_C \Pr(C|a) \sum_E \Pr(E|C) \\
&\qquad \textstyle\sum_F \Pr(F|d)\ \Pr(h|E,F) \sum_G \Pr(G)\ \Pr(\text{-}i|F,G) \\
&\qquad \textstyle\sum_J \Pr(J|h,\text{-}i) \\
&\qquad \textstyle\sum_K \Pr(K|\text{-}i) \\
&+ \\
&\Pr(a)\Pr(\text{-}b)\ \Pr(d|a,\text{-}b)\ \textstyle\sum_C \Pr(C|a) \sum_E \Pr(E|C) \\
&\qquad \textstyle\sum_F \Pr(F|d)\ \Pr(h|E,F) \sum_G \Pr(G)\ \Pr(\text{-}i|F,G) \\
&\qquad \textstyle\sum_J \Pr(J|h,\text{-}i) \\
&\qquad \textstyle\sum_K \Pr(K|\text{-}i) \\
&+ \\
&\Pr(\text{-}a)\Pr(b)\ \Pr(d|\text{-}a,b)\ \textstyle\sum_C \Pr(C|\text{-}a) \sum_E \Pr(E|C) \\
&\qquad \textstyle\sum_F \Pr(F|d)\ \Pr(h|E,F) \sum_G \Pr(G)\ \Pr(\text{-}i|F,G) \\
&\qquad \textstyle\sum_J \Pr(J|h,\text{-}i) \\
&\qquad \textstyle\sum_K \Pr(K|\text{-}i) \\
&+ \\
&\Pr(\text{-}a)\Pr(\text{-}b)\ \Pr(d|\text{-}a,\text{-}b)\ \textstyle\sum_C \Pr(C|\text{-}a) \sum_E \Pr(E|C) \\
&\qquad \textstyle\sum_F \Pr(F|d)\ \Pr(h|E,F) \sum_G \Pr(G)\ \Pr(\text{-}i|F,G) \\
&\qquad \textstyle\sum_J \Pr(J|h,\text{-}i) \\
&\qquad \textstyle\sum_K \Pr(K|\text{-}i)
\end{aligned}
$$

Repeated subterm

Repeated subterm

Solution: If we store the value of the subterms, we need only compute them once.

Dynamic Programming

=

$\Pr(a)\Pr(b)\ \Pr(d|a,b) \sum_C \Pr(C|a) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d)\ \Pr(h|E,F) \sum_G \Pr(G)\ \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

+

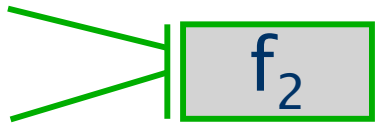
$\Pr(a)\Pr(-b)\ \Pr(d|a,-b) \sum_C \Pr(C|a) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d)\ \Pr(h|E,F) \sum_G \Pr(G)\ \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

+

$\Pr(-a)\Pr(b)\ \Pr(d|-a,b) \sum_C \Pr(C|-a) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d)\ \Pr(h|E,F) \sum_G \Pr(G)\ \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

+

$\Pr(-a)\Pr(-b)\ \Pr(d|-a,-b) \sum_C \Pr(C|-a) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d)\ \Pr(h|E,F) \sum_G \Pr(G)\ \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$



$$= c_1 f_1 + c_2 f_1 + c_3 f_2 + c_4 f_2$$

$$c_1 = \Pr(a)\Pr(b)\ \Pr(d|a,b)$$

$$c_2 = \Pr(a)\Pr(-b)\ \Pr(d|a,-b)$$

$$c_3 = \Pr(-a)\Pr(b)\ \Pr(d|-a,b)$$

$$c_4 = \Pr(-a)\Pr(-b)\ \Pr(d|-a,-b)$$

Dynamic Programming

$$f_1 = \sum_C \Pr(C|a) \sum_E \Pr(E|C) \sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G) \sum_J \Pr(J|h,-i) \sum_K \Pr(K|-i)$$

$$= \Pr(c|a) \sum_E \Pr(E|c) \sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G) \sum_J \Pr(J|h,-i) \sum_K \Pr(K|-i)$$

$$+ \Pr(-c|a) \sum_E \Pr(E|-c) \sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G) \sum_J \Pr(J|h,-i) \sum_K \Pr(K|-i)$$

Repeated subterm

Dynamic Programming

- So within the computation of the subterms we obtain more repeated smaller subterms.
- he core idea of dynamic programming is to remember all "smaller" computations, so that they can be reused.
- This can convert an exponential computation into one that takes only polynomial time.
- Variable elimination is a dynamic programming technique that computes the sum from the bottom up (starting with the smaller subterms and working its way up to the bigger terms).

Relevant

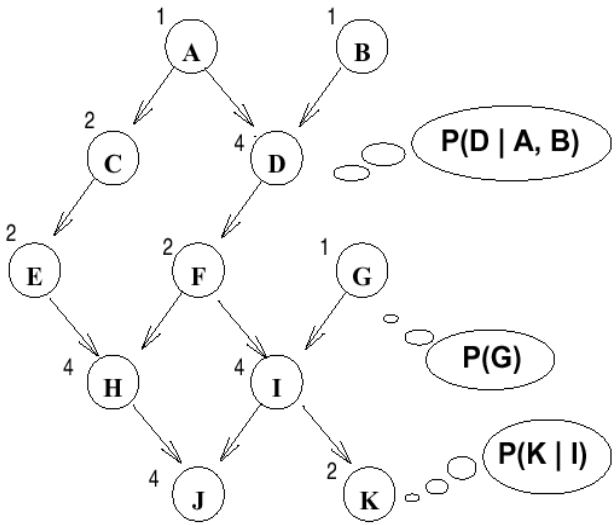
A brief aside .... note that in the sum

$\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

we have that $\sum_K \Pr(K|-i) = 1$ (Why?), thus

$\sum_J \Pr(J|h,-i) \sum_K \Pr(K|-i) = \sum_J \Pr(J|h,-i)$

Furthermore $\sum_J \Pr(J|h,-i) = 1$.



So we could drop these last two terms from the computation:
J and K are irrelevant given our query D and evidence -i and -h.
For now we keep these terms.

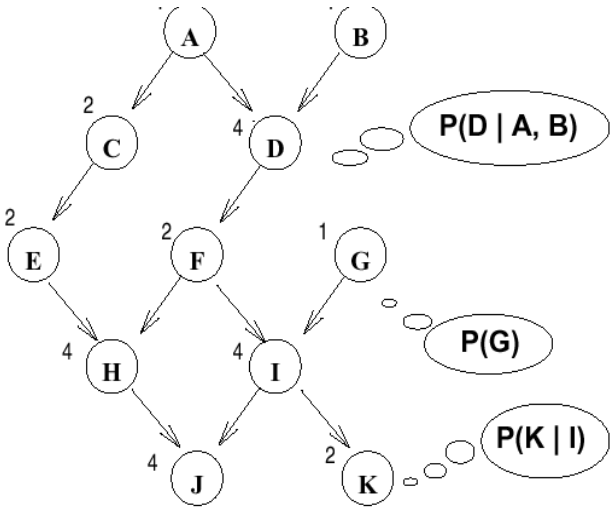
Variable Elimination (VE)

VE works from the inside out, summing out K, then J, G, ...

- Recall, when we tried to sum out G

$$\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$$
$$\sum_J \Pr(J|h,-i)$$
$$\sum_K \Pr(K|-i)$$

$$c_1 c_2 \sum_G \Pr(G) \Pr(-i|F,G)$$
$$= c_1 c_2 (\Pr(g)\Pr(-i|F,g) + \Pr(-g)\Pr(-i|F,-g))$$



we found that Pr(-i|F,-g) depends on the value of F, it wasn't a single number.

- However, we can still continue with the computation by computing two different numbers, one for each value of F

Variable Elimination (VE)

$\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$
$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$
$\sum_J \Pr(J|h,-i)$
$\sum_K \Pr(K|-i)$

- $t(-f) = c_1 c_2 \sum_G \Pr(G) \Pr(-i|-f,G)$
  $t(f) = c_1 c_2 (\sum_G \Pr(G) \Pr(-i|f,G)$
  - $t(-f) = c_1 c_2 (\Pr(g)\Pr(-i|-f,g) + \Pr(-g)\Pr(-i|-f,-g))$
  - $t(f) = c_1 c_2 (\Pr(g)\Pr(-i|f,g) + \Pr(-g)\Pr(-i|f,-g))$
- Now we sum out F

Variable Elimination (VE)

- $\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$
  $\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$
  $\sum_J \Pr(J|h,-i)$
  $\sum_K \Pr(K|-i)$

$c_1 c_2 \sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$

$= c_1 c_2 (\Pr(f|d) \Pr(h|E,f)(\sum_G \Pr(G) \Pr(-i|f,G)) + \Pr(-f|d) \Pr(h|E,-f)(\sum_G \Pr(G) \Pr(-i|-f,G)))$

$= c_1 c_2 \sum_F \Pr(F|d) \Pr(h|E,F) t(F)$

$t(f)$, $t(-f)$

Variable Elimination (VE)

- $c_1 c_2$ (Pr(f|d) Pr(h|E,f) t(f)
  + Pr(-f|d)Pr(h|E,-f) t(-f)

This is a function of E, so we obtain <u>two new numbers</u>

s(e) = $c_1 c_2$ (Pr(f|d) Pr(h|e,f)t(f)
+ Pr(-f|d)Pr(h|e,-f) t(-f)

s(-e) = $c_1 c_2$(Pr(f|d) Pr(h|-e,f)t(f)
+ Pr(-f|d)Pr(h|-e,-f) t(-f)

Variable Elimination (VE)

$$\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(\text{-}i|F,G)$$
$$\sum_J \Pr(J|h,\text{-}i)$$
$$\sum_K \Pr(K|\text{-}i)$$

- On summing out E we obtain two numbers, or a function of C.
- Then a function of B, then a function of A.
- On finally summing out A we obtain the single number we wanted to compute which is Pr(d,h,-i).
- Now we can repeat the process to compute Pr(-d,h,-i).
- Instead, we can regard D as a variable in the computation.
- This way, summing out A will yield a function of D.

Variable Elimination (VE)

- In general, at each stage VE will compute a table of numbers: one for each different instantiation of the variables in the sum.
- The size of these tables is exponential in the number of variables appearing in the sum, *e.g.*

$$\sum_F Pr(F|D)Pr(h|E,F)t(F)$$

depends on the value of D and E, thus we will obtain $|Dom[D]||Dom[E]|$ different numbers in the resulting table.

Factors

- We call these tables of values computed by VE factors.
- Note that the original CPTs are also table of values.
  Thus we also all them factors
- Each factor is a function of some variables, *e.g.*, $P(C|A) = f(A, C)$: it maps each value of its arguments to a number.
- A tabular representation is exponential in the number of variables in the factor.
- Notation: $f(\mathbf{X}, \mathbf{Y})$ denotes a factor over the variables $\mathbf{X} \cup \mathbf{Y}$ (where $\mathbf{X}$ and $\mathbf{Y}$ are sets of variables)
- If we examine the inside-out summation process we see that various operations occur on factors.

The Product of Two Factors

- Let f(**X**,**Y**) & g(**Y**,**Z**) be two factors with variables **Y** in common
- The ***product*** of f and g, denoted h = f × g (or sometimes just h = fg), is defined:

$$h(\mathbf{X},\mathbf{Y},\mathbf{Z}) = f(\mathbf{X},\mathbf{Y}) \times g(\mathbf{Y},\mathbf{Z})$$

f(A,B)		g(B,C)		h(A,B,C)			
ab	0.9	bc	0.7	abc	0.63	ab~c	0.27
a~b	0.1	b~c	0.3	a~bc	0.08	a~b~c	0.02
~ab	0.4	~bc	0.8	~abc	0.28	~ab~c	0.12
~a~b	0.6	~b~c	0.2	~a~bc	0.48	~a~b~c	0.12

Summing a Variable Out of a Factor

- Let f(X,**Y**) be a factor with variable X (**Y** is a set)
- We ***sum out*** variable X from f to produce a new factor h = Σ_X f, which is defined:

$$h(\mathbf{Y}) = \Sigma_{x \in Dom(X)} f(x, \mathbf{Y})$$

f(A,B)		h(B)	
ab	0.9	b	1.3
a~b	0.1	~b	0.7
~ab	0.4		
~a~b	0.6		

No error in the table. Here $f(A, B)$ is not $P(AB)$, but $P(B|A)$.

Restricting a Factor

- Let f(X,**Y**) be a factor with variable X (**Y** is a set)
- We ***restrict*** factor f *to* X=*a* by setting X to the value *a* and "deleting" incompatible elements of f's domain . Define $h = f_{X=a}$ as: h(**Y**) = f(a,**Y**)

f(A,B)		$h(B) = f_{A=a}$	
ab	0.9	b	0.9
a~b	0.1	~b	0.1
~ab	0.4		
~a~b	0.6		

The VE Algorithm

Given a Bayes Net with CPTs F, query variable Q, evidence variables **E** (observed to have values e), and remaining variables **Z**. Compute Pr(Q|**E**)

1. Replace each factor $f \in F$ that mentions a variable(s) in **E** with its restriction $f_{\mathbf{E}=e}$ (this might yield a "constant" factor)
2. For each Z_j– in the order given –eliminate $Z_j \in \mathbf{Z}$ as follows:
   1. Let $f_1, f_2, \ldots, f_k$ be the factors in F that include Z_j
   2. Compute new factor $g_j = \sum_{Z_j} f_1 \times f_2 \times \ldots \times f_k$
   3. Remove the factors f_i from F and add new factor g_j to F
3. The remaining factors refer only to the query variable Q. Take their product and normalize to produce Pr(Q|**E**).

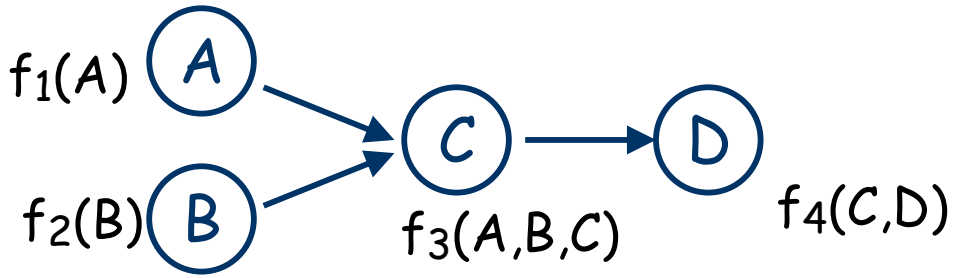
VE: Example

Factors: $f_1(A)$ $f_2(B)$ $f_3(A,B,C)$ $f_4(C,D)$

Query: P(A)?

Evidence: D = d

Elim. Order: C, B



Restriction: replace $f_4(C,D)$ with $f_5(C) = f_4(C,d)$

Step 1: Eliminating C: Compute & Add $f_6(A,B) = \Sigma_C f_5(C) f_3(A,B,C)$

Remove: $f_3(A,B,C)$, $f_5(C)$

Step 2: Eliminating B: Compute & Add $f_7(A) = \Sigma_B f_6(A,B) f_2(B)$

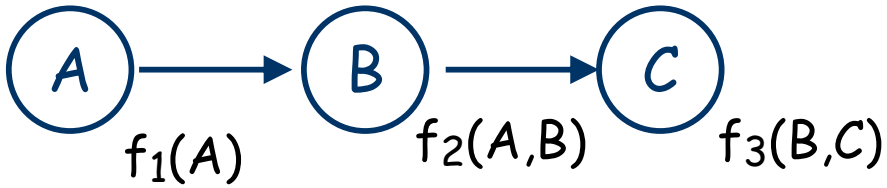
Remove: $f_6(A,B)$, $f_2(B)$

Last factors: $f_7(A)$, $f_1(A)$. The product $f_1(A) \times f_7(A)$ is (unnormalized) posterior. So… $P(A|d) = \alpha f_1(A) \times f_7(A)$ where $\alpha = 1/\sum_A f_1(A)f_7(A)$ ← **Note the Normalization Constant!**

Numeric example

Here's an example with some numbers

Eliminate A then B



0.85=0.9*0.9 + 0.1*0.4
0.15=0.9*0.1 + 0.1*0.6

Eliminate A Eliminate B

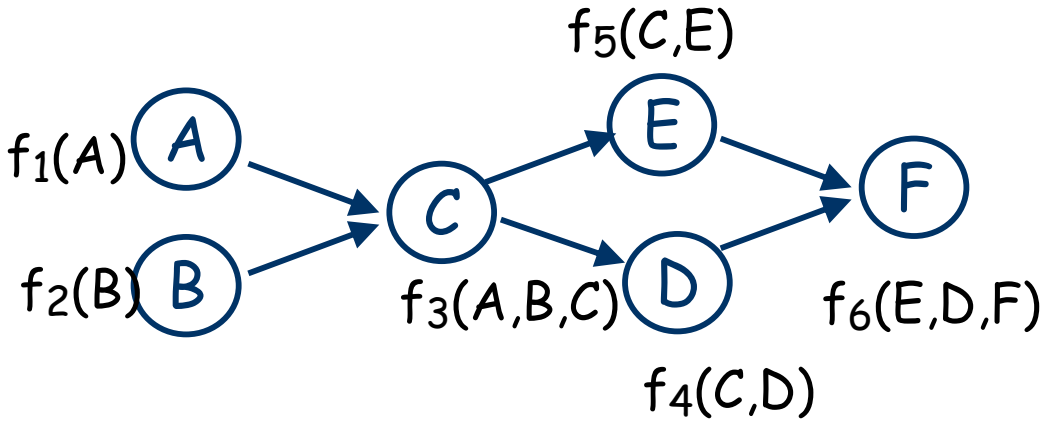
$f_1(A)$		$f_2(A,B)$		$f_3(B,C)$		$f_4(B)$ $\Sigma_A f_2(A,B)f_1(A)$		$f_5(C)$ $\Sigma_B f_3(B,C) f_4(B)$	
a	**0.9**	a**b**	**0.9**	bc	0.7	**b**	**0.85**	c	0.625
~a	**0.1**	a**~b**	**0.1**	b~c	0.3	**~b**	**0.15**	~c	0.375
		~a**b**	**0.4**	~bc	0.2				
		~a**~b**	**0.6**	~b~c	0.8				

VE as Buckets Elimintation

The bucket elimination framework is a unifying algorithmic framework that generalizes dynamic programming to accommodate algorithms for many complex problem-solving and reasoning activities $\cdots$including the variable elimination algorithm for probabilistic inference

We will use buckets as a notational device to do Variable Elimination

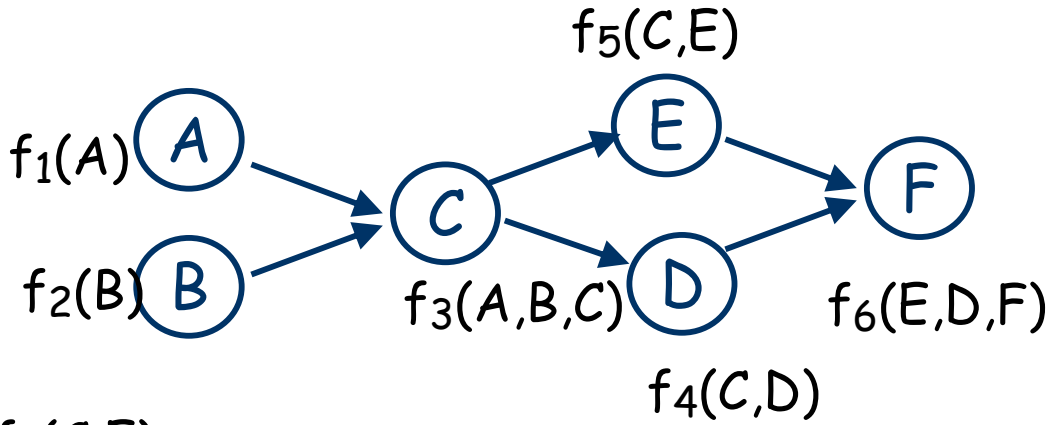
VE Ordering:
C,F,A,B,E,D



1. C:
2. F:
3. A:
4. B:
5. E:
6. D:

STEP 1: Place Original Factors in first applicable bucket.

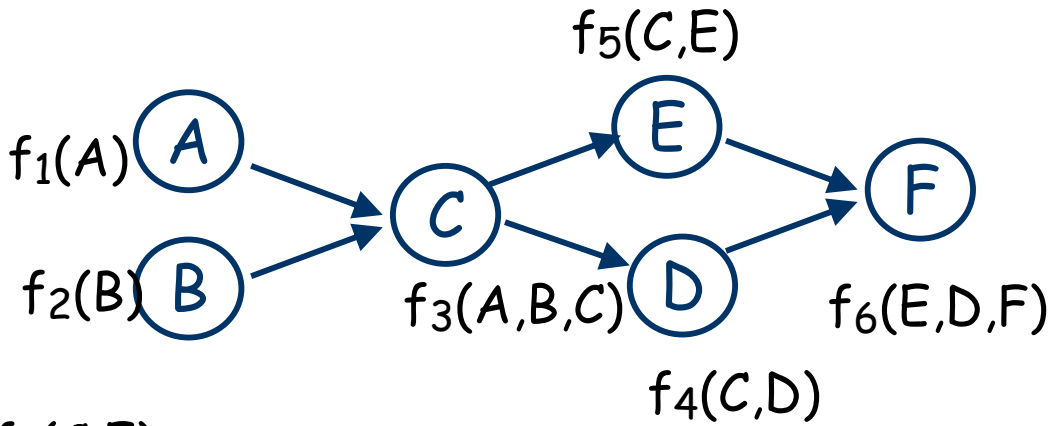
VE Ordering:
C,F,A,B,E,D



1. C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$
2. F: $f_6(E,D,F)$
3. A: $f_1(A)$
4. B: $f_2(B)$
5. E:
6. D:

STEP 2: Eliminate variables in order, placing new factor in 1st applicable bucket

VE Ordering:
C,F,A,B,E,D



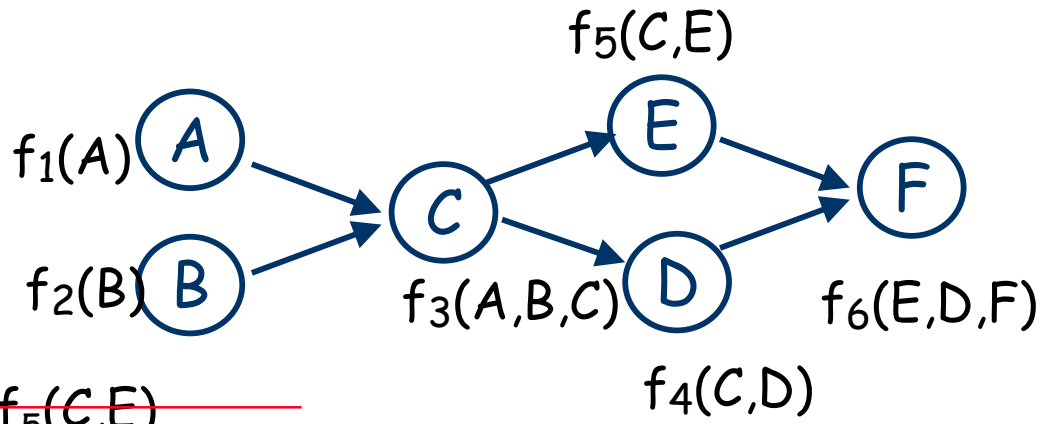
1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. F: $f_6(E,D,F)$
3. A: $f_1(A)$, $f_7(A,B,D,E)$
4. B: $f_2(B)$
5. E:
6. D:

1. Eliminating C:

$$\Sigma_C f_3(A,B,C), f_4(C,D), f_5(C,E) = f_7(A,B,D,E)$$

Eliminate F, placing new factor f8 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D



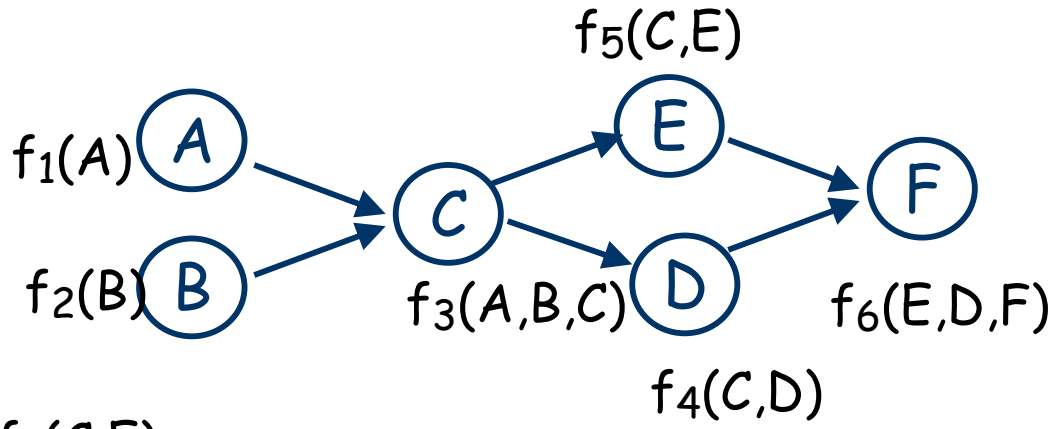
1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. ~~F: $f_6(E,D,F)$~~
3. A: $f_1(A)$, $f_7(A,B,D,E)$
4. B: $f_2(B)$
5. E: $f_8(E,D)$
6. D:

2. Eliminating F:

$\Sigma_F f_6(E,D,F) = f_8(E,D)$

Eliminate A, placing new factor f9 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D



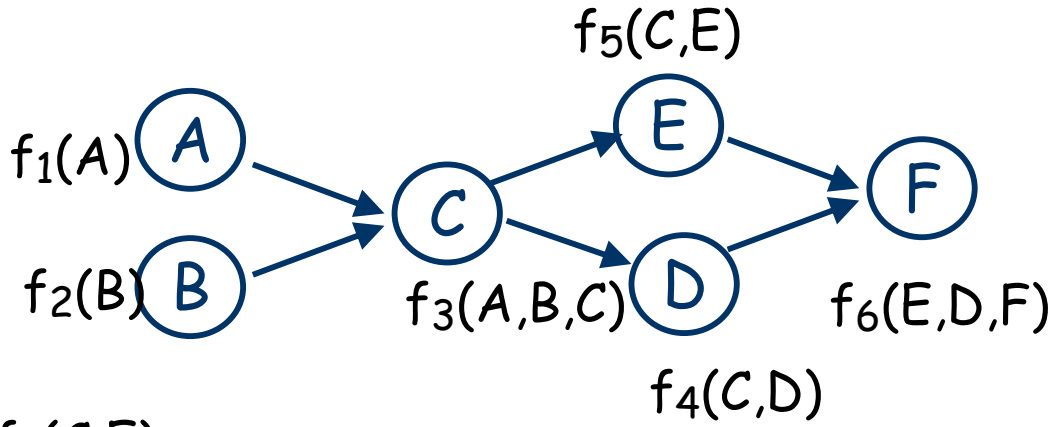
1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. ~~F: $f_6(E,D,F)$~~
3. ~~A: $f_1(A)$, $f_7(A,B,D,E)$~~
4. B: $f_2(B)$, $f_9(B,D,E)$
5. E: $f_8(E,D)$
6. D:

3. Eliminating A:

$\Sigma_A f_1(A), f_7(A,B,D,E) = f_9(B,D,E)$

Eliminate B, placing new factor f10 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D



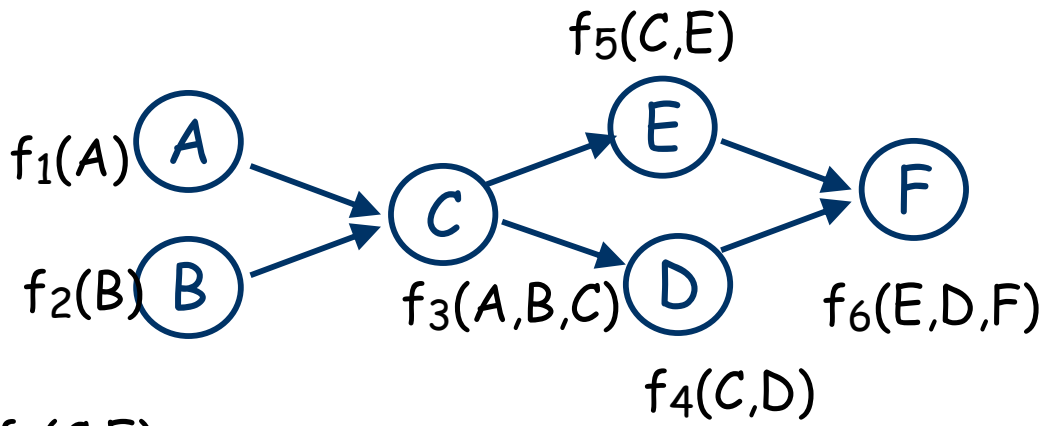
1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. ~~F: $f_6(E,D,F)$~~
3. ~~A: $f_1(A)$, $f_7(A,B,D,E)$~~
4. ~~B: $f_2(B)$, $f_9(B,D,E)$~~
5. E: $f_8(E,D)$, $f_{10}(D,E)$
6. D:

4. Eliminating B:

$\Sigma_B f_2(B), f_9(B,D,E)$
$= f_{10}(D,E)$

Eliminate E, placing new factor f11 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D



1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. ~~F: $f_6(E,D,F)$~~
3. ~~A: $f_1(A)$, $f_7(A,B,D,E)$~~
4. ~~B: $f_2(B)$, $f_9(B,D,E)$~~
5. ~~E: $f_8(E,D)$, $f_{10}(D,E)$~~
6. D: $f_{11}(D)$

5. Eliminating E:

$\Sigma_E f_8(E,D), f_{10}(D,E)$
$= f_{11}(D)$

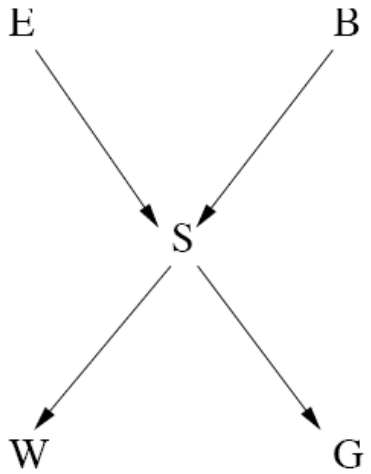
f_{11} is the final answer, once we normalize it.

Exercise

E – Earthquake, B – Burglary

S – Sound of alarm heard

W – Dr. Watson Calls G – Mrs Gibbons Calls



P(E)	e	-e
	1/10	9/10

P(B)	b	-b
	1/10	9/10

P(S|E,B)	s	-s
e ∧ b	9/10	1/10
e ∧ -b	2/10	8/10
-e ∧ b	8/10	2/10
-e ∧ -b	0	1

P(W|S)	w	-w
s	8/10	2/10
-s	2/10	8/10

P(G|S)	g	-g
s	1/2	1/2
-s	0	1

- What is $P(g|s)$?
- What is $P(s|g)$?
- What is $P(g|s, w)$?

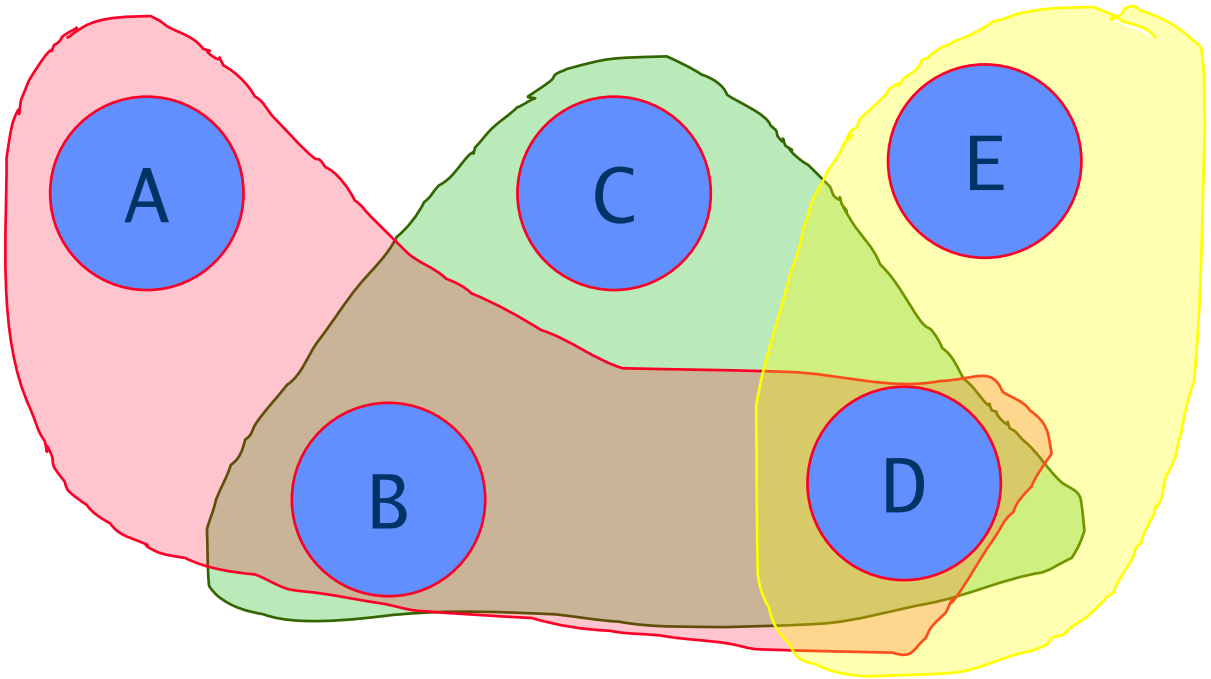
- What is $P(G|W)$? (i.e., the four probability values $P(g|w)$, $P(-g|w)$, $P(g|-w)$, and $P(-g|-w)$).
- Query variable is G.
- First run of VE, evidence is $W = w$.
- Second run of VE, evidence is $W = -w$.
- Use same ordering for both runs of VE: E, B, S, G.

Complexity of VE

- Variable elimination (VE) starts with the set of CPTs (tables) from the original Bayes Net (BN).
- As it eliminates variables it produces new intermediate tables (factors).
- The complexity of VE is determined by the size of the largest such intermediate table.
- We would like to find a variable elimination ordering that minimizes this.
- A table can be viewed as a hyperedge in a hypergraph.

Hypergraphs

- Hypergraph has vertices just like an ordinary graph, but instead of edges between two vertices, it contains hyperedges.
- A hyperedge is a set of vertices (potentially more than one)

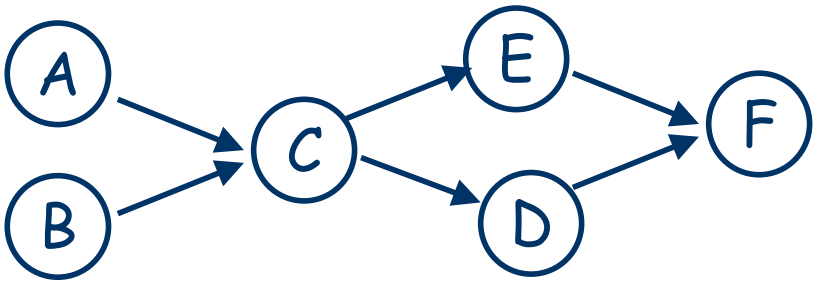


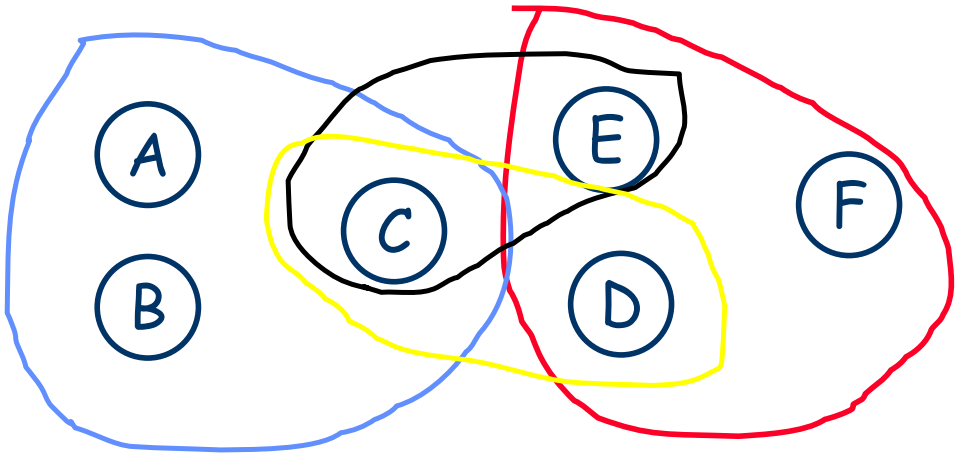
{A,B,D}
{B,C,D}
{E,D}

Hypergraph of Bayes Net

- The set of vertices are precisely the nodes of the Bayes net.
- The hyperedges are the variables appearing in each CPT, *i.e.*, $\{X_i\} \cup Par(X_i)$

Pr(A,B,C,D,E,F) =
Pr(A)Pr(B)
X Pr(C|A,B)
X Pr(E|C)
X Pr(D|C)
X Pr(F|E,D).

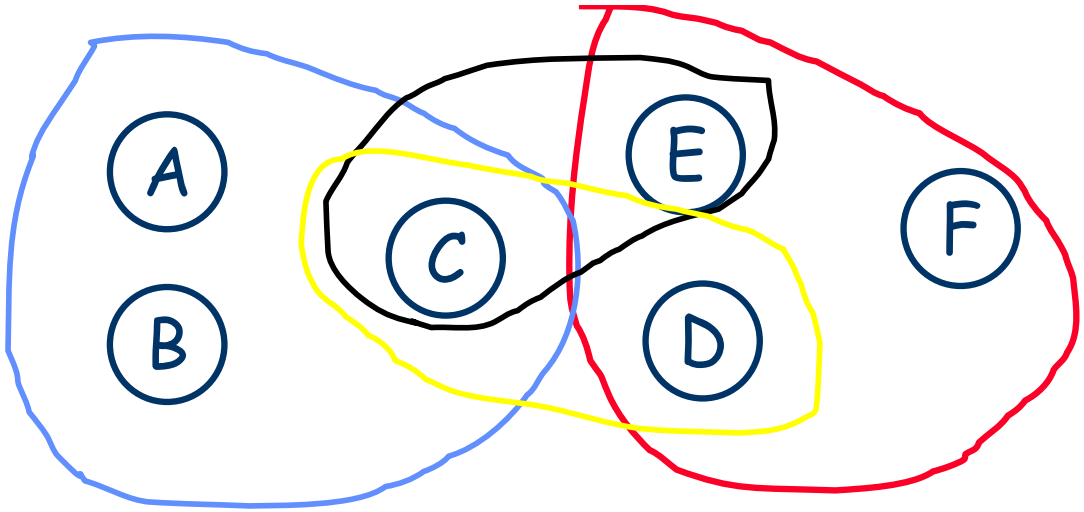




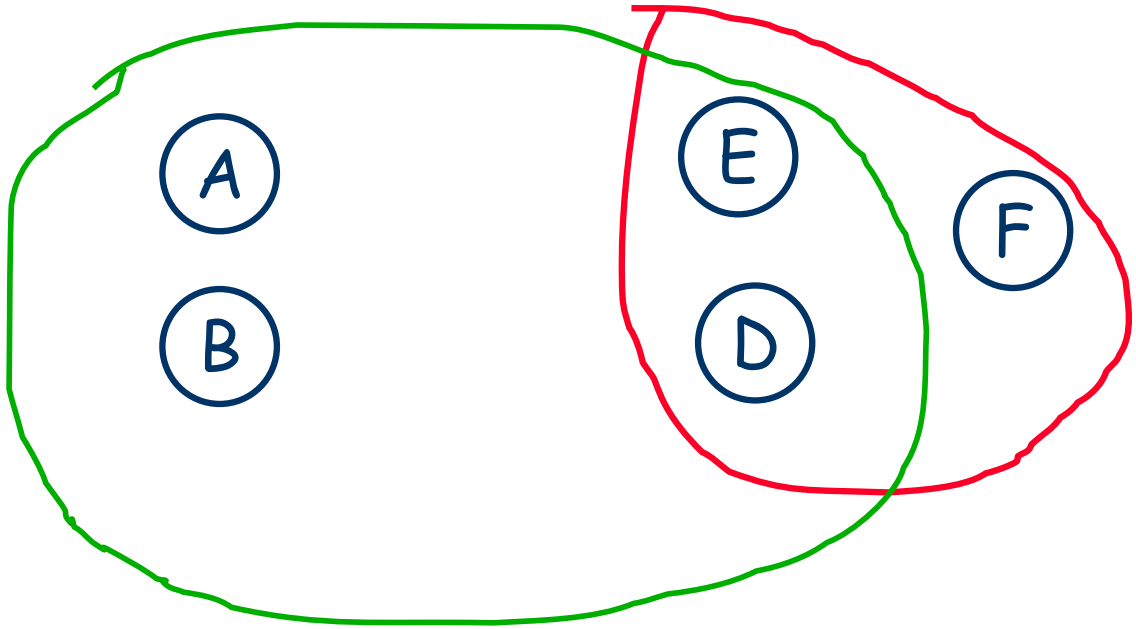
Variable Elimination in the HyperGraph

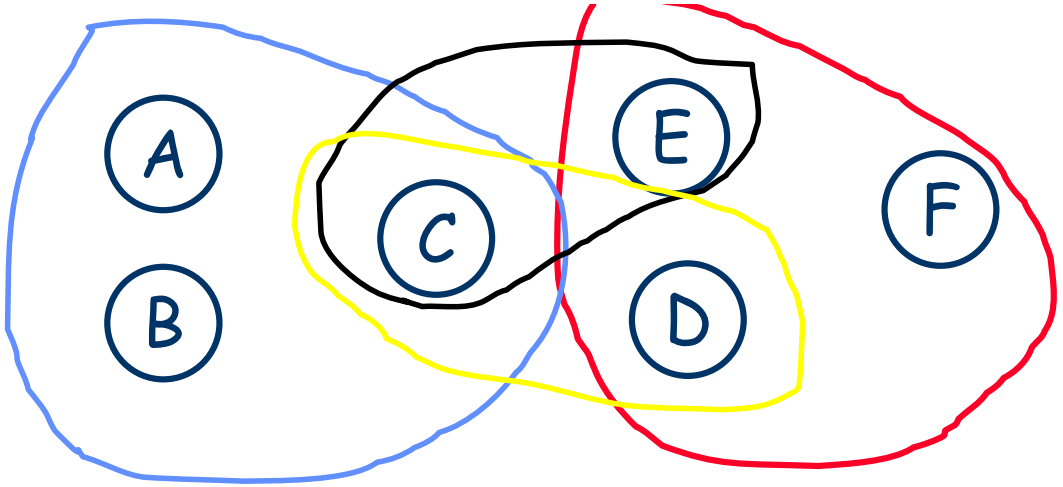
To eliminate variable X_i in the hypergraph we

- Remove the vertex X_i
- Create a new hyperedge H_i equal to the union of all of the hyperedges that contain X_i minus X_i
- Remove all hyperedges containing X_i from the hypergraph.
- Add the new hyperedge H_i to the hypergraph.

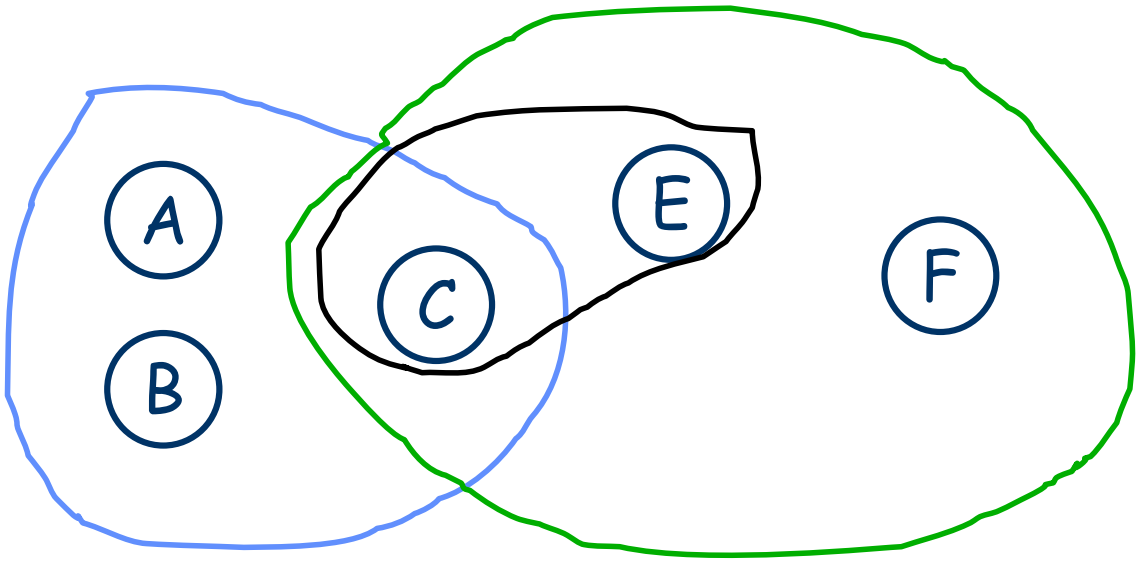


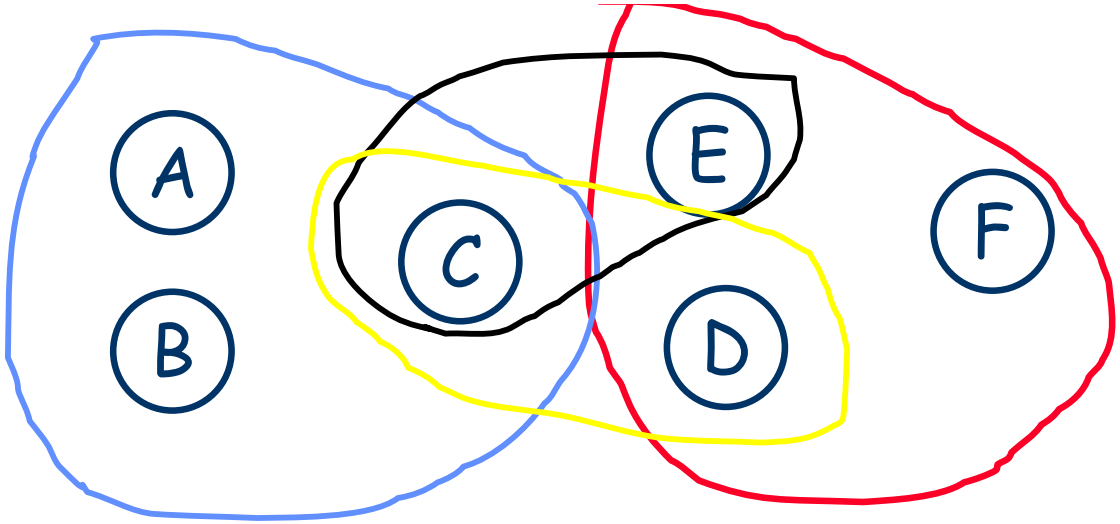
From above
Eliminate C



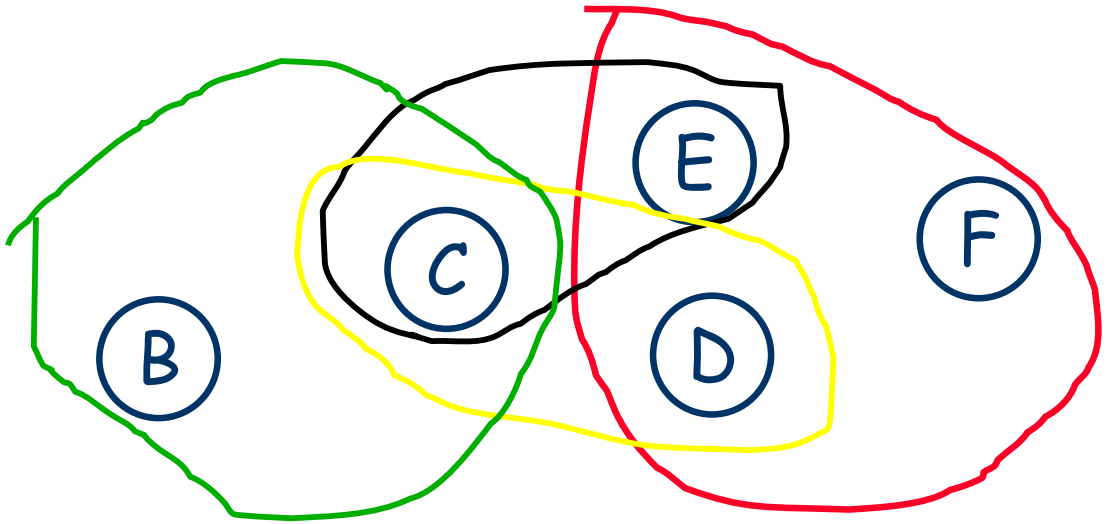


From above
Eliminate D



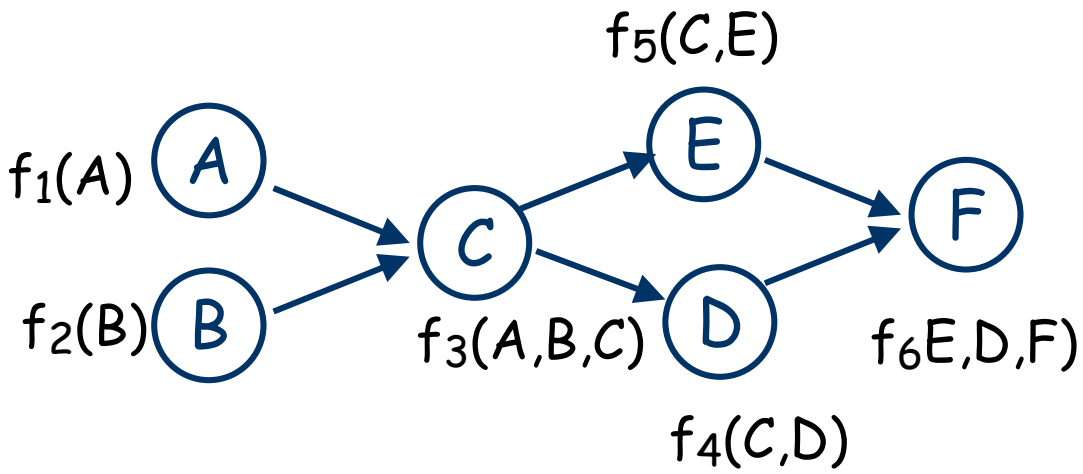


From above
Eliminate A

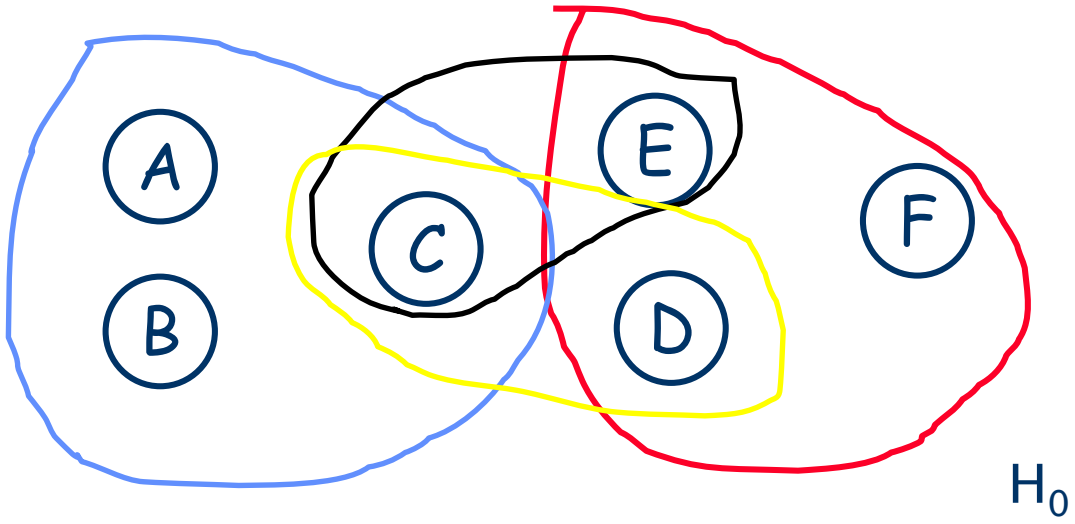


Variable Elimination – looking at the hypergraphs

VE Ordering:
C,F,A,B,E,D

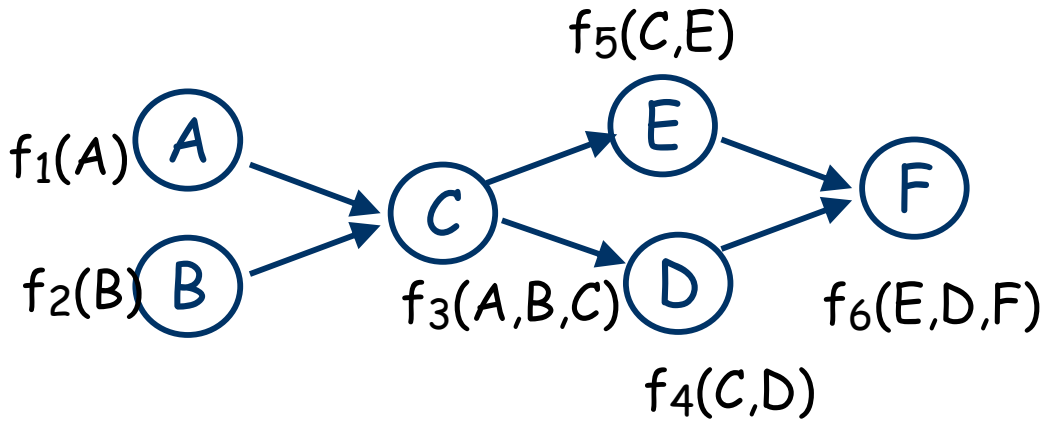


1. C:
2. F:
3. A:
4. B:
5. E:
6. D:



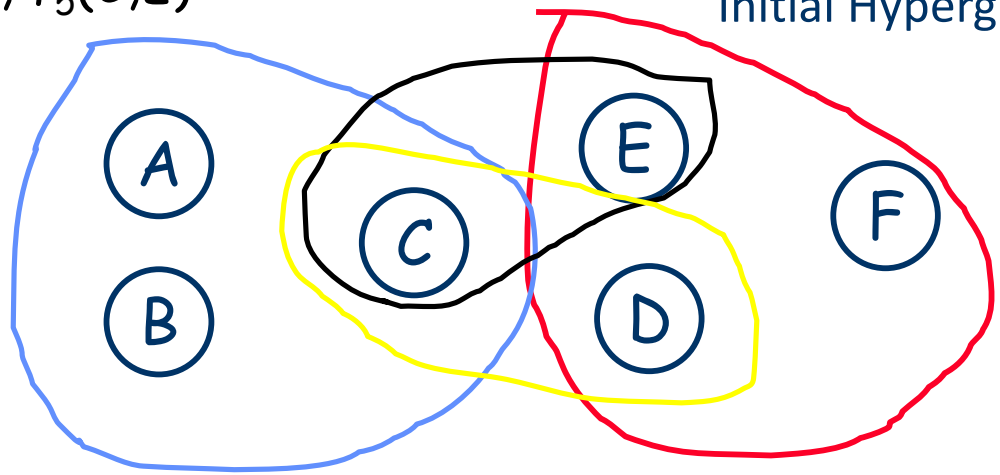
Place Original Factors in first applicable bucket.

VE Ordering:
C,F,A,B,E,D



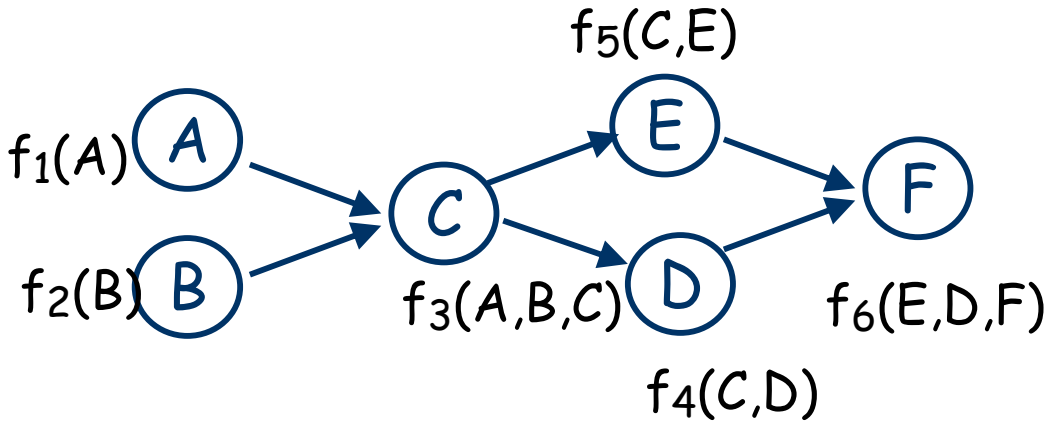
1. C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$
2. F: $f_6(E,D,F)$
3. A: $f_1(A)$
4. B: $f_2(B)$
5. E:
6. D:

Initial Hypergraph H_0

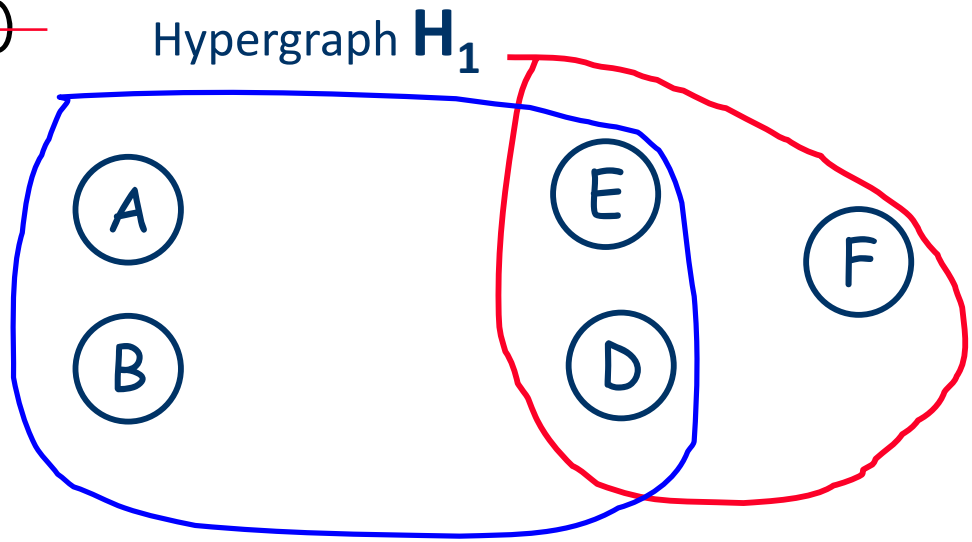


Eliminate C, placing new factor f7 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D

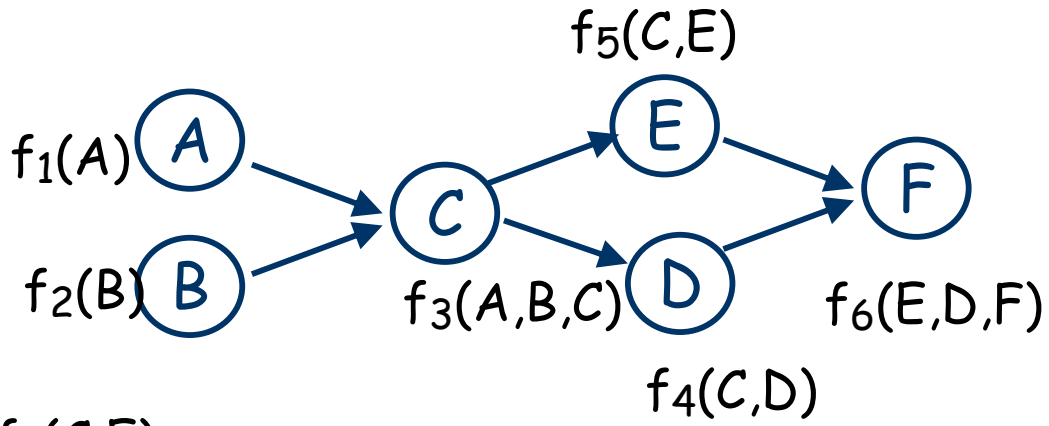


1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. F: $f_6(E,D,F)$
3. A: $f_1(A)$, $f_7(A,B,D,E)$
4. B: $f_2(B)$
5. E:
6. D:



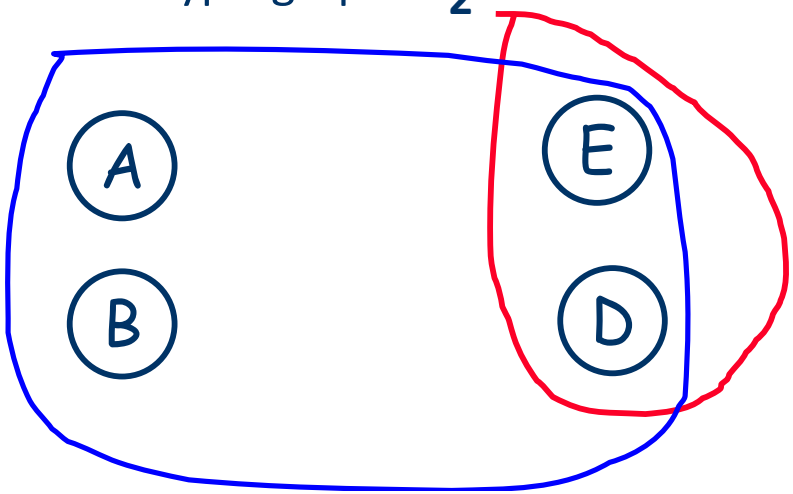
Eliminate F, placing new factor f8 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D



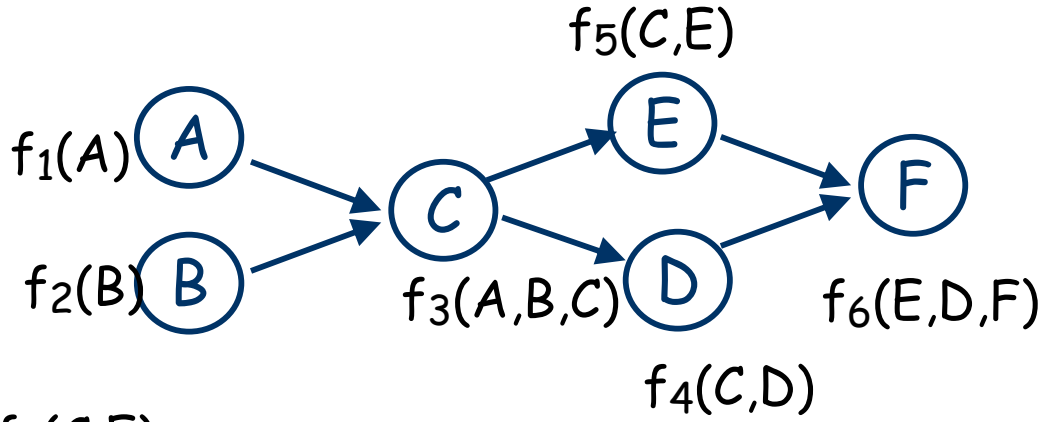
1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. ~~F: $f_6(E,D,F)$~~
3. A: $f_1(A)$, $f_7(A,B,D,E)$
4. B: $f_2(B)$
5. E: $f_8(E,D)$
6. D:

Hypergraph $\mathbf{H_2}$

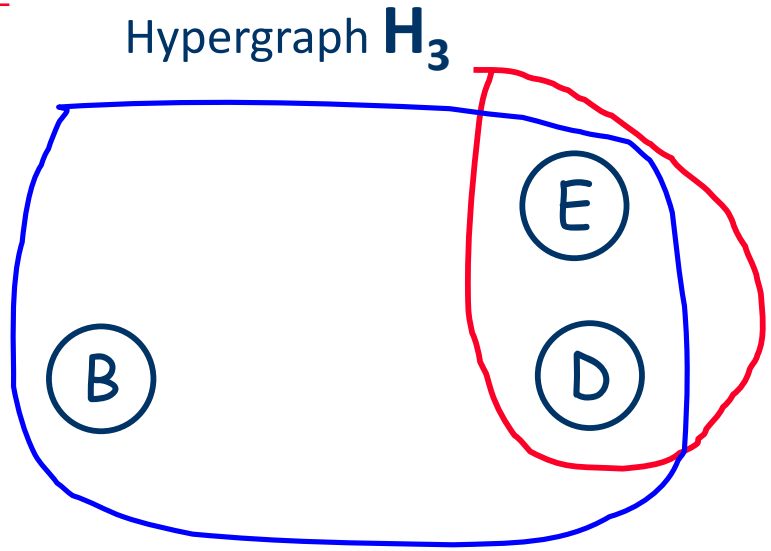


Eliminate A, placing new factor f9 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D

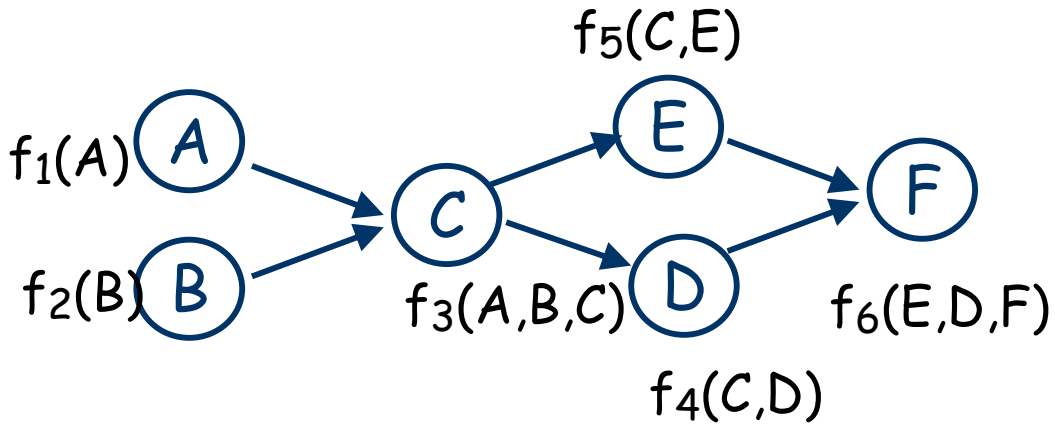


1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. ~~F: $f_6(E,D,F)$~~
3. ~~A: $f_1(A)$, $f_7(A,B,D,E)$~~
4. B: $f_2(B)$, $f_9(B,D,E)$
5. E: $f_8(E,D)$
6. D:



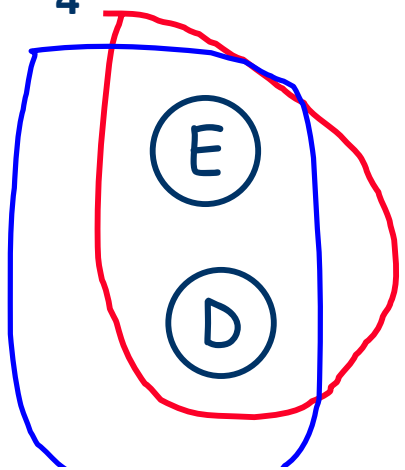
Eliminate B, placing new factor f10 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D



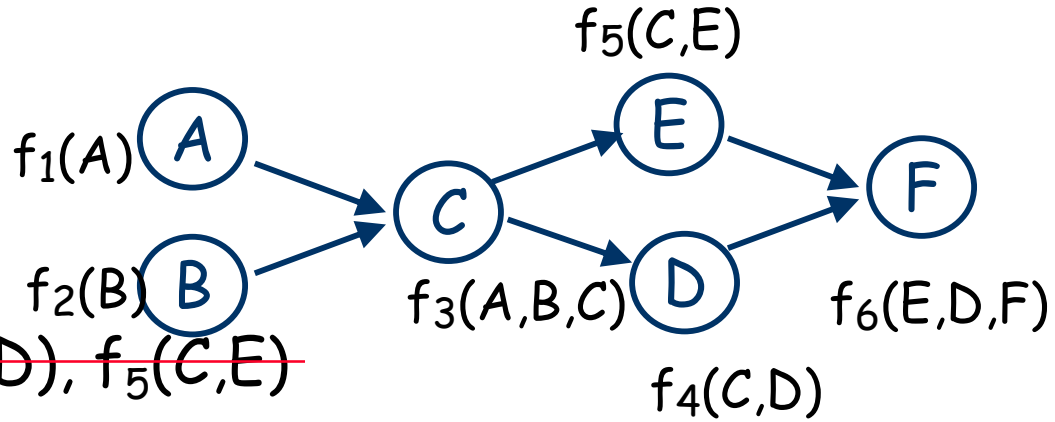
1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. ~~F: $f_6(E,D,F)$~~
3. ~~A: $f_1(A)$, $f_7(A,B,D,E)$~~
4. ~~B: $f_2(B)$, $f_9(B,D,E)$~~
5. E: $f_8(E,D)$, $f_{10}(D,E)$
6. D:

Hypergraph $\mathbf{H_4}$



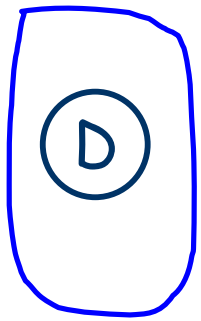
Eliminate E, placing new factor f11 in first applicable bucket.

VE Ordering:
C,F,A,B,E,D



1. ~~C: $f_3(A,B,C)$, $f_4(C,D)$, $f_5(C,E)$~~
2. ~~F: $f_6(E,D,F)$~~
3. ~~A: $f_1(A)$, $f_7(A,B,D,E)$~~
4. ~~B: $f_2(B)$, $f_9(B,D,E)$~~
5. ~~E: $f_8(E,D)$, $f_{10}(D,E)$~~
6. D: $f_{11}(D)$

Hypergraph $\mathbf{H_5}$



Elimination width

- Given an ordering of the variables and an initial hypergraph H. Eliminating these variables yields a sequence of hypergraphs $H_0, H_1, H_2, \ldots, H_n$,
  where H_n contains only one vertex (the query variable).
- The elimination width is the maximum size (number of variables) of any hyperedge in any of the hypergraphs $H_0, H_1, H_2, \ldots, H_n$.
- The elimination width of the previous example was 4 ($\{A, B, E, D\}$ in H1 and H2).

Tree width

- Given a hypergraph H with vertices $\{X_1, X_2, \ldots, X_n\}$, the tree width of H is $k-1$, where k is the minimum elimination width of any of the $n!$ different orderings of the variables
- Thus VE has best case complexity of $2^{O(w)}$, where w is the tree width of the initial Bayes Net.
- In the worst case, the tree width can be equal to the number of variables.

Tree width

Exponential in the tree width is the best that VE can do.

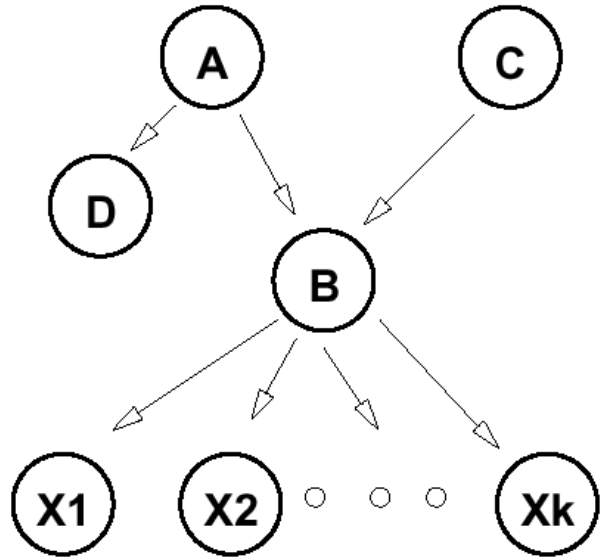
- Finding an ordering with elimination width equal to tree width is NP-Hard.
- So in practice there is no point in trying to speed up VE by finding the best possible elimination ordering.
- Heuristics are used to find orderings with good (low) elimination widths.
- In practice, this can be very successful. Elimination widths can often be relatively small, 8-10 even when the network has 1000s of variables.
- Thus VE can be much!! more efficient than simply summing the probability of all possible events (which is exponential in the number of variables).

Finding Good Orderings (sometimes)

- A polytree is a singly connected Bayes Net: in particular there is only one path between any two nodes.
- A node can have multiple parents, but there are no cycles.
- Good orderings are easy to find for polytrees
  - At each stage eliminate a singly connected node.
  - Because we have a polytree we are assured that a singly connected node will exist at each elimination stage.
  - The size of the factors in the tree never increase.

Elimination Ordering: Polytrees

Treewidth of a polytree = maximum number of parents among all nodes.



Eliminating singly connected nodes

allows VE to run in **time linear in size of network**

E.g., in this network,

- eliminate D, A, C, X1,…; or
- eliminate X1,… Xk, D, A, C; or
- mix up…

Result: no factor ever larger than original CPTs

BUT E.g.,

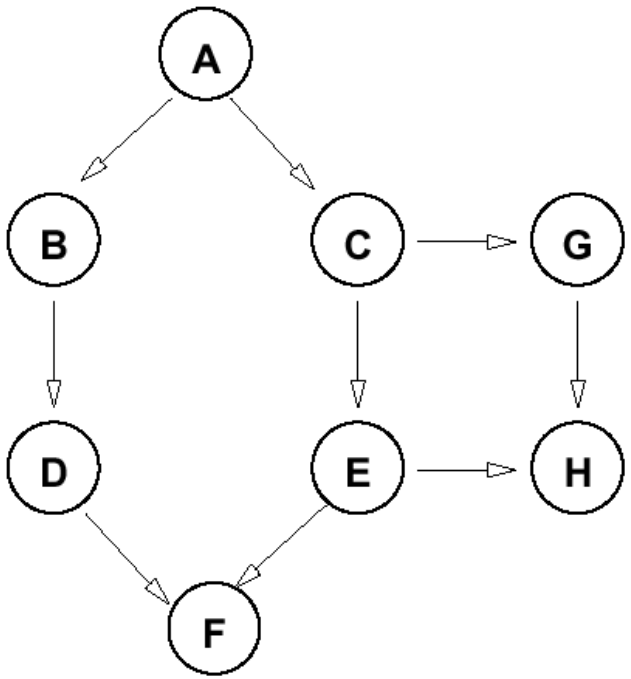
- eliminating B before these

Result: factors that include all of A,C, X1,… Xk !!!

Effect of Different Orderings

Given the following BN with **query variable** is **D**.

This BN is **not a polytree!**



Consider different **variable elimination orderings**

A,F,H,G,B,C,E:

- Good ordering

E,C,A,B,G,H,F:

- Bad ordering

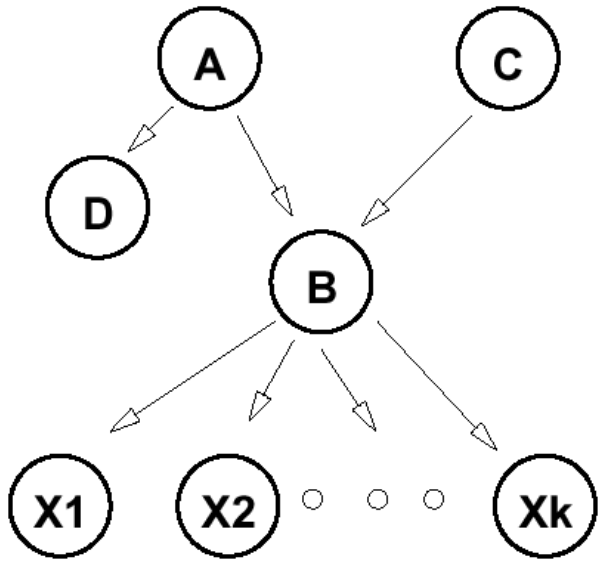
Elimination Order: Min Fill Heuristic

Min-fill Heuristic:

"always eliminate next the variable that creates the smallest size factor."

This is a reasonably effective heuristic for determining an elimination order for VE

- B creates a factor of size k+2
- A creates a factor of size 2
- D creates a factor of size 1



The heuristic **always solves polytrees in linear time**.

Relevant

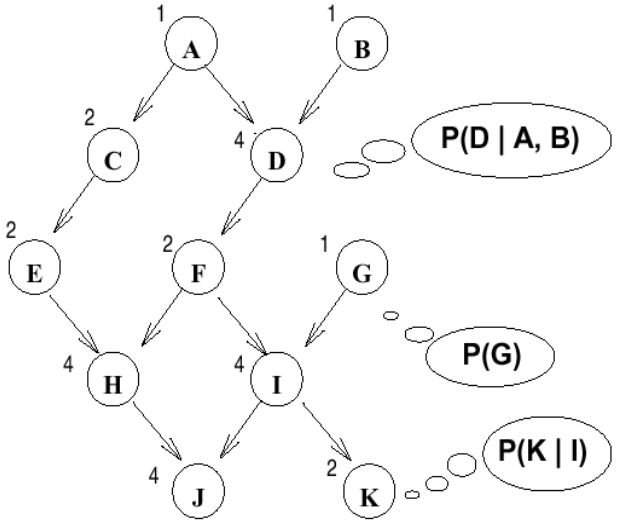
A brief aside .... note that in the sum

$$\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C)$$
$$\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G)$$
$$\sum_J \Pr(J|h,-i)$$
$$\sum_K \Pr(K|-i)$$

we have that $\sum_K \Pr(K|-i) = 1$ (Why?), thus

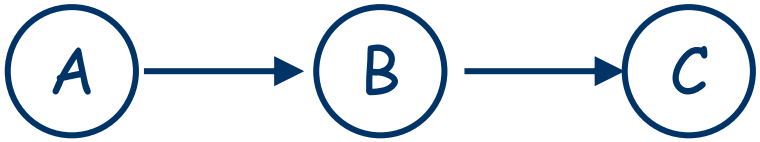
$$\sum_J \Pr(J|h,-i) \sum_K \Pr(K|-i) = \sum_J \Pr(J|h,-i)$$

Furthermore $\sum_J \Pr(J|h,-i) = 1$.



So we could drop these last two terms from the computation:
J and K are irrelevant given our query D and evidence -i and -h.

Relevance



Certain variables have no impact on the query.

In network ABC, computing Pr(A) with no evidence requires elimination of B and C.

- But when you sum out these vars, you compute a trivial factor (whose value are all ones); for example:
- eliminating C: $f_4(B) = \Sigma_C f_3(B,C) = \Sigma_C \Pr(C|B)$
- 1 for any value of B (e.g., $\Pr(c|b) + \Pr(\sim c|b) = 1$)

Observation: B or C *are irrelevant* to the query. No need to think about them.

Relevance

Observation: We can restrict attention to ***relevant*** variables.

When is a variable relevant to a query?

Given

- query **Q**,
- evidence **E**

The following variables are **relevant** to the evaluation of Q given E:

- Q itself is relevant
- if any node **Z** is relevant, its parents are relevant
- if e$\in$**E** is a descendent of a relevant node, then E is relevant

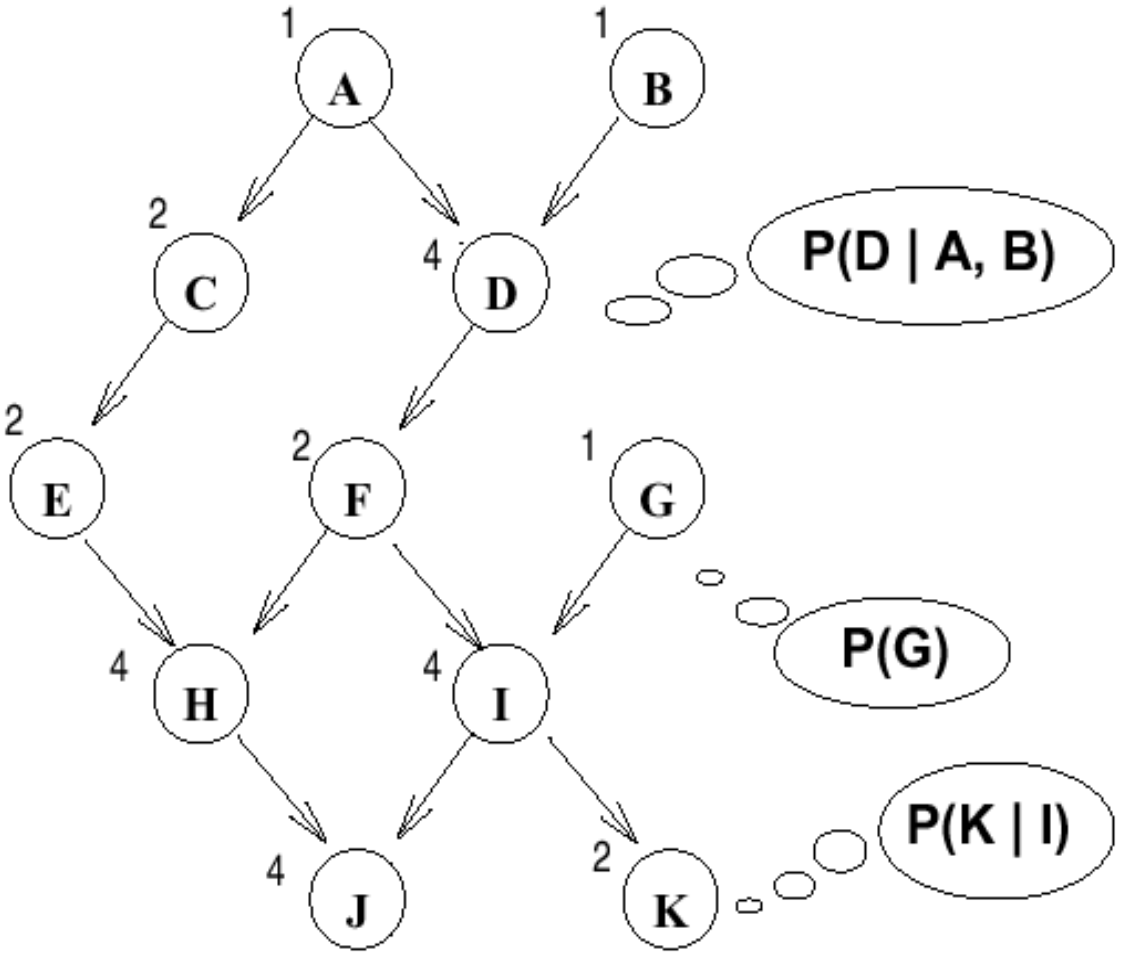
---

When evaluating query Q, we can restrict our attention to the *subnetwork comprising only relevant variables*

An example

Query: $P(D|h, -i)$

What variables are relevant?



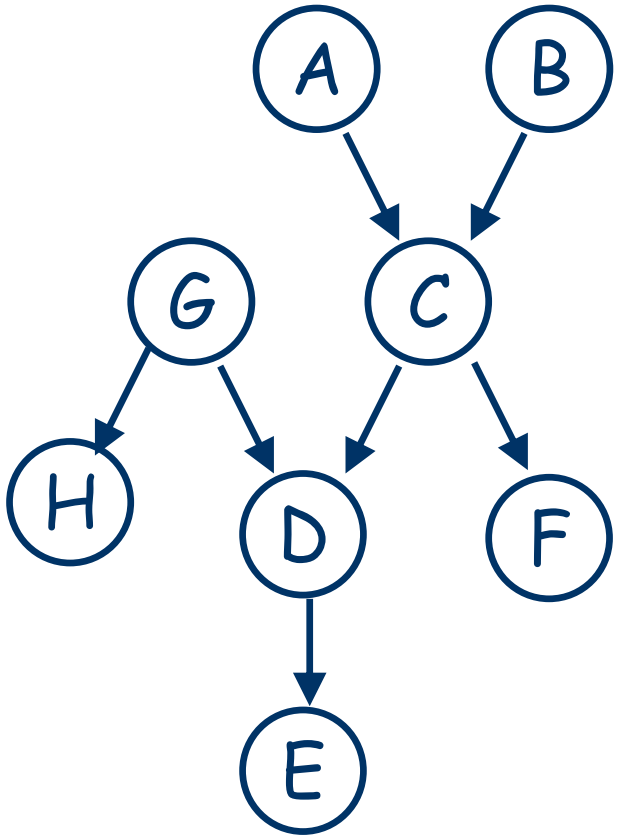
Relevance: Examples

Query: P(F)

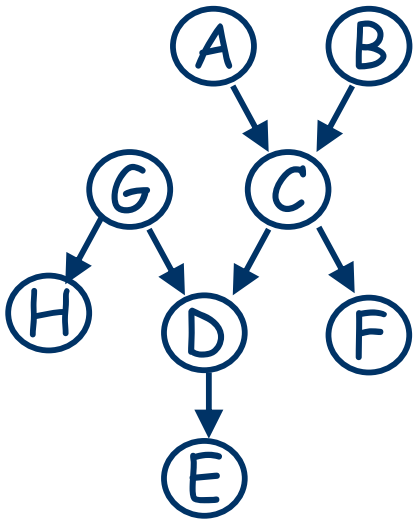
- relevant: F, C, B, A

Query: P(F|E)

- relevant: F, C, B, A
- **also: E, hence D, G**
- intuitively, we need to compute P(C|E) to compute P(F|E)



Relevance: Examples



Query: P(F|H)

- relevant F,C,A,B.

Pr(A,B,C,D,E,F,G,H)
= Pr(A)Pr(B)Pr(C|A,B)Pr(F|C) Pr(G)Pr(h|G) Pr(D|G,C) Pr(E|D)
= ... Pr(G)Pr(h|G)Pr(D|G,C) $\sum_E$Pr(E|D) = a table of 1's
= ... Pr(G)Pr(h|G)$\sum_D$ Pr(D|G,C) = a table of 1's
= [Pr(A)Pr(B)Pr(C|A,B)Pr(F|C)] [Pr(G)Pr(h|G)]

[Pr(G)Pr(h|G)] $\neq 1$

but irrelevant once we normalize, multiplies each value of F equally

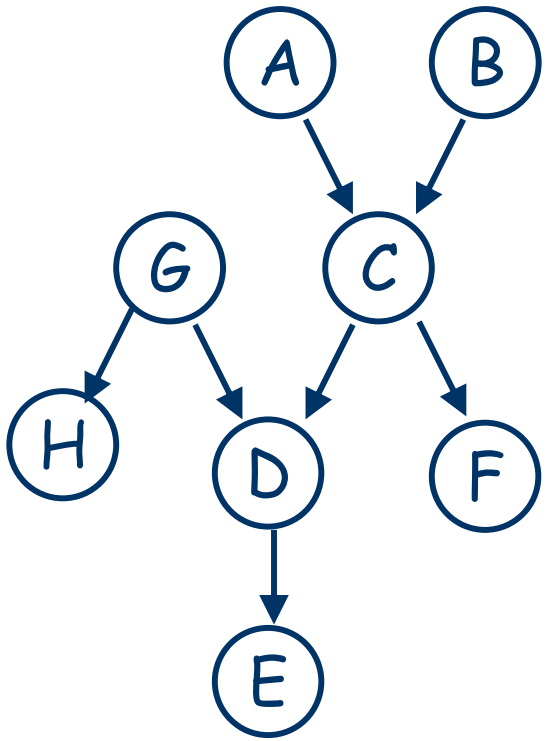
So D, E, G are actually irrelevant

Relevance: Examples

Query: $P(F|E, C)$

- algorithm says all variables are relevant, except H;
- but really none except C, F (since C cuts of all influence of others)

The algorithm is overestimating relevant set



Exercise

- Which of the following are asserted by the network structure?
  - $P(B, I, M) = P(B)P(I)P(M)$.
  - $P(J|G) = P(J|G, I)$.
  - $P(M|G, B, I) = P(M|G, B, I, J)$.

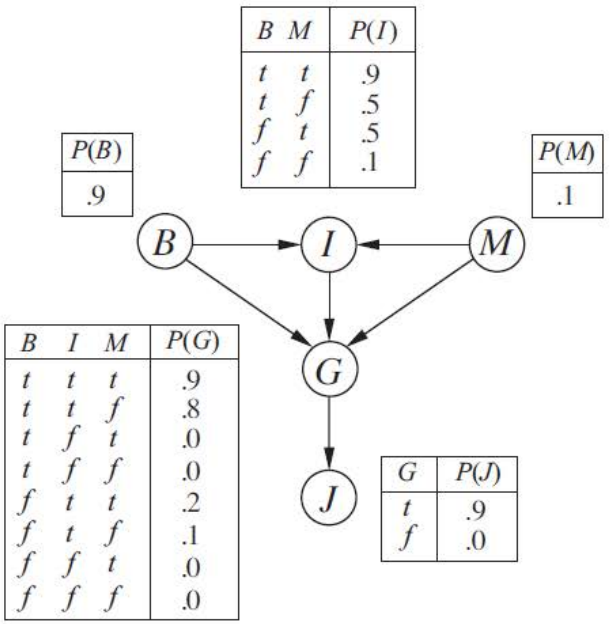


Figure 14.23 A simple Bayes net with Boolean variables $B = BrokeElectionLaw$, $I = Indicted$, $M = PoliticallyMotivatedProsecutor$, $G = FoundGuilty$, $J = Jailed$.

Exercise

- Calculate the value of $P(b, i, \neg m, g, j)$.
- Calculate the probability that someone goes to jail given that they broke the law, have been indicted, and face a politically motivated prosecutor.

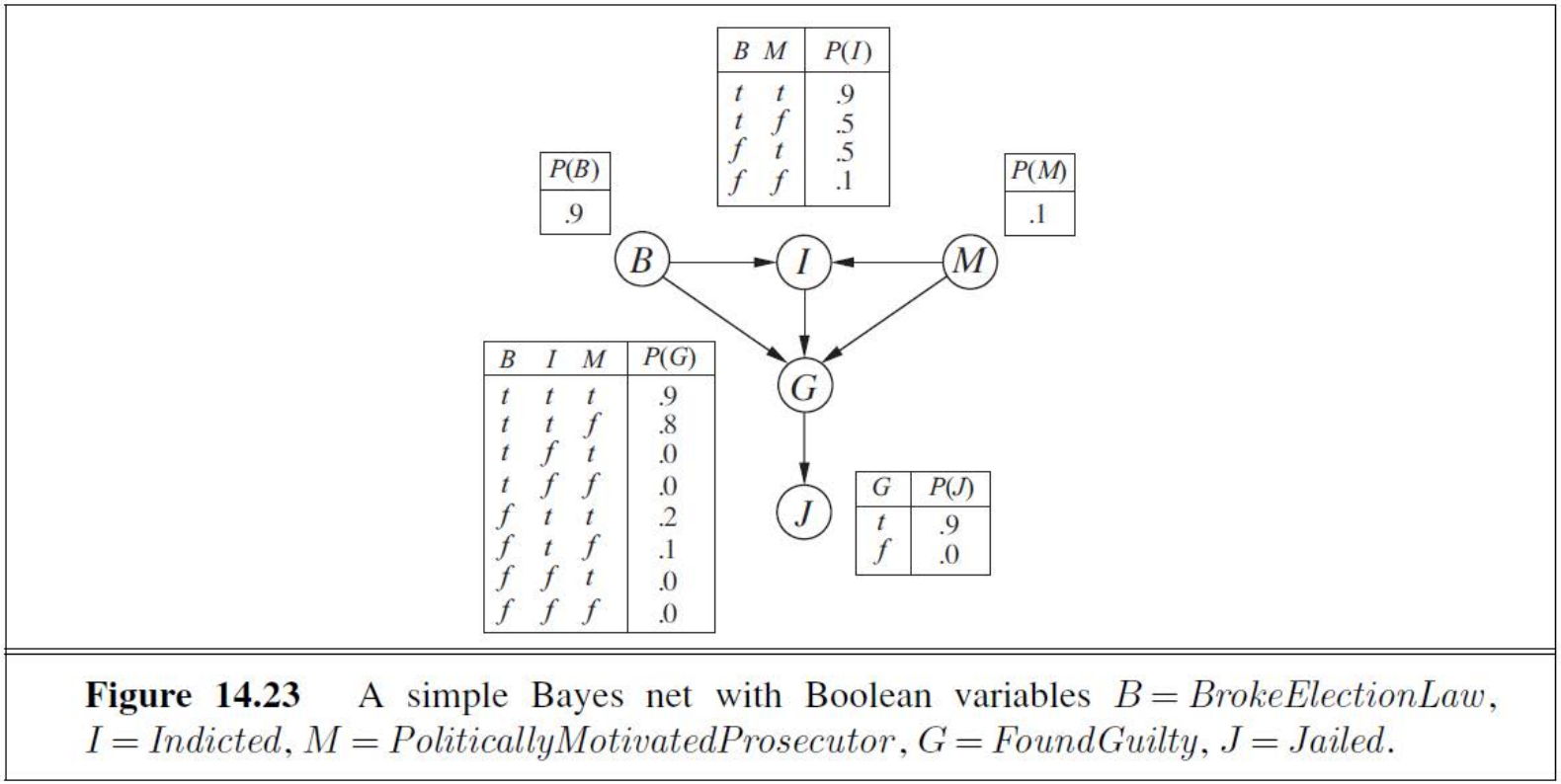


Figure 14.23 A simple Bayes net with Boolean variables $B = BrokeElectionLaw$, $I = Indicted$, $M = PoliticallyMotivatedProsecutor$, $G = FoundGuilty$, $J = Jailed$.