Reasoning under Uncertainty

This material is covered in chapters 13 and 14 of Russell and Norvig 2\textsuperscript{nd} and 3\textsuperscript{rd} edition.

- Chapter 13 gives some basic background on probability from the point of view of AI.
- Chapter 14 talks about Bayesian Networks, which will be a main topic for us.
Uncertainty

• With First Order Logic we examined a mechanism for representing true facts and for reasoning to new true facts.

• The emphasis on truth is sensible in some domains.

• But in many domain it is not sufficient to deal only with true facts. We have to “gamble”.

• E.g., we don’t know for certain what the traffic will be like on a trip to the airport.
Uncertainty

But how do we gamble *rationally*?

- If we must arrive at the airport at 9pm on a week night we could “safely” leave for the airport ½ 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 hour or more to get to the airport.
  - Relatively high probability of it taking 1.5 hours.
Uncertainty

• To act rationally under uncertainty we must be able to evaluate how likely certain things are.
  • An FOL fact \( F \) is only useful if it is known to be true or false.
  • But we need to be able to evaluate how likely it is that \( F \) is true.
• By weighing likelihoods of events (probabilities) we can develop mechanisms for acting rationally under uncertainty.
Dental Diagnosis example.

• In FOL we might formulate
  • \( \forall P. \text{symptom}(P, \text{toothache}) \rightarrow \text{disease}(P, \text{cavity}) \lor \text{disease}(P, \text{gumDisease}) \lor \text{disease}(P, \text{foodStuck}) \lor \cdots \)

• When do we stop?
  • Cannot list all possible causes.
  • We also want to rank the possibilities. We don’t want to start drilling for a cavity before checking for more likely causes first.
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.

\[
Pr(F) = \sum_{e \in F} Pr(e)
\]

• \( Pr(U) = 1 \), i.e., sum over all events is 1.

• Therefore: \( Pr(\{\}) = 0 \) and

\[
Pr(A \cup B) = Pr(A) + Pr(B) - Pr(A \cap B)
\]
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

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

Note if \( A \cap B = \{\} \) then \( \Pr(A \cup B) = \Pr(A) + \Pr(B) \)
Probability over Feature Vectors

• We will work with a universe consisting of a set of vectors of feature values.

• Like CSPs, we have
  1. a set of variables $V_1, V_2, \ldots, V_n$
  2. a finite domain of values for each variable, $\text{Dom}[V_1], \text{Dom}[V_2], \ldots, \text{Dom}[V_n]$.

• The universe of events $U$ is the set of all vectors of values for the variables
  $\langle d_1, d_2, \ldots, d_n \rangle: d_i \in \text{Dom}[V_i]$
Probability over Feature Vectors

• This event space has size
  \[ \prod_i |\text{Dom}[V_i]| \]
  i.e., the product of the domain sizes.

• E.g., if \(|\text{Dom}[V_i]| = 2\) 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 \(V_3 = d\).
  • ...

• E.g.
  • \(Pr(\{V_1 = a\}) = \sum_{x \in \text{Dom}[V_3]} Pr(\{V_1 = a, V_3 = x\}).\)
### Probability over Feature Vectors

- If we had Pr of every atomic event (full instantiation of the variables) we could compute the probability of any other set.

- E.g.
  - \{V_1 = a\} = set of all events where \( V_1 = a \)
  - \( \Pr(\{V_1 = a\}) = \)

\[
\sum_{x_2 \in \text{Dom}[V_2]} \sum_{x_3 \in \text{Dom}[V_3]} \cdots \sum_{x_n \in \text{Dom}[V_n]} \Pr(V_1=a, V_2=x_2, V_3=x_3, \ldots, V_n=x_n)
\]
Probability over Feature Vectors

Problem:

• This is an exponential number of atomic probabilities to specify.

• Requires summing up an exponential number of items.

• For evaluating the probability of sets containing a particular subset of variable assignments we can do much better. Improvements come from the use of probabilistic independence, especially conditional independence.
Conditional Probabilities.

- In logic one has implication to express “conditional” statements.
  - \( \forall X. \text{apple}(X) \rightarrow \text{goodToEat}(X) \)
  - This assertion only provides useful information about “apples”.
- With probabilities one has access to a different way of capturing conditional information: conditional probabilities.
- It turns out that conditional probabilities are essential for both representing and reasoning with probabilistic information.
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 $A$:

$$\Pr(B|A) = \frac{\Pr(B \cap A)}{\Pr(A)}$$
Conditional Probabilities

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

• Conditioning on A, corresponds to restricting one’s attention to the events in A.
• We now consider A to be the whole set of events (a new universe of events):
  \[ \Pr(A|A) = 1. \]
• Then we assign all other sets a probability by taking the probability mass that “lives” in A (Pr(B ∩ A)), and normalizing it to the range [0,1] by dividing by Pr(A).
Conditional Probabilities

B’s probability in the new universe A is 0.8.
Conditional Probabilities

• A conditional probability is a probability function, but now over $A$ instead of over the entire space $U$.

  • $\Pr(A|A) = 1$
  • $\Pr(B|A) \in [0,1]$
  • $\Pr(C \cup B|A) = \Pr(C|A) + \Pr(B|A) - \Pr(C \cap B|A)$
Properties and Sets

• In First Order Logic, properties like tall(X), are interpreted as a set of individuals (the individuals that are tall).

• Similarly any set of events A can be interpreted as a property: the set of events with property A.

• When we specify big(X) \( \land \) tall(X) in FOL, we interpret this as the set of individuals that lie in the intersection of the sets big(X) and tall(X).
Properties and Sets

- Similarly, big(X) ∨ tall(X) is the set of individuals that is the union of big(X) and tall(X).

- Hence, we often write
  - $A \lor B$ to represent the set of events $A \cup B$
  - $A \land B$ to represent the set of events $A \cap B$
  - $\neg A$ to represent the set of events $U - A$
    (i.e., the complement of $A$ wrt the universe of events $U$)
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 that its density on the whole space.

• In the first case we say that B is independent of A. While in the second case B is dependent on A.
Independence

Independent Dependent

Density of B

Sheila McIlraith, CSC384, University of Toronto, Winter 2011
Independence Definition

A and B are independent properties

\[ \Pr(B \mid A) = \Pr(B) \]

A and B are dependent.

\[ \Pr(B \mid A) \neq \Pr(B) \]
Implications for Knowledge

• Say that we have picked an element from the entire set. Then we find out that this element has property A (i.e., is a member of the set A).

• Does this tell us anything more about how likely it is that the element also has property B?

• If B is independent of A then we have learned nothing new about the likelihood of the element being a member of B.
Independence

• E.g., we have a feature vector, we don’t know which one. We then find out that it contains the feature $V_1 = a$.
  • I.e., we know that the vector is a member of the set $\{V_1 = a\}$.
  • Does this tell us anything about whether or not $V_2 = a$, $V_3 = c$, ..., etc?
  • This depends on whether or not these features are independent/dependent of $V_1 = a$. 
Conditional Independence

- Say we have already learned that the 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?
  \[ \Pr(B|A \land C) \] expresses the probability of this being true under the additional information.
Conditional Independence

• If

\[ \Pr(B|A \land C) = \Pr(B|A) \]

then we have not gained any additional information from knowing that the element is also a member of the set 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 to whether or not B is true).
  • Conditional independence is independence in the conditional probability space \( \Pr(\bullet | A) \).
  • Note we could have \( \Pr(B|C) \neq \Pr(B) \). But once we learn A, C becomes irrelevant.
Computational Impact of Independence

- We will see in more detail how independence allows us to speed up computation. But the fundamental insight is that

If A and B are independent properties then

\[ \Pr(A \land B) = \Pr(B) \times \Pr(A) \]

Proof:

\[ \Pr(B | A) = \Pr(B) \quad \text{independence} \]
\[ \Pr(A \land B)/\Pr(A) = \Pr(B) \quad \text{definition} \]
\[ \Pr(A \land B) = \Pr(B) \times \Pr(A) \]
Computational Impact of Independence

• This property allows us to “break” up the computation of a conjunction “Pr(A \& B)” into two separate computations “Pr(A)” and “Pr(B)”. 

• Dependent on how we express our probabilistic knowledge this yield great computational savings.
Computational Impact of Independence

• Similarly for conditional independence.

\[ \Pr(B|C \land A) = \Pr(B|A) \]
\[ \Pr(B \land C|A) = \Pr(B|A) \times \Pr(C|A) \]

Proof:
\[ \Pr(B|C \land A) = \Pr(B|A) \] (independence)
\[ \Pr(B \land C \land A)/\Pr(C \land A) = \Pr(B \land A)/\Pr(A) \] (defn.)
\[ \Pr(B \land C \land A)/\Pr(A) = \Pr(C \land A)/\Pr(A) \times \Pr(B \land A)/\Pr(A) \]
\[ \Pr(B \land C|A) = \Pr(B|A) \times \Pr(C|A) \] (defn.)
Computational Impact of Independence

• Conditional independence allows us to break up our computation into distinct parts
  \[ \Pr(B \land C | A) = \Pr(B | A) \times \Pr(C | A) \]

• And it also allows us to ignore certain pieces of information
  \[ \Pr(B | A \land C) = \Pr(B | A) \]
Bayes Rule

• Bayes rule is a simple mathematical fact. But it has great implications wrt how probabilities can be reasoned with.

• \( \Pr(Y \mid X) = \Pr(X \mid Y)\Pr(Y)/\Pr(X) \)

\[
\Pr(Y \mid X) = \frac{\Pr(Y \land X)}{\Pr(X)} \\
= \frac{\Pr(Y \land X)}{\Pr(X)} \cdot \frac{\Pr(Y)}{\Pr(Y)} \\
= \frac{\Pr(Y \land X)}{\Pr(Y)} \cdot \frac{\Pr(Y)}{\Pr(X)} \\
= \Pr(X \mid Y)\Pr(Y)/\Pr(X)
\]
Bayes Rule

• Bayes rule allows us to use a supplied conditional probability in both directions.

• E.g., from treating patients with heart disease we might be able to estimate the value of

\[ \text{Pr( high\_Cholesterol | heart\_disease)} \]

• With Bayes rule we can turn this around into a predictor for heart disease

\[ \text{Pr(heart\_disease | high\_Cholesterol)} \]

• Now with a simple blood test we can determine “high\_Cholesterol” use this to help estimate the likelihood of heart disease.
Bayes Rule

• For this to work we have to deal with the other factors as well

\[
\Pr(\text{heart\_disease} \mid \text{high\_Cholesterol}) \\
= \Pr(\text{high\_Cholesterol} \mid \text{heart\_disease}) \\
\times \frac{\Pr(\text{heart\_disease})}{\Pr(\text{high\_Cholesterol})}
\]

• We will return to this later.
Bayes Rule Example

• Disease ∈ \{malaria, cold, flu\}; Symptom = fever
  • Must compute \( \Pr(D \mid \text{fever}) \) to prescribe treatment

• Why not assess this quantity directly?
  • \( \Pr(\text{mal} \mid \text{fever}) \) is not natural to assess;
    \( \Pr(\text{fever} \mid \text{mal}) \) reflects the underlying “causal” mechanism
  • \( \Pr(\text{mal} \mid \text{fever}) \) is not “stable”: a malaria epidemic changes this quantity (for example)

• So we use Bayes rule:
  • \( \Pr(\text{mal} \mid \text{fever}) = \Pr(\text{fever} \mid \text{mal}) \Pr(\text{mal}) / \Pr(\text{fever}) \)
Bayes Rule

- $\Pr(\text{mal} \mid \text{fever}) = \frac{\Pr(\text{fever} \mid \text{mal})\Pr(\text{mal})}{\Pr(\text{fever})}$

- What about $\Pr(\text{mal})$?
  - This is the prior probability of Malaria, i.e., before you exhibited a fever, and with it we can account for other factors, e.g., a malaria epidemic, or recent travel to a malaria risk zone.
Bayes Rule

- \( \Pr(\text{mal} \mid \text{fever}) = \Pr(\text{fever} \mid \text{mal})\Pr(\text{mal})/\Pr(\text{fever}) \)

- What about \( \Pr(\text{fever})? \)
  - We compute
    \[
    \Pr(\text{fever} \mid \text{mal})\Pr(\text{mal}),
    \Pr(\text{fever} \mid \text{cold})\Pr(\text{cold}),
    \Pr(\text{fever} \mid \text{flu})\Pr(\text{flu}).
    \]
  - I.e., solve the same problem for each possible clause of fever.

- \( \Pr(\text{fever} \mid \text{mal})\Pr(\text{mal}) \)
  \[
  = \Pr(\text{fever} \land \text{mal})/\Pr(\text{mal}) \times \Pr(\text{mal})
  = \Pr(\text{fever} \land \text{mal})
  \]

- Similarly we can obtain
  \[
  \Pr(\text{fever} \land \text{cold})
  \]
  \[
  \Pr(\text{fever} \land \text{flu})
  \]
Bayes Rule

• Say that in our example, m, c and fl are only possible causes of fever (Pr(fev|¬m ∧ ¬c ∧ ¬fl) = 0) and they are mutually exclusive. Then

\[ Pr(\text{fev}) = Pr(m \& \text{fev}) + Pr(c \& \text{fev}) + Pr(fl \& \text{fev}) \]

• So we can sum the numbers we obtained for each disease to obtain the last factor in Bayes Rule: Pr(fever).
  • Then we divide each answer by Pr(fever) to get the final probabilities Pr(mal|fever), Pr(cold|fever), Pr(flu|fever).
  • This summing and dividing by the sum ensures that Pr(mal|fever) + Pr(cold|fever) + Pr(flu|fever) = 1, and is called normalizing.
Chain Rule

\[ \Pr(A_1 \land A_2 \land \ldots \land A_n) = \]
\[ \Pr(A_1 | A_2 \land \ldots \land A_n) \times \Pr(A_2 | A_3 \land \ldots \land A_n) \]
\[ \times \ldots \times \Pr(A_{n-1} | A_n) \times \Pr(A_n) \]

Proof:

\[ \Pr(A_1 | A_2 \land \ldots \land A_n) \times \Pr(A_2 | A_3 \land \ldots \land A_n) \]
\[ \times \ldots \times \Pr(A_{n-1} | A_n) \]
\[ = \frac{\Pr(A_1 \land A_2 \land \ldots \land A_n)}{\Pr(A_2 \land \ldots \land A_n)} \times \frac{\Pr(A_2 \land \ldots \land A_n)}{\Pr(A_3 \land \ldots \land A_n)} \times \ldots \times \frac{\Pr(A_{n-1} \land A_n)}{\Pr(A_n)} \times \Pr(A_n) \]
Variable Independence

• Recall that we will be mainly dealing with probabilities over feature vectors.

• We have a set of variables, each with a domain of values.

• It could be that \{V_1=a\} and \{V_2=b\} are independent:

\[
Pr(V_1=a \land V_2=b) = Pr(V_1=a) \times Pr(V_2=b)
\]

• It could also be that \{V_1=b\} and \{V_2=b\} are not independent:

\[
Pr(V_1=b \land V_2=b) \neq Pr(V_1=b) \times Pr(V_2=b)
\]
Variable Independence

- However we will generally want to deal with the situation where we have variable independence.

- Two variables $X$ and $Y$ are conditionally independent given variable $Z$ iff
  \[ \forall x,y,z. \ x \in \text{Dom}(X) \land y \in \text{Dom}(Y) \land z \in \text{Dom}(Z) \]
  \[
  \rightarrow \ X=x \text{ is conditionally independent of } \ Y=y \text{ given } Z = z
  \]
  \[
  \equiv \Pr(X=x \land Y=y|Z=z)
  \]
  \[
  = \Pr(X=x|Z=z) \times \Pr(Y=y|Z=z)
  \]

- Also applies to sets of more than two variables

- Also to unconditional case ($X,Y$ independent)
Variable Independence

• If you know the value of Z (\textit{whatever} it is), learning Y’s value (\textit{whatever} it is) does not influence your beliefs about any of X’s values.

• these definitions differ from earlier ones, which talk about particular sets of events being independent. Variable independence is a concise way of stating a number of individual independencies.
What does independence buys us?

• Suppose (say, 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)

• How? Simply specify $Pr(X_1), \ldots, Pr(X_n)$ (i.e., $Pr(X_i=\text{true})$ for all $i$)
  • from this I can recover probability of any primitive event easily (or any conjunctive query).
    e.g. $Pr(X_1 \neg X_2 X_3 X_4) = Pr(X_1) \cdot (1-Pr(X_2)) \cdot Pr(X_3) \cdot Pr(X_4)$
  • we can condition on observed value $X_k$ (or $\neg X_k$) trivially
    $Pr(X_1 \neg X_2 | X_3) = Pr(X_1) \cdot (1-Pr(X_2))$
The Value of Independence

• Complete independence reduces both representation of joint and inference from $O(2^n)$ to $O(n)$!
• Unfortunately, such complete mutual independence is very 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
An Aside on Notation

• Pr(X) for variable X (or set of variables) refers to the *(marginal)* distribution over X.
  • It specifies Pr(X=d) for all \(d \in \text{Dom}[X]\)

• Note

\[ \sum_{d \in \text{Dom}[X]} \text{Pr}(X=d) = 1 \]

(every vector of values must be in one of the sets \(\{X=d\} \ d \in \text{Dom}[X]\))

• Also

\[ \text{Pr}(X=d_1 \land X=d_2) = 0. \] for all \(d_1, d_2 \in \text{Dom}[X] \ d_1 \neq d_2 \]

(no vector of values contains two different values for X).
An Aside on Notation

• $\Pr(X | Y)$ refers to family of conditional distributions over $X$, one for each $y \in \text{Dom}(Y)$.
  • For each $d \in \text{Dom}[Y]$, $\Pr(X | Y)$ specifies a distribution over the values of $X$:
    $\Pr(X=d_1 | Y=d)$, $\Pr(X=d_2 | Y=d)$, ..., $\Pr(X=d_n | Y=d)$
    for $\text{Dom}[X] = \{d_1, d_2, ..., d_n\}$.

• Distinguish between $\Pr(X)$—which is a distribution—and $\Pr(x_i)$ ($x_i \in \text{Dom}[X]$)—which is a number. Think of $\Pr(X)$ as a function that accepts any $x_i \in \text{Dom}(X)$ as an argument and returns $\Pr(x_i)$.

• Similarly, think of $\Pr(X | Y)$ as a function that accepts any $x_i \in \text{Dom}[X]$ and $y_k \in \text{Dom}[Y]$ and returns $\Pr(x_i | y_k)$. Note that $\Pr(X | Y)$ is not a single distribution; rather it denotes the family of distributions (over $X$) induced by the different $y_k \in \text{Dom}(Y)$.
Exploiting Conditional Independence

• Let’s see what conditional independence buys us

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

\[
\begin{align*}
E & \rightarrow C & A & \rightarrow B & \rightarrow H \\
E & - Craig woke too early & A & - Craig is angry & H & - Craig hospitalized \\
C & - Craig needs coffee & B & - Craig burst a blood vessel
\end{align*}
\]
Cond’l Independence in our Story

- 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(~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 an aneurysm.
  - So H is independent of E, and C, and A, given B.
Similarly:
• B is independent of E, and C, given A
• A is independent of E, given C

This means that:
• \( \Pr(H \mid B, \{A,C,E\}) = \Pr(H \mid B) \)
  • i.e., for any subset of \{A,C,E\}, this relation holds
• \( \Pr(B \mid A, \{C,E\}) = \Pr(B \mid A) \)
• \( \Pr(A \mid C, \{E\}) = \Pr(A \mid C) \)
• \( \Pr(C \mid E) \) and \( \Pr(E) \) don’t “simplify”
Cond’l Independence in our Story

• 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)

• We can specify the full joint by specifying five local conditional distributions: Pr(H|B); Pr(B|A); Pr(A|C); Pr(C|E); and Pr(E)
**Example Quantification**

- Specifying the joint requires only 9 parameters (if we note that half of these are “1 minus” the others), instead of 31 for explicit representation
  - linear in number of vars instead of exponential!
  - linear generally if dependence has a chain structure
Inference is Easy

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

$$P(a) = \sum_{c_i \in \text{Dom}(C)} \Pr(a | c_i) \Pr(c_i)$$

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

These are all terms specified in our local distributions!
Inference is Easy

- Computing $P(a)$ in more concrete terms:
  - $P(c) = P(c|e)P(e) + P(c|\sim e)P(\sim e)$
    - $= 0.8 \times 0.7 + 0.5 \times 0.3 = 0.78$
  - $P(\sim c) = P(\sim c|e)P(e) + P(\sim c|\sim e)P(\sim e) = 0.22$
    - $P(\sim c) = 1 - P(c)$, as well
  - $P(a) = P(a|c)P(c) + P(a|\sim c)P(\sim c)$
    - $= 0.7 \times 0.78 + 0.0 \times 0.22 = 0.546$
  - $P(\sim a) = 1 - P(a) = 0.454$
Bayesian Networks

• 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* 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) \( \Pr(X_i \mid \text{Par}(X_i)) \)
    for each \( X_i \)

- Key notions (see text for defn’s, all are intuitive):
  - parents of a node: \( \text{Par}(X_i) \)
  - children of node
  - descendents of a node
  - ancestors of a node
  - family: set of nodes consisting of \( X_i \) and its parents
  - CPTs are defined over families in the BN
Example (Binary valued Variables)

- A couple CPTS are “shown”
- Explicit joint requires $2^{11} - 1 = 2047$ parameters
- BN requires only 27 parameters (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.

\[ \text{Pr}(X_1, X_2, \ldots, X_n) = \text{Pr}(X_n \mid \text{Par}(X_n)) \times \text{Pr}(X_{n-1} \mid \text{Par}(X_{n-1})) \times \ldots \times \text{Pr}(X_1 \mid \text{Par}(X_1)) \]

• This equation hold for any set of values \(d_1, d_2, \ldots, d_n\) for the variables \(X_1, X_2, \ldots, X_n\).
Semantics of Bayes Nets.

- E.g., say we have $X_1$, $X_2$, $X_3$ each with domain $\text{Dom}[X_i] = \{a, b, c\}$ and we have
  \[
  \Pr(X_1, X_2, X_3) = \Pr(X_3 | X_2) \Pr(X_2) \Pr(X_1)
  \]

Then

\[
\begin{align*}
\Pr(X_1 = a, X_2 = a, X_3 = a) &= \Pr(X_3 = a | X_2 = a) \Pr(X_2 = a) \Pr(X_1 = a) \\
\Pr(X_1 = a, X_2 = a, X_3 = b) &= \Pr(X_3 = b | X_2 = a) \Pr(X_2 = a) \Pr(X_1 = a) \\
\Pr(X_1 = a, X_2 = a, X_3 = c) &= \Pr(X_3 = c | X_2 = a) \Pr(X_2 = a) \Pr(X_1 = a) \\
\Pr(X_1 = a, X_2 = b, X_3 = a) &= \Pr(X_3 = a | X_2 = b) \Pr(X_2 = b) \Pr(X_1 = a) \\
\end{align*}
\]

...
Example (Binary valued Variables)

\[ \Pr(a,b,c,d,e,f,g,h,i,j,k) = \]

\[ \Pr(a) \times \Pr(b) \times \Pr(c|a) \times \Pr(d|a,b) \times \Pr(e|c) \times \Pr(f|d) \times \Pr(g) \times \Pr(h|e,f) \times \Pr(i|f,g) \times \Pr(j|h,i) \times \Pr(k|i) \]
Semantics of Bayes Nets.

• Note that this means we can compute the probability of any setting of the variables using only the information contained in the CPTs of the network.
Constructing a Bayes Net

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

- Take any ordering of the variables (say, the order given). From the chain rule we obtain.

$$
Pr(X_1, ..., X_n) = Pr(X_n|X_1, ..., X_{n-1})Pr(X_{n-1}|X_1, ..., X_{n-2})...Pr(X_1)
$$

- Now for each $X_i$ go through its conditioning set $X_1, ..., X_{i-1}$, 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.

- The final product will specify a Bayes net.
Constructing a Bayes Net

• The end result will be a product decomposition/Bayes net
  \[ \Pr(X_n \mid \text{Par}(X_n)) \Pr(X_{n-1} \mid \text{Par}(X_{n-1})) \ldots \Pr(X_1) \]

• Now we specify the numeric values associated with each term \( \Pr(X_i \mid \text{Par}(X_i)) \) in a CPT.

• Typically we represent the CPT as a table mapping each setting of \( \{X_i, \text{Par}(X_i)\} \) to the probability of \( X_i \) taking that particular value given that the variables in \( \text{Par}(X_i) \) have their specified values.

• If each variable has \( d \) different values.
  • We will need a table of size \( d^{\left| \{X_i, \text{Par}(X_i)\} \right|} \).
  • That is, exponential in the size of the parent set.

• Note that the original chain rule
  \[ \Pr(X_1, \ldots, X_n) = \Pr(X_n \mid X_1, \ldots, X_{n-1}) \Pr(X_{n-1} \mid X_1, \ldots, X_{n-2}) \ldots \Pr(X_1) \]
  requires as much space to represent as specifying the probability of each individual event.
Causal Intuitions

• The BN can be constructed using an arbitrary ordering of the variables.
• However, some orderings will yield BN’s 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”.

![Diagram showing a Bayes net with nodes for Malaria, Flu, Cold, and Aches, with directed edges connecting them.](image-url)
Causal Intuitions

• Suppose we build the BN for distribution \( P \) using the opposite ordering
  • i.e., we use ordering Aches, Cold, Flu, Malaria

\[
Pr(A,C,F,M) = Pr(M|A,C,F) \cdot Pr(F|A,C) \cdot Pr(C|A) \cdot Pr(A)
\]

• We can’t reduce \( Pr(M|A,C,F) \). Probability of Malaria is clearly affected by knowing aches. What 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!
Causal Intuitions

\[ \Pr(A,C,F,M) = \Pr(M | A,C,F) \Pr(F | A,C) \Pr(C | A) \Pr(A) \]

- 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).

Sheila McIlraith, CSC384, University of Toronto, Winter 2011
Bayes Net Examples

• I'm at work, neighbor John calls to say my alarm is ringing, but neighbor 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

# of Params: $1 + 1 + 4 + 2 + 2 = 10$ (vs. $2^5 - 1 = 31$)
Example of Constructing Bayes Network

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

\[ P(J \mid M) = P(J) ? \]
Example continue...

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

$$P(J \mid M) = P(J)? \quad \text{No}$$

$$P(A \mid J, M) = P(A \mid J)? \quad P(A \mid J, M) = P(A)?$$
Example continue...

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

\[ P(J \mid M) = P(J)? \text{ No} \]
\[ P(A \mid J, M) = P(A \mid J)? \quad P(A \mid J, M) = P(A)? \text{ No} \]
\[ P(B \mid A, J, M) = P(B \mid A)? \]
\[ P(B \mid A, J, M) = P(B)? \]
Example continue...

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

\[
P(J \mid M) = P(J) \ ? \ No
\]
\[
P(A \mid J, M) = P(A \mid J) \ ? \ P(A \mid J, M) = P(A) \ ? \ No
\]
\[
P(B \mid A, J, M) = P(B \mid A) \ ? \ Yes
\]
\[
P(B \mid A, J, M) = P(B) \ ? \ No
\]
\[
P(E \mid B, A, J, M) = P(E \mid A) ?
\]
\[
P(E \mid B, A, J, M) = P(E \mid A, B) ?
\]
Example continue...

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

\[ P(J \mid M) = P(J)? \quad \text{No} \]
\[ P(A \mid J, M) = P(A \mid J)? \quad P(A \mid J, M) = P(A)? \quad \text{No} \]
\[ P(B \mid A, J, M) = P(B \mid A)? \quad \text{Yes} \]
\[ P(B \mid A, J, M) = P(B)? \quad \text{No} \]
\[ P(E \mid B, A, J, M) = P(E \mid A)? \quad \text{No} \]
\[ P(E \mid B, A, J, M) = P(E \mid A, B)? \quad \text{Yes} \]
Deciding conditional independence is hard in non-causal directions!

(Causal models and conditional independence seem hardwired for humans!)

Network is less compact: $1 + 2 + 4 + 2 + 4 = 13$ numbers needed!
Inference in Bayes Nets

• Given a Bayes net
  \[ \Pr(X_1, X_2, ..., X_n) = \Pr(X_n \mid \text{Par}(X_n)) \times \Pr(X_{n-1} \mid \text{Par}(X_{n-1})) \times \cdots \times \Pr(X_1 \mid \text{Par}(X_1)) \]

• And some evidence \( E = \{ \text{a set of values for some of the variables} \} \) we want to compute the new probability distribution
  \[ \Pr(X_k \mid E) \]

• That is, we want to figure our \( \Pr(X_k = d \mid E) \) for all \( d \in \text{Dom}[X_k] \)
Inference in Bayes Nets

• Other types of examples are, computing probability of different diseases given symptoms, computing probability of hail storms given different metrological evidence, etc.

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

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

\[ Pr(B,E,A,Mc,Jc) = Pr(E) \times Pr(B) \times Pr(A|E,B) \times Pr(Mc|A) \times P(Jc|A) \]

• And, e.g., we want to compute things like
  \[ Pr(B=\text{True} | Mc=\text{true}, Jc=\text{false}, E=\text{false}) \]
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.
Example (Binary valued Variables)

\[
\Pr(A,B,C,D,E,F,G,H,I,J,K) = \\
\Pr(A) \times \Pr(B) \times \Pr(C|A) \times \Pr(D|A,B) \times \Pr(E|C) \times \Pr(F|D) \times \Pr(G) \times \Pr(H|E,F) \times \Pr(I|F,G) \times \Pr(J|H,I) \times \Pr(K|I)
\]
Example

\[ \Pr(A,B,C,D,E,F,G,H,I,J,K) = \]
\[ \Pr(A) \times \Pr(B) \times \Pr(C|A) \times \Pr(D|A,B) \times \Pr(E|C) \times \Pr(F|D) \times \Pr(G) \times \]
\[ \Pr(H|E,F) \times \Pr(I|F,G) \times \Pr(J|H,I) \times \Pr(K|I) \]

Now, say \( E = \{H=\text{true}, I=\text{false}\} \), and we want to know
\[ \Pr(D|h,-i) \quad (h: \text{H is true, } -i: \text{I is false}) \]

• Write as a sum for each value of \( D \)

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

2. \( \Pr(d,h,-i) + \Pr(-d,h,-i) = \Pr(h,-i) \)

3. \( \Pr(d|h,-i) = \frac{\Pr(d,h,-i)}{\Pr(h,-i)} \)
   \( \Pr(-d|h,-i) = \frac{\Pr(-d,h,-i)}{\Pr(h,-i)} \)

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

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

Use Bayes Net product decomposition to rewrite summation:

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

Now rearrange summations so that we are not summing over that do not depend on the summed variable.
Example

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

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

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

• Now start computing.

\[
\sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C) \\
\sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G) \\
\sum_j \Pr(J|h,-i) \\
\sum_k \Pr(K|-i)
\]

\[
\sum_k \Pr(K|-i) = \Pr(k|-i) + \Pr(-k|-i) = c_1
\]

\[
\sum_j \Pr(J|h,-i) \ c_1 = c_1 \sum_j \Pr(J|h,-i) \\
= c_1 (\Pr(j|h,-i) + \Pr(-j|h,-i)) \\
= c_1 c_2
\]
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.
Example

- So, Let’s Try eliminate in outside->inside order:

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

= 

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

+

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

Sheila McIlraith, CSC384, University of Toronto, Winter 2011
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_I \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_I \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_I \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_I \Pr(J|h,-i) 
\sum_K \Pr(K|-i)
Example

Yikes! 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.
Example

\[=\]

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

\[+\]

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

\[+\]

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

\[+\]

\[
\text{Pr}(-a)\text{Pr}(-b) \text{Pr}(d|-a,-b) \sum_C \text{Pr}(C|-a) \sum_E \text{Pr}(E|C) \\
\sum_F \text{Pr}(F|d) \text{Pr}(h|E,F) \sum_G \text{Pr}(G) \text{Pr}(-i|F,G) \\
\sum_J \text{Pr}(J|h,-i) \\
\sum_K \text{Pr}(K|-i) 
\]
Dynamic Programming

• 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

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

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

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

• So within the computation of the subterms we obtain more repeated smaller subterms.
• The 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).
• A brief aside is to also note that in the sum
\[ \sum_A \Pr(A) \sum_B \Pr(B) \Pr(d|A,B) \sum_C \Pr(C|A) \sum_E \Pr(E|C) \]
\[ \sum_F \Pr(F|d) \Pr(h|E,F) \sum_G \Pr(G) \Pr(-i|F,G) \]
\[ \sum_J \Pr(J|h,-i) \]
\[ \sum_K \Pr(K|\neg i) \]

we have that \[ \sum_K \Pr(K|\neg i) = 1 \] (Why?), thus
\[ \sum_J \Pr(J|h,-i) \sum_K \Pr(K|\neg i) = \sum_J \Pr(J|h,-i) \]

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

So we could drop these last two terms from the computation--
-J and K are not relevant given our query D and our evidence –i and –h. For now we keep these terms.
Variable Elimination (VE)

- VE works from the inside out, summing out K, then J, then G, ..., as we tried to before.
- When we tried to sum out G

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

\[
c_1c_2 \sum_G \Pr(G) \Pr(-i|F,G) \\
= c_1c_2(\Pr(g)\Pr(-i|F,g) + \Pr(-g)\Pr(-i|F,-g))
\]

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 (-f, f)!
Variable Elimination (VE)

- \( t(-f) = c_1 c_2 \sum_G \Pr(G) \Pr(-i|-f,G) \)

\[
\begin{align*}
\Pr(-i|f,G) &= \Pr(g) \Pr(-i|f,g) + \Pr(-g) \Pr(-i|f,-g) \\
\sum_G \Pr(G) &= \frac{1}{Z(f)} \\
\sum_G \Pr(G) \Pr(-i|f,G) &= \frac{1}{Z(f)} \Pr(g) \Pr(-i|f,g) + \frac{1}{Z(f)} \Pr(-g) \Pr(-i|f,-g) \\
\sum_G \Pr(G) \Pr(-i|-f,G) &= \frac{1}{Z(f)} \Pr(g) \Pr(-i|-f,g) + \frac{1}{Z(f)} \Pr(-g) \Pr(-i|-f,-g) \\
\end{align*}
\]

- \( t(f) = c_1 c_2 (\sum_G \Pr(G) \Pr(-i|f,G) \)

- 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)

- \( c_1c_2(Pr(f|d) \ Pr(h|E,f)t(f) + Pr(-f|d)Pr(h|E,-f)t(-f) \)

- This is a function of E, so we obtain two new numbers

\[
s(e) = c_1c_2(Pr(f|d) \ Pr(h|e,f)t(f) + Pr(-f|d)Pr(h|e,-f)t(-f) \)
\]

\[
s(-e) = c_1c_2(Pr(f|d) \ Pr(h|-e,f)t(f) + Pr(-f|d)Pr(h|-e,-f)t(-f) \)
\]
Variable Elimination (VE)

• 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 $\text{Pr}(d,h,-i)$.

• Now we can repeat the process to compute $\text{Pr}(-d,h,-i)$.

• But instead of doing it twice, we can simply regard D as a variable in the computation.

• This will result in some computations depending on the value of D, and we obtain a different number for each value of D.

• Proceeding in this manner, summing out A will yield a function of D. (I.e., a number for each value of D).
Variable Elimination (VE)

• In general, at each stage VE will compute a table of numbers: one number for each different instantiation of the variables that are 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 \(|\text{Dom}[D]| \times |\text{Dom}[E]|\) different numbers in the resulting table.
Factors

• we call these tables of values computed by VE factors.
• Note that the original probabilities that appear in the summation, e.g., $P(C|A)$, are also tables of values (one value for each instantiation of $C$ and $A$).
• Thus we also call the original CPTs 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.
Operations on Factors

• If we examine the inside-out summation process we see that various operations occur on factors.

• 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)
The Product of Two Factors

• Let \( f(\text{X,Y}) \) & \( g(\text{Y,Z}) \) be two factors with variables \( \text{Y} \) in common

• The *product* of \( f \) and \( g \), denoted \( h = f \times g \) (or sometimes just \( h = fg \)), is defined:

\[
h(\text{X,Y,Z}) = f(\text{X,Y}) \times g(\text{Y,Z})
\]

<table>
<thead>
<tr>
<th>( f(\text{A,B}) )</th>
<th>( g(\text{B,C}) )</th>
<th>( h(\text{A,B,C}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>ab</td>
<td>0.9</td>
<td>abc</td>
</tr>
<tr>
<td>a~b</td>
<td>0.1</td>
<td>b~c</td>
</tr>
<tr>
<td>~ab</td>
<td>0.4</td>
<td>~bc</td>
</tr>
<tr>
<td><del>a</del>b</td>
<td>0.6</td>
<td><del>b</del>c</td>
</tr>
</tbody>
</table>
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 $h = \sum_X f$, which is defined:

$$h(Y) = \sum_{x \in \text{Dom}(X)} f(x, Y)$$

<table>
<thead>
<tr>
<th>$f(A, B)$</th>
<th>$h(B)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ab$</td>
<td>0.9</td>
</tr>
<tr>
<td>$a\sim b$</td>
<td>0.1</td>
</tr>
<tr>
<td>$\sim ab$</td>
<td>0.4</td>
</tr>
<tr>
<td>$\sim a\sim b$</td>
<td>0.6</td>
</tr>
</tbody>
</table>
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: $h(Y) = f(a,Y)$

<table>
<thead>
<tr>
<th></th>
<th>$f(A,B)$</th>
<th>$h(B) = f_{A=a}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ab$</td>
<td>0.9</td>
<td>$b$</td>
</tr>
<tr>
<td>$a\sim b$</td>
<td>0.1</td>
<td>$\sim b$</td>
</tr>
<tr>
<td>$\sim ab$</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>$\sim a\sim b$</td>
<td>0.6</td>
<td></td>
</tr>
</tbody>
</table>
Variable Elimination the Algorithm

Given query var Q, evidence vars \( \mathbf{E} \) (set of variables observed to have values \( \mathbf{e} \)), remaining vars \( \mathbf{Z} \). Let \( \mathbf{F} \) be original CPTs.

1. Replace each factor \( f \in \mathbf{F} \) that mentions a variable(s) in \( \mathbf{E} \) with its restriction \( f_{\mathbf{E}=\mathbf{e}} \) (this might yield a “constant” factor)
2. For each \( Z_j \) — in the order given — *eliminate* \( Z_j \in \mathbf{Z} \) as follows:
   (a) Compute new factor \( g_j = \sum_{Z_j} f_1 \times f_2 \times \ldots \times f_k \), where the \( f_i \) are the factors in \( \mathbf{F} \) that include \( Z_j \)
   (b) Remove the factors \( f_i \) (that mention \( Z_j \)) from \( \mathbf{F} \) and add new factor \( g_j \) to \( \mathbf{F} \)
3. The remaining factors refer only to the query variable Q. Take their product and normalize to produce \( \Pr(Q|\mathbf{E}) \)
**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 $f_6(A,B) = \sum_C f_5(C) f_3(A,B,C)$  
Remove: $f_3(A,B,C)$, $f_5(C)$

**Step 2:** **Eliminating B:** Compute & Add $f_7(A) = \sum_B f_6(A,B) f_2(B)$  
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 $\alpha = 1/\sum_A f_1(A)f_7(A)$
**Numeric Example**

• Here’s the example with some numbers

![Diagram](attachment:image.png)

<table>
<thead>
<tr>
<th></th>
<th>$f_1(A)$</th>
<th>$f_2(A,B)$</th>
<th>$f_3(B,C)$</th>
<th>$f_4(B)$</th>
<th>$f_5(C)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>0.9</td>
<td>0.9</td>
<td>bc</td>
<td>0.7</td>
<td>b</td>
</tr>
<tr>
<td>$\sim a$</td>
<td>0.1</td>
<td>a$\sim b$</td>
<td>0.1</td>
<td>$b\sim c$</td>
<td>0.3</td>
</tr>
<tr>
<td>$\sim ab$</td>
<td>0.4</td>
<td>$\sim bc$</td>
<td>0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\sim a\sim b$</td>
<td>0.6</td>
<td>$\sim b\sim c$</td>
<td>0.8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
VE: Buckets as a Notational Device

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

1. C:
2. F:
3. A:
4. B:
5. E:
6. D:
**VE:** 1. Place Original Factors in first applicable bucket.

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:
VE: 2. Eliminate the variables in order, placing new factor in first applicable bucket.

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: 
\[ \Sigma_C f_3(A, B, C), f_4(C, D), f_5(C, E) = f_7(A, B, D, E) \]
VE: Eliminate F, placing new factor \( f_8 \) in first applicable bucket.

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. \( D: \)

\[
\sum_F f_6(E,D,F) = f_8(E,D)
\]
**VE:** Eliminate A, placing new factor f9 in first applicable bucket.

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

1. C: f3(A, B, C), f4(C, D), f5(C, E)
2. F: f6(E, D, F)
3. A: f1(A), f7(A, B, D, E)
4. B: f2(B), f9(B, D, E)
5. E: f8(E, D)
6. D:

3. Eliminating A:
\[ \Sigma_A f_1(A), f_7(A, B, D, E) = f_9(B, D, E) \]
**VE:** Eliminate B, placing new factor $f_{10}$ in first applicable bucket.

**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 $B$:
$$\sum_B f_2(B), f_9(B, D, E) = f_{10}(D, E)$$
**VE:** Eliminate E, placing new factor f11 in first applicable bucket.

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

1. C: f₃(A,B,C), f₄(C,D), f₅(C,E)
2. F: f₆(E,D,F)
3. A: f₁(A), f₇(A,B,D,E)
4. B: f₂(B), f₉(B,D,E)
5. E: f₈(E,D), f₁₀(D,E)
6. D: f₁₁(D)

5. Eliminating E:

\[ \Sigma_E f₈(E,D), f₁₀(D,E) = f₁₁(D) \]

f₁₁ is the final answer, once we normalize it.
Complexity of Variable Elimination

- Hypergraph of Bayes Net.
  - Hypergraph has vertices just like an ordinary graph, but instead of edges between two vertices $X\leftrightarrow Y$ it contains hyperedges.
  - A hyperedge is a set of vertices (i.e., potentially more than one)
Complexity of Variable Elimination

- 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.
    - \( \{X_i\} \cup \text{Par}(X_i) \)
Complexity of Variable Elimination

- \( \text{Pr}(A,B,C,D,E,F) = \text{Pr}(A)\text{Pr}(B) \times \text{Pr}(C|A,B) \times \text{Pr}(E|C) \times \text{Pr}(D|C) \times \text{Pr}(F|E,D) \).
Variable Elimination in the HyperGraph

- To eliminate variable $X_i$ in the hypergraph we
  - 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 of the hyperedges containing $X$ from the hypergraph.
  - Add the new hyperedge $H_i$ to the hypergraph.
Complexity of Variable Elimination

- From above
  - Eliminate C
Complexity of Variable Elimination

• From above
   Eliminate D
Complexity of Variable Elimination

• From above
  Eliminate A
Variable Elimination

• Notice that when we start VE we have a set of factors consisting of the reduced CPTs. The unassigned variables for the vertices and the set of variables each factor depends on forms the hyperedges of a hypergraph $H_1$.

• If the first variable we eliminate is $X$, then we remove all factors containing $X$ (all hyperedges) and add a new factor that has as variables the union of the variables in the factors containing $X$ (we add a hyperdege that is the union of the removed hyperedges minus $X$).
VE Factors

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

1. C:
2. F:
3. A:
4. B:
5. E:
6. D:
VE: Place Original Factors in first applicable bucket.

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:
**VE:** Elim C & place new factor f7 in 1st applicable bucket.

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

1. C: f3(A,B,C), f4(C,D), f5(C,E)
2. F: f6(E,D,F)
3. A: f1(A), f7(A,B,D,E)
4. B: f2(B)
5. E:
6. D:

Sheila McIlraith, CSC384, University of Toronto, Winter 2011
**VE:** Elim F & place new factor f8 in 1st applicable bucket.

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

1. C: f3(A, B, C), f4(C, D), f5(C, E)
2. F: f6(E, D, F)
3. A: f1(A), f7(A, B, D, E)
4. B: f2(B)
5. E: f8(E, D)
6. D:
**VE: Elim A & place new factor f9 in 1st applicable bucket.**

**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:

Sheila McIlraith, CSC384, University of Toronto, Winter 2011
VE: Elim B & pace new factor f10 in 1st applicable bucket.

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

1. C: f3(A,B,C), f4(C,D), f5(C,E)
2. F: f6(E,D,F)
3. A: f1(A), f7(A,B,D,E)
4. B: f2(B), f9(B,D,E)
5. E: f8(E,D), f10(D,E)
6. D:

Sheila McIlraith, CSC384, University of Toronto, Winter 2011
VE: Elim E & place new factor f11 in 1st applicable bucket

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

1. C: f3(A, B, C), f4(C, D), f5(C, E)
2. F: f6(E, D, F)
3. A: f1(A), f7(A, B, D, E)
4. B: f2(B), f9(B, D, E)
5. E: f8(E, D), f10(D, E)
6. D: f11(D)
Elimination Width

• Given an ordering $\pi$ of the variables and an initial hypergraph $\mathcal{H}$ eliminating these variables yields a sequence of hypergraphs

$$\mathcal{H} = H_0, H_1, H_2, \ldots, H_n$$

• Where $H_n$ contains only one vertex (the query variable).

• The elimination width $\pi$ is the maximum size (number of variables) of any hyperedge in any of the hypergraphs $H_0, H_1, \ldots, H_n$.

• The elimination width of the previous example was 4 ($\{A,B,E,D\}$ in $H_1$ and $H_2$).
Elimination Width

• If the elimination width of an ordering $\pi$ is $k$, then the complexity of VE using that ordering is $2^{O(k)}$.

• Elimination width $k$ means that at some stage in the elimination process a factor involving $k$ variables was generated.

• That factor will require $2^{O(k)}$ space to store
  • space complexity of VE is $2^{O(k)}$

• And it will require $2^{O(k)}$ operations to process (either to compute in the first place, or when it is being processed to eliminate one of its variables).
  • Time complexity of VE is $2^{O(k)}$

• NOTE, that $k$ is the elimination width of this particular ordering.
Tree Width

• Given a hypergraph $\mathcal{H}$ with vertices $\{X_1, X_2, \ldots, X_n\}$ the tree width of $\mathcal{H}$ is the MINIMUM elimination width of any of the $n!$ different orderings of the $X_i$ minus 1.

• Thus VE has best case complexity of $2^{O(\omega)}$ where $\omega$ 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 that has 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).
  • Sometimes, however, the treewidth is equal to the number of variables.
Finding Good Orderings

• A *polytrees* is a singly connected Bayes Net: in particular there is only one path between any two nodes.

• A node can have multiple parents, but we have 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 is equal to the 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
  • eliminating B before these gives factors that include all of A,C, X1,... Xk !!!
Effect of Different Orderings

• Suppose query variable is D. Consider different orderings for this network (not a polytree!)
  • A,F,H,G,B,C,E:
    • good
  • E,C,A,B,G,H,F:
    • bad
Min Fill Heuristic

• A fairly effective heuristic is always **eliminate next the variable that creates the smallest size factor.**
• This is called the **min-fill heuristic.**
• 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.
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: \( f_4(B) = \sum_C f_3(B,C) = \sum_C Pr(C|B) \)
    • 1 for any value of B (e.g., \( Pr(c|b) + Pr(\neg c|b) = 1 \))

• No need to think about B or C for this query
Relevance

• Can restrict attention to relevant variables. Given query q, evidence E:
  • q itself is relevant
  • if any node Z is relevant, its parents are relevant
  • if e∈E is a descendant of a relevant node, then E is relevant

• We can restrict our attention to the subnetwork comprising only relevant variables when evaluating a query Q
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 \( P(F|E) \)
• Query: \( P(F|H) \)
  • relevant \( F, C, A, B \).

\[
\Pr(A)\Pr(B)\Pr(C|A,B)\Pr(F|C) \Pr(G)\Pr(h|G) \Pr(D|G,C) \Pr(E|D)
= \ldots \Pr(G)\Pr(h|G)\Pr(D|G,C) \sum_E \Pr(E|D) = \text{a table of 1's}
= \ldots \Pr(G)\Pr(h|G)\sum_D \Pr(D|G,C) = \text{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 \text{ but irrelevant once we normalize, multiplies each value of } F \text{ equally}
Relevance: Examples

• Query: $P(F|E,C)$
  • algorithm says all vars relevant; but really none except C, F (since C cuts off influence of others)
  • algorithm is overestimating relevant set
Independence in a Bayes Net

• Another piece of information we can obtain from a Bayes net is the “structure” of relationships in the domain.

• The structure of the BN means: every $X_i$ is conditionally independent of all of its nondescendants given its parents:

$$\Pr(X_i \mid S \cup \text{Par}(X_i)) = \Pr(X_i \mid \text{Par}(X_i))$$

for any subset $S \subseteq \text{NonDescendents}(X_i)$
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 $E$?
  • 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 $Y$. (We'll define *blocks* next.)

• $X$ and $Y$ are conditionally independent given evidence $E$ if $E$ *d*-separates $X$ and $Y
  • thus BN gives us an easy way to tell if two variables are independent (set $E = \emptyset$) or cond. independent given $E$. 
Blocking in D-Separation

• Let P be an \textit{undirected} path from X to Y in a BN. Let E be a set of variables. \textit{We say E blocks path P iff there is some} node Z on the path such that:

• \textbf{Case 1:} one arc on P \textit{goes into} Z and one \textit{goes out} of Z, and \( Z \in E \); or

• \textbf{Case 2:} both arcs on P leave Z, and \( Z \in E \); or

• \textbf{Case 3:} both arcs on P enter Z and \textit{neither Z, nor any of its descendents}, 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
Recall: D-Separation

**D-separation**: A set of variables $E$ \textit{d-separates} $X$ and $Y$ if it \textit{blocks every undirected path} in the BN between $X$ and $Y$. 
D-Separation: Intuitions

- Subway and Thermometer are dependent; but are independent given Flu (since Flu blocks the only path)
D-Separation: Intuitions

- Aches and Fever are dependent; but are independent given Flu (since Flu blocks the only path). Similarly for Aches and Therm (dependent, but indep. given Flu).
D-Separation: Intuitions

- Flu and Mal are independent (given no evidence): Fever blocks the path, since it is not in evidence, nor is its descendant Therm.
- Flu and Mal are dependent given Fever (or given Therm): nothing blocks path now.

What's the intuition?
D-Separation: Intuitions

- Subway, ExoticTrip are independent;
- They are dependent given Therm;
- They are independent given Therm and Malaria. This for exactly the same reasons for Flu/Mal above.
D-Separation Example

• In the following network determine if A and E are independent given the evidence:

1. A and E given no evidence? N
2. A and E given {C}? N
3. A and E given {G,C}? Y
5. A and E given {G,F}? N
6. A and E given {F,D}? Y
7. A and E given {F,D,H}? N
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