


# Artificial Intelligence

## Probabilistic Reasoning



Mountains & Minds 1

---

---

---

---

---



---

---

---

# Perfect Information?

- So far, we have assumed models are **truth**.
- We have also assumed facts are **truth**.
- We also required **complete** information.
- What if our knowledge is **incomplete**?
- What is our truth is **uncertain**?
- What happens if our truth **conflicts**?
- These questions apply to all approaches.



Mountains & Minds 2

---

---

---

---

---


---

---

---

# Why Reason Probabilistically?

- Agents are resource bound.
- With incomplete information, must “reason under uncertainty.”
- Probability theory has firm theoretical and practical foundation.
- Can model both “chance” items (i.e., randomness) and incomplete information (i.e., uncertainty).



Mountains & Minds 3

---

---

---

---

---

---

---

---

## Probabilistic Agents

- We take a “Bayesian” view of the world.
- Agents have *degrees of belief* about facts.
- Rather than reasoning about truth or falsity of a fact, reason about the belief that a fact is likely or unlikely.
- Represent degrees of belief as the probability that a fact is true, given the evidence.

---

---

---

---

---

---

---

---

## Probability Review

- **Def:** A *random variable* is a variable that can take on some value (from a set of legal values).
- **Def:** A *probability measure* is a function  $P: \Omega^n \rightarrow [0,1]$  such that the following axioms hold.
  - $P(\Omega) = 1$
  - If  $A \subseteq \Omega$ ,  $P(A) \in [0,1]$
  - For sets  $A, B \subseteq \Omega$ ,  
 $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ .

---

---

---

---

---

---

---

---

## Random Variables

- For random variable  $X$ , we consider probabilities of the form  $P(X = x) = p$ , where  $0 \leq p \leq 1$ .
- This defines a probability *distribution* over the values of the random variable.
- For Boolean random variables, we have
  - $P(X = \text{True}) = P(X)$
  - $P(X = \text{False}) = P(\neg X) = 1 - P(X)$

---

---

---

---

---

---

---

---

## Unconditional Probabilities

- Also sometimes called “prior” or a *priori* probabilities.
- Probability of a random variable taking a value in the absence of any information.
- Examples
  - $P(\text{Weather} = \text{sunny}) = 0.4$
  - $P(\text{Weather} = \text{overcast}) = 0.2$
  - $P(\text{Weather} = \text{rainy}) = 0.4$

---

---

---

---

---

---

---

---

## Conditional Probabilities

- Also sometimes called “posterior” or a *posteriori* probabilities or the “likelihood” (depending on their role).
- Defined by the mathematical expression:

$$P(a | b) = \frac{P(a \wedge b)}{P(b)}$$

---

---

---

---

---

---

---

---

## Marginalizing and Conditioning

- One common task is extracting the distribution over a subset of the variables or over a single variable.
- The process of extracting the sub-distribution is called *marginalizing*.

$$P(\mathbf{Y}) = \sum_{\mathbf{z}} P(\mathbf{Y}, \mathbf{z})$$

- Doing this with conditional probabilities is called *conditioning*.

$$P(\mathbf{Y}) = \sum_{\mathbf{z}} P(\mathbf{Y} | \mathbf{z}) P(\mathbf{z})$$

- Note: Given the definition of conditional probability, the two are equivalent.

---

---

---

---

---

---

---

---

## Product Rule

- Simple algebraic manipulation of definition of conditional probability yields,  $P(a \wedge b) = P(a | b) P(b)$ .
- An expanded form (called the conditionalized form or the factored form) of the product rule yields  $P(a, b | e) = P(a | b, e) P(b | e)$ .
- Generalizing,  
 $P(v_1, \dots, v_n) = P(v_1) \times P(v_2 | v_1) \times \dots \times P(v_n | v_2, \dots, v_{n-1})$ .
- This is also called the *chain rule*.

---

---

---

---

---

---

---

---

## Bayes' Rule

- Bayes' Rule follows from two instances of the product rule
  - $P(a \wedge b) = P(a | b) P(b)$
  - $P(b \wedge a) = P(b | a) P(a)$
- Since  $P(a \wedge b) = P(b \wedge a)$ , we know
  - $P(a | b) P(b) = P(b | a) P(a)$
- Therefore,

$$P(b | a) = \frac{P(a | b)P(b)}{P(a)} = \frac{P(a | b)P(b)}{P(a | b)P(b) + P(a | \neg b)P(\neg b)}$$

---

---

---

---

---

---

---

---

## A Puzzle

- Sandy takes a routine blood test, testing for a disease that strikes 1 in 15,000 people.
- The test is never wrong if it comes back negative, so if a person has the disease, the test will be positive.
- However, it is expected to be wrong in 1 out of 200 cases when testing people without the disease (i.e., come back positive when it should be negative).
- The results of the test are positive.
- What is the probability that Sandy has the disease?

---

---

---

---

---

---

---

---

## A Puzzle

- Assume 15,000 people are selected at random. Then 1 of these is expected to actually have the disease. Since false negatives don't occur, that person will test positive.
- Of the remaining 14,999 people, 74.995 people are expected to test positive, even though they don't have the disease.

$$\frac{1}{200} * 14,999 = 74.995$$

---

---

---

---

---

---

---

---

## A Puzzle

- Using Bayes' Rule

$$P(Dis | Pos) = \frac{P(Pos | Dis)P(Dis)}{P(Pos)} = \frac{1 * \frac{1}{15000}}{\frac{1}{200}} = \frac{0.000067}{P(Pos)}$$

$$P(-Dis | Pos) = \frac{P(Pos | -Dis)P(-Dis)}{P(Pos)} = \frac{\frac{1}{200} * \frac{14999}{15000}}{\frac{1}{200}} = \frac{0.005}{P(Pos)}$$

---

---

---

---

---

---

---

---

## A Puzzle

- Since  $P(Dis|Pos) + P(-Dis|Pos) = 1$ ,  
 $\frac{0.000067}{P(Pos)} + \frac{0.005}{P(Pos)} = \frac{0.005067}{P(Pos)} = 1$

$$P(Pos) = 0.005067$$

$$P(Dis | Pos) = \frac{0.000067}{0.005067} = 0.0132$$

$$P(-Dis | Pos) = \frac{0.005}{0.005067} = 0.9868$$

---

---

---

---

---

---

---

---

## Probabilistic Inference

- We will start by considering inference with the full joint probability distribution.
- We will illustrate the process with a small example.
- Assume we have three Boolean variables: *Toothache*, *Cavity*, and *Catch* (i.e. dentist's tool catches the tooth).
- We represent the joint probability distribution with a table.

---

---

---

---

---

---

---

---

## Full Joint Distribution

	<i>toothache</i>		<i>¬toothache</i>	
	<i>catch</i>	<i>¬catch</i>	<i>catch</i>	<i>¬catch</i>
<i>cavity</i>	0.108	0.012	0.072	0.008
<i>¬cavity</i>	0.016	0.064	0.144	0.576

---

---

---

---

---

---

---

---

## Full Joint Distribution

	<i>toothache</i>		<i>¬toothache</i>	
	<i>catch</i>	<i>¬catch</i>	<i>catch</i>	<i>¬catch</i>
<i>cavity</i>	0.108	0.012	0.072	0.008
<i>¬cavity</i>	0.016	0.064	0.144	0.576

$$P(\text{toothache}) = 0.108 + 0.012 + 0.016 + 0.064 = 0.2$$

---

---

---

---

---

---

---

---

## Full Joint Distribution

	toothache		¬toothache	
	catch	¬catch	catch	¬catch
cavity	0.108	0.012	0.072	0.008
¬cavity	0.016	0.064	0.144	0.576

$$\begin{aligned}
 P(\text{toothache} \vee \text{cavity}) &= 0.108 + 0.012 + 0.016 + 0.064 \\
 &\quad + 0.072 + 0.008 \\
 &= 0.28
 \end{aligned}$$

---

---

---

---

---

---

---

---

---

---

## Full Joint Distribution

	toothache		¬toothache	
	catch	¬catch	catch	¬catch
cavity	0.108	0.012	0.072	0.008
¬cavity	0.016	0.064	0.144	0.576

$$\begin{aligned}
 P(\text{cavity} \mid \text{toothache}) &= \frac{P(\text{cavity} \wedge \text{toothache})}{P(\text{toothache})} \\
 &= \frac{0.108 + 0.012}{0.108 + 0.012 + 0.016 + 0.064} = 0.6
 \end{aligned}$$

---

---

---

---

---

---

---

---

---

---

## Inference by Enumeration

- Consider a query on a single variable.
- Let  $X$  be the query variable,  $\mathbf{E}$  be the set of evidence variables,  $\mathbf{e}$  be the observed values for  $\mathbf{E}$ , and  $\mathbf{Y}$  be the remaining unobserved variables.
- Then we can evaluate the query  $\mathbf{P}(X \mid \mathbf{e})$  as
 
$$\mathbf{P}(X \mid \mathbf{e}) = \alpha \mathbf{P}(X, \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$
- Note we are summing over all possible values of the unobserved variables  $\mathbf{Y}$ .

---

---

---

---

---

---

---

---

---

---

## Inference by Enumeration

```
function ENUM-JOINT-ASK( $X$ ,  $e$ ,  $\mathbf{P}$ ) returns a distribution over  $X$ 
inputs:    $X$ , the query variable
           $e$ , the observed values for variables  $\mathbf{E}$ 
           $\mathbf{P}$ , the joint distribution on variables  $\{X\} \cup \mathbf{E} \cup \mathbf{Y}$ 

 $Q(X) \leftarrow$  a distribution over  $X$ , initially empty
for each value  $x_i$  of  $X$  do
   $Q(x_i) \leftarrow$  ENUM-JOINT( $x_i$ ,  $e$ ,  $\mathbf{Y}$ , [],  $\mathbf{P}$ )
return NORMALIZE( $Q(X)$ )

function ENUM-JOINT( $x$ ,  $e$ ,  $vars$ ,  $vals$ ,  $\mathbf{P}$ ) returns a real number
if EMPTY?( $vars$ ) then return  $\mathbf{P}(x, e, vals)$ 
 $Y \leftarrow$  FIRST( $vars$ )
return  $\sum_y$  ENUM-JOINT( $x, e, REST(vars), [y|vals], \mathbf{P}$ )
```

---

---

---

---

---

---

---

---

## Independence

- We say two propositions  $a$  and  $b$  are independent if any of the following hold.
  - $P(a \mid b) = P(a)$
  - $P(b \mid a) = P(b)$
  - $P(a \wedge b) = P(a) P(b)$
- Similarly, two variables  $X$  and  $Y$  are independent if any of the following hold.
  - $P(X \mid Y) = P(X)$
  - $P(Y \mid X) = P(Y)$
  - $P(X \wedge Y) = P(X) P(Y)$

---

---

---

---

---

---

---

---

## Conditional Independence

- We say two random variables  $X$  and  $Y$  are conditionally independent given a third random variable  $Z$  if any of the following hold:
  - $P(X, Y \mid Z) = P(X \mid Z) P(Y \mid Z)$
  - $P(X \mid Y, Z) = P(X \mid Z)$
  - $P(Y \mid X, Z) = P(Y \mid Z)$
- We will use this later to reduce the size of the joint distribution.

---

---

---

---

---


---

---

---

## Wumpus World (Again)

- Let's return to the Wumpus World and see how probabilities help with uncertainty.
- Consider the problem of determining the probability of each square having a pit, given three measurements on breeze ([1,1], [1,2], and [2,1]).
- In this case, we want a single Boolean variable  $P_{ij}$  for each square that is true *iff* square  $[i, j]$  contains a pit.
- We also have a single Boolean variable  $B_{ij}$  for each square that is true *iff* square  $[i, j]$  is breezy. We will only include these variables following an observation.


25  
Mountains & Minds


---

---

---

---

---


---

---

---

## Full Joint Distribution

- We now need to specify the full joint distribution  $\mathbf{P}(P_{1,1}, \dots, P_{4,4}, B_{1,1}, B_{1,2}, B_{2,1})$ .
- Using the product rule, we get
  - $\mathbf{P}(P_{1,1}, \dots, P_{4,4}, B_{1,1}, B_{1,2}, B_{2,1}) = \mathbf{P}(B_{1,1}, B_{1,2}, B_{2,1} \mid P_{1,1}, \dots, P_{4,4}) \mathbf{P}(P_{1,1}, \dots, P_{4,4})$ .
- The first term evaluates to 1.0 if the breezes are adjacent to the pits indicated and 0.0 otherwise.
- The second term is the prior probability of the pit configuration.


26  
Mountains & Minds


---

---

---

---

---


---

---

---

## Full Joint Distribution

- We will assume each square contains a pit with probability 0.2 (for the sake of discussion).
- We will also assume the assignment of a pit to a square is independent of any other pit.
- Then, for  $n$  pits,  $\mathbf{P}(P_{1,1}, \dots, P_{4,4}) = 0.2^n \times 0.8^{16-n}$ .
- More generally, the probability of any particular configuration is computed as

$$\mathbf{P}(P_{1,1}, \dots, P_{4,4}) = \prod_{i,j=1,1}^{4,4} \mathbf{P}(P_{i,j})$$

27  
Mountains & Minds


---

---

---

---

---

---

---

---

## Example

- We can use ENUM-JOINT-ASK to ask about square [1,3].
- Let  $b = \neg b_{1,1} \wedge b_{1,2} \wedge b_{2,1}$
- Let *Unknown* be the composite variable consisting of the  $P_{i,j}$  variables for squares other than the *Known* squares and the query square [1,3].
- Then we have
  - $\mathbf{P}(P_{1,3} \mid \text{Known}, b) = \alpha \sum_{\text{Unknown}} \mathbf{P}(P_{1,3} \mid \text{Unknown}, \text{Known}, b)$

---

---

---

---

---

---

---

---

## Example

- Recall that we have 16 squares, 12 of which are unknown.
- Therefore, for ENUM-JOINT-ASK, we need to add  $2^{12} = 4,096$  terms.
- In general, the summation will grow exponentially with the number of squares.
- Note, however, that the squares not adjacent to [1,3] and the observed squares are irrelevant.
- Let *Fringe* be the squares adjacent to known squares (other than the query).
- Let *Other* be the remaining unknown squares.

---

---

---

---

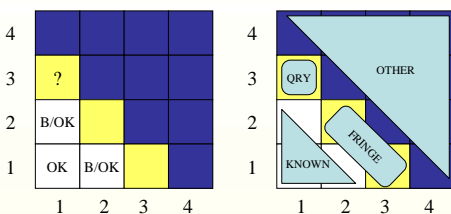
---

---

---

---

## Example




---

---

---

---

---

---

---

---

## Key Insight

- The key insight to exploit is the following:
  - The observed breezes are *conditionally independent* of the other variables, given the known, fringe, and query variables.
- Exploiting this, we get

$$\begin{aligned}
 P(P_{1,3} | \text{Known}, b) &= \alpha \sum_{\text{Unknown}} P(b | P_{1,3}, \text{Known}, \text{Unknown}) P(P_{1,3}, \text{Known}, \text{Unknown}) \\
 &= \alpha \sum_{\text{Fringe}} \sum_{\text{Other}} P(b | P_{1,3}, \text{Known}, \text{Fringe}, \text{Other}) P(P_{1,3}, \text{Known}, \text{Fringe}, \text{Other}) \\
 &= \alpha \sum_{\text{Fringe}} P(b | P_{1,3}, \text{Known}, \text{Fringe}) P(P_{1,3}, \text{Known}, \text{Fringe}, \text{Other})
 \end{aligned}$$

---

---

---

---

---

---

---

---

---

---

## Simplifying

- Note that now the first term does not depend on the other variables, so we can move the summation inwards.

$$P(P_{1,3} | \text{Known}, b) = \alpha \sum_{\text{Fringe}} P(b | P_{1,3}, \text{Known}, \text{Fringe}) \sum_{\text{Other}} P(P_{1,3}, \text{Known}, \text{Fringe}, \text{Other})$$

- Given the independence of the *Other* squares, we can simplify further.

$$\begin{aligned}
 P(P_{1,3} | \text{Known}, b) &= \alpha \sum_{\text{Fringe}} P(b | P_{1,3}, \text{Known}, \text{Fringe}) \sum_{\text{Other}} P(P_{1,3}) P(\text{Known}) P(\text{Fringe}) P(\text{Other}) \\
 &= \alpha P(\text{Known}) P(P_{1,3}) \sum_{\text{Fringe}} P(b | P_{1,3}, \text{Known}, \text{Fringe}) P(\text{Fringe}) \sum_{\text{Other}} P(\text{Other}) \\
 &= \alpha P(P_{1,3}) \sum_{\text{Fringe}} P(b | P_{1,3}, \text{Known}, \text{Fringe}) P(\text{Fringe})
 \end{aligned}$$

---

---

---

---

---

---

---

---

---

---

## Consistent Models

- So now consider the possibility that [1,3] has a pit. Determine  $P(\text{Fringe})$ .

3	P		
2	B/OK	P	
1	OK	B/OK	P
	1	2	3

0.2 x 0.2 = 0.04

3	P		
2	B/OK	P	
1	OK	B/OK	P
	1	2	3

0.2 x 0.8 = 0.16

3	P		
2	B/OK	P	
1	OK	B/OK	P
	1	2	3

0.8 x 0.2 = 0.16

---

---

---

---

---

---

---

---

---

---

## Consistent Models

- The consider the case where [1,3] does *not* have a pit. Determine  $P(\text{Fringe})$ .

3			
2	B/OK	P	
1	OK	B/OK	P
	1	2	3

$0.2 \times 0.2 = 0.04$

3			
2	B/OK	P	
1	OK	B/OK	
	1	2	3

$0.2 \times 0.8 = 0.16$

---

---

---

---

---

---

---

---

---

---

## Simplified

- The end result is a sum over the *Fringe* variables.
- Note that the *Fringe* consists only of  $P_{2,2}$ , and  $P_{3,1}$ .
- Thus we need to sum over the combinations of these two variables with  $b$ , i.e.,
  - $P(P_{1,3} | \text{Known}, b)$
  - $= \alpha' ((0.2 \times 0.2 \times 0.2 + 0.2 \times 0.2 \times 0.8 + 0.2 \times 0.8 \times 0.2),$
  - $(0.8 \times 0.2 \times 0.2 + 0.8 \times 0.2 \times 0.8))$
  - $= \alpha' (0.2 (0.04 + 0.16 + 0.16), 0.8 (0.04 + 0.16))$
  - $\approx (0.31, 0.69)$
- Interpreting, this means [1,3] (and [3,1] by symmetry) contains a pit with roughly 31% probability.
- Similarly,  $P(P_{2,2} | \text{Known}, b) \approx (0.86, 0.14)$ .

---

---

---

---

---

---

---

---

---

---

## Problem

- Recall, to use Bayes' Rule, you must have the entire joint probability distribution for all propositions in the domain.
- This is because  $E$  can be any set of evidence propositions.
- The *joint probability distribution* of a set of random variables  $W_i$  assigns probability  $\Pr(w_1, \dots, w_n)$  for all possible values of  $W_i$ .
- This requires  $2^n$  values!
- Conditional independence allows us to reduce the number of values required.

---

---

---

---

---

---

---

---

---

---

## Bayesian Networks

- **Def:** A *Bayesian belief network* is a graph in which
  - Nodes correspond to random variables
  - Directed links connect pairs of nodes and indicate a causal link or influence.
  - Each node has a conditional probability table that quantifies the effects parents of the node have on the node.
  - The graph is acyclic.

---

---

---

---

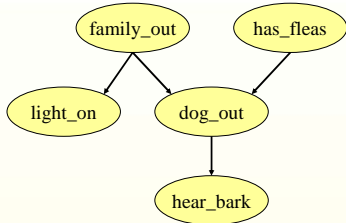
---

---

---

---

## Example Bayesian Net



---

---

---

---

---

---

---

---

## Topology

- The topology of the network captures independence assumptions:
  - A node, given its parents, is conditionally independent of the rest of the network.
- Probability distributions are defined using
  - Prior probabilities for root nodes (i.e., no parents).
  - Conditional probabilities giving all possible combinations of parents for non-root nodes.

---

---

---

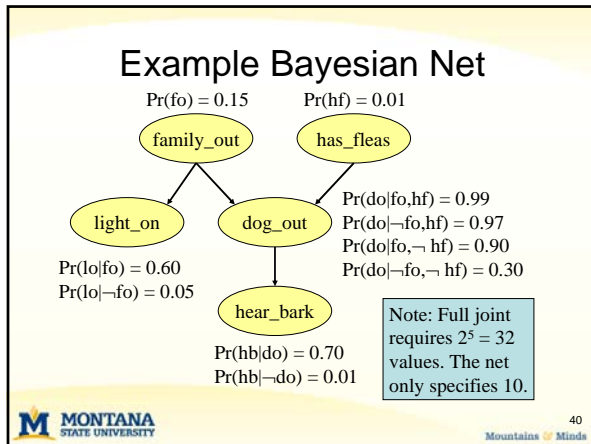
---

---

---

---

---




---

---

---

---

---

---

---

---

---

---

### Constructing Bayesian Networks

- Given a full joint probability distribution, rewrite it in terms of a conditional probability, using the product rule.
  - $P(x_1, \dots, x_n) = P(x_n | x_{n-1}, \dots, x_1) P(x_{n-1}, \dots, x_1)$
- Repeat this process, reducing each conjunctive probability to a conditional probability.
  - $P(x_1, \dots, x_n)$   
 $= P(x_n | x_{n-1}, \dots, x_1) P(x_{n-1} | x_{n-2}, \dots, x_1) \dots P(x_2 | x_1) P(x_1)$   
 $= \prod_i P(x_i | x_{i-1}, \dots, x_1)$
- Note, from conditional independence,
  - $P(x_i | x_{i-1}, \dots, x_1) = P(x_i | \text{Parents}(X_i))$ ,  
 provided  $\text{Parents}(X_i) \subseteq \{X_{i-1}, \dots, X_1\}$

MONTANA STATE UNIVERSITY
41 Mountains & Minds

---

---

---

---

---

---

---

---

---

---

### Constructing Bayesian Networks

- Choose the set of relevant variables  $X_i$  that describe the domain.
- Choose an ordering of the variables based on causal relationships.
- While variables remain, do
  - Pick a variable  $X_i$  and add a node to the net.
  - Set  $\text{Parents}(X_i)$  to minimal set of nodes already in net, satisfying conditional independence.
  - Define conditional probability table for  $X_i$ .

MONTANA STATE UNIVERSITY
42 Mountains & Minds

---

---

---

---

---

---

---

---

---

---

## Conditional Independence

- Consider how conditional independence is represented in the Bayesian network.
- The following provide two equivalent ways of interpreting conditional independence.
  - A node is conditionally independent of its non-descendants, given its parents.
  - A node is conditionally independent of all other nodes in the network, given its parents, children, and children's parents (i.e., given its *Markov blanket*).

---

---

---

---

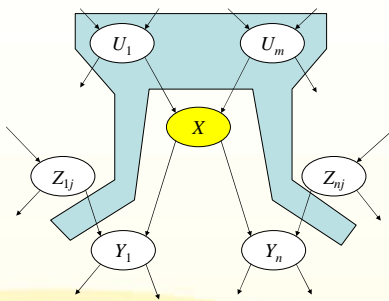
---

---

---

---

## Conditional Independence



---

---

---

---

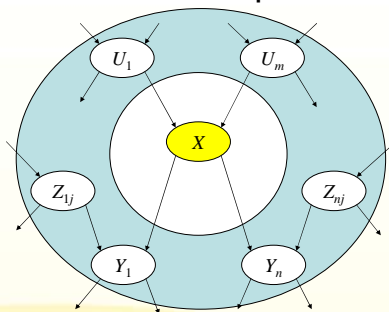
---

---

---

---

## Conditional Independence



---

---

---

---

---

---

---

---

## Direction-Dependent Separation

- We can define conditional independence further using the concept of  $d$ -separation.
- If every *undirected* path from a node in a set of nodes  $X$  to a node in a set of nodes  $Y$  is  $d$ -separated by a set of evidence  $E$ , then  $X$  and  $Y$  are conditionally independent given  $E$ .
- A set of evidence nodes,  $E$ ,  $d$ -separates two sets of nodes  $X$  and  $Y$  if every undirected path from a node in  $X$  to a node in  $Y$  is *blocked* given  $E$ .

---

---

---

---

---

---

---

---

## Path Blocking

- A path is blocked given a set of nodes  $E$  if there exists a node  $Z$  on the path for which one of the following holds:
  - $Z$  is in  $E$  and  $Z$  has one arrow on the path leading in and one arrow leading out.
  - $Z$  is in  $E$  and  $Z$  has both path arrows leading out.
  - Neither  $Z$  nor any descendant of  $Z$  is in  $E$ , and both path arrows lead in to  $Z$ .

---

---

---

---

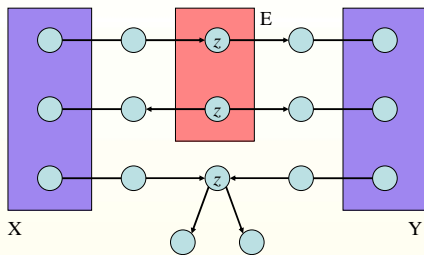
---

---

---

---

## Path Blocking




---

---

---

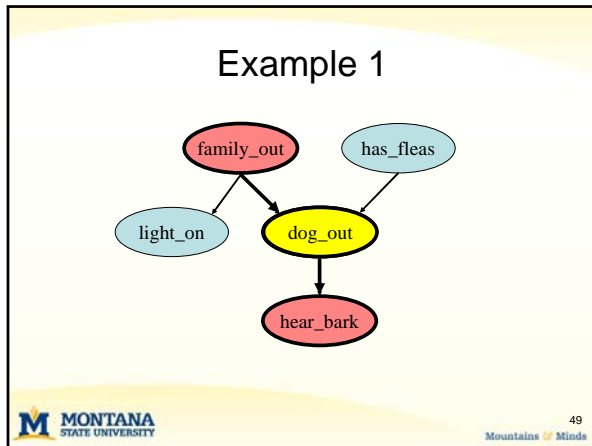
---

---

---

---

---




---

---

---

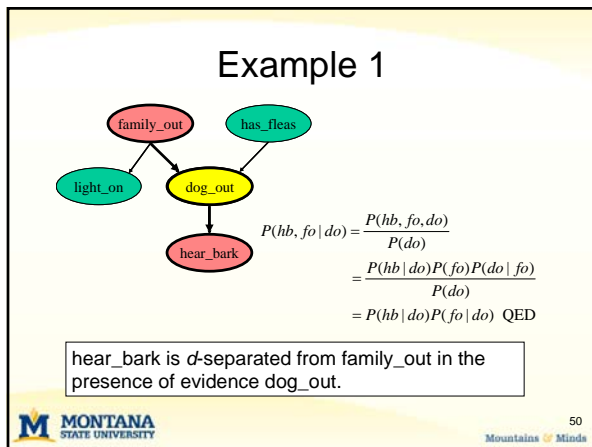
---

---

---

---

---




---

---

---

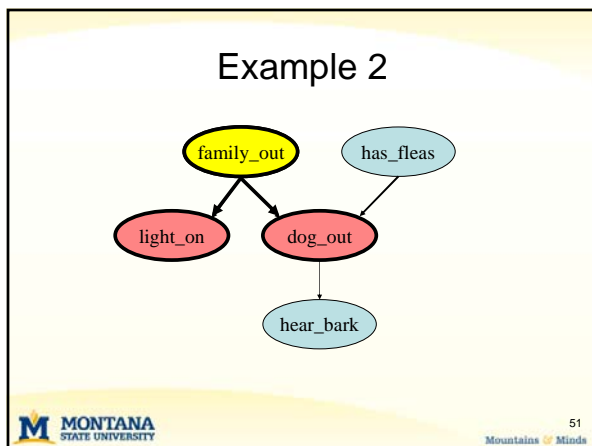
---

---

---

---

---




---

---

---

---

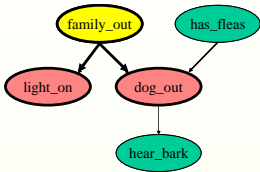
---

---

---

---

### Example 2



$$P(lo, do | fo) = P(lo | fo)P(do | fo)$$

QED

light\_on is  $d$ -separated from dog\_out in the presence of evidence family\_out.

---

---

---

---

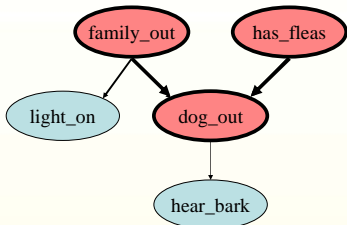
---

---

---

---

### Example 3




---

---

---

---

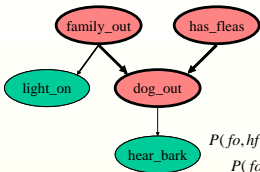
---

---

---

---

### Example 3



$$P(fo, hf, do) = P(fo)P(hf)P(do | fo, hf)$$

$$P(fo, hf) = P(fo)P(hf) \sum_{do} P(do | fo, hf)$$

$$= P(fo)P(hf) \quad \text{QED}$$

family\_out is  $d$ -separated from has\_fleas in the absence of evidence dog\_out.

---

---

---

---

---

---

---

---

## Inference in Bayesian Nets

- Diagnostic Inference: Infer from effect to cause, thus determining the causes.
- Causal Inference: Infer from cause to effect, thus determining how a cause is manifest.
- Intercausal Inference: Infer between causes of a common effect.
- Mixed Inference: Two or more of above.

---

---

---

---

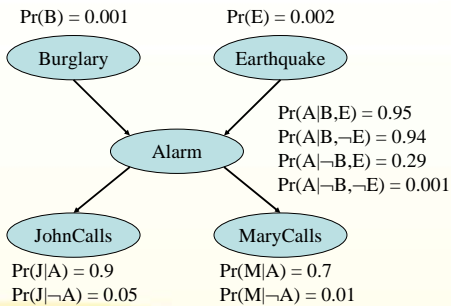
---

---

---

---

## Another Network




---

---

---

---

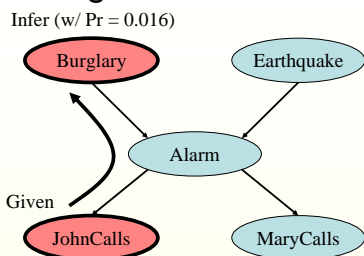
---

---

---

---

## Diagnostic Inference




---

---

---

---

---


---

---

---

## Some Applicable Rules

- We want  $P(B|J)$ .
- Using Baye's Rule, we get
  - $P(B|J) = P(J|B) P(B) / P(J)$
- To find  $P(J)$ 
  - $P(J) = P(J|-A) P(-A) + P(J|A) P(A)$
- To find  $P(A)$ 
  - $P(A) = P(A|-B) P(-B) + P(A|B) P(B)$
- To find  $P(A|B)$ 
  - $P(A|B) = P(A|B,-E) P(-E) + P(A|B,E) P(E)$
- To find  $P(J|B)$ 
  - $P(J|B) = P(J|-A) P(-A|B) + P(J|A) P(A|B)$


58 Mountains & Minds

---

---

---

---

---


---

---

---

## Computing $P(B|J)$

- Start with  $P(J|B)$
- This requires  $P(A|B)$ 
  - $P(A|B) = 0.94 \times 0.998 + 0.95 \times 0.002 = 0.94$
  - $P(A|-B) = 0.001 \times 0.998 + 0.29 \times 0.002 = 0.0016$
  - $P(J|B) = 0.05 \times 0.06 + 0.9 \times 0.94 = 0.85$
- To find  $P(J)$  we first need  $P(A)$ 
  - $P(A) = 0.0016 \times 0.999 + 0.94 \times 0.001 = 0.0025$
  - $P(J) = 0.05 \times 0.997 + 0.9 \times 0.0025 = 0.052$
- Using Baye's Rule, we get
  - $P(B|J) = 0.85 \times 0.001 / 0.052 = 0.016$


59 Mountains & Minds

---

---

---

---

---

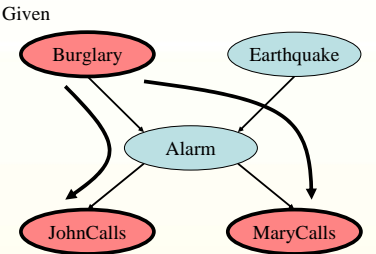
---

---

---

## Causal Inference


Given



```

    graph TD
      Burglary((Burglary)) --> Alarm((Alarm))
      Earthquake((Earthquake)) --> Alarm
      Alarm --> JohnCalls((JohnCalls))
      Alarm --> MaryCalls((MaryCalls))
      Burglary --> JohnCalls
      Earthquake --> MaryCalls
  
```

Infer (w/ Pr = 0.86)      Infer (w/ Pr = 0.67)


60 Mountains & Minds

---

---

---

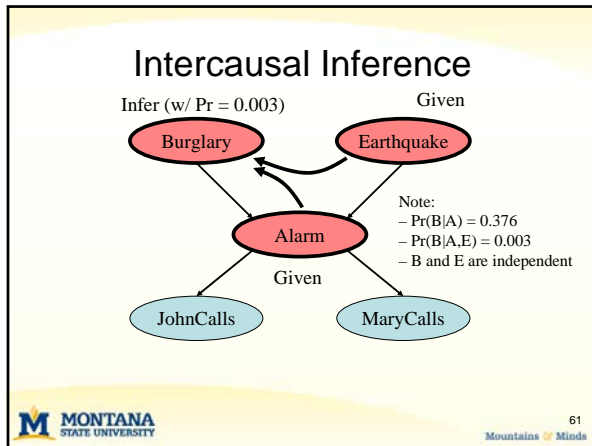
---

---

---

---

---




---

---

---

---

---

---

---

---

### Notation

$E_X^+$	causal support for $X$ —the variables above $X$ that are connected to $X$ through its parents.
$E_X^-$	evidential support for $X$ —the variables below $X$ that are connected to $X$ through its children.
$E_{U_i \setminus X}$	all the evidence connected to node $U_i$ except via the path from $X$ .
$E_{U_i \setminus X}^+$	all the evidence connected to node $U_i$ through its parents except via the path from $X$ .
$E_{U_i \setminus X}^-$	all the evidence connected to node $U_i$ through its children except via the path from $X$ .
$E_X = E_{X \setminus}$	the total evidence $E$ of $X$ (i.e., all evidence connected to $X$ ).

MONTANA STATE UNIVERSITY      Mountains & Minds      62

---

---

---

---

---

---

---

---

- ### Inference Process
- Express probability table  $\Pr(X|E)$  in terms of contributions of  $E_X^+$  and  $E_X^-$
  - Compute contribution of  $E_X^+$  by computing its effects on the parents of  $X$  and passing effect to  $X$  (recursive).
  - Compute contribution of  $E_X^-$  by computing its effects on the children of  $X$  and passing effect to  $X$  (recursive).
- MONTANA STATE UNIVERSITY      Mountains & Minds      63

---

---

---

---

---

---

---

---

## Main Function

```

function BELIEF-NET-ASK( $X$ )
returns a probability distribution over the values of  $X$ 

  inputs:  $X$ , a random variable (i.e. the query)

  SUPPORT-EXCEPT( $X$ , null);
  
```

---

---

---

---

---

---

---

---

---

---

## Function SUPPORT-EXCEPT

```

function SUPPORT-EXCEPT( $X$ ,  $V$ ) returns  $P(X | E_{X|V})$ 

  if EVIDENCE?( $X$ ) then return observed point distribution for  $X$ 
  else
    calculate  $P(E_{X|V}^- | X) = \text{EVIDENCE-EXCEPT}(X, V)$ 
     $U \leftarrow \text{PARENTS}(X)$ 
    if  $U$  is empty
      then return  $\alpha P(E_{X|V}^- | X) P(X)$ 
    else
      for each  $U_i$  in  $U$ 
        calculate and store
           $P(U_i | E_{U_i|X}) = \text{SUPPORT-EXCEPT}(U_i, X)$ 
      return  $\alpha P(E_{X|V}^- | X) \sum_u P(X | u) \prod_i P(U_i | E_{U_i|X})$ 
  
```

---

---

---

---

---

---

---

---

---

---

## Function EVIDENCE-EXCEPT

```

function EVIDENCE-EXCEPT( $X$ ,  $V$ ) returns  $P(E_{X|V}^- | X)$ 

   $Y \leftarrow \text{CHILDREN}(X) - V$ 
  if  $Y$  is empty
    then return a uniform distribution
  else
    for each  $Y_i$  in  $Y$  do
      calculate  $P(E_{Y_i}^- | Y_i) = \text{EVIDENCE-EXCEPT}(Y_i, \text{null})$ 
     $Z_i \leftarrow \text{PARENTS}(Y_i) - X$ 
    for each  $Z_j$  in  $Z_i$ 
      calculate
         $P(Z_j | E_{Z_j|Y_i}) = \text{SUPPORT-EXCEPT}(Z_j, Y_i)$ 
    return  $\beta \prod_i \sum_{y_i} P(E_{Y_i}^- | Y_i) \sum_{z_i} P(y_i | X, z_i) \prod_j P(z_j | E_{z_j|Y_i})$ 
  
```

---

---

---

---

---

---

---

---

---

---

## Key Expressions

SUPPORT-EXCEPT:

$$\alpha \mathbf{P}(E_{X|V}^- | X) \sum_{\mathbf{u}} \mathbf{P}(X | \mathbf{u}) \prod_i \mathbf{P}(U_i | E_{U_i|X})$$

Normalizer  
 From EVIDENCE-EXCEPT  
 Conditional Probability Table  
 From recursive call to SUPPORT-EXCEPT

67  
Mountains & Minds

---

---

---

---

---

---

---

---

## Key Expressions

EVIDENCE-EXCEPT:

$$\beta \prod_i \sum_{y_i} P(E_{Y_i}^- | y_i) \sum_{z_i} \mathbf{P}(y_i | X, z_i) \prod_j P(z_{ij} | E_{z_{ij}|Y_i})$$

Normalizer  
 From recursive call to EVIDENCE-EXCEPT  
 Conditional Probability Table  
 From SUPPORT-EXCEPT

68  
Mountains & Minds

---

---

---

---

---

---

---

---

## Limitation of Algorithm

- Only works on networks defined as *polytree*.
- A polytree is a DAG for which there exists at most one *undirected* path between any two nodes in the network.
- Solving multiply-connected networks is NP-Hard.

69  
Mountains & Minds

---

---

---

---

---

---

---

---

## General Exact Inference

- Recall the general query expression.
  - $P(X | \mathbf{e}) = \alpha \sum_y P(X, \mathbf{e}, y)$
- We used this to define ENUM-JOINT-ASK.
- Given a Bayesian network, we can answer a query by computing sums of products of conditional probabilities (ENUM-ASK).
- This algorithm can be used on *any* Bayesian network (whether polytree or not).
- Unfortunately, due to *NP*-hardness of BN inference, the algorithm is exponential in the size of the network.

---

---

---

---

---

---

---

---

## General Exact Inference

```
function ENUM-ASK( $X, \mathbf{e}, bn$ ) returns a distribution over  $X$   
inputs:  $X$ , the query variable  
        $\mathbf{e}$ , the observed values for variables  $\mathbf{E}$   
        $bn$ , a Bayesian net with variables  $\{X\} \cup \mathbf{E} \cup \mathbf{Y}$ 
```

```
 $Q(X) \leftarrow$  a distribution over  $X$ , initially empty  
for each value  $x_i$  of  $X$  do  
  extend  $\mathbf{e}$  with value  $x_i$  for  $X$   
   $Q(x_i) \leftarrow$  ENUM-ALL( $vars(bn), \mathbf{e}$ )  
return NORMALIZE( $Q(X)$ )
```

```
function ENUM-ALL( $vars, \mathbf{e}$ ) returns a real number  
if EMPTY?( $vars$ ) then return 1.0  
 $Y \leftarrow$  FIRST( $vars$ )  
if  $Y$  has value  $y$  in  $\mathbf{e}$   
  then return  $P(y | parents(Y)) \times$  ENUM-ALL( $REST(vars), \mathbf{e}$ )  
  else return  $\sum_y P(y | parents(Y)) \times$  ENUM-ALL( $REST(vars), \mathbf{e}_y$ )  
  where  $\mathbf{e}_y$  is  $\mathbf{e}$  extended with  $Y = y$ 
```

---

---

---

---

---

---

---

---

## Variable Elimination

- The enumeration idea can be improved.
- ENUM-JOINT-ASK has complexity  $O(n2^n)$  and does not use a Bayesian network.
- Using a Bayesian network, ENUM-ASK has complexity  $O(2^n)$  on *all* networks.
- We will now refine ENUM-ASK in a way to eliminate redundant calculations.
- The resulting algorithm, ELIM-ASK, has  $O(2^n)$  complexity in the worst case but has complexity  $O(n)$  on polytrees.

---

---

---

---

---

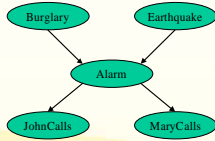
---

---

---

## Example

- The basic idea is to consider the query in terms of the previous form.
  - $P(X | \mathbf{e}) = \alpha \sum_y P(X, \mathbf{e}, y)$
- The elements are expanded and solved from right to left.
- As an example, consider the query  $P(\text{Burglary} | \text{JohnCalls}, \text{MaryCalls})$  on the network below.




---

---

---

---

---

---

---

---

## Example

- We can rewrite the query as follows:
  - $P(\text{Burglary} | \text{JohnCalls}, \text{MaryCalls})$ , we denote  $P(B | j, m)$ .
  - We capitalize  $B$  to remind us that it is our query variable.
  - $P(B | j, m) = \alpha P(B, j, m) = \alpha \sum_e \sum_a P(B, e, a, j, m)$ , where  $e$  denotes "earthquake" and  $a$  denotes "alarm."
  - Using the structure of the network, we get  $P(B | j, m) = \alpha \sum_e \sum_a P(B) P(e) P(a | B, e) P(j | a) P(m | a)$
  - Rearranging terms to move constants outside of sums:  $P(B | j, m) = \alpha P(B) \sum_e P(e) \sum_a P(a | B, e) P(j | a) P(m | a)$
  - Now consider the factors,  $M = P(m | a)$ ,  $J = P(j | a)$ ,  $A = P(a | B, e)$ ,  $E = P(e)$ , and  $B = P(B)$ , in this, right-to-left order.

---

---

---

---

---

---

---

---

## Example

- The factor for  $M$ ,  $P(m | a)$  does not require summing over  $M$  since  $m$  is fixed.
- Store probability, given each value of  $a$ , in a two-element vector.
  - $\mathbf{f}_M(A) = (P(m | a) P(m | \neg a))^T$
- Do the same for  $J$ 
  - $\mathbf{f}_J(A) = (P(j | a) P(j | \neg a))^T$
- The factor for  $A$  is  $P(a | B, e)$ , which is a  $2 \times 2 \times 2$  matrix,  $\mathbf{f}_A(A, B, E)$ .

---

---

---

---

---

---

---

---

## Example

- Next sum  $A$  out from the product of the three factors:
  - $f_{AJM}(B, E) = \sum_a f_A(a, B, E) \times f_J(a) \times f_M(a)$
  - $= f_A(a, B, E) \times f_J(a) \times f_M(a) + f_A(-a, B, E) \times f_J(-a) \times f_M(-a)$
- Then process  $E$  in the same way, summing  $E$  out
  - $f_{EAJM}(B) = f_E(e) \times f_{AJM}(B, e) + f_E(-e) \times f_{AJM}(B, -e)$
- Finally, compute the answer by multiplying the factor for  $B$  by the accumulated matrix.
  - $P(B | j, m) = \alpha f_B(B) \times f_{EAJM}(B)$
- All factor multiplications are via “pointwise product.”

---

---

---

---

---

---

---

---

---

---

## Pointwise Product

- Assume we have two factors,  $f_1$  and  $f_2$ .
- The “pointwise product” of these factors yield a new factor  $f = f_1 \times f_2$  determined as follows.
- The variables of the new factor  $f$  is the *union* of the variables in  $f_1$  and  $f_2$ .

A	B	$f_1(A, B)$	B	C	$f_2(B, C)$	A	B	C	$f(A, B, C)$
T	T	0.3	T	T	0.2	T	T	T	$0.3 \times 0.2$
T	F	0.7	F	T	0.6	T	T	F	$0.3 \times 0.8$
F	T	0.9	T	F	0.8	T	F	T	$0.7 \times 0.6$
F	F	0.1	F	F	0.4	T	F	F	$0.7 \times 0.4$
						F	T	T	$0.9 \times 0.2$
						F	T	F	$0.9 \times 0.8$
						F	F	T	$0.1 \times 0.6$
						F	F	F	$0.1 \times 0.4$

---

---

---

---

---

---

---

---

---

---

## Variable Elimination

```

function ELIM-ASK( $X, e, bn$ ) returns a distribution over  $X$ 
inputs:  $X$ , the query variable
        $e$ , the observed values for variables  $E$ 
        $bn$ , a Bayesian net specifying joint distribution  $P(X_1, \dots, X_n)$ 

factors ← []
vars ← REVERSE(VARS[bn])
for each var in vars do
    factors ← [MAKE-FACTORS(var, e) | factors]
    if var is a hidden variable then
        factors ← SUM-OUT(var, factors)
return NORMALIZE(POINTWISE-PRODUCT(factors))
    
```

---

---

---

---

---

---

---

---

---

---

## Multiply-Connected Networks

- Clustering
  - Transform network into probabilistically equivalent polytree by merging nodes in multiple paths. Exact solution. (Join Trees)
- Conditioning
  - Instantiate variables to definite values and evaluate polytree. Approximate solution.
- Stochastic Simulation
  - Generate several domain models at random and estimate probabilities. Approximate solution.

---

---

---

---

---

---

---

---