Uncertain knowledge and Reasoning

CS 332 Artificial Intelligence

Teaching Scheme **Credits: 2 + 1 = 3**

Theory: 3 Hrs / Week **Practical:** 2 Hrs / Week

- **Course Objectives:**

1) To understand the concept of Artificial Intelligence (AI)
2) To learn various peculiar search strategies for AI
3) To develop a mind to solve real world problems unconventionally with optimality

- **Course Outcomes:**

1) Identify and apply suitable Intelligent agents for various AI applications
2) Design smart system using different informed search / uninformed search or heuristic approaches.
3) Identify knowledge associated and represent it by ontological engineering to plan a strategy to solve given problem.

Syllabus

- Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Independence,
- Bayes’ Rule and Its Use, Representing Knowledge in an Uncertain Domain,
- The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Exact Inference in Bayesian Networks. Approximate inference in Bayesian Networks,
- Other Approaches to Uncertain reasoning-Fuzzy sets and Fuzzy logic.

Acting under uncertainty

- Agents may need to handle uncertainty, due to partial observability, nondeterminism, or a combination of the two. An agent may never know for certain what state it’s in or where it will end up after a sequence of actions.

- To act rationally under uncertainty we must be able to evaluate how likely are certain things.
- With FOL a 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.

Motivation

- Uncertainty arises through:
  - Noisy measurements
  - Finite size of data sets
  - Ambiguity: The word bank can mean (1) a financial institution, (2) the side of a river, or (3) tilting an airplane. Which meaning was intended, based on the words that appear nearby?
  - Limited Model Complexity
- Probability theory provides a consistent framework for the quantification and manipulation of uncertainty
- Allows us to make optimal predictions given all the information available to us, even though that information may be incomplete or ambiguous

Sample Space

- In probability theory, the sample space is also called as a sample description space or possibility space.
- The sample space (Ω) is the set of possible outcomes of an experiment or random trial. Sample Points ω in Ω are called sample outcomes, realizations, or elements. Subsets of Ω are called Events.

- **Example**. If we toss a coin twice then
  - Sample space (Ω) = {HH,HT, TH, TT}.
  - **The event that the first toss is heads is A** = {HH,HT}

- We say that events A1 and A2 are disjoint (mutually exclusive) if Ai ∩ Aj = {}
  - Example: first flip being heads and first flip being tails

A set Ω with outcomes S_1,S_2…..S_n, must meet some conditions in order to be a sample space:

- The outcomes must be **mutually exclusive**, i.e. if S_j occurs, then no other S_i will take place,
- The outcomes must be **collectively exhaustive**, i.e. on every experiment (or random trial) there will always take place some outcome for The sample space Ω must have the **right granularity** depending on what the experimenter is interested in relevant information must be removed from the sample space and the right abstraction must be chosen

$$\forall i, j = 1, 2, \ldots, n \quad i \neq j$$

$$s_i \in \Omega \quad i \in \{1, 2, \ldots, n\}$$

Probability

- A **probability** is a number that reflects the chance or likelihood that a particular event will occur.
- **Probabilities** can be expressed as proportions that range from 0 to 1, and they can also be expressed as percentages ranging from 0% to 100%.
- A probability of 0 indicates that there is no chance that a particular event will occur, whereas a probability of 1 indicates that an event is certain to occur.
- A probability of 0.45 (45%) indicates that there are 45 chances out of 100 of the event occurring.

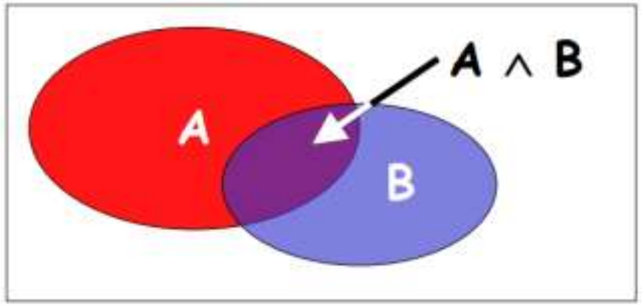
Axioms of Probability

- We will assign a real number P(A) to every event A, called the probability of A.
- To qualify as a probability, P must satisfy three axioms:
  - Axiom 1: $P(A) \geq 0$ for every A
  - Axiom 2: $P(\Omega) = 1$
  - Axiom 3: If A1,A2, . . . are disjoint then

$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i)$$

- The probability of disjunction is:

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$$

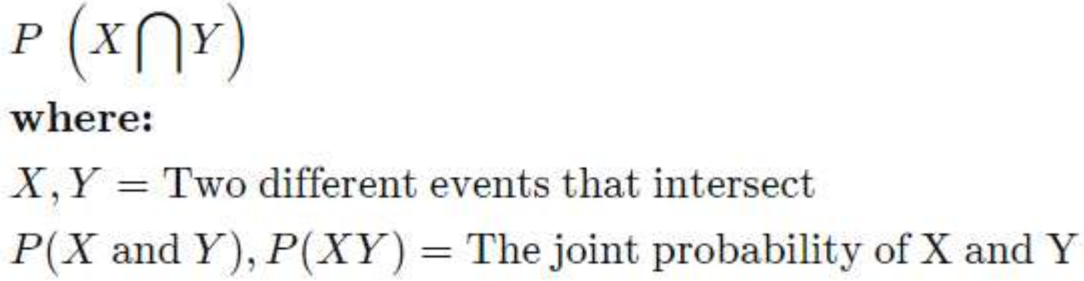


Joint and Conditional Probabilities

- Joint Probability:
  - Joint probability is a statistical measure that calculates the likelihood of two events occurring together and at the same point in time.
  - Joint probability is the probability of event Y occurring at the same time that event X occurs.
- P(X,Y)
- Probability of X and Y

The Formula for Joint Probability:

Notation for joint probability can take a few different forms. The following formula represents the probability of events intersection:



$$P\left(X \bigcap Y\right)$$

where:

$X, Y =$ Two different events that intersect

$P(X \text{ and } Y), P(XY) =$ The joint probability of X and Y

Independent and Conditional Probabilities

- Assuming that P(B) > 0, the **conditional** probability of A given B:
- $P(A|B)=P(AB)/P(B)$
- $P(AB) = P(A|B)P(B) = P(B|A)P(A)$
  - Product Rule

- Two events A and B are **independent** if
- $P(AB) = P(A)P(B)$
  - Joint = Product of Marginals

- Two events A and B are **conditionally independent** given C if they are independent after conditioning on C
- $P(AB|C) = P(B|AC)P(A|C) = P(B|C)P(A|C)$

Unconditional Probability

- If we select a child at random (by simple random sampling), then each child has the same probability (equal chance) of being selected, and the probability is 1/N, where N=the population size.
- Thus, the probability that any child is selected is 1/5,290 = 0.0002.
- In most sampling situations we are generally not concerned with sampling a specific individual but instead we concern ourselves with the probability of sampling certain types of individuals.
- For example, what is the probability of selecting a boy or a child 7 years of age? The following formula can be used to compute probabilities of selecting individuals with specific attributes or characteristics.

Conditional Probability:

- One can be interested in the probability of an event given the occurrence of another event.
- The probability of one event given the occurrence of another event is called the conditional probability. The conditional probability of one to one or more random variables is referred to as the conditional probability distribution.
- P(X|Y)
  - The above statement can be interpreted as:
    - Probability of X given Y

Example

- 60% of ML students pass the final and 45% of ML students pass both the final and the midterm
- What percent of students who passed the final also passed the midterm?

Example

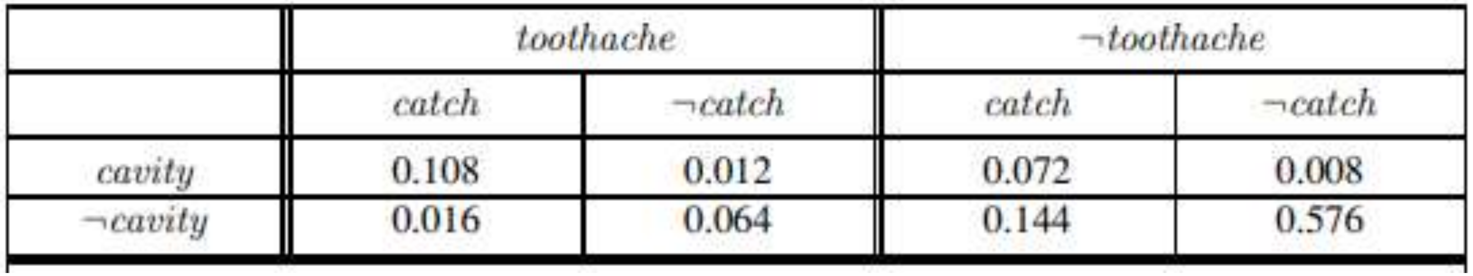
- 60% of ML students pass the final and 45% of ML students pass both the final and the midterm
- What percent of students who passed the final also passed the midterm?

- Reworded: What percent of students passed the midterm given they passed the final?
- P(M|F) = P(M,F) / P(F)
  - = 0.45 / 0.60
  - = 0.75

Inference using full joint distribution

- **Probabilistic inference:** The computation of posterior probabilities for query propositions given observed evidence.
- We use the full joint distribution as the “knowledge base” from which answers to all questions may be derived.
- The **full joint probability distribution** specifies the probability of each complete assignment of values to random variables.
  - It is usually too large to create or use in its explicit form,
  - However, when it is available it can be used to answer queries simply by adding up entries for the possible worlds corresponding to the query propositions.

- **Example**: A domain consisting of just the three Boolean variables Toothache, Cavity, and Catch (the dentist’s nasty steel probe catches in my tooth).



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

- **Marginalization / summing out:** The process of extracting the distribution over some subset of variables or a single variable (to get the **marginal probability**), by summing up the probabilities for each possible value of the other variables, thereby taking them out of the equation.
- The general **marginalization rule** for any sets of variables **Y** and **Z**:

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

- EX: The marginal probability of cavity is P(cavity) = 0.108+0.012+0.072+0.008

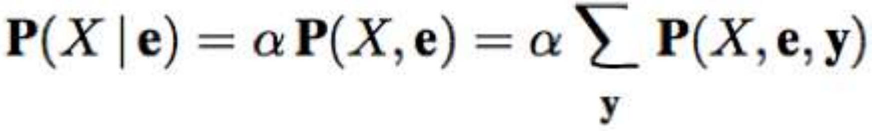
- **The conditioning rule:**

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

- Ex:Just to check, we can also compute the probability that there is no cavity, given a toothache:
- P(¬cavity |toothache) = P(¬cavity ∧ toothache) P(toothache) = 0.016 + 0.064 0.108 + 0.012 + 0.016 + 0.064 = 0.4

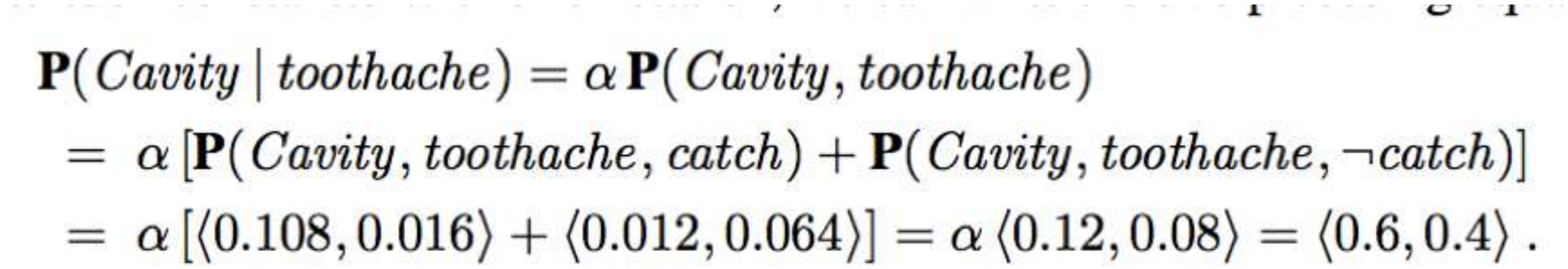
- **Normalization constant** (α): 1/P(evidence) can be viewed as a normalization constant for the distribution P(event | evidence), ensuring that it adds up to 1.

- X (Cavity) is a single variable. Let **E** (Toothache) be the list of evidence variables, **e** be the list of observed values for them, **Y** (Catch) be the remaining unobserved variables. The query **P**(X|e) can be evaluated as:



$$\mathbf{P}(X \mid \mathbf{e}) = \alpha\, \mathbf{P}(X, \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$

- Ex:



$$\begin{aligned}
\mathbf{P}(Cavity \mid toothache) &= \alpha\, \mathbf{P}(Cavity, toothache) \\
&= \alpha\, [\mathbf{P}(Cavity, toothache, catch) + \mathbf{P}(Cavity, toothache, \neg catch)] \\
&= \alpha\, [\langle 0.108, 0.016\rangle + \langle 0.012, 0.064\rangle] = \alpha\, \langle 0.12, 0.08\rangle = \langle 0.6, 0.4\rangle\,.
\end{aligned}$$

Drawbacks of inference by enumeration:

- The worst-case time complexity is O(d^n),
  - where d is the number of values in domains of each random variable
  - To store full joint probability distribution we need O(d^n) space

Independence

- A and B are independent iff:

$$P(A \mid B) = P(A)$$

$$P(B \mid A) = P(B)$$

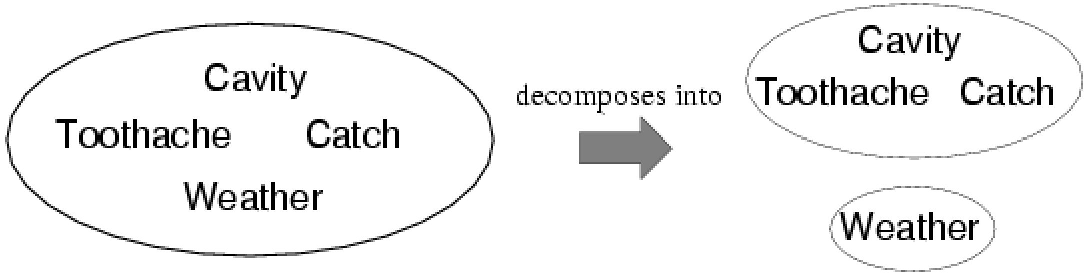
These two constraints are logically equivalent

- Therefore, if A and B are independent:

$$P(A \mid B) = \frac{P(A \wedge B)}{P(B)} = P(A)$$

$$P(A \wedge B) = P(A)P(B)$$

- **Independence** assertions can dramatically reduce the amount of information necessary to specify the full joint distribution.
- If the complete set of variables can be divided into independent subsets, then the full joint distribution can be factored into separate joint distributions on those subsets.



Weather is independent of Cavity, Toothache, catch

- **Absolute independence** between subsets of random variables allows the full joint distribution to be factored into smaller joint distributions, greatly reducing its complexity. Absolute independence seldom occurs in practice.

Bayes’ Rule

It is a mathematical formula for determining conditional probability

$P(A|B) = P(AB) /P(B)$ (Conditional Probability)

$P(A|B) = P(B|A)P(A) /P(B)$ (Product Rule)

$P(A|B) = P(B|A)P(A) / \Sigma P(B|A)P(A)$ (Law of Total Probability)

$$P(A|B) = \frac{P(A)\,P(B|A)}{P(B)}$$

$$P(B) = \sum_j P(B \mid A_j)P(A_j)$$

Key terms in Bayes Rule

- **Prior Probability:**
  - The **prior probability** of a **random event** is the **unconditional probability** that is assigned before any relevant evidence is taken into account.
- **Posterior Probability**:
  - A posterior probability, in Bayesian statistics, is the revised or updated probability of an event occurring after taking into consideration new information. The posterior probability is calculated by updating the **prior probability** using **Bayes' theorem**.
- **Likelihood**: Likelihood refers to finding the best distribution of the data given a particular value of some feature or some situation in the data.

Bayes’ Rule

$$P(A|B) = \frac{P(A)\,P(B|A)}{P(B)}$$

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}.$$

$$\text{Posterior} = \frac{\text{Likelihood} * \text{Prior}}{\text{Evidence}}$$

$$\text{Posterior probability} \propto \text{Likelihood} \times \text{Prior probability}$$

Example

- Suppose you have tested positive for a disease; what is the probability that you actually have the disease?
- It depends on the accuracy and sensitivity of the test, and on the background (prior) probability of the disease.
- $P(T=1|D=1) = 0.95$ (true positive)
- $P(T=1|D=0) = 0.10$ (false positive)
- $P(D=1) = 0.01$ (prior)

- $P(D=1|T=1) = ?$

Example

- $P(T=1|D=1) = .95$ (true positive)
- $P(T=1|D=0) = .10$ (false positive)
- $P(D=1) = .01$ (prior)

Bayes’ Rule

- $P(D|T) = P(T|D)P(D) / P(T)$

$= .95 * .01 / .1085$

$= .087$

Law of Total Probability

- $P(T) = \Sigma\ P(T|D)P(D)$

$= P(T|D=1)P(D=1) + P(T|D=0)P(D=0)$

$= .95*.01 + .1*.99$

$= .1085$

The probability that you have the disease given you tested positive is 8.7%

Applying Bayes' rule: The simple case

- It allows us to compute the single term P(b | a) in terms of three terms: P(a | b), P(b), and P(a).
- That seems like two steps backwards, but Bayes' rule is useful in practice because there are many cases where we do have good probability estimates for these three numbers and need to compute the fourth.
- Often, we perceive as evidence the effect of some unknown cause and we would like to determine that cause.
- In that case, Bayes' rule becomes
  - P(cause | effect) = (P(effect | cause)P(cause))/P(effect) .

- The conditional probability P(effect | cause) quantifies the relationship in the **causal direction**, whereas P(cause | effect) describes the **diagnostic direction.**
- **Ex:** A doctor knows that the disease meningitis causes the patient to have a stiff neck, say, 70% of the time. The doctor also knows some unconditional facts: the prior probability that a patient has meningitis is 1/50,000, and the prior probability that any patient has a stiff neck is 1%. Letting s be the proposition that the patient has a stiff neck and m be the proposition that the patient has meningitis, we have
- $P(s \mid m)=0.7$, $P(m)=1/50000$, $P(s)=0.01$, $P(m \mid s) =(P(s \mid m)P(m)) /P(s) = 0.7 \times 1/50000 /0.01 = 0.0014$.
- That is, we expect less than 1 in 700 patients with a stiff neck to have meningitis.

Use of Bayes Rule: Combining evidence

- We have seen that Bayes’ rule can be useful for answering probabilistic queries conditioned on one piece of evidence
- What happens when we have two or more pieces of evidence?
  - For example, what can a dentist conclude if her nasty steel probe catches in the aching tooth of a patient? If we know the full joint distribution (Figure 13.3), we can read off the answer: P(Cavity |toothache ∧ catch) = α <0.108, 0.016> ==<0.871, 0.129>
- We know, however, that such an approach does not scale up to larger numbers of variables. We can try using Bayes’ rule to reformulate the problem:
- P(Cavity |toothache ∧ catch) = α P(toothache ∧ catch | Cavity) P(Cavity) .

- For the previous reformulation to work, there is the need to know the conditional probabilities of the conjunction (toothache ∧catch) for each value of Cavity.

Representing Knowledge in an Uncertain Domain

- We have learned knowledge representation using first-order logic and propositional logic with certainty, which means we were sure about the predicates. With this knowledge representation, we might write A→B, which means if A is true then B is true,
- Consider a situation where we are not sure about whether A is true or not then we cannot express this statement, this situation is called uncertainty.
- So to represent uncertain knowledge, where we are not sure about the predicates, we need uncertain reasoning or probabilistic reasoning.

Causes of uncertainty:

- Following are some leading causes of uncertainty to occur in the real world.
    - Information occurred from unreliable sources.
    - Experimental Errors
    - Equipment fault
    - Temperature variation
    - Climate change.

Probabilistic reasoning:

- Probabilistic reasoning is a way of knowledge representation where we apply the concept of probability to indicate the uncertainty in knowledge.
- In probabilistic reasoning, we combine probability theory with logic to handle the uncertainty.
- We use probability in probabilistic reasoning because it provides a way to handle the uncertainty that is the result of someone's laziness and ignorance.
- In the real world, there are lots of scenarios, where the certainty of something is not confirmed, such as "It will rain today," "behavior of someone for some situations," "A match between two teams or two players." These are probable sentences for which we can assume that it will happen but not sure about it, so here we use probabilistic reasoning.

Need of probabilistic reasoning in AI:

- When there are unpredictable outcomes.
- When specifications or possibilities of predicates becomes too large to handle.
- When an unknown error occurs during an experiment.

In **probabilistic reasoning**, there are two ways to solve problems with uncertain knowledge:

- **Bayes' rule**
- **Bayesian Statistics**

Introduction to Bayesian Network:

- A method to represent dependencies among variables and specify more concisely any full joint probability distribution is another data structure, called Bayesian network.
- This is another method of reducing the complexity of certainty factors. It is a directed graph in which each node is annotated with quantitative probability information.

Bayesian network

A Bayesian network is a directed graph in which each node is annotated with quantitative probability information. The full specification is as follows:

- Each node corresponds to a random variable, which may be discrete or continuous.
- A set of directed links or arrows connects pairs of nodes. If there is an arrow from node X to node Y , X is said to be a parent of Y. The graph has no directed cycles (and hence is a directed acyclic graph, or DAG.
- Each node X_i has a conditional probability distribution $\mathbf{P(X_i \mid Parents(X_i))}$that quantifies the effect of the parents on the node.

Introduction to BN Contd..

- The **topology of the network**—the set of nodes and links, specifies the conditional independence relationships which hold in the domain.
- The intuitive meaning of an arrow in a properly constructed network is usually that X has a direct influence on Y.
- After the topology of the Bayesian network is laid down
  - we need to specify a conditional probability distribution for each variable, given its parents.
  - Then, the combination of the topology and the conditional distributions suffices to specify (implicitly) the full joint distribution for all the variables

Bayesian networks

- A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions

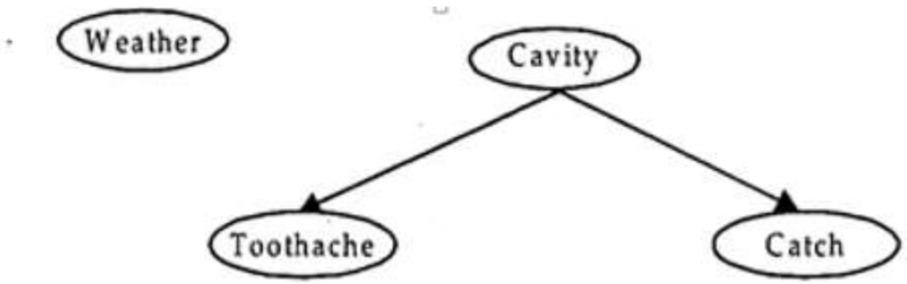
- Syntax:
  - a set of nodes, one per variable
  - a directed, acyclic graph (link ≈ "directly influences")
  - a conditional distribution for each node given its parents:

$$\mathbf{P}\,(X_i \mid \text{Parents}\,(X_i))$$

- In the simplest case, conditional distribution represented as a conditional probability table (CPT) giving the distribution over X_i for each combination of parent values
- A node is independent of its nondescendents given its parents.

Example

- Take the world consisting of simple variables Toothache, Cavity, Catch and Weather. Assuming that weather has no influence on toothache, weather is independent of the other variables; toothache and catch are conditionally independent, given a cavity.
- Topology of network encodes conditional independence assertions:



Bayesian Network: Weather variable is independent from other three variables. Toothache and catch are conditional independent given cavity.

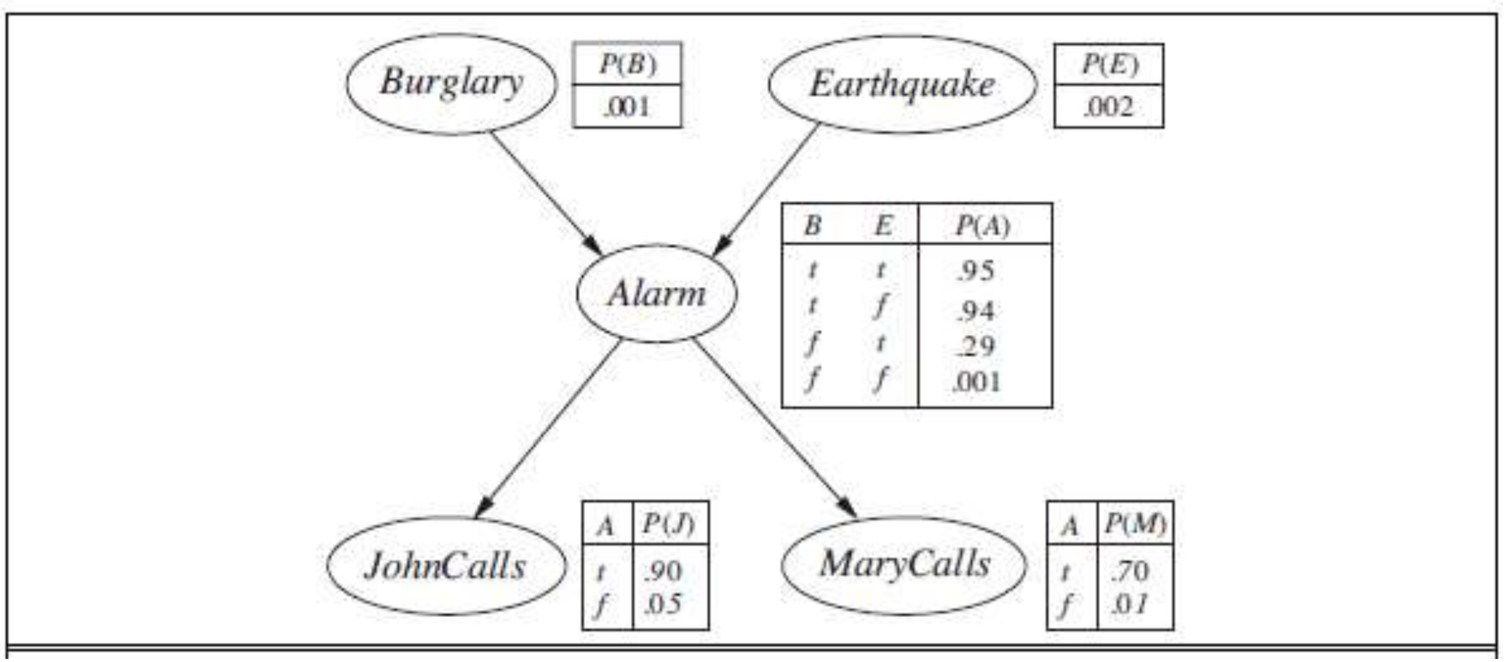
Example

- I'm at work, neighbor Jhon 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*, *MaryCalls*, *JhonCalls*

- 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

Example contd.



P(B)
.001

P(E)
.002

B	E	P(A)
t	t	.95
t	f	.94
f	t	.29
f	f	.001

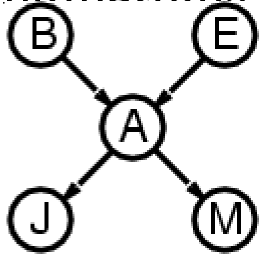
A	P(J)
t	.90
f	.05

A	P(M)
t	.70
f	.01

A typical Bayesian network, showing both the topology and the conditional probability tables (CPTs). In the CPTs, the letters B, E, A, J, and M stand for Burglary, Earthquake, Alarm, JohnCalls, and MaryCalls , respectively.

Compactness

- A CPT for Boolean X_i with k Boolean parents has 2^k rows for the combinations of parent values



- Each row requires one number p for $X_i = true$
(the number for $X_i = false$ is just $1-p$)

- If each variable has no more than k parents, the complete network requires $O(n \cdot 2^k)$ numbers

- I.e., grows linearly with n, vs. $O(2^n)$ for the full joint distribution

- For burglary net, $1 + 1 + 4 + 2 + 2 = 10$ numbers (vs. $2^5-1 = 31$)

Semantics of Bayesian Networks:

- **There are two ways in which we can understand Semantics of Bayesian networks:**
- 1. See the network as representation of the joint probability distribution. This is useful in understanding how to construct networks.
- 2. See the networks as an encoding of a collection of conditional independence statements. This is useful in designing inference procedures. However, the two ways are equivalent.

Representing the Full Joint Distribution:

- We explain it, by calculating the probability that the alarm has sounded, but neither the burglary nor an earthquake has occurred, and both Mary and Jhon telephone you.

$$\begin{aligned} P(j \wedge m \wedge a \wedge \neg b \neg e) &= P(j|a)\, P(m\,|a)\, P(a\,|\neg b \wedge \neg e)\, P(\neg b)\, P(\neg e) \\ &= 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998 \\ &= 0.00062 \end{aligned}$$

- We can generalize the example by writing: $P(x_1, \ldots, x_n) = \prod_{j=1}^{n} P(x_i\,|\,parents\,(x_i))$

- Where parents (x) denotes the specific values of the variables in the parents (x).
- If the Bayesian network is a representation of the joint distribution then it too can be used to answer any query, as earlier in the case of inference through probabilities. This does it really so and that too more efficiently. An extension of Bayesian net work is called Decision Net Work or Influence Diagram.

Constructing Bayesian networks

- 1. Choose an ordering of variables $X_1, \dots, X_n$
- 2. For $i = 1$ to n
  - add X_i to the network
  - select parents from $X_1, \dots, X_{i-1}$ such that

$$\mathbf{P}(X_i \mid Parents(X_i)) = \mathbf{P}(X_i \mid X_1, \dots X_{i-1})$$

This choice of parents guarantees:

$$\begin{aligned} \mathbf{P}(X_1, \dots, X_n) &= \pi_{i=1}^{n} \mathbf{P}(X_i \mid X_1, \dots, X_{i-1}) && \text{(chain rule)} \\ &= \pi_{i=1}^{n} \mathbf{P}(X_i \mid Parents(X_i)) && \text{(by construction)} \end{aligned}$$

The process of creating Network structure

- **Adding MaryCalls**: No parents.
- **Adding JohnCalls:** If Mary calls, that probably means the alarm has gone off, which of course would make it more likely that John calls. Therefore, JohnCalls needs MaryCalls as a parent.
- **Adding Alarm**: Clearly, if both call, it is more likely that the alarm has gone off than if just one or neither calls, so we need both MaryCalls and JohnCalls as parents.
- **Adding Burglary:** If we know the alarm state, then the call from John or Mary might give us information about our phone ringing or Mary's music, but not about burglary:

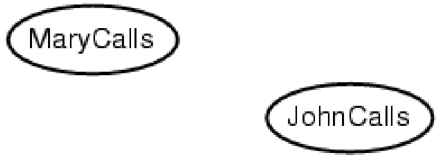
P(Burglary | Alarm, JohnCalls ,MaryCalls) = P(Burglary | Alarm) .

Hence we need just Alarm as parent.

- **Adding Earthquake:** If the alarm is on, it is more likely that there has been an earthquake.
- (The alarm is an earthquake detector of sorts.) But if we know that there has been a burglary, then that explains the alarm, and the probability of an earthquake would be only slightly above normal. Hence, we need both Alarm and Burglary as parents

Example

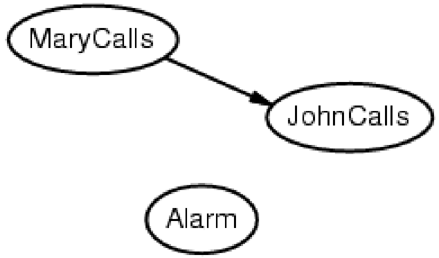
- Suppose we have MaryCalls, JohnCalls, Alarm, Burglary, Earthquake attributes
- Suppose we choose the ordering *M*, *J*, *A*, *B*, *E*



P$(J \mid M) =$ ***P***(J)?

Example

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

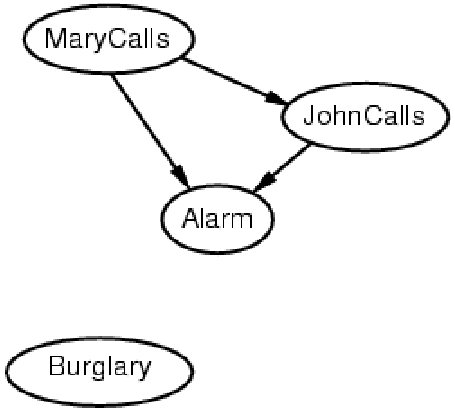


$\boldsymbol{P}(J \mid M) = \boldsymbol{P}(J)$ **No**

$\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A \mid J)$? $\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A)$?

Example

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



$\mathbf{P}(J \mid M) = \mathbf{P}(J)$ **No**

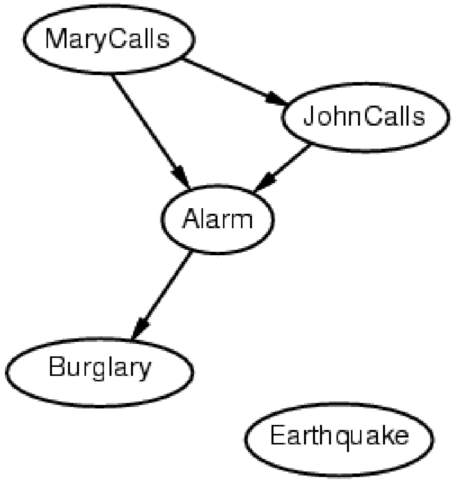
$\mathbf{P}(A \mid J, M) = \mathbf{P}(A \mid J)$? $\mathbf{P}(A \mid J, M) = \mathbf{P}(A)$? **No**

$\mathbf{P}(B \mid A, J, M) = \mathbf{P}(B \mid A)$?

$\mathbf{P}(B \mid A, J, M) = \mathbf{P}(B)$?

Example

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



P*(J | M)* = ***P****(J)* **No**

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

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

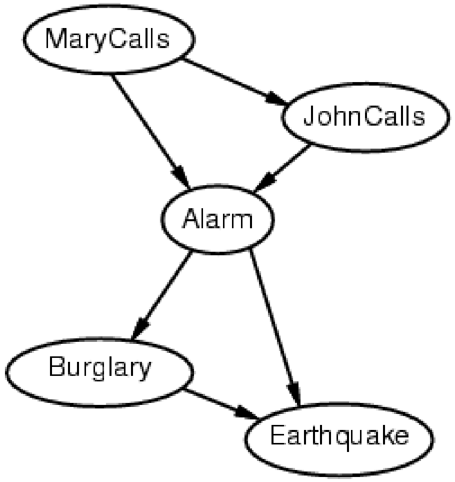
P*(B | A, J, M)* = ***P****(B)*? **No**

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

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

Example

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



P(*J* | *M*) = ***P***(*J*) **No**

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

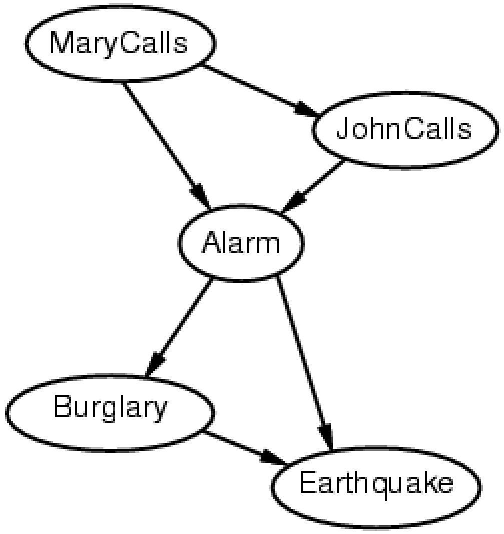
P(*B* | *A*, *J*, *M*) = ***P***(*B* | *A*)? **Yes**

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

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

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

Example contd.



- Deciding conditional independence is hard in noncausal directions
- (Causal models and conditional independence seem hardwired for humans!)
- Network is less compact: 1 + 2 + 4 + 2 + 4 = 13 numbers needed

Some Applications of BN

- Medical diagnosis
- Troubleshooting of hardware/software systems
- Fraud/uncollectible debt detection
- Data mining
- Analysis of genetic sequences
- Data interpretation, computer vision, image understanding

EFFICIENT REPRESENTATION OF CONDITIONAL DISTRIBUTIONS

- Even if the maximum number of parents k is smallish, filling in the CPT for a node requires up to $O(2^k)$ numbers and perhaps a great deal of experience with all the possible conditioning cases.
- In fact, this is a worst-case scenario in which the relationship between the parents and the child is completely arbitrary. Usually, such relationships are describable by a **canonical distribution** that fits some standard pattern. In such cases, the complete table can be specified by naming the pattern and perhaps supplying a few parameters—much easier than supplying an exponential number of parameters.

- The simplest example is provided by **deterministic nodes**. A deterministic node has its value specified exactly by the values of its parents, with no uncertainty.
- Uncertain relationships can often be characterized by so-called **noisy** logical relation ships.
- A conditional distribution is a probability distribution for a **sub-population**. In other words, it shows the probability that a randomly selected item in a sub-population has a characteristic you're interested in.
- For example, if you are fever (the population) you might want to know how many people have Malaria (the sub-population).

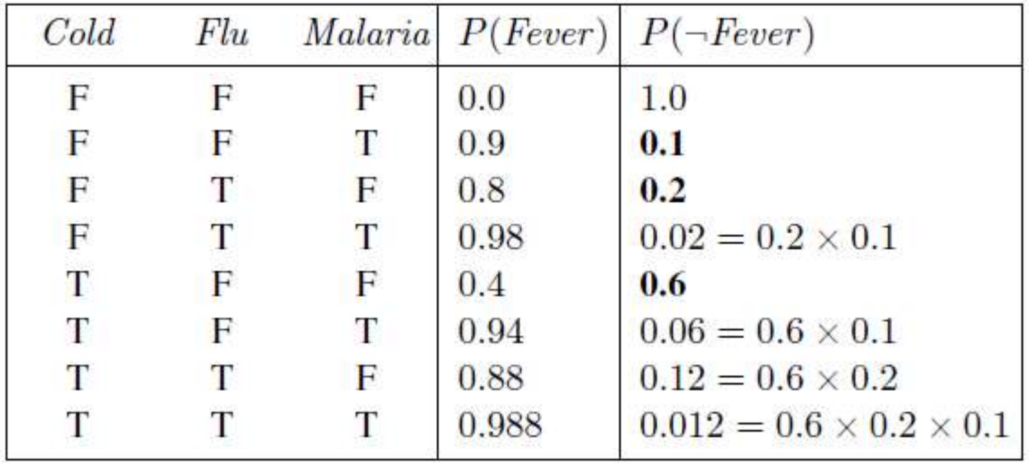
Let us suppose these individual inhibition probabilities are as follows:

$q_{cold} = P(\neg\text{fever} \mid \text{cold}, \neg\text{flu}, \neg\text{malaria}) = 0.6$,
$q_{flu} = P(\neg\text{fever} \mid \neg\text{cold}, \text{flu}, \neg\text{malaria}) = 0.2$,
$q_{malaria} = P(\neg\text{fever} \mid \neg\text{cold}, \neg\text{flu}, \text{malaria}) = 0.1$.

Then, from this information and the noisy-OR assumptions, the entire CPT can be built. The general rule is that

$$P(x_i \mid parents(X_i)) = 1 - \prod_{\{j:X_j = true\}} q_j \;,$$

where the product is taken over the parents that are set to true for that row of the CPT. The following table illustrates this calculation:

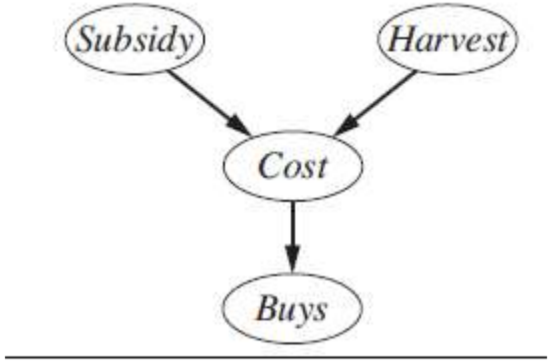


Cold	*Flu*	*Malaria*	$P(Fever)$	$P(\neg Fever)$
F	F	F	0.0	1.0
F	F	T	0.9	**0.1**
F	T	F	0.8	**0.2**
F	T	T	0.98	$0.02 = 0.2 \times 0.1$
T	F	F	0.4	**0.6**
T	F	T	0.94	$0.06 = 0.6 \times 0.1$
T	T	F	0.88	$0.12 = 0.6 \times 0.2$
T	T	T	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

Bayesian nets with continuous variables

- Many real-world problems involve continuous quantities, such as height, mass, temperature, and money; in fact, much of statistics deals with random variables whose domains are continuous.
- Continuous variables have an infinite number of possible values, so it is impossible to specify conditional probabilities explicitly for each value.
- One possible way to handle continuous variables is to avoid them by using **discretization i.e.** dividing up possible values into a fixed set of intervals.

For example, temperatures could be divided into (<0₀C), (0₀C−100₀C), and (>100₀C). Discretization is sometimes an adequate solution, but often results in a considerable loss of accuracy and very large CPTs.



A network with both discrete and continuous variables is called a hybrid Bayesian HYBRID BAYESIAN network.

In the example, Subsidy and Harvest is discrete variable and Cost and Buys are the continuous variable.

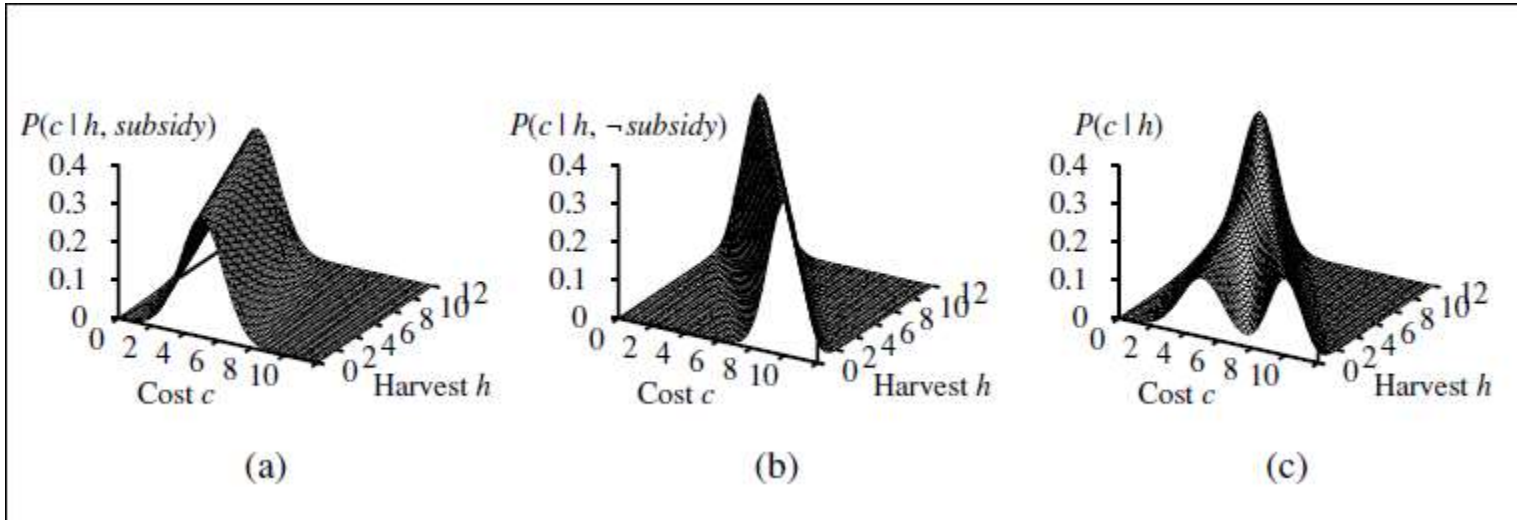
For the Cost variable, we need to specify **P**(Cost | harvest, Subsidy).
The discrete parent is handled by enumeration—that is, by specifying both P(Cost | Harvest, subsidy) and P(Cost | Harvest,¬subsidy).

- We need two distributions, one for subsidy and one for ¬subsidy, with different parameters:

$$P(c\,|\,h, subsidy) \;=\; N(a_t h + b_t, \sigma_t^2)(c) = \frac{1}{\sigma_t\sqrt{2\pi}}\, e^{-\frac{1}{2}\left(\frac{c-(a_t h+b_t)}{\sigma_t}\right)^2}$$

$$P(c\,|\,h, \neg subsidy) \;=\; N(a_f h + b_f, \sigma_f^2)(c) = \frac{1}{\sigma_f\sqrt{2\pi}}\, e^{-\frac{1}{2}\left(\frac{c-(a_f h+b_f)}{\sigma_f}\right)^2}$$

For this example, then, the conditional distribution for Cost is specified by naming the linear Gaussian distribution and providing the parameters a_t, b_t, σ_t, a_f, b_f, and σ_f



The graphs in (a) and (b) show the probability distribution over Cost as a function of Harvest size, with Subsidy true and false, respectively. Graph (c) shows the distribution P(Cost |Harvest), obtained by summing over the two subsidy cases.

Notice that in each case the slope is negative, because cost decreases as supply increases.

- The linear Gaussian conditional distribution has some special properties. A network containing only continuous variables with linear Gaussian distributions has a joint distribution that is a multivariate Gaussian distribution over all the. Furthermore, the posterior distribution given any evidence also has this property.
- When discrete variables are added as parents (not as children) of continuous variables, the network defines a **conditional Gaussian**, or CG, distribution: given any assignment to the discrete variables, the distribution over the continuous variables is a multivariate Gaussian.

- Now we turn to the distributions for discrete variables with continuous parents. Consider, for example, the Buys node in previous Figure. It seems reasonable to assume that the customer will buy if the cost is low and will not buy if it is high and that the probability of buying varies smoothly in some intermediate region.
- In other words, the conditional distribution is like a “soft” threshold function. One way to make soft thresholds is to use the *integral* of the standard normal distribution:

$$\Phi(x) = \int_{-\infty}^{x} N(0,1)(x)dx \; .$$

- Then the probability of Buys given Cost might be $P(buys \mid Cost = c) = \Phi((-c + \mu)/\sigma)$,
- which means that the cost threshold occurs around μ, the width of the threshold region is proportional to σ, and the probability of buying decreases as cost increases.

Inference in Bayesian Networks

- Now that we know what the semantics of Bayes nets are; what it means when we have one, we need to understand how to use it. Typically, we’ll be in a situation in which we have some evidence, that is, some of the variables are instantiated, and we want to infer something about the probability distribution of some other variables.
- **Inference** in graphical models, in which some of the nodes in a graph are clamped to observed values, and we wish to compute the posterior distributions of one or more subsets of other nodes

- The basic task for any probabilistic inference system is to compute the posterior probability EVENT distribution for a set of **query variables**, given some observed **event**—that is, some assignment of values to a set of **evidence variables**.
- In the burglary network, we might observe the event in which JohnCalls =true and MaryCalls =true. We could then ask for, say, the probability that a burglary has occurred:

$$\mathbf{P}(Burglary \mid JohnCalls = true, MaryCalls = true) = 0.284, 0.716 .$$

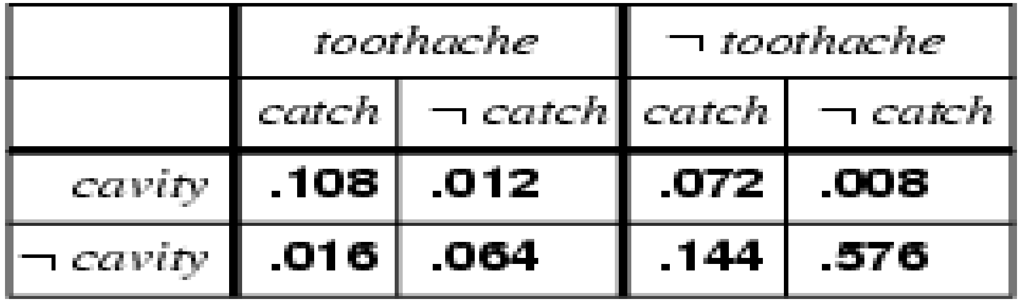
Inference in Bayesian Nets

- Objective: calculate posterior prob of a variable x conditioned on evidence Y and marginalizing over Z (unobserved vars)

- Exact methods:
  - Enumeration
  - Factoring
  - Variable elimination
- Approximate Methods: sampling

Inference by enumeration

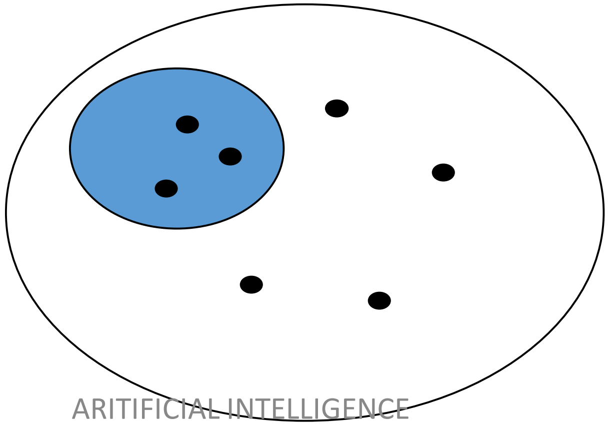
Start with the joint probability distribution:



	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

For any proposition a, sum the atomic events where it is true: $P(a) = \Sigma_{\omega \text{ s.t. } a=true} P(\omega)$

P(a)=1/7 + 1/7 + 1/7 = 3/7



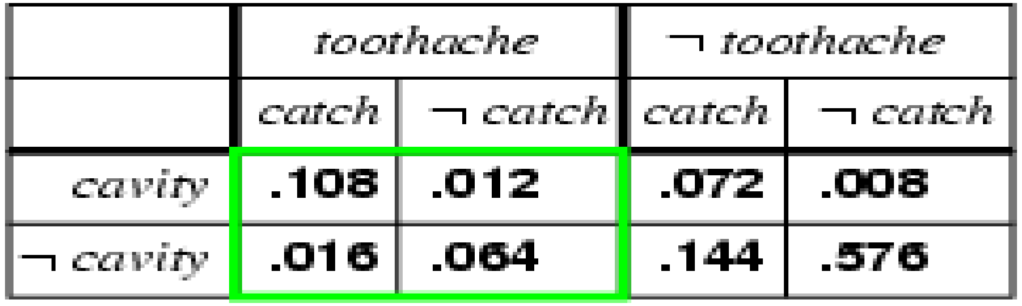
Algorithm

```
function ENUMERATION-ASK(X, e, bn) returns a distribution over X
inputs: X, the query variable
        e, observed values for variables E
        bn, a Bayes net with variables {X} ∪ E ∪ Y /* Y = hidden variables */
Q(X)←a distribution over X, initially empty
for each value xi of X do
Q(xi)←ENUMERATE-ALL(bn.VARS, exi )
            where exi is e extended with X = xi
return NORMALIZE(Q(X))
```

```
function ENUMERATE-ALL(vars, e) returns a real number
if EMPTY?(vars) then return 1.0
Y ←FIRST(vars)
if Y has value y in e
then return P(y | parents(Y )) × ENUMERATE-ALL(REST(vars), e)
else returny P(y | parents(Y )) × ENUMERATE-ALL(REST(vars), ey)
where ey is e extended with Y = y
```

Inference by enumeration

Start with the joint probability distribution:



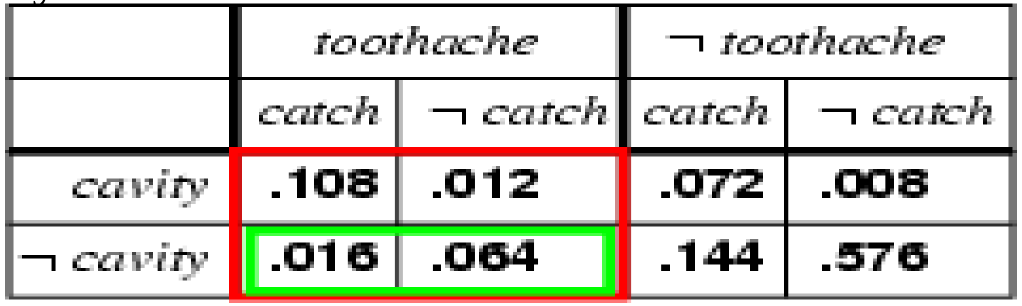
	toothache		¬ *toothache*	
	catch	¬ *catch*	*catch*	¬ *catch*
cavity	.108	.012	.072	.008
¬ *cavity*	.016	.064	.144	.576

For any proposition a, sum the atomic events where it is true: $P(a) = \Sigma_{\omega:\omega \text{ s.t. } a=\text{true}}\ P(\omega)$

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

Inference by enumeration

Start with the joint probability distribution:



	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

Can also compute conditional probabilities:

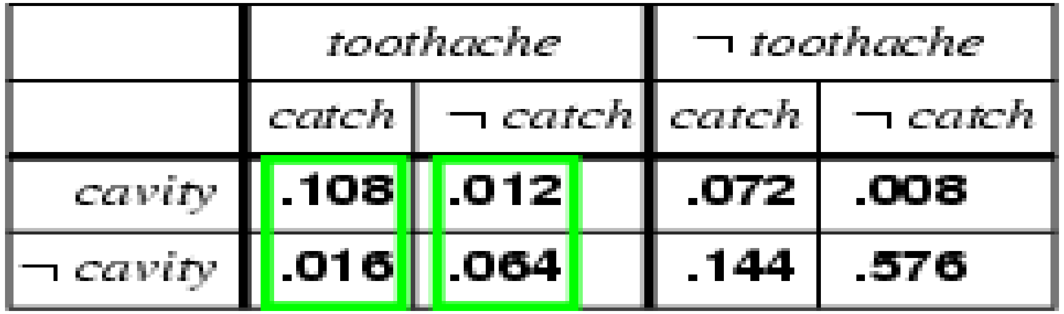
$P(\neg cavity \mid toothache) \quad = P(\neg cavity \wedge toothache) / P(toothache)$

$= \quad (0.016+0.064\ ) / (0.108 + 0.012 + 0.016 + 0.064)$

$= 0.4$

Calculate (i) $P(cavity)$ (ii) $P(cavity \mid toothache)$

Inference by enumeration



	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

Denominator can be viewed as a normalization constant α

$$\mathbf{P}(Cavity \mid toothache) = \alpha \times \mathbf{P}(Cavity, toothache)$$
$$= \alpha \times [\mathbf{P}(Cavity, toothache, catch) + \mathbf{P}(Cavity, toothache, \neg\, catch)]$$
$$= \alpha \times [\langle 0.108, 0.016\rangle + \langle 0.012, 0.064\rangle]$$
$$= \alpha \times \langle 0.12, 0.08\rangle = \langle 0.6, 0.4\rangle$$

General idea: compute distribution on query variable by fixing evidence variables and summing over hidden variables

Inference by enumeration

Typically, we are interested in

the posterior joint distribution of the query variables **Y**

given specific values **e** for the evidence variables **E**

Let the hidden variables be **H** = **X** - **Y** - **E**

Then the required summation of joint entries is done by summing out the hidden variables:

$$\mathbf{P}(\mathbf{Y} \mid \mathbf{E} = \mathbf{e}) = \alpha\mathbf{P}(\mathbf{Y},\mathbf{E} = \mathbf{e}) = \alpha\Sigma_{h}\mathbf{P}(\mathbf{Y},\mathbf{E}= \mathbf{e}, \mathbf{H} = \mathbf{h})$$

The terms in the summation are joint entries because **Y**, **E** and **H** together exhaust the set of random variables

Obvious problems:

1. Worst-case time complexity $O(d^n)$ where d is the largest arity
2. Space complexity $O(d^n)$ to store the joint distribution
3. How to find the numbers for $O(d^n)$ entries

Factors

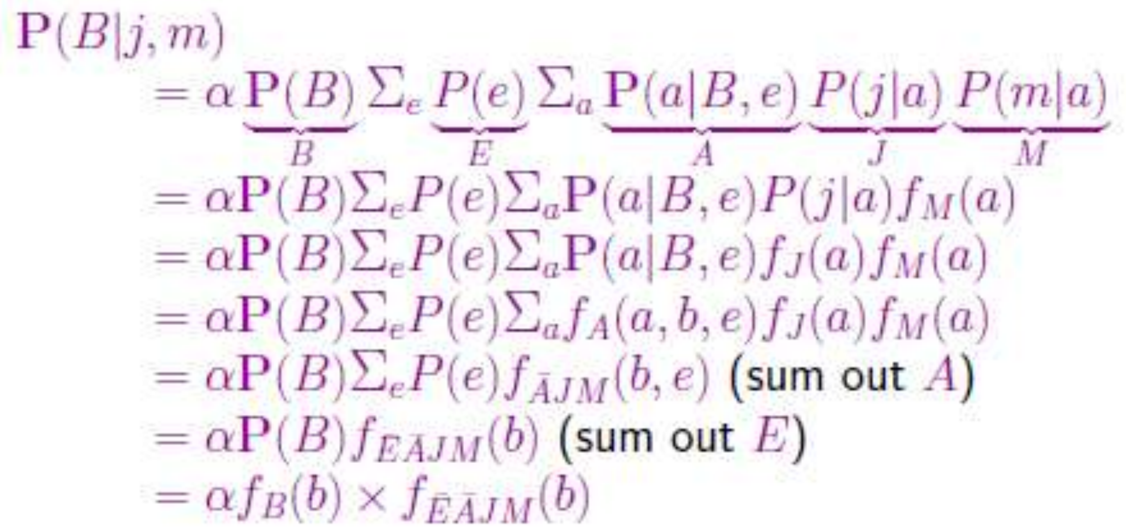
- A factor is a multi-dimensional table, like a CPT
- $f_{\underline{A}JM}(B,E)$
  - 2x2 table with a “number” for each combination of B,E
  - Specific values of J and M were used
  - A has been summed out
- $f(J,A)=P(J|A)$ is 2x2:
- $f_J(A)=P(j|A)$ is 1x2: $\{p(j|a),p(j|\neg a)\}$

$p(j\|a)$	$p(j\|\neg a)$
$p(\neg j\|a)$	$p(\neg j\|\neg a)$

Use of factors in variable elimination:

- The **enumeration** algorithm can be improved substantially by eliminating repeated calculations.
- The idea is simple: do the calculation once and save the results for later use.

- Variable elimination works by evaluating expressions in *right-to-left* order
- Intermediate results are stored, and summations over
- each variable are done only for those portions of the expression that depend on the variable.
- Let us illustrate this process for the burglary network.



$$
\begin{aligned}
&\mathbf{P}(B|j,m)\\
&\quad = \alpha \underbrace{\mathbf{P}(B)}_{B} \Sigma_e \underbrace{P(e)}_{E} \Sigma_a \underbrace{\mathbf{P}(a|B,e)}_{A} \underbrace{P(j|a)}_{J} \underbrace{P(m|a)}_{M}\\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a \mathbf{P}(a|B,e) P(j|a) f_M(a)\\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a \mathbf{P}(a|B,e) f_J(a) f_M(a)\\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a f_A(a,b,e) f_J(a) f_M(a)\\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) f_{\bar{A}JM}(b,e) \ \text{(sum out } A\text{)}\\
&\quad = \alpha \mathbf{P}(B) f_{E\bar{A}JM}(b) \ \text{(sum out } E\text{)}\\
&\quad = \alpha f_B(b) \times f_{\bar{E}\bar{A}JM}(b)
\end{aligned}
$$

Algorithm variable elimination

- **function** ELIMINATION-ASK(X, **e**, bn) **returns** a distribution over X
- **inputs**: X, the query variable
- **e**, observed values for variables **E**
- bn, a Bayesian network specifying joint distribution **P**(X1, . . . , Xn)
- factors ←[ ]
- **for each** var **in** ORDER(bn.VARS) **do**
  - factors ←[MAKE-FACTOR(var , **e**)|factors]
  - **if** var is a hidden variable **then** factors ←SUM-OUT(var, factors)
- **return** NORMALIZE(POINTWISE-PRODUCT(factors))

Pointwise product

- The pointwise product of two factors **f**1 and **f**2 yields a new factor **f** whose variables are the *union* of the variables in **f**1 and **f**2 and whose elements are given by the product of the corresponding elements in the two factors.
- given 2 factors that share some variables:
  - f1(X1..Xi,Y1..Yj), f2(Y1..Yj,Z1..Zk)
- Resulting table has dimensions of union of variables, f1*f2=F(X1..Xi,Y1..Yj,Z1..Zk)
- each entry in F is a truth assignment over vars and can be computed by multiplying entries from f1 and f2

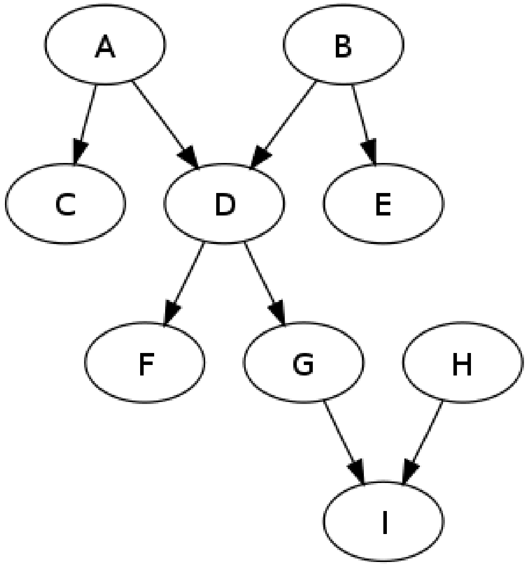
A	B	f1(A,B)
T	T	0.3
T	F	0.7
F	T	0.9
F	F	0.1

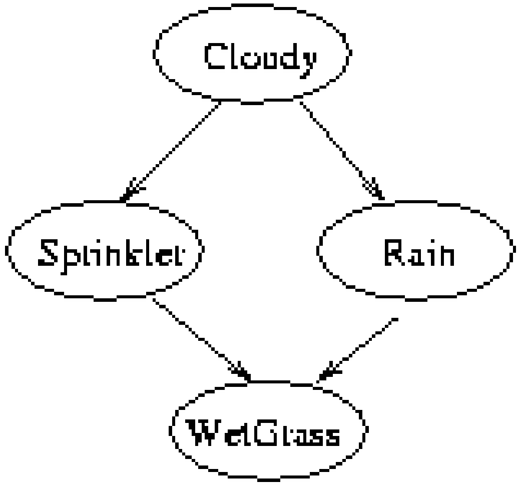
B	C	f2(B,C)
T	T	0.2
T	F	0.8
F	T	0.6
F	F	0.4

A	B	C	F(A,B,C)
T	T	T	0.3x0.2
T	T	F	0.3x0.8
T	F	T	0.7x0.6
T	F	F	0.7x0.4
F	T	T	0.9x0.2
F	T	F	0.9x0.8
F	F	T	0.1x0.6
F	F	F	0.1x0.4

Computational Complexity

- Belief propagation is linear in the size of the BN for polytrees
- Belief propagation is NP-hard for trees with “cycles”





Approximate Inference

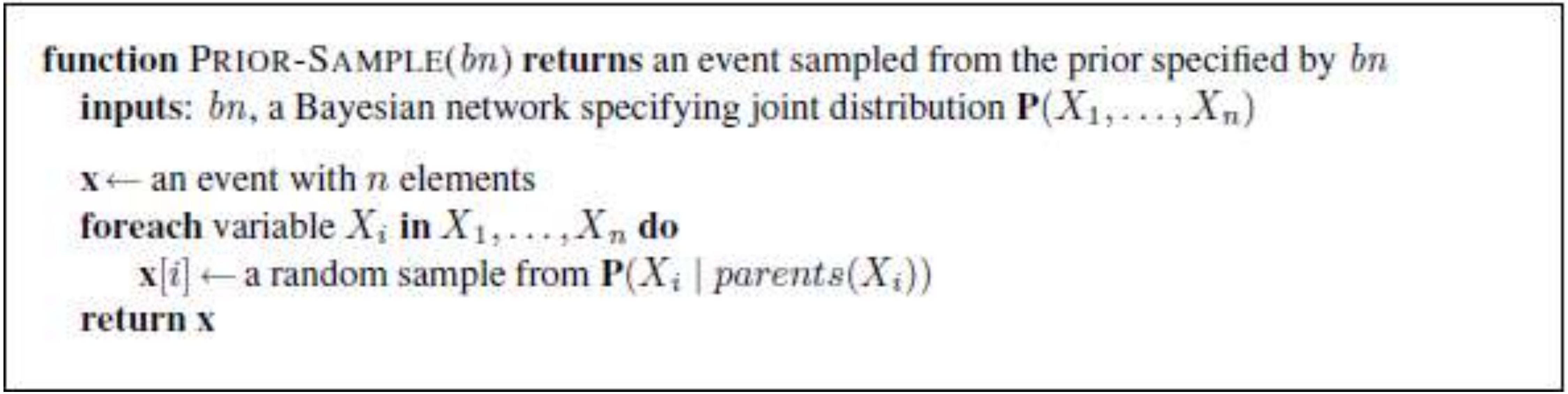
- Given the intractability of exact inference in large, multiply connected networks, it is essential to consider approximate inference methods.
- Simple Sampling: logic sample
- Use BayesNetwork as a generative model
  - Eg. generate million or more models, via topological order.
- Generates examples with appropriate distribution.
- Now use examples to estimate probabilities.

Direct sampling

- Create an independent atomic event
  - for each var *in topological order*, choose a value conditionally dependent on parents
    1. sample from p(Cloudy)=<0.5,0.5>; suppose T
    2. sample from p(Sprinkler|Cloudy=T)=<0.1,0.9>, suppose F
    3. sample from P(Rain|Cloudy=T)=<0.8,0.2>, suppose T
    4. sample from P(WetGrass|Sprinkler=F,Rain=T)=<0.9,0,1>, suppose T

    event: <Cloudy,¬Sprinkler,Rain,WetGrass>
- repeat
- in the limit, each event occurs with frequency proportional to its joint probability, P(Cl,Sp,Ra,Wg)= P(Cl)*P(Sp|Cl)*P(Ra|Cl)*P(Wg|Sp,Ra)
- averaging: P(Ra,Cl) = Num(Ra=T&Cl=T)/|Sample|

Algorithm for sampling



```
function PRIOR-SAMPLE($bn$) returns an event sampled from the prior specified by $bn$
  inputs: $bn$, a Bayesian network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$

  $\mathbf{x} \leftarrow$ an event with $n$ elements
  foreach variable $X_i$ in $X_1, \ldots, X_n$ do
      $\mathbf{x}[i] \leftarrow$ a random sample from $\mathbf{P}(X_i \mid parents(X_i))$
  return $\mathbf{x}$
```

Rejection sampling

- **Rejection sampling** is a general method for producing samples from a hard-to-sample distribution given an easy-to-sample distribution.
- it can be used to compute conditional probabilities—that is, to determine P(X | **e**).
- To condition upon evidence variables **e**, average over samples that satisfy **e**
- P(j,m|¬e,¬b)

`<e,b,-a,-j,m>`

`<e,-b,a,-j,-m>`

`<-e,b,a,j,m>`

`<-e,-b,-a,-j,m>`

`<-e,-b,a,-j,-m>`

`<e,b,a,j,m>`

`<-e,-b,a,j,-m>`

`<e,-b,a,j,m>`

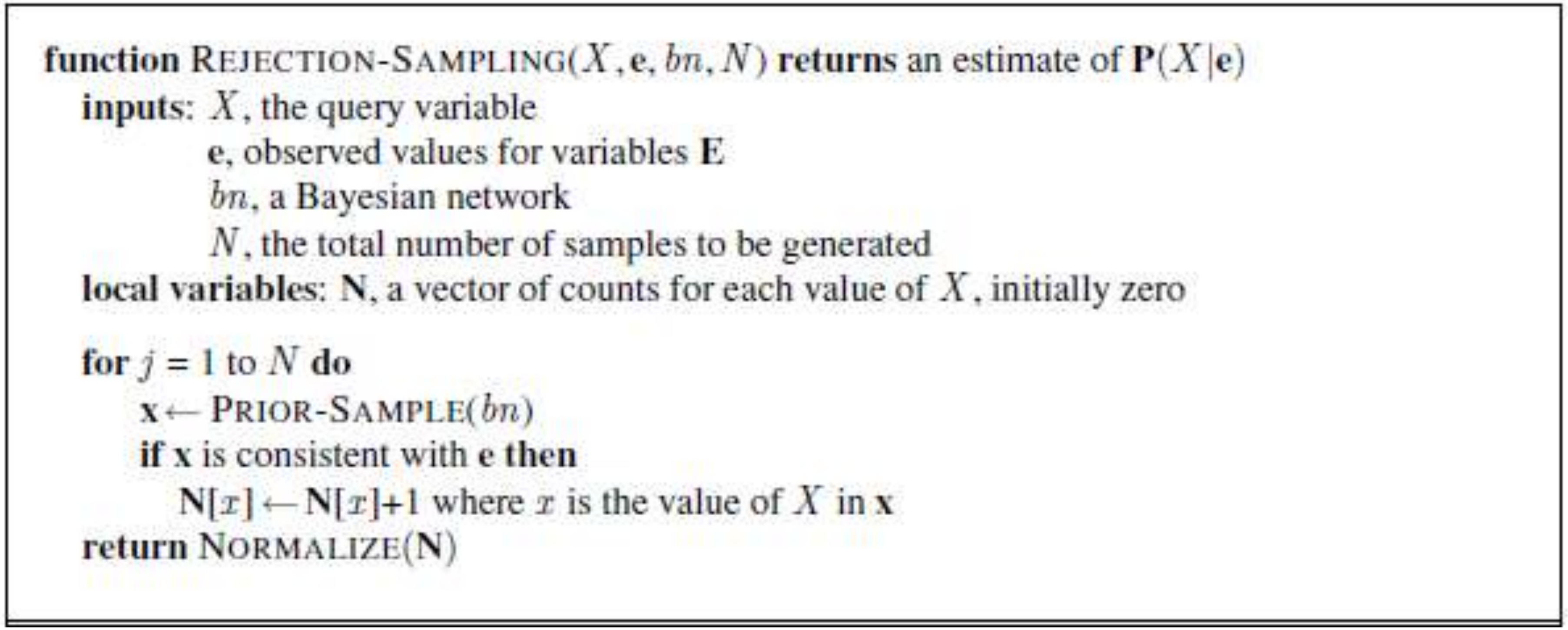
`...`

- First, it generates samples from the prior distribution specified by the network. Then, it rejects all those that do not match the evidence.
- Finally, the estimate

$$\hat{\mathbf{P}}(X \mid \mathbf{e}) = \alpha\, \mathbf{N}_{PS}(X, \mathbf{e}) = \frac{\mathbf{N}_{PS}(X, \mathbf{e})}{N_{PS}(\mathbf{e})}$$

- N_{PS} $(x_1, \ldots, x_n)$ be the number of times the specific event $x_1, \ldots, x_n$ occurs in the set of samples.
- That is, rejection sampling produces a consistent estimate of the true probability.

Algorithm for Rejection sampling



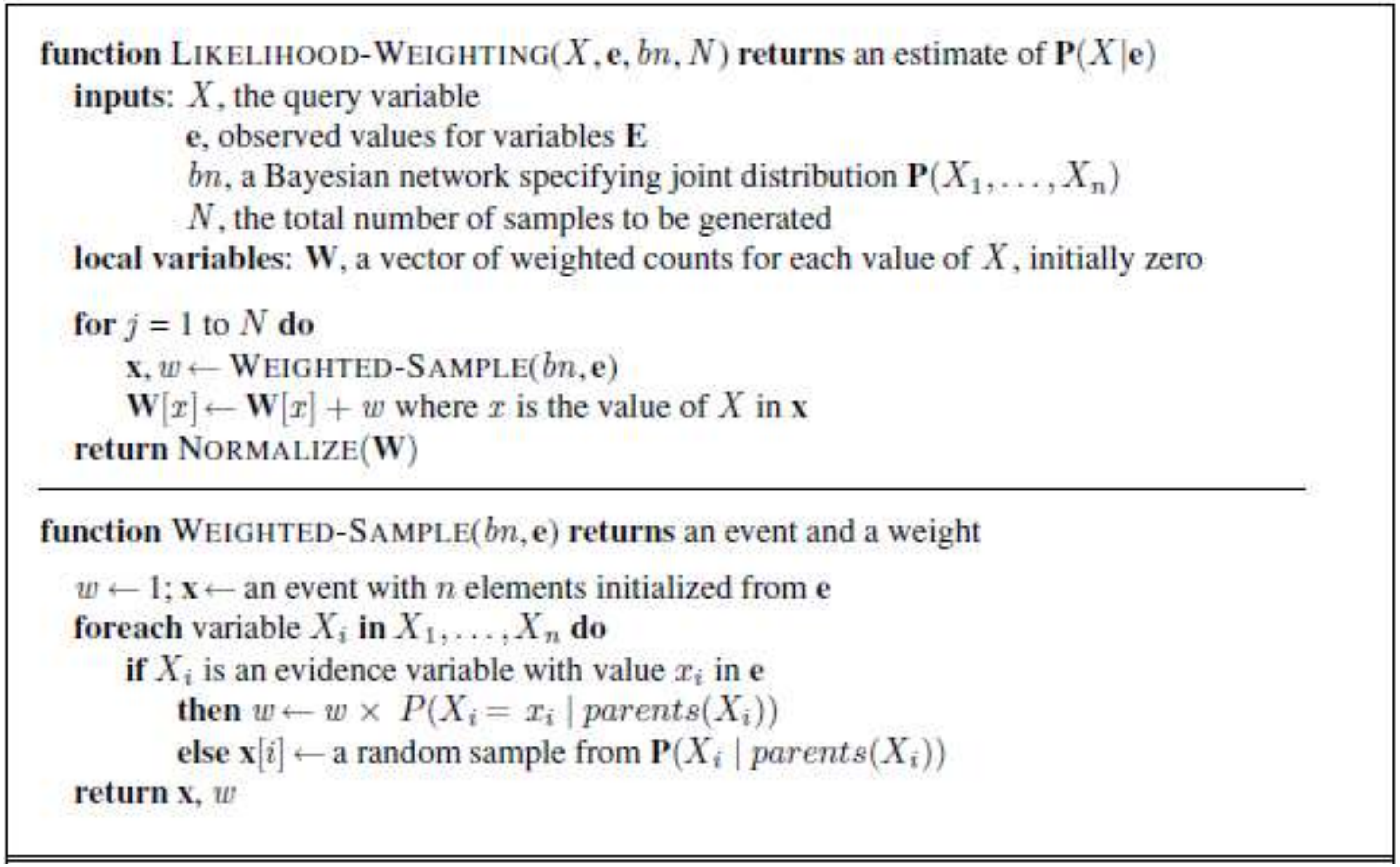
```
function REJECTION-SAMPLING($X$, e, $bn$, $N$) returns an estimate of $\mathbf{P}(X|\mathbf{e})$
   inputs: $X$, the query variable
           e, observed values for variables E
           $bn$, a Bayesian network
           $N$, the total number of samples to be generated
   local variables: N, a vector of counts for each value of $X$, initially zero

   for $j$ = 1 to $N$ do
      x ← PRIOR-SAMPLE($bn$)
      if x is consistent with e then
         N[$x$] ← N[$x$]+1 where $x$ is the value of $X$ in x
   return NORMALIZE(N)
```

Likelihood weighting

- **Likelihood weighting** avoids the inefficiency of rejection sampling by generating only events that are consistent with the evidence **e**.
- It is a particular instance of the general statistical technique of **importance sampling**, tailored for inference in Bayesian networks.
- LIKELIHOOD-WEIGHTING fixes the values for the evidence variables **E** and samples only the nonevidence variables. This guarantees that each event generated is consistent with the evidence.
- P(j|e) – earthquakes only occur 0.2% of the time, so can only use ~2/1000 samples to determine frequency of JohnCalls
- During sample generation, when reach an evidence variable e_i, force it to be known value accumulate weight $w=\Pi\ p(e_i|parents(e_i))$ now every sample is useful ("consistent")
- when calculating averages over samples **x**, weight them: $P(j|e) = \alpha\Sigma_{consistent}\ w(\mathbf{x})=<\Sigma_{J=T}\ w(\mathbf{x}), \Sigma_{J=F}\ w(\mathbf{x})>$

Algorithm for Likelihood weighting



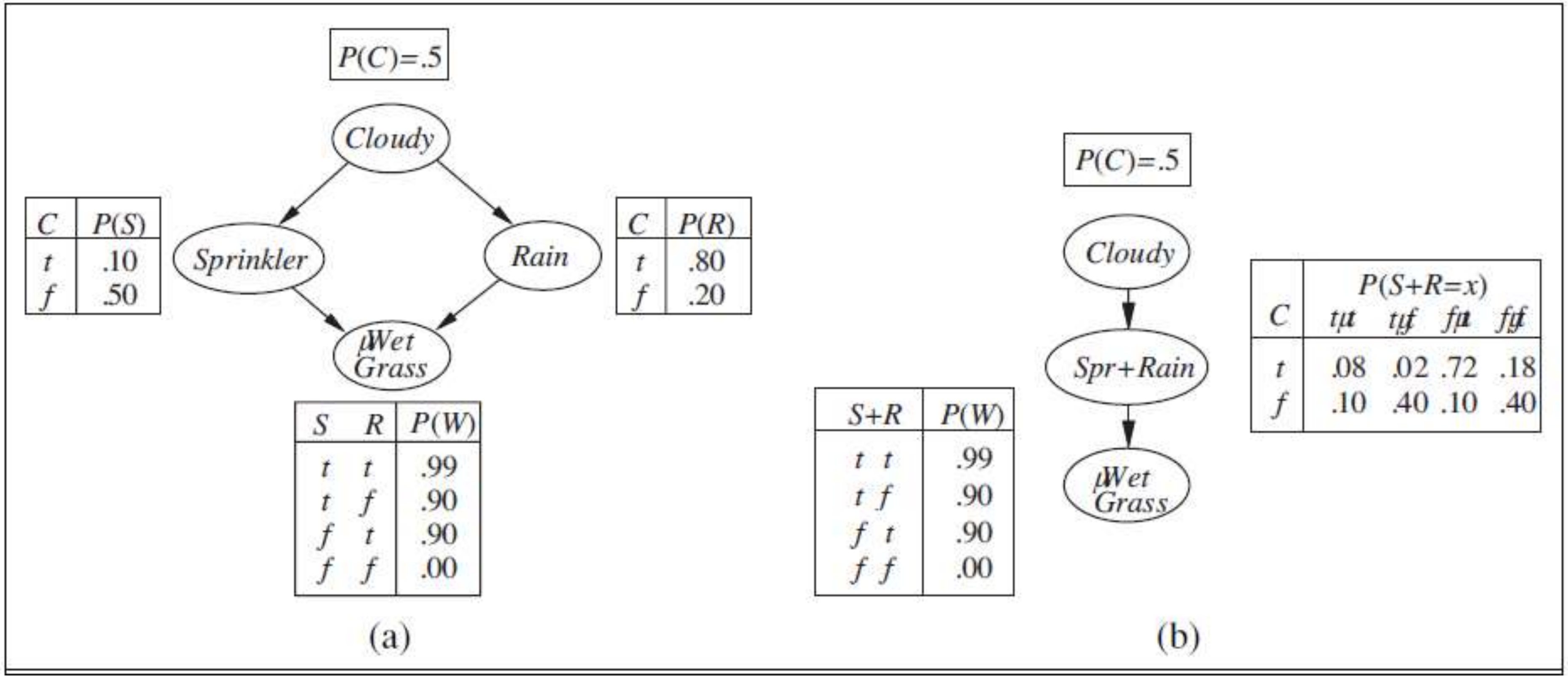
```
function LIKELIHOOD-WEIGHTING($X$, $\mathbf{e}$, $bn$, $N$) returns an estimate of $\mathbf{P}(X|\mathbf{e})$
  inputs: $X$, the query variable
          $\mathbf{e}$, observed values for variables $\mathbf{E}$
          $bn$, a Bayesian network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$
          $N$, the total number of samples to be generated
  local variables: $\mathbf{W}$, a vector of weighted counts for each value of $X$, initially zero

  for $j$ = 1 to $N$ do
     $\mathbf{x}, w \leftarrow$ WEIGHTED-SAMPLE($bn$, $\mathbf{e}$)
     $\mathbf{W}[x] \leftarrow \mathbf{W}[x] + w$ where $x$ is the value of $X$ in $\mathbf{x}$
  return NORMALIZE($\mathbf{W}$)
```

```
function WEIGHTED-SAMPLE($bn$, $\mathbf{e}$) returns an event and a weight

  $w \leftarrow 1$; $\mathbf{x} \leftarrow$ an event with $n$ elements initialized from $\mathbf{e}$
  foreach variable $X_i$ in $X_1, \ldots, X_n$ do
     if $X_i$ is an evidence variable with value $x_i$ in $\mathbf{e}$
        then $w \leftarrow w \times P(X_i = x_i \mid parents(X_i))$
        else $\mathbf{x}[i] \leftarrow$ a random sample from $\mathbf{P}(X_i \mid parents(X_i))$
  return $\mathbf{x}$, $w$
```

Example



C	P(S)
t	.10
f	.50

C	P(R)
t	.80
f	.20

S	R	P(W)
t	t	.99
t	f	.90
f	t	.90
f	f	.00

(a)

S+R	P(W)
t t	.99
t f	.90
f t	.90
f f	.00

C	P(S+R=x)			
	t t	t f	f t	f f
t	.08	.02	.72	.18
f	.10	.40	.10	.40

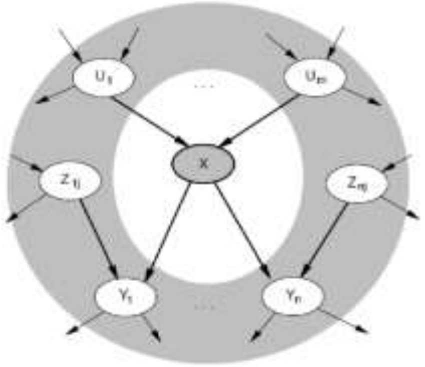
(b)

- Let us apply the algorithm to , with the query **P**(Rain |Cloudy =true, WetGrass =true) and the ordering *Cloudy*, *Sprinkler*, *Rain*, *WetGrass*. (Any topological ordering will do.) The process goes as follows: First, the weight w is set to 1.0. Then an event is generated:
  - 1. Cloudy is an evidence variable with value true. Therefore, we set w ← w×P(Cloudy =true) = 0.5 .
  - 2. Sprinkler is not an evidence variable, so sample from **P**(Sprinkler |Cloudy =true) = 0.1, 0.9; suppose this returns false.
  - 3. Similarly, sample from **P**(Rain |Cloudy =true) = 0.8, 0.2; suppose this returns true.
  - 4. WetGrass is an evidence variable with value true. Therefore, we set
  - w ← w×P(WetGrass =true | Sprinkler =false, Rain =true) = 0.45 .
- **Here WEIGHTED-SAMPLE returns the event [true, false, true, true] with weight 0.45, and this is tallied under Rain =true.**

Gibbs sampling (MCMC)

- Start with a random assignment to vars
  - set evidence vars to observed values
- Iterate {
  - pick a non-evidence variable, X
  - define Markov blanket of X, mb(X)
    - parents, children, and parents of children
  - re-sample value of X from conditional distrib.
    - P(X|mb(X))=αP(X|parents(X))*Π P(y|parents(X)) for y∈children(X)

  }

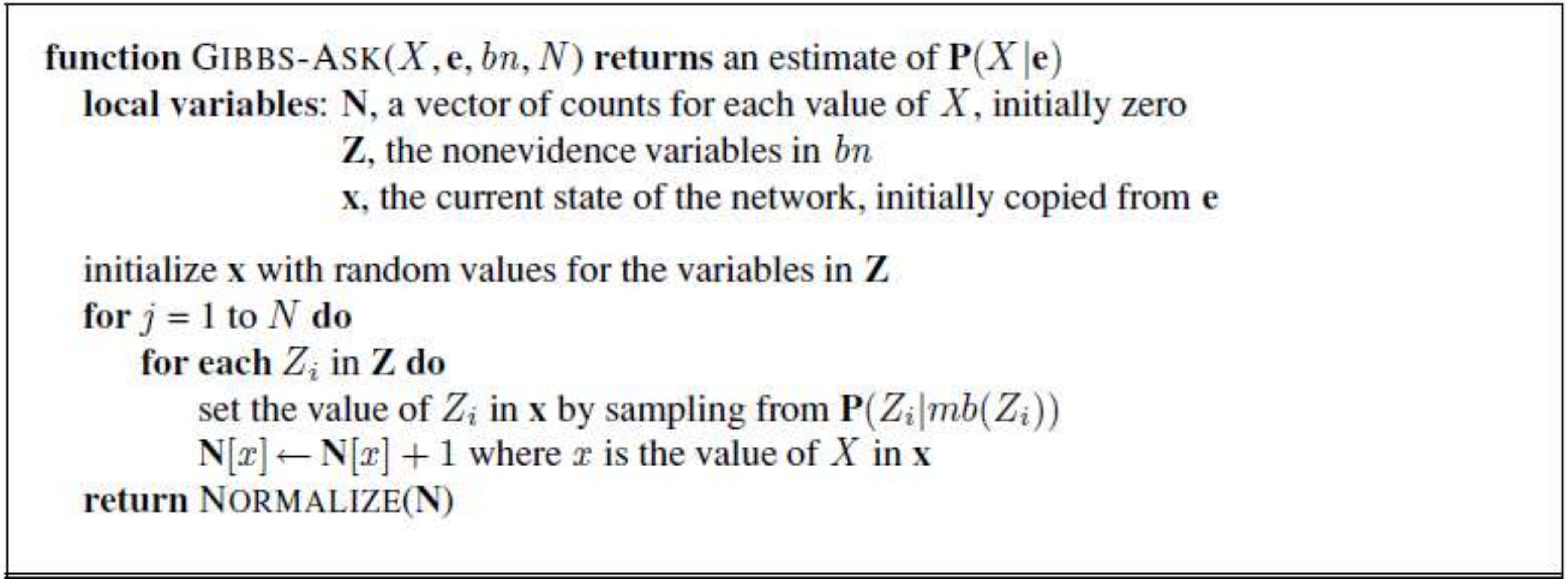


- The Markov blanket of a variable consists of its parents, children, and children’s parents.
- Generates a large sequence of samples, where each might “flip a bit” from previous sample
- In the limit, this converges to joint probability distribution (samples occur for frequency proportional to joint PDF)

Example

- Consider the query **P**(Rain | Sprinkler =true, WetGrass =true) applied to the network in previous graph.
- The evidence variables Sprinkler and WetGrass are fixed to their observed values and the non evidence variables Cloudy and Rain are initialized randomly—
- let us say to true and false respectively. Thus, the initial state is [true, true, false, true].
- Now the non evidence variables are sampled repeatedly in an arbitrary order. For example:
  - 1. Cloudy is sampled, given the current values of its Markov blanket variables: in this case, we sample from **P**(Cloudy | Sprinkler =true, Rain =false). (Shortly, we will show how to calculate this distribution.) Suppose the result is Cloudy =false. Then the new current state is [false, true, false, true].
  - 2. Rain is sampled, given the current values of its Markov blanket variables: in this case, we sample from **P**(Rain |Cloudy =false, Sprinkler =true, WetGrass =true). Suppose this yields Rain =true. The new current state is [false, true, true, true].

Algorithm for Gibbs sampling



```
function GIBBS-ASK($X$, $\mathbf{e}$, $bn$, $N$) returns an estimate of $\mathbf{P}(X|\mathbf{e})$
  local variables: $\mathbf{N}$, a vector of counts for each value of $X$, initially zero
                   $\mathbf{Z}$, the nonevidence variables in $bn$
                   $\mathbf{x}$, the current state of the network, initially copied from $\mathbf{e}$

  initialize $\mathbf{x}$ with random values for the variables in $\mathbf{Z}$
  for $j = 1$ to $N$ do
     for each $Z_i$ in $\mathbf{Z}$ do
        set the value of $Z_i$ in $\mathbf{x}$ by sampling from $\mathbf{P}(Z_i|mb(Z_i))$
        $\mathbf{N}[x] \leftarrow \mathbf{N}[x] + 1$ where $x$ is the value of $X$ in $\mathbf{x}$
  return NORMALIZE($\mathbf{N}$)
```

Other Approaches to Uncertain reasoning- Fuzzy sets and Fuzzy logic.

Fuzzy set

- **Fuzzy set theory** is a means of specifying how well an object satisfies a vague description.
- For example, consider the proposition “Nate is tall.” Is this true if Nate is 5 10? Most people would hesitate to answer “true” or “false,” preferring to say, “sort of.” Note that this is not a question of uncertainty about the external world—we are sure of Nate’s height.
- The issue is that the linguistic term “tall” does not refer to a sharp demarcation of objects into two classes—there are *degrees* of tallness.
- For this reason, *fuzzy set theory is not a method for uncertain reasoning at all.*
- Rather, fuzzy set theory treats Tall as a fuzzy predicate and says that the truth value of Tall (Nate) is a number between 0 and 1, rather than being just true or false.
- The name “fuzzy set” derives from the interpretation of the predicate as implicitly defining a set of its members—a set that does not have sharp boundaries.

What is Fuzzy Logic?

- The inventor of fuzzy logic is Lotfi Zadeh
- Fuzzy Logic (FL) is a method of reasoning that resembles human reasoning. The approach of FL imitates the way of decision making in humans that involves all intermediate possibilities between digital values YES and NO.
- The conventional logic block that a computer can understand takes precise input and produces a definite output as TRUE or FALSE, which is equivalent to human’s YES or NO.

- Inventor of Fuzzy logic, observed that unlike computers, the human decision making includes a range of possibilities between YES and NO, such as −

CERTAINLY YES
POSSIBLY YES
CANNOT SAY
POSSIBLY NO
CERTAINLY NO

- The fuzzy logic works on the levels of possibilities of input to achieve the definite output.

- **Fuzzy logic** is a method for reasoning with logical expressions describing membership in fuzzy sets. For example, the complex sentence Tall (Nate) ∧ Heavy(Nate) has a fuzzy truth value that is a function of the truth values of its components. The standard rules for evaluating the fuzzy truth, T, of a complex sentence are
  - $T(A \land B) = \min(T(A), T(B))$
  - $T(A \lor B) = \max(T(A), T(B))$
  - $T(\neg A) = 1 - T(A)$.

- Fuzzy logic is therefore a truth-functional system—a fact that causes serious difficulties.
    - For example, suppose that T(Tall (Nate))=0.6 and T(Heavy(Nate))=0.4. Then we have T(Tall (Nate) ∧ Heavy(Nate))=0.4, which seems reasonable, but we also get the result
    - T(Tall (Nate) ∧ ¬Tall (Nate))=0.4, which does not. Clearly, the problem arises from the inability of a truth-functional approach to take into account the correlations or anti-correlations among the component propositions.

- **Fuzzy control** is a methodology for constructing control systems in which the mapping between real-valued input and output parameters is represented by fuzzy rules.
- Fuzzy control has been very successful in commercial products such as automatic transmissions, video cameras, and electric shavers.

- Fuzzy predicates can also be given a probabilistic interpretation in terms of **random sets**—that is, random variables whose possible values are sets of objects.
- For example, Tall is a random set whose possible values are sets of people. The probability P(Tall =S1),
  - where S1 is some particular set of people, is the probability that exactly that set would be identified as “tall” by an observer. Then the probability that “Nate is tall” is the sum of the probabilities of all the sets of which Nate is a member.

- Both the hybrid Bayesian network approach and the random sets approach appear to capture aspects of fuzziness without introducing degrees of truth.