Bayesian Networks

CHAPTER 14
HASSAN KHOSRAVI
SPRING 2011

- Definition of Bayesian networks
  - Representing a joint distribution by a graph
  - Can yield an efficient factored representation for a joint distribution
- Inference in Bayesian networks
  - Inference = answering queries such as P(Q | e)
  - Intractable in general (scales exponentially with num variables)
  - But can be tractable for certain classes of Bayesian networks
  - Efficient algorithms leverage the structure of the graph

Computing with Probabilities: Law of Total Probability

Law of Total Probability (aka “summing out” or marginalization)

$$P(a) = \Sigma_b \; P(a, b)$$

$$= \Sigma_b \; P(a \mid b)\, P(b)$$ where B is any random variable

Why is this useful?

given a joint distribution (e.g., P(a,b,c,d)) we can obtain any “marginal” probability (e.g., P(b)) by summing out the other variables, e.g.,

$$P(b) = \Sigma_a \, \Sigma_c \, \Sigma_d \; P(a, b, c, d)$$

Less obvious: we can also compute any conditional probability of interest given a joint distribution, e.g.,

$$P(c \mid b) = \Sigma_a \Sigma_d P(a, c, d \mid b)$$
$$= 1 / P(b) \;\; \Sigma_a \Sigma_d P(a, c, d, b)$$

where $1 / P(b)$ is just a normalization constant

Thus, the joint distribution contains the information we need to compute any probability of interest.

Computing with Probabilities: The Chain Rule or Factoring

We can always write

$$P(a, b, c, \ldots z) = P(a \mid b, c, \ldots. z)\, P(b, c, \ldots z)$$

(by definition of joint probability)

Repeatedly applying this idea, we can write

$$P(a, b, c, \ldots z) = P(a \mid b, c, \ldots. z)\, P(b \mid c, .. z)\, P(c \mid .. z) .. P(z)$$

This factorization holds for any ordering of the variables

This is the chain rule for probabilities

Conditional Independence

- 2 random variables A and B are conditionally independent given C iff

  P(a, b | c) = P(a | c) P(b | c)   for all values a, b, c

- More intuitive (equivalent) conditional formulation
  - A and B are conditionally independent given C iff
    P(a | b, c) = P(a | c)   OR   P(b | a, c) =P(b | c),   for all values a, b, c

Are A, B, and C independent?

P(A=1, B=1, C=1) = 2/10   p(A=1) p(B=1) p(C=1) = ½ * 6/10 * ½= 3/20

Are A and B given C conditionally independent of each other?

P(A=1, B=1| C=1) =2 /5   P(A=1|C=1) p(B=1|C=1) = 2/5 *3/5= 6/25

A	B	C
0	0	1
0	1	0
1	1	1
1	1	0
0	1	1
0	1	0
0	0	1
1	0	0
1	1	1
1	0	0

- Intuitive interpretation:

  P(a | b, c) = P(a | c) tells us that learning about b, given that we already know c, provides no change in our probability for a,

  i.e., b contains no information about a beyond what c provides

- Can generalize to more than 2 random variables
  - E.g., K different symptom variables X1, X2, … XK, and C = disease
  - $P(X1, X2, \ldots XK \mid C) = \prod P(Xi \mid C)$
  - Also known as the naïve Bayes assumption

"…probability theory is more fundamentally concerned with the <u>structure</u> of reasoning and causation than with numbers."

Glenn Shafer and Judea Pearl
***Introduction to Readings in Uncertain Reasoning*,**
Morgan Kaufmann, 1990

Bayesian Networks

- Full joint probability distribution can answer questions about domain
  - Intractable as number of variables grow
  - Unnatural to have probably of all events unless large amount of data is available
- Independence and conditional independence between variables can greatly reduce number of parameters.
- We introduce a data structure called Bayesian Networks to represent dependencies among variables.

Example

- You have a new burglar alarm installed at home
- Its reliable at detecting burglary but also responds to earthquakes
- You have two neighbors that promise to call you at work when they hear the alarm
- John always calls when he hears the alarm, but sometimes confuses alarm with telephone ringing
- Marry listens to loud music and sometimes misses the alarm

Example

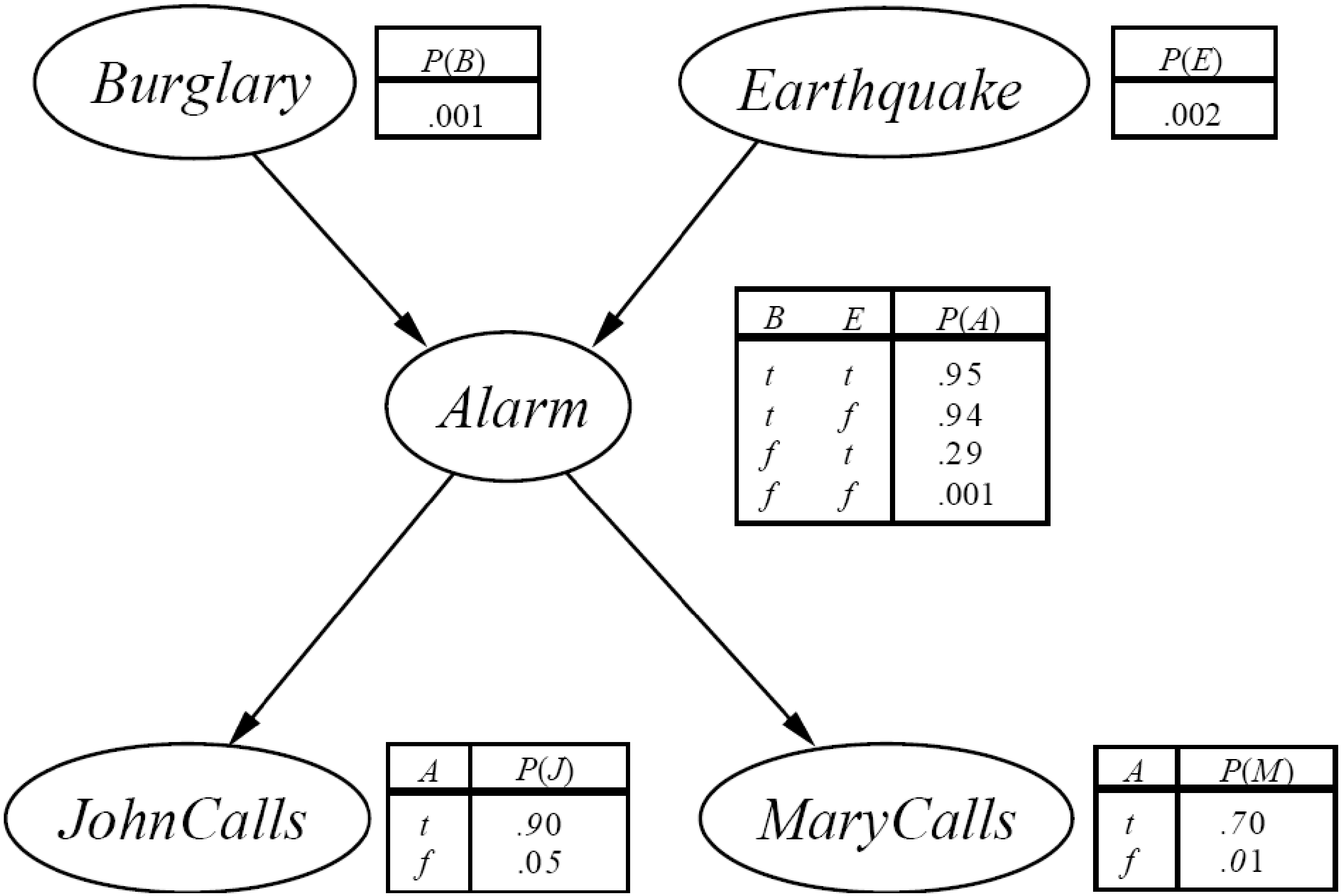
- Consider the following 5 binary variables:
  - B = a burglary occurs at your house
  - E = an earthquake occurs at your house
  - A = the alarm goes off
  - J = John calls to report the alarm
  - M = Mary calls to report the alarm

  - What is P(B | M, J) ? (for example)

  - We can use the full joint distribution to answer this question
    - Requires $2^5 = 32$ probabilities

    - Can we use prior domain knowledge to come up with a Bayesian network that requires fewer probabilities?

The Resulting Bayesian Network



$P(B)$
.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

Bayesian Network

- A Bayesian Network is a graph in which each node is annotated with probability information. The full specification is as follows
  - A set of random variables makes up the nodes of the network
  - A set of directed links or arrows connects pair of nodes. $X \rightarrow Y$ reads X is the parent of Y
  - Each node X has a conditional probability distribution $P(X|parents(X))$
  - The graph has no directed cycles (directed acyclic graph)

- $P(M, J, A, E, B) = P(M | J, A, E, B) p(J, A, E, B) = P(M|A)\, p(J, A, E, B)$

$$= P(M|A)\, p(J|A,E,B) p(A,E,B) = P(M|A)\, p(J|A) p(A,E,B)$$

$$= P(M|A)\, p(J|A) p(A|E,B) P(E,B)$$

$$= P(M|A)\, p(J|A) p(A|E,B) P(E) P(B)$$

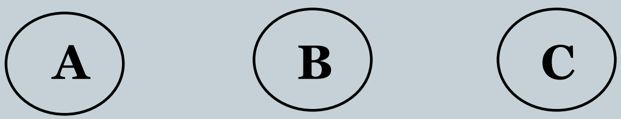
In general,

$$p(X_1, X_2, \ldots X_N) = \prod p(X_i \mid \text{parents}(X_i))$$

The full joint distribution

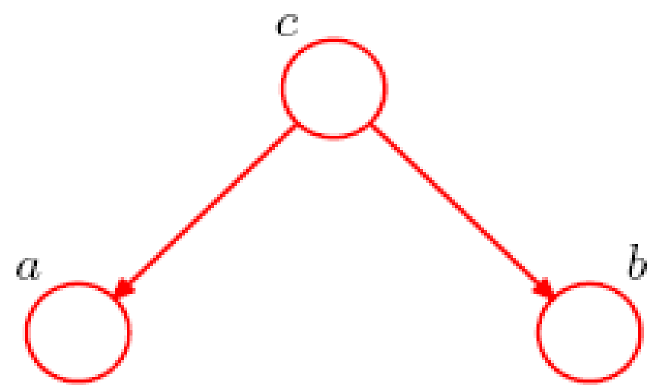
The graph-structured approximation

Examples of 3-way Bayesian Networks



Marginal Independence:
p(A,B,C) = p(A) p(B) p(C)

A Tale of Three Graphs - Part 1

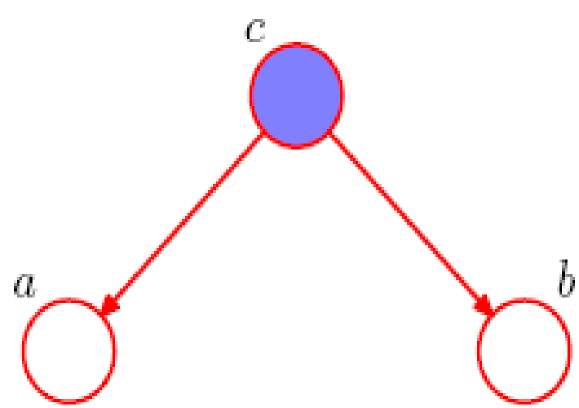


- The graph above means

$$
\begin{aligned}
p(a, b, c) &= p(a|c)p(b|c)p(c) \\
p(a, b) &= \sum_c p(a|c)p(b|c)p(c) \\
&\neq p(a)p(b) \text{ in general}
\end{aligned}
$$

- So a and b not independent

A Tale of Three Graphs - Part 1

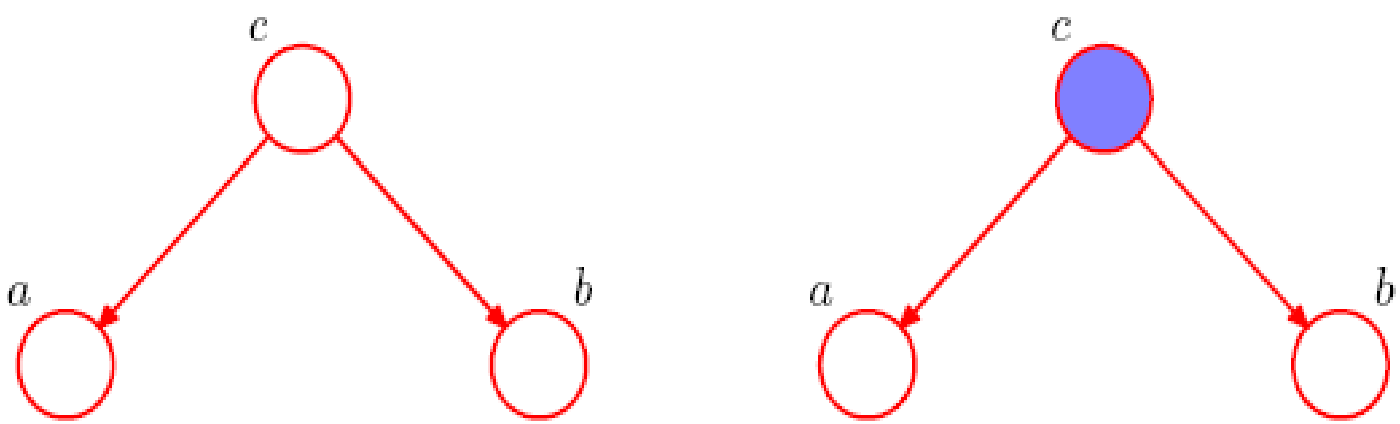


- However, conditioned on c

$$p(a,b|c) = \frac{p(a,b,c)}{p(c)} = \frac{p(a|c)p(b|c)p(c)}{p(c)} = p(a|c)p(b|c)$$

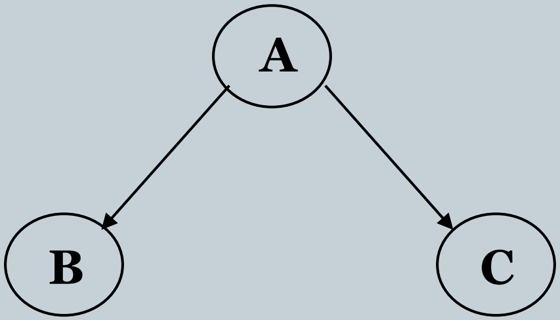
- So $a \perp\!\!\!\perp b|c$

A Tale of Three Graphs - Part 1



- Note the path from a to b in the graph
  - When c is not observed, path is open, a and b not independent
  - When c is observed, path is blocked, a and b independent
- In this case c is tail-to-tail with respect to this path

Examples of 3-way Bayesian Networks

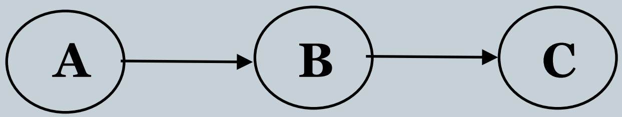


Conditionally independent effects:
p(A,B,C) = p(B|A)p(C|A)p(A)

B and C are conditionally independent Given A

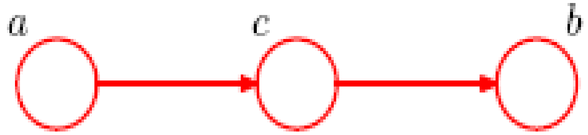
e.g., A is a disease, and we model B and C as conditionally independent symptoms given A

Examples of 3-way Bayesian Networks



Markov dependence:
$p(A,B,C) = p(C|B)\ p(B|A)p(A)$

A Tale of Three Graphs - Part 2

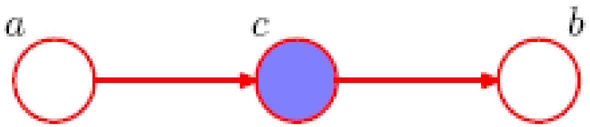


- The graph above means

$$p(a, b, c) \quad = \quad p(a)p(b|c)p(c|a)$$

- Again a and b not independent

A Tale of Three Graphs - Part 2

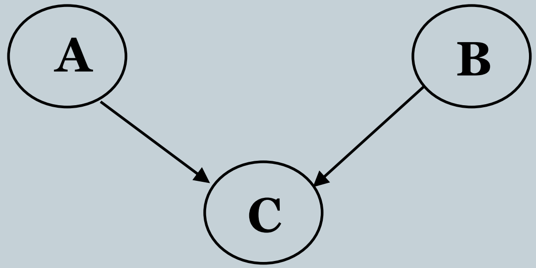


- However, conditioned on c

$$
\begin{aligned}
p(a, b|c) &= \frac{p(a, b, c)}{p(c)} = \frac{p(a)p(b|c)}{p(c)} p(c|a) \\
&= \frac{p(a)p(b|c)}{p(c)} \underbrace{\frac{p(a|c)p(c)}{p(a)}}_{\text{Bayes' Theorem}} \\
&= p(a|c)p(b|c)
\end{aligned}
$$

- So $a \perp\!\!\!\perp b|c$

Examples of 3-way Bayesian Networks

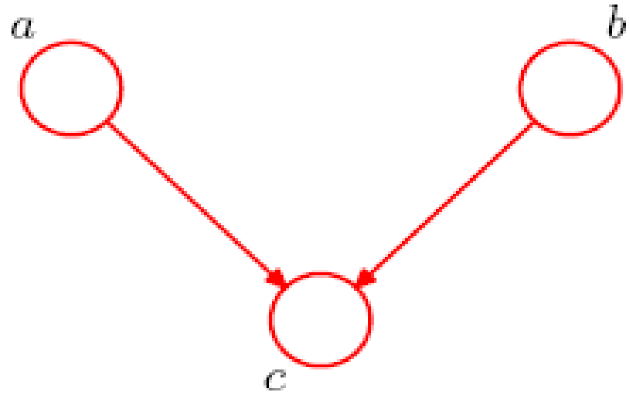


Independent Causes:
p(A,B,C) = p(C|A,B)p(A)p(B)

"Explaining away" effect:
Given C, observing A makes B less likely
e.g., earthquake/burglary/alarm example

A and B are (marginally) independent but become dependent once C is known

A Tale of Three Graphs - Part 3

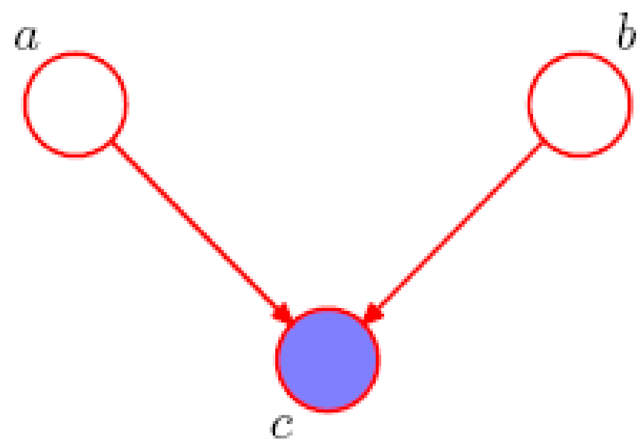


- The graph above means

$$
\begin{aligned}
p(a,b,c) &= p(a)p(b)p(c|a,b) \\
p(a,b) &= \sum_c p(a)p(b)p(c|a,b) \\
&= p(a)p(b)
\end{aligned}
$$

- This time a and b are independent

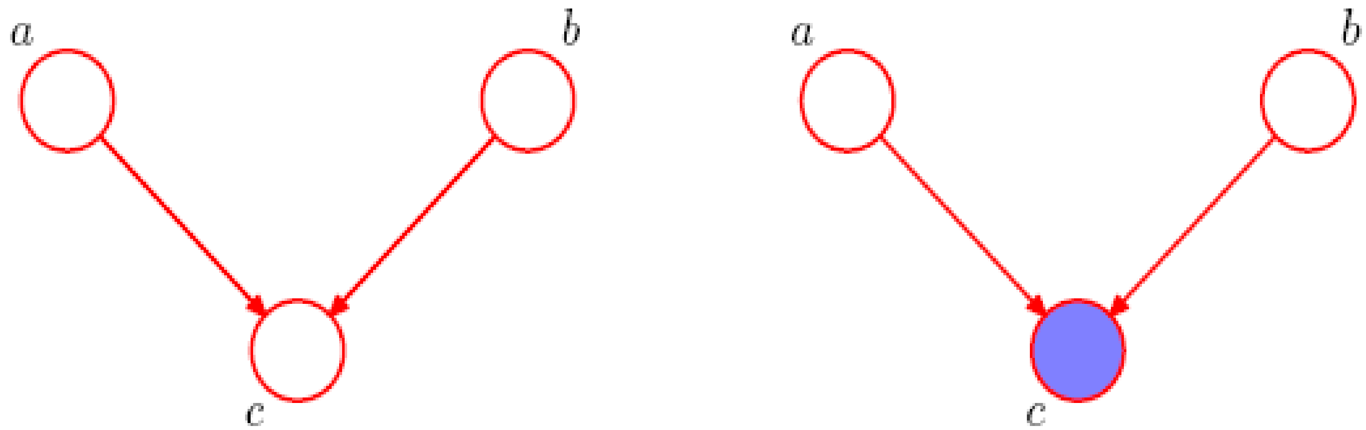
A Tale of Three Graphs - Part 3



- However, conditioned on c

$$\begin{aligned} p(a,b|c) &= \frac{p(a,b,c)}{p(c)} = \frac{p(a)p(b)p(c|a,b)}{p(c)} \\ &\neq p(a|c)p(b|c) \text{ in general} \end{aligned}$$

- So $a \not\perp\!\!\!\perp b | c$



- Frustratingly, the behaviour here is different
  - When c is not observed, path is blocked, a and b independent
  - When c is observed, path is unblocked, a and b not independent
- In this case c is head-to-head with respect to this path
- Situation is in fact more complex, path is unblocked if any descendent of c is observed

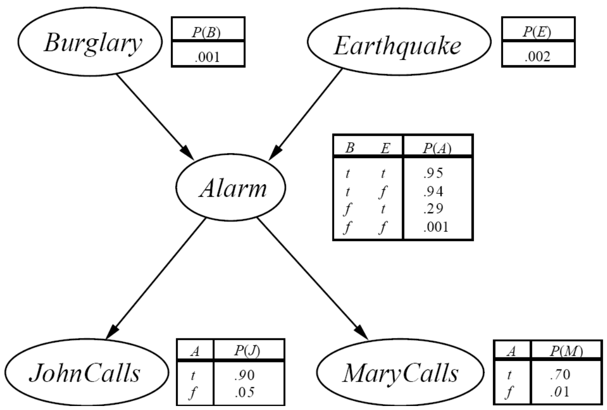
Constructing a Bayesian Network: Step 1

- Order the variables in terms of causality (may be a partial order)

  e.g., {E, B} -> {A} -> {J, M}

Constructing this Bayesian Network: Step 2

- P(J, M, A, E, B) =
  P(J | A) P(M | A) P(A | E, B) P(E) P(B)

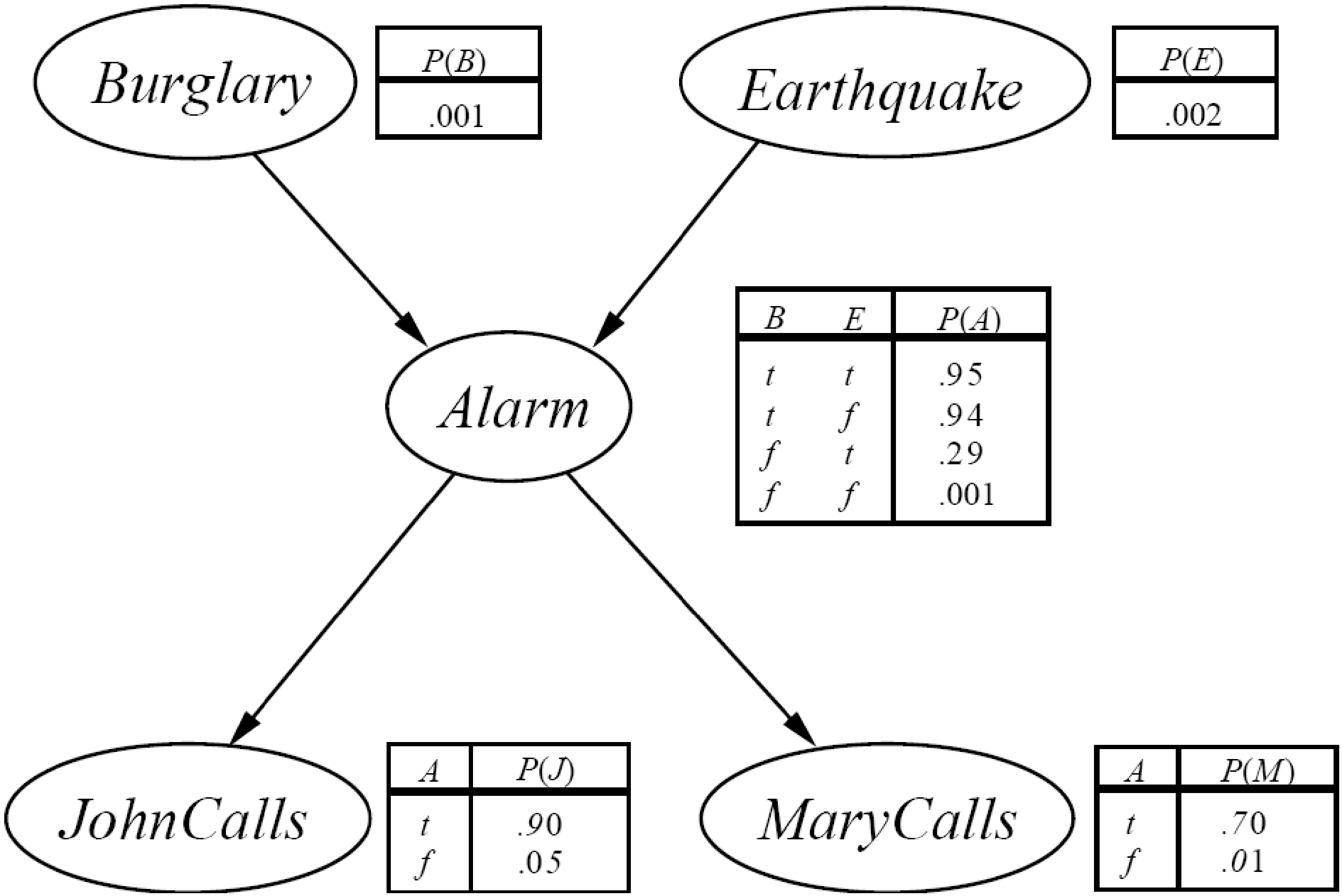


- There are 3 conditional probability tables (CPDs) to be determined:
  P(J | A), P(M | A), P(A | E, B)
  - Requiring 2 + 2 + 4 = 8 probabilities

- And 2 marginal probabilities P(E), P(B) -> 2 more probabilities

- Where do these probabilities come from?
  - Expert knowledge
  - From data (relative frequency estimates)
  - Or a combination of both - see discussion in Section 20.1 and 20.2 (optional)

The Bayesian network



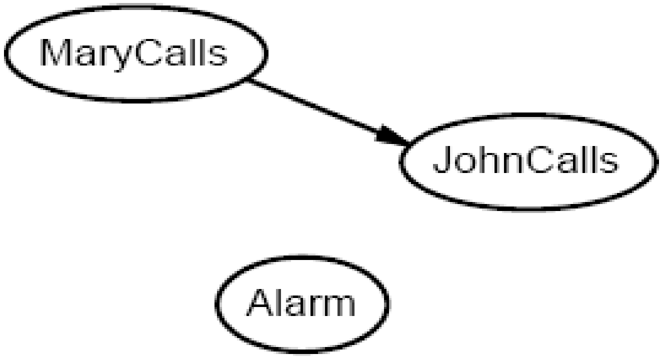
Number of Probabilities in Bayesian Networks

- Consider n binary variables
- Unconstrained joint distribution requires $O(2^n)$ probabilities
- If we have a Bayesian network, with a maximum of k parents for any node, then we need $O(n\ 2^k)$ probabilities
- Example
  - Full unconstrained joint distribution
    - $n = 30$: need 10^9 probabilities for full joint distribution
  - Bayesian network
    - $n = 30$, $k = 4$: need 480 probabilities

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

$P(J|M) = P(J)$?

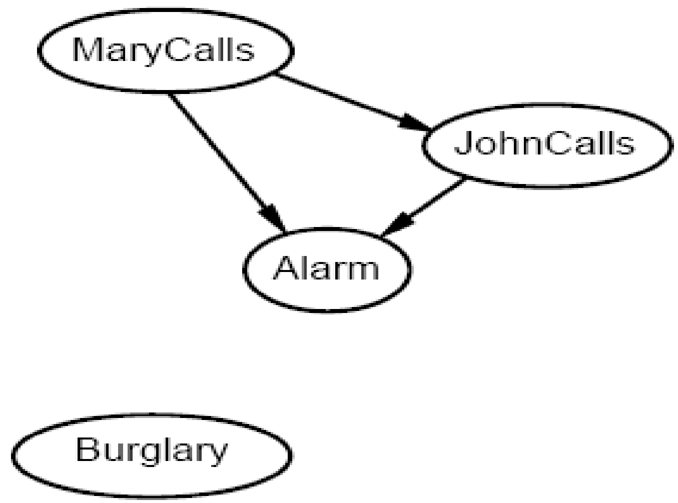
Suppose we choose the ordering M, J, A, B, E



$P(J|M) = P(J)$? No

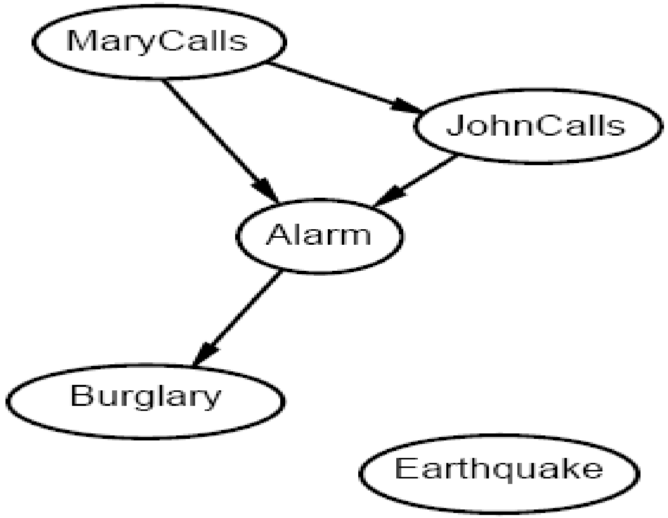
$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$?

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



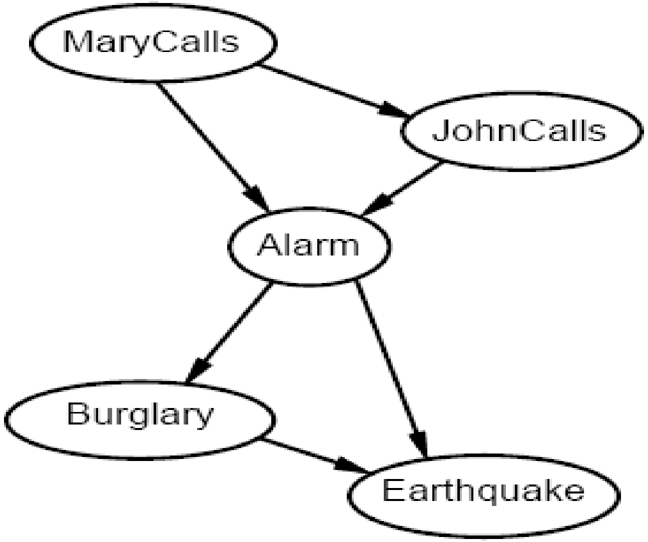
$P(J|M) = P(J)$? No
$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? No
$P(B|A, J, M) = P(B|A)$?
$P(B|A, J, M) = P(B)$?

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



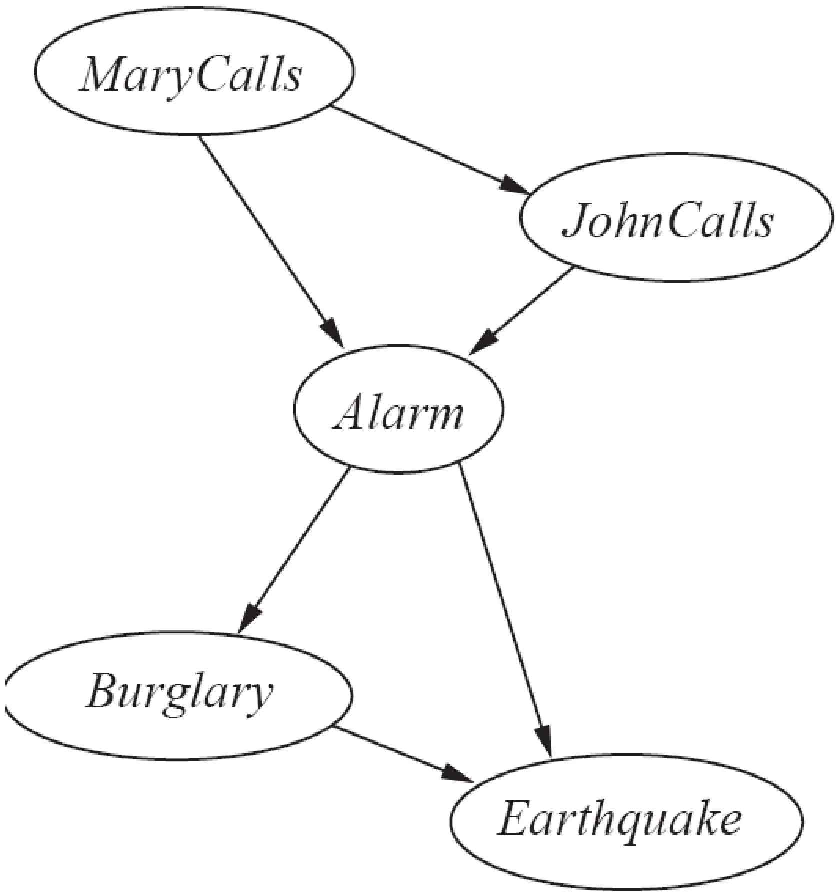
$P(J|M) = P(J)$? No
$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? No
$P(B|A, J, M) = P(B|A)$? Yes
$P(B|A, J, M) = P(B)$? No
$P(E|B, A, J, M) = P(E|A)$?
$P(E|B, A, J, M) = P(E|A, B)$?

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

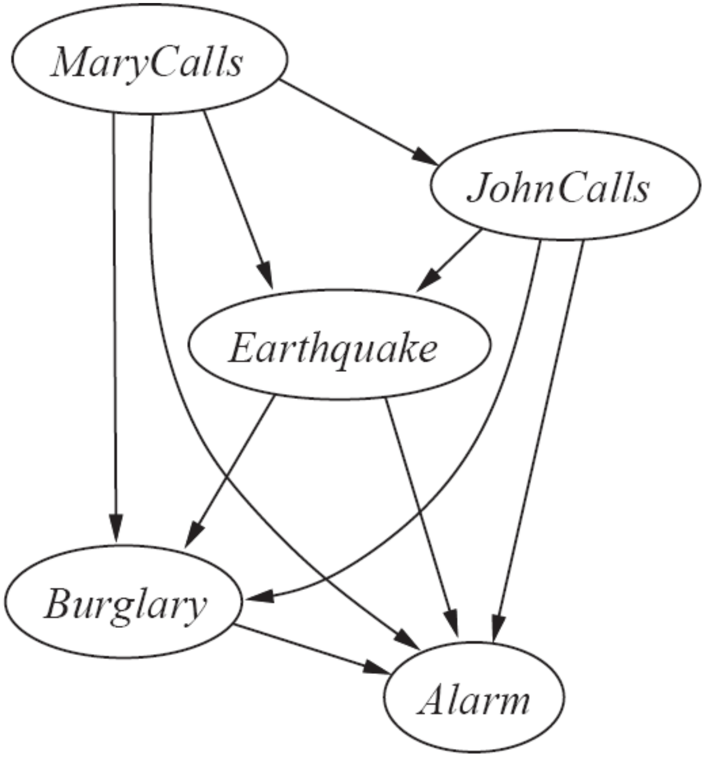
The Bayesian Network from a different Variable Ordering



(a)

The Bayesian Network from a different Variable Ordering

Order of {M, J,E,B,A }



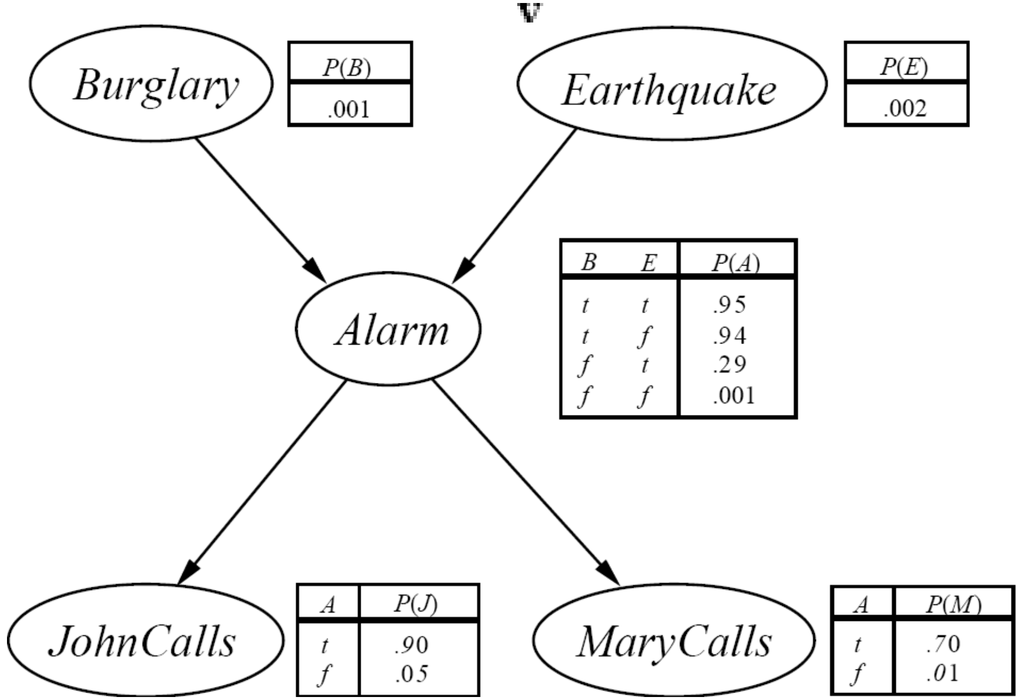
(b)

Inference in Bayesian Networks

Exact inference in BNs

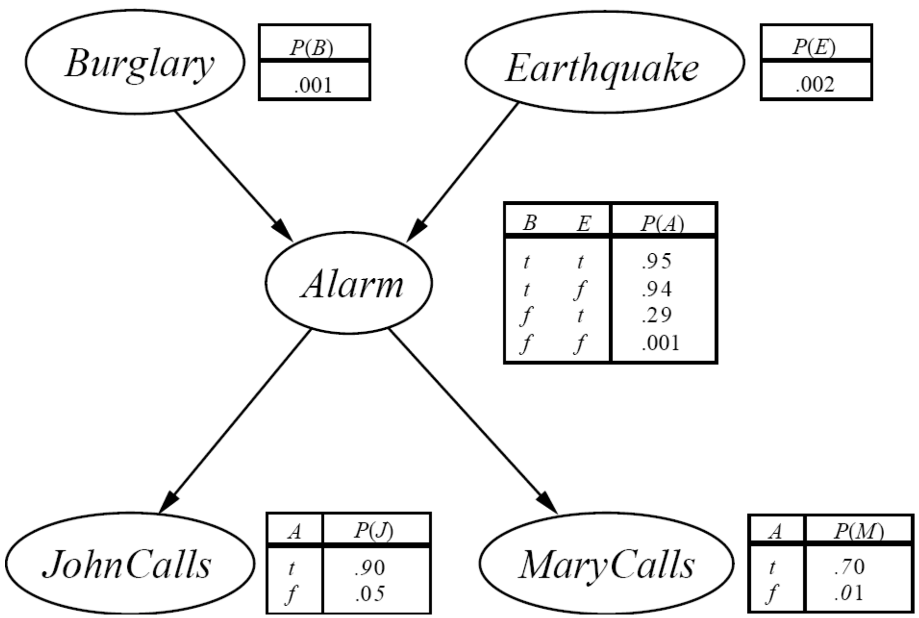
- A query P(X|e) can be answered using marginlization.

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



Inference by enumeration

$$\mathbf{P}(B|j,m) = \alpha\,\mathbf{P}(B,j,m) = \alpha \sum_e \sum_a \mathbf{P}(B,e,a,j,m)\,.$$



$$P(b|j,m) = \alpha \sum_e \sum_a P(b)P(e)P(a|b,e)P(j|a)P(m|a)$$

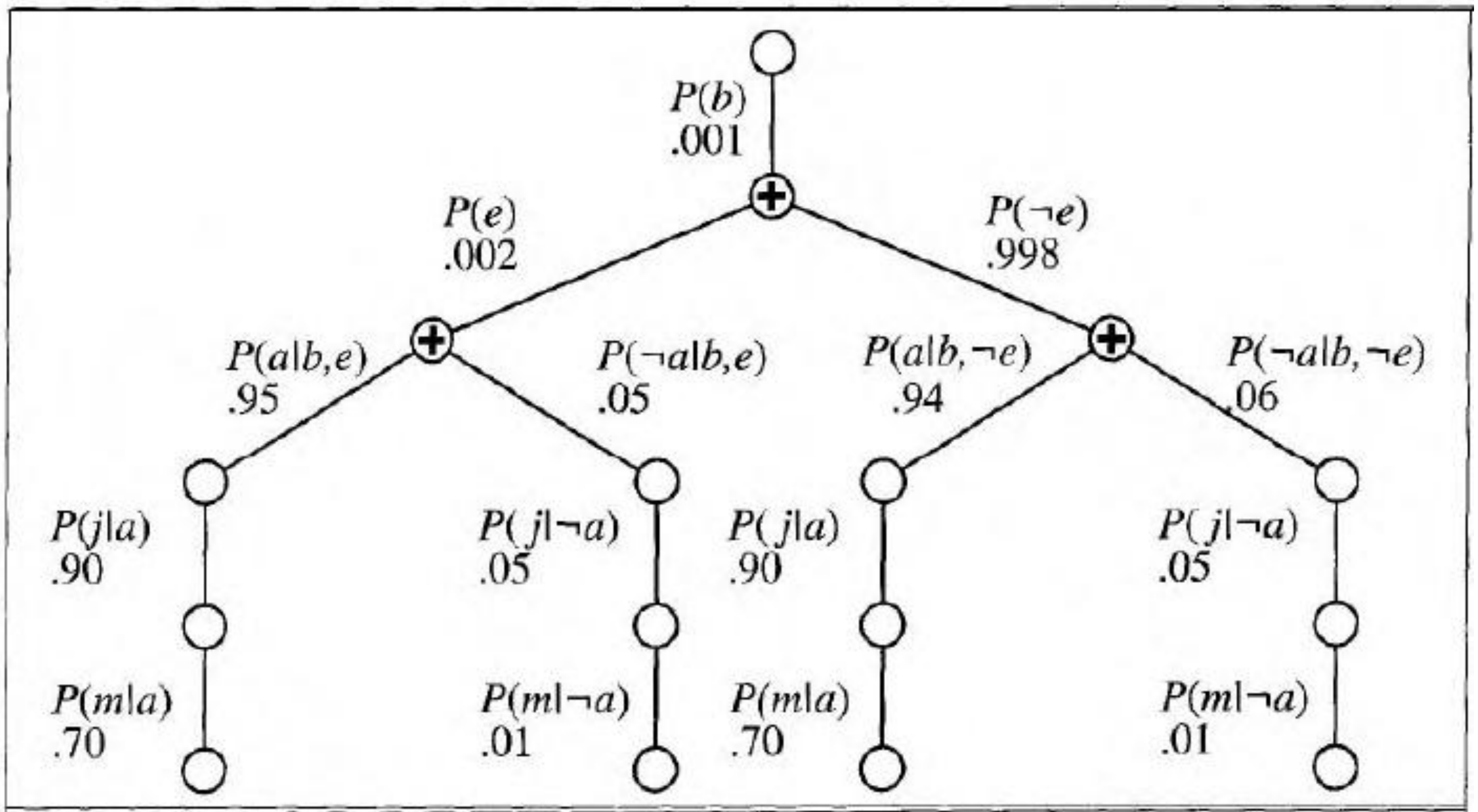
$$P(b|j,m) = \alpha \sum_{e} \sum_{a} P(b)P(e)P(a|b,e)P(j|a)P(m|a)$$

- We have to add 4 terms each have 5 multiplications.
- With n Booleans complexity is $O(n2^n)$
- Improvement can be obtained

$$P(b|j,m) = \alpha P(b) \sum_{e} P(e) \sum_{a} P(a|b,e)P(j|a)P(m|a) .$$

$$\mathbf{P}(B|j,m) = \alpha \langle 0.00059224, 0.0014919 \rangle \approx \langle 0.284, 0.716 \rangle .$$

Inference by enumeration



- What is the problem? Why is this inefficient ?

Variable elimination

$$\mathbf{P}(B|j,m) = \alpha\, \underbrace{\mathbf{P}(B)}_{B} \sum_e \underbrace{P(e)}_{E} \sum_a \underbrace{\mathbf{P}(a|B,e)}_{A} \underbrace{P(j|a)}_{J} \underbrace{P(m|a)}_{M} .$$

- Store values in vectors and reuse them.

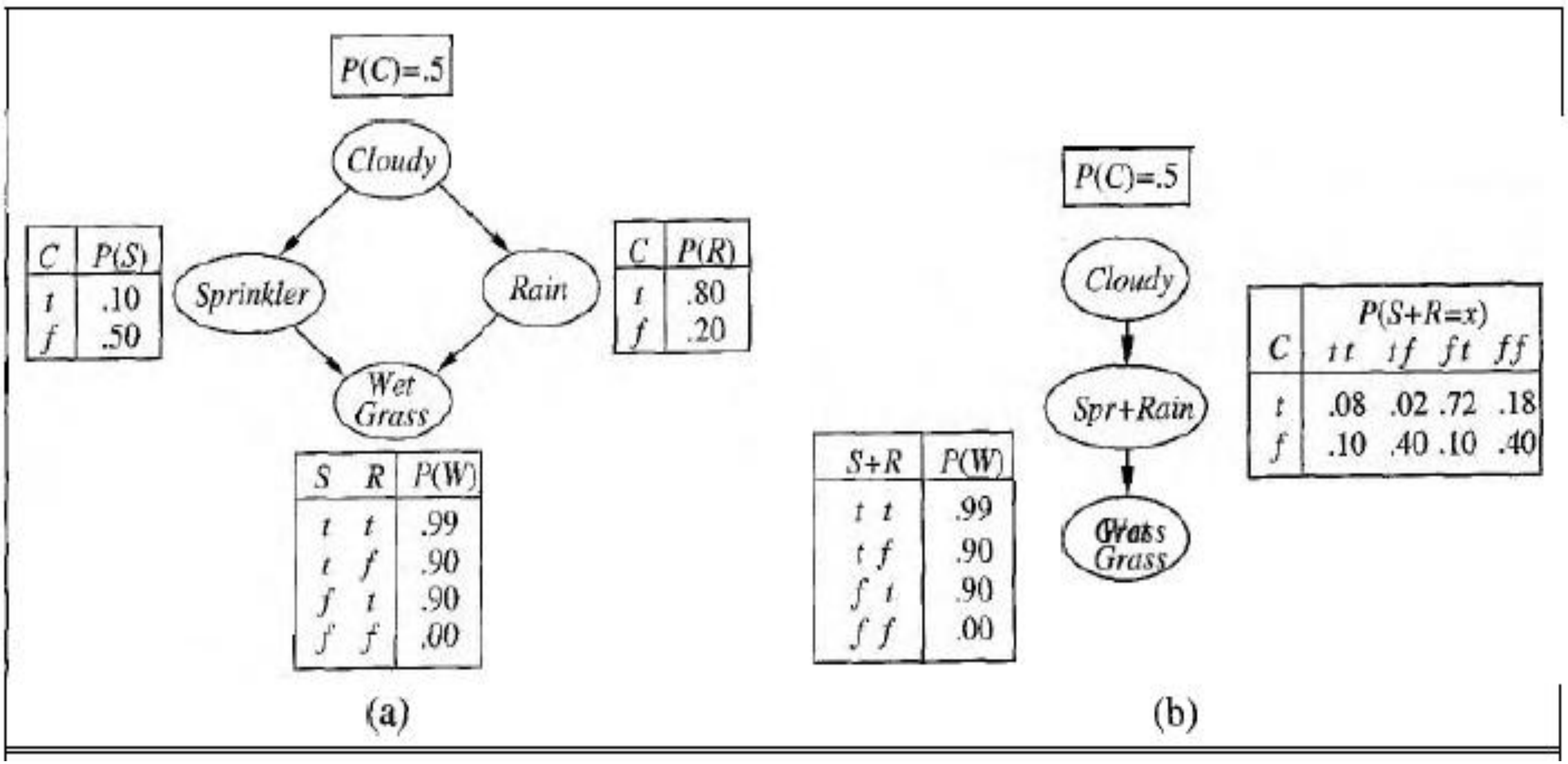
$$\mathbf{f}_M(A) = \begin{pmatrix} P(m|a) \\ P(m|\neg a) \end{pmatrix}$$

Complexity of exact inference

- Polytree: there is at most one undirected path between any two nodes. Like Alarm.
- Time and space complexity in such graphs is linear in n
- However for multi-connected graphs (still dags) its exponential in n.

Clustering Algorithm

- If we want to find posterior probabilities for many queries.



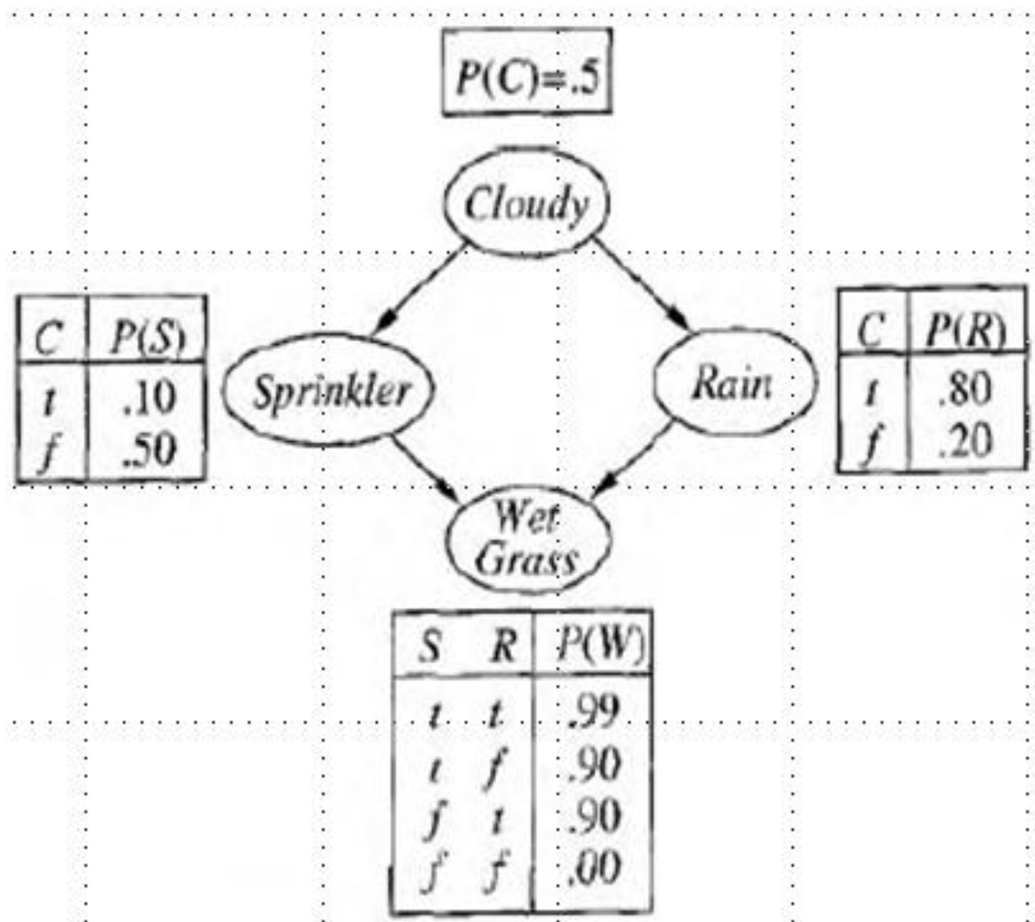
Approximate inference in BNs

- Give that exact inference is intractable in large networks. It is essential to consider approximate inference models
  - Discrete sampling method
  - Rejection sampling method
  - Likelihood weighting
  - MCMC algorithms

Discrete sampling method

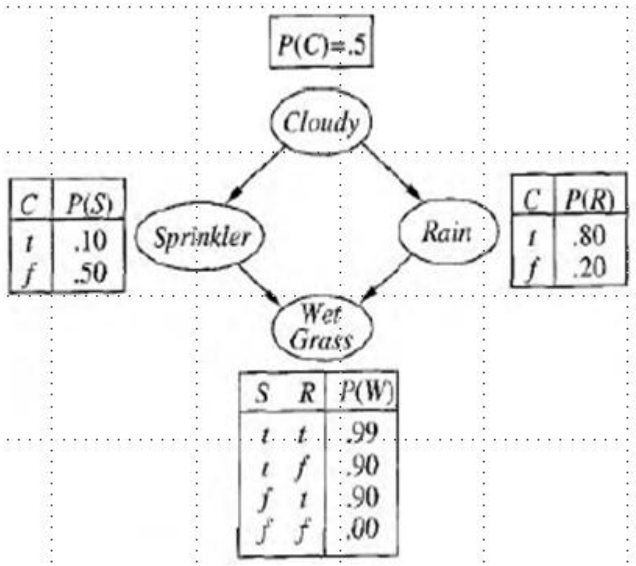
- Example : unbiased coin
- Sampling this distribution
  - Flipping the coin.. Flip the coin 1000 times
  - Number of heads / 1000 is an approximation of p(head)

Discrete sampling method



Discrete sampling method

- P(cloudy)= < 0.5 , 0.5 > suppose T
- P(sprinkler|cloudy=T)= < 0.1 , 0.9 > suppose F
- P(rain|cloudy =T) = < 0.8 , 0.2 > suppose T
- P(W| Sprinkler=F, Rain=T) = < 0.9 , 0.1 > suppose T



- [True, False, True, True]

Discrete sampling method

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

x ← an event with n elements
for i = 1 **to** n **do**
x_i ← a random sample from $\mathbf{P}(X_i \mid parents(X_i))$
return x

Figure 14.12 A sampling algorithm that generates events from a Bayesian network.

Discrete sampling method

- Consider p(T, F, T, T)= 0.5 * 0.9 * 0.8 * 0.9 = 0.324.
- Suppose we generate 1000 samples
  - p(T, F, T, T) = 350/1000
  - P(T) = 550/1000

$$\lim_{N \to \infty} \frac{N_{PS}(x_1, \ldots, x_n)}{N} = S_{PS}(x_1, \ldots, x_n) = P(x_1, \ldots, x_n) .$$

  - Problem?

Rejection sampling in BNs

- Is a general method for producing samples from a hard to sample distribution.
  - Suppose $p(X|e)$. Generate samples from prior distribution then reject the ones that do not match evidence.

Rejection sampling in BNs

function REJECTION-SAMPLING(X, **e**, **bn**, N) **returns** an estimate of $P(X|\mathbf{e})$
  inputs: X, the query variable
    e, evidence specified as an event
    bn, a Bayesian network
    N, the total number of samples to be generated
  local variables: **N**, a vector of counts over X, initially zero

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

Figure 14.13 The rejection sampling algorithm for answering queries given evidence in a Bayesian network.

Rejection sampling in BNs

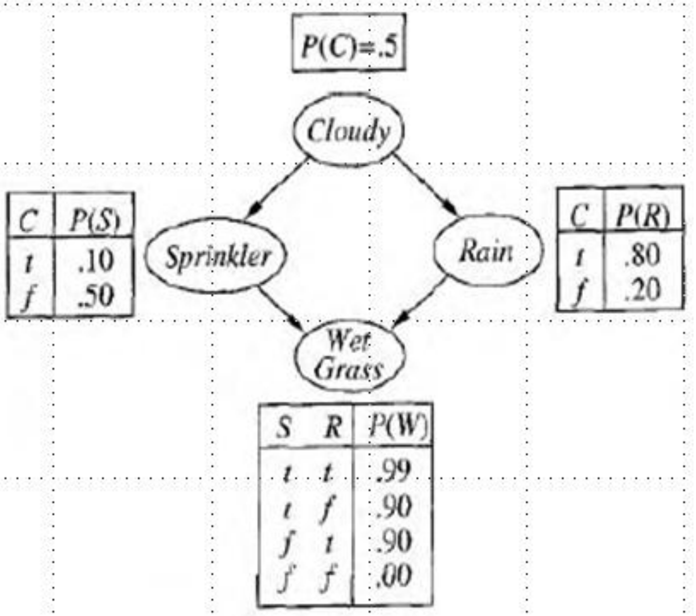
- P(rain | sprinkler =T) using 1000 samples
  - Suppose 730 of them sprinkler = false of the 270
  - 80 rain = true and 190 rain = false
  - P(rain |sprinkler =true) Normalize(8,19) = $\langle 0.296, 0.704 \rangle$
  - Problem?
    - Rejects so many samples
    - Hard to sample rear events
    - P(rain | redskyatnight=T)

Likelihood weighting

P(C,S,W,R) = P(C) * P(S|C) * P(R|C) * P(W|S,R)

Now suppose we want samples that S and W are true

Z= {C,W}   e={S,W}



P(C,S,W,R) = $\prod p(z_i \mid parents(z_i)) \prod p(e_j \mid parents(e_j))$

Likelihood weighting

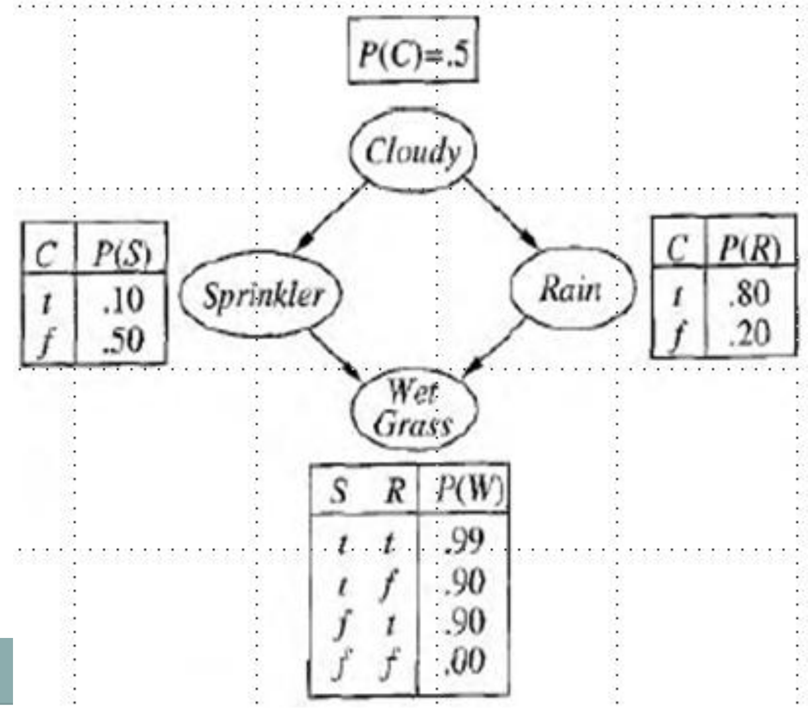
P(C,S,W,R) = $\prod p(z_i \mid parents(z_i)) \prod p(e_j \mid parents(e_j))$

Sample this part $S_{WS}(\mathbf{z}, \mathbf{e}) = \prod_{i=1}^{l} P(z_i | parents(Z_i))$

Calculate this part from model $w(\mathbf{z}, \mathbf{e}) = \prod_{i=1}^{m} P(e_i | parents(E_i))$.

Likelihood weighting

- Generate only events that are consistent with evidence
  - Fix values for Evidence and only sample query variables.
  - Weight the samples based on the likelihood of the event according to the evidence.
  - P(rain|sp=T, WG=T)
    - Sample p(cl) → <0.5, 0.5 >
    - w <- w * p(sp=T||cl =T)=0.1
    - p(rain|cloudy=T) <0.8, 0.2>
    - w <- w * p(WG|SP =T, R=T)= 0.099



Likelihood weighting

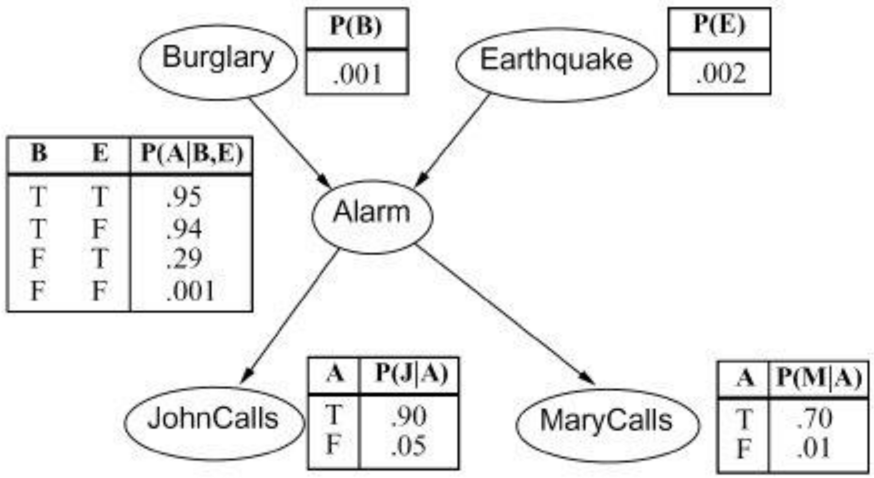
- Examining sampling distribution over variables that are not part of evidence

$\mathbf{P}(Rain|Sprinkler = true, WetGrass = true)$.

w is set to 1.0. Then an event is generated:

1. Sample from $\mathbf{P}(Cloudy) = \langle 0.5, 0.5 \rangle$; suppose this returns *true*.
2. *Sprinkler* is an evidence variable with value *true*. Therefore, we set
$$w \leftarrow w \times P(Sprinkler = true|Cloudy = true) = 0.1 .$$
3. Sample from $\mathbf{P}(Rain|Cloudy = true) = \langle 0.8, 0.2 \rangle$; suppose this returns *true*.
4. *WetGrass* is an evidence variable with value *true*. Therefore, we set
$$w \leftarrow w \times P(WetGrass = true|Sprinkler = true, Rain = true) = 0.099$$

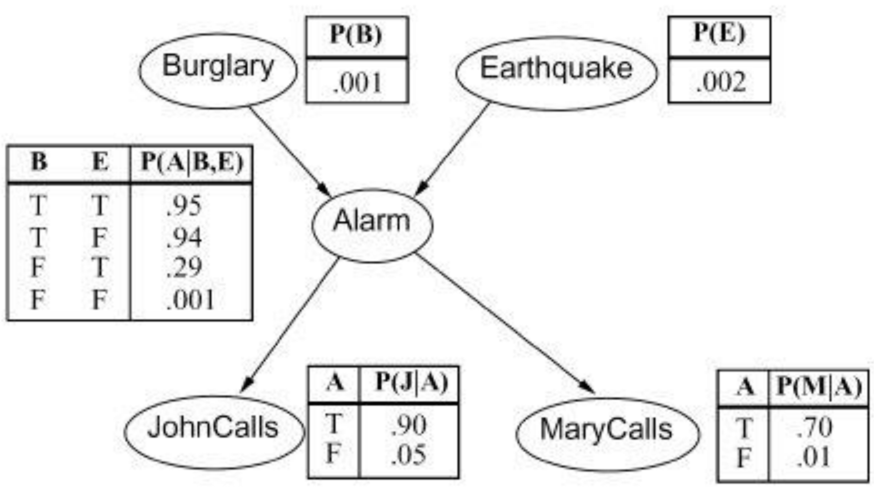
example



We keep the following table

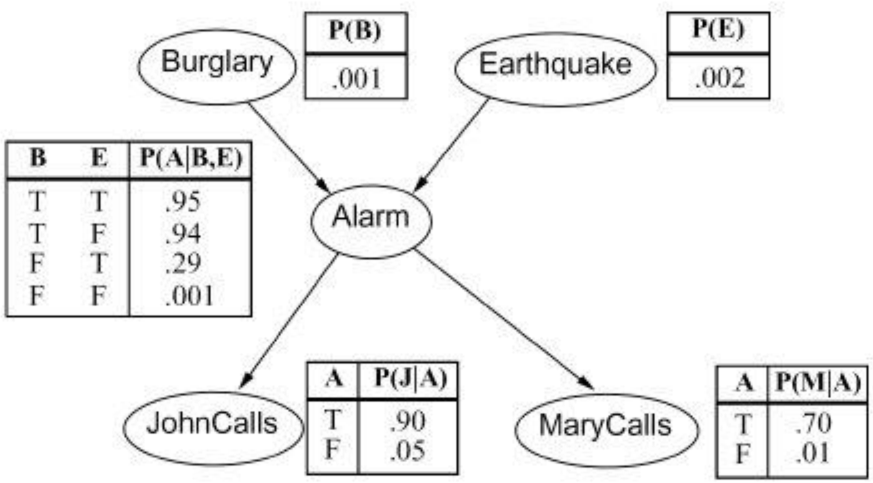
W						
Sample	**Key**					**Weight**
1	~b	~e	~a	~j	~m	0.997

If the same key happens more than once, we add weights



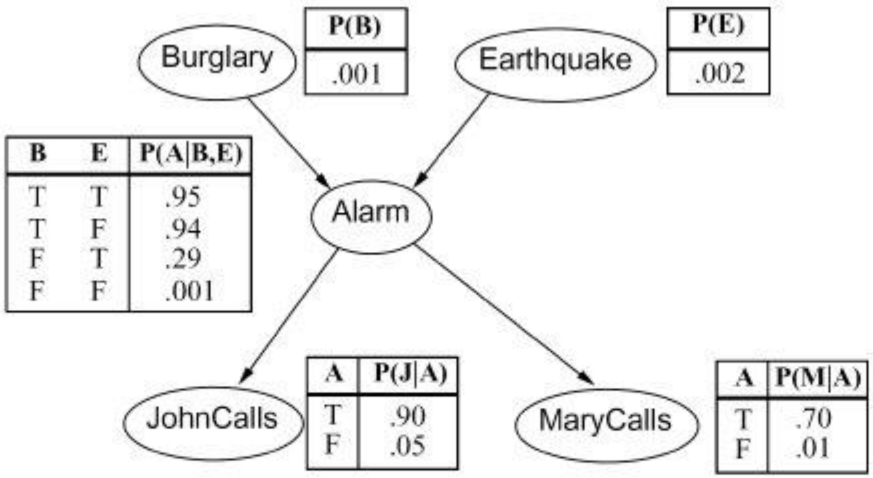
Evidence is *Burglary=false* and *Earthquake=false*

W						
Sample	**Key**					**Weight**
1	~b	~e	~a	~j	~m	0.997



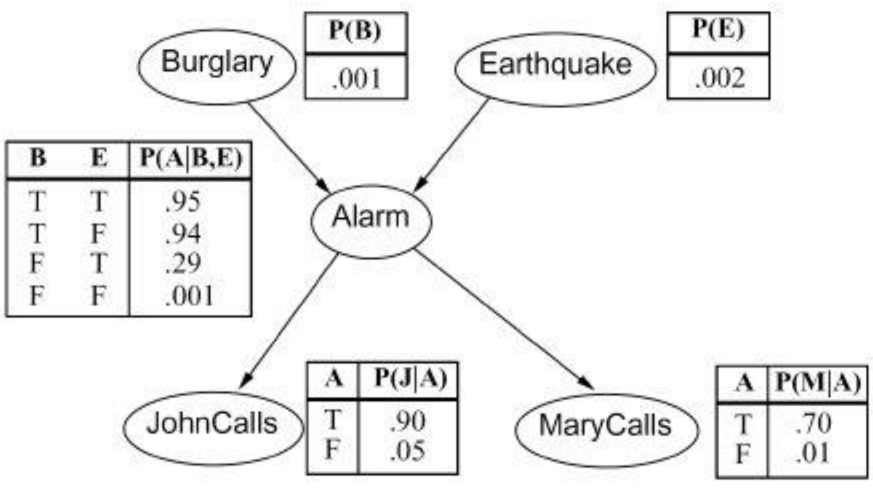
Evidence is *Alarm=false* and *JohnCalls=true*.

W						
Sample	**Key**					**Weight**
1	~b	~e	~a	~j	~m	0.997
2	~b	~e	~a	j	~m	0.05



Evidence is *JohnCalls=true* and *MaryCalls=true*.

W						
Sample	**Key**					**Weight**
1	~b	~e	~a	~j	~m	0.997
2	~b	~e	~a	j	~m	0.05
3	~b	~e	a	j	m	0.63



Evidence is *Burglary=true* and *Earthquake=false*.

W						
Sample	**Key**					**Weight**
1	~b	~e	~a	~j	~m	0.997
2	~b	~e	~a	j	~m	0.10
3	~b	~e	a	j	m	0.63
4	b	~e	~a	~j	~m	0.001

Using Likelihood Weights

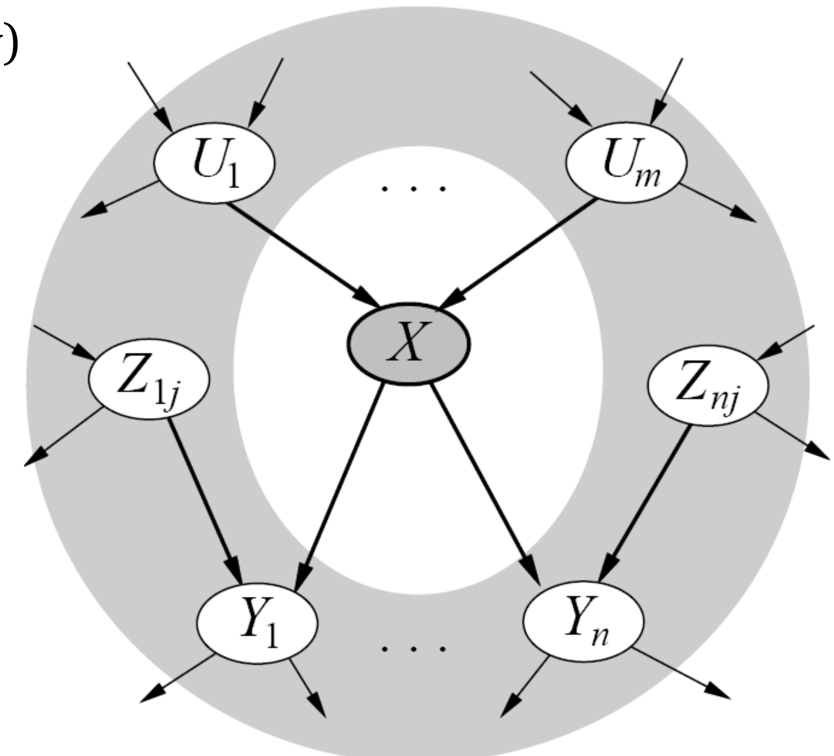
W						
Sample	**Key**					**Weight**
1	~b	~e	~a	~j	~m	0.997
2	~b	~e	~a	j	~m	0.10
3	~b	~e	a	j	m	0.63
4	b	~e	~a	~j	~m	0.001

P(*Burglary*=*true*) = (0.001) / (0.997 + 0.10 + 0.63 + 0.001) = 0.00058

p(Alarm =true | johncall = true) = 0.63 / (0.10 + 0.63) = 0.63 / 0.73 = 0.863

Given a graph, can we "read off" conditional independencies?

A node is conditionally independent of all other nodes in the network given its Markov blanket (in gray)



The MCMC algorithm

- Markov Chain Monte Carlo
  - Assume that calculating p(x|markovblanket(x)) is easy
  - Unlike other samplings which generate events from scratch, MCMC makes a random change to the preceding event.
  - At each step a value is generated for one of the non evidence variables condition on its markov blanket.

The MCMC algorithm

- Example estimate P(R|SP =T , WG=T) using MCMC
- Initialize the other variables randomly consistent with query [T, T, F, T]
- Sample non evidence variables.
  - P(C| S =T, R=F) → [40,60] assume cl =F
  - [F,T,F,T]
  - P(R|CL= F, SP =T, WG=T) → assume rain =T
  - [F,T,T,T]
  - Sample CL again..