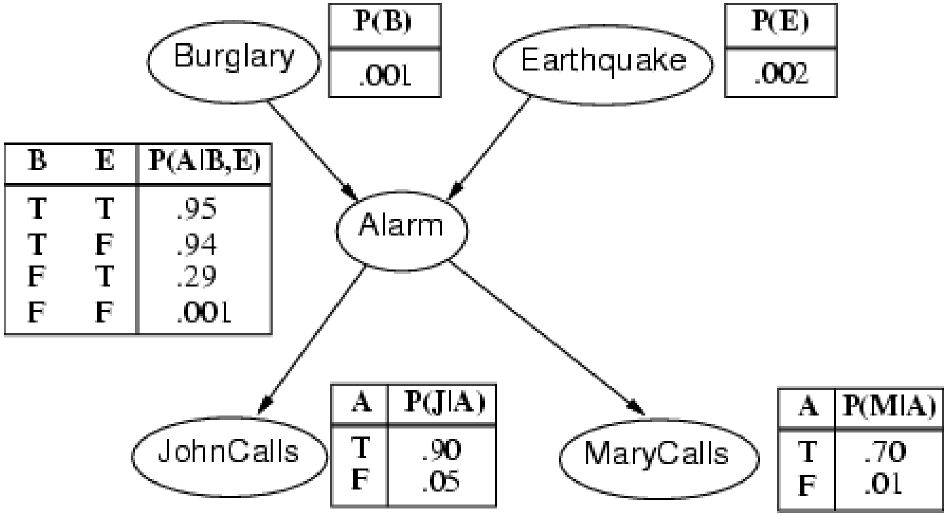
CSE 473

Lecture 22
(Chapters 14 & 15)

Probabilistic Inference in Bayesian Networks



Example: Burglars and Earthquakes

- You are at a “Done with the AI class” party.
- Neighbor John calls to say your home alarm has gone off (but neighbor Mary doesn't).
- Sometimes your alarm is set off by minor earthquakes.
- Question: Is your home being burglarized?

Recall: Probabilistic Inference

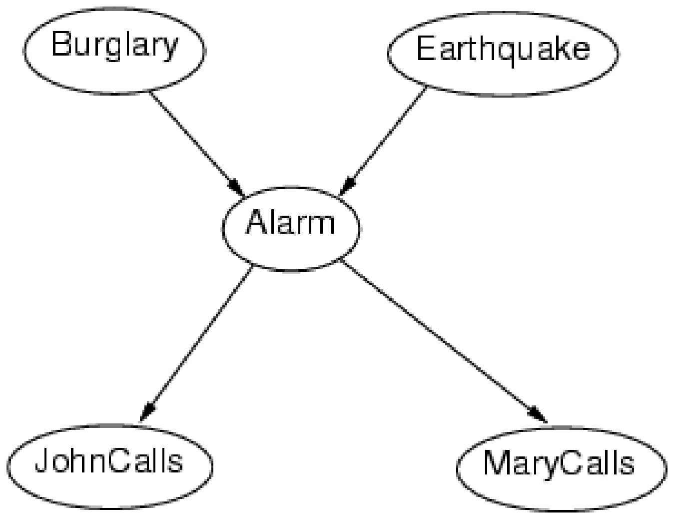
- Variables: *Burglary*, *Earthquake*, *Alarm*, *JohnCalls*, *MaryCalls (shorthand: B, E, A, J, M)*
- Full joint distribution allows inference of all types of probabilities
  - E.g. Given random variables A, B, E, J, M, we want P(B|J,M):

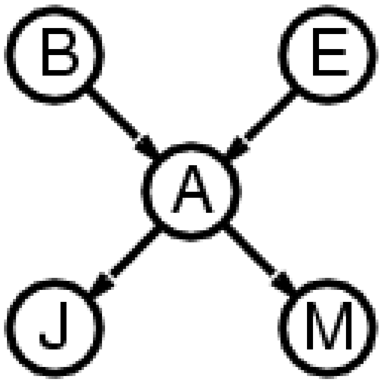
$$P(B|J,M) = \alpha\, P(B,J,M) = \alpha \sum_{E,A} P(B,J,M,E,A)$$

- Problem: Full joint requires you to specify 2*2*2*2*2 = 32 values

Bayesian Network Idea

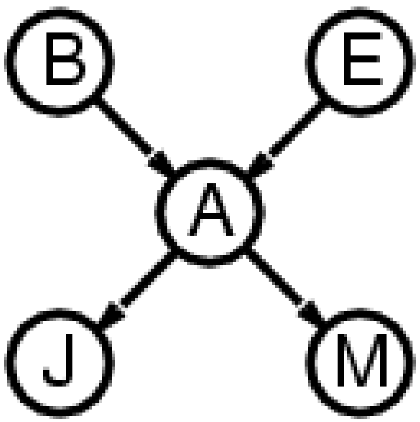
- Network topology reflects "causal" knowledge:
  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause Mary to call
  - The alarm can cause John to call





Bayesian networks

- Simple graphical notation for conditional independence assertions
  - In many cases, allows *compact specification* of full joint distributions



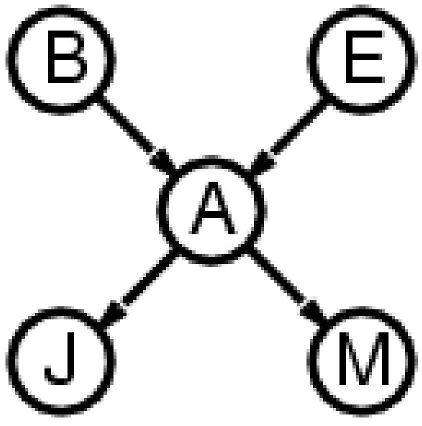
$P(J,M,A,B,E) =$

$\prod_i P(X_i \mid Parents(X_i)) =$

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

Only requires 2+2+4+1+1=10 values

Why is joint = $\pi_i\ P(X_i \mid Parents(X_i))$?



Keep applying definition of conditional probability and use network topology to simplify:

$P(J,M,A,B,E) =$

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

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

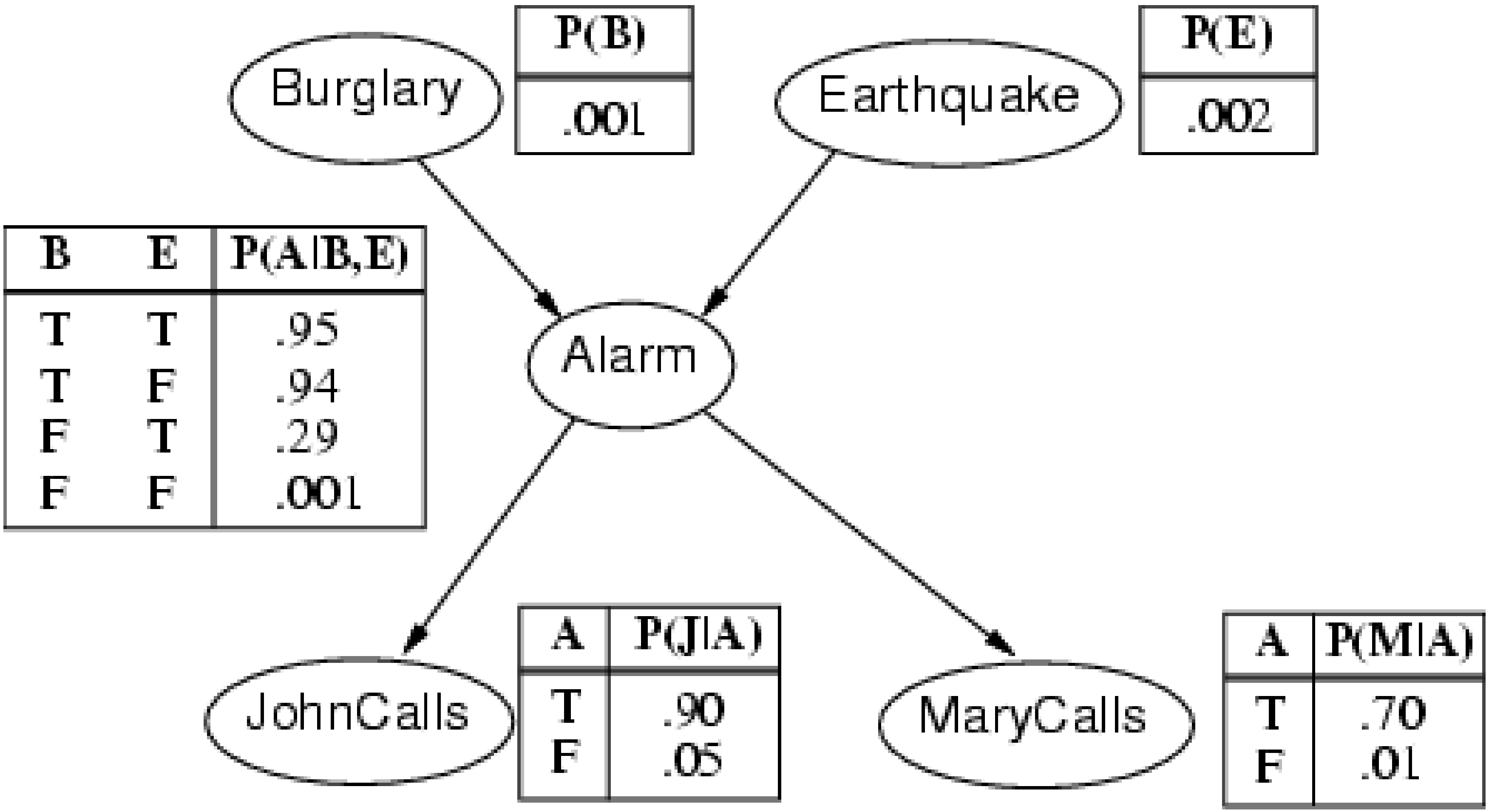
$= P(J|A)\ P(M|A,B,E)\ P(A,B,E)$

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

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

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

Full Specification of Bayesian Network for Burglars and Earthquakes



P(B)
.001

P(E)
.002

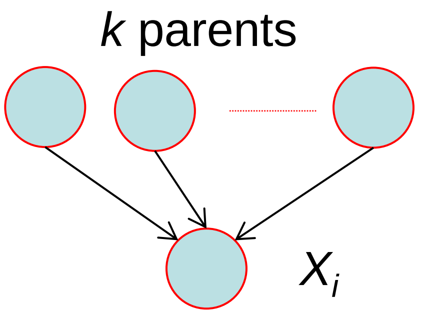
B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

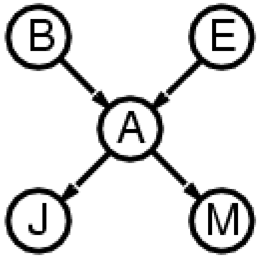
A	P(M|A)
T	.70
F	.01

Compact Representation of Probabilities in Bayesian Network

- A conditional probability table (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 other number for X_i = *false* is just $1-p$)



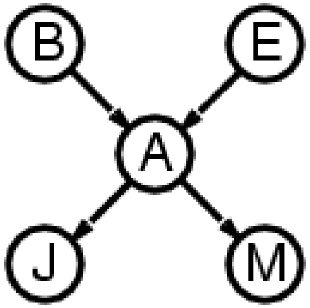
- If each variable has no more than k parents, an n-variable network requires $O(n \cdot 2^k)$ numbers
  - This grows linearly with n vs. $O(2^n)$ for full joint distribution
- For burglar network, 1+1+4+2+2 = 10 numbers (vs. 2^5-1 = 31 numbers) for full joint distribution



Bayesian Network Semantics

- Full joint distribution is defined as product of local conditional distributions:

$$P(X_1, \dots, X_n) = \pi_{i=1}^{n} P(X_i \mid Parents(X_i))$$



- e.g., Joint probability of all variables being true = ?

$$P(j \wedge m \wedge a \wedge b \wedge e)$$

$$= P(j \mid a)\, P(m \mid a)\, P(a \mid b, e)\, P(b)\, P(e)$$

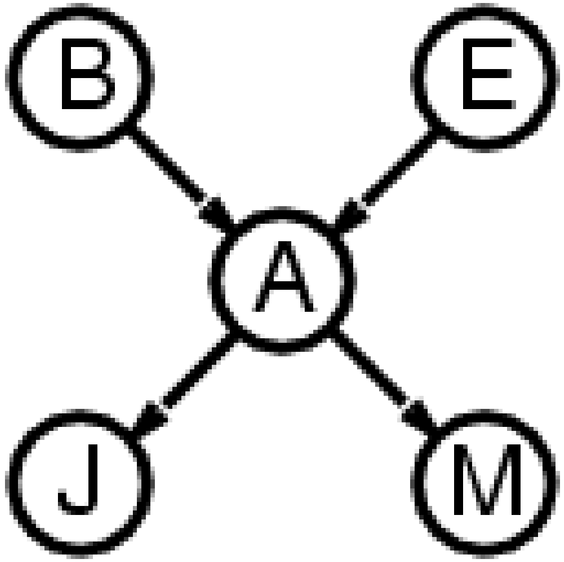
- Similarly, $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

$$= P(j \mid a)\, P(m \mid a)\, P(a \mid \neg b, \neg e)\, P(\neg b)\, P(\neg e)$$

Probabilistic Inference in BNs

- The graphical independence representation yields efficient inference schemes
- We generally want to compute
  - $P(X|E)$ where E is the evidence from sensory measurements (known values for variables)
  - Sometimes, may want to compute just $P(X)$
- One simple inference algorithm:
  - *variable elimination (VE)*

Compute P(B=true | J=true, M=true)

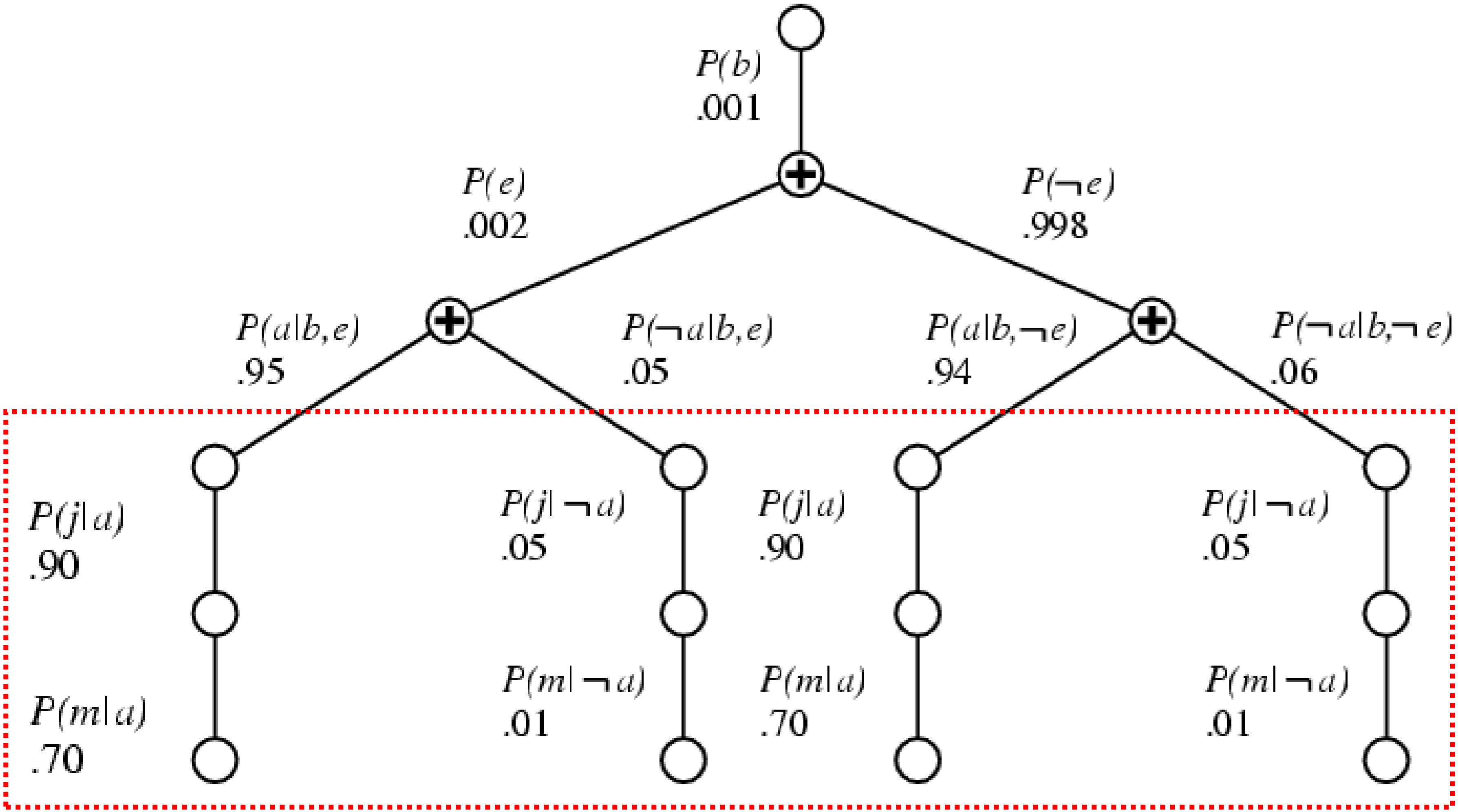


$$P(b|j,m) = \alpha \textstyle\sum_{e,a} P(b,j,m,e,a)$$

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

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

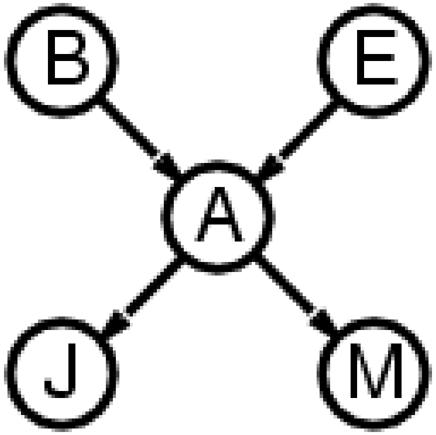
Structure of Computation



$P(b|j,m) = \alpha P(b) \Sigma_e P(e) \Sigma_a P(a|b,e)P(j|a)P(m|a)$

Repeated computations ⇒ use dynamic programming

Can we derive a general inference algorithm?



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

- Join all factors containing *a*
- Sum out *a* to get new function of b,e,j,m only

Variable Elimination (VE) Algorithm

- Eliminate variables one-by-one until there is a factor with only the query variables:
  1. *join* all factors containing that variable, multiplying probabilities
  2. *sum out* the influence of the variable

Function of b,j,m

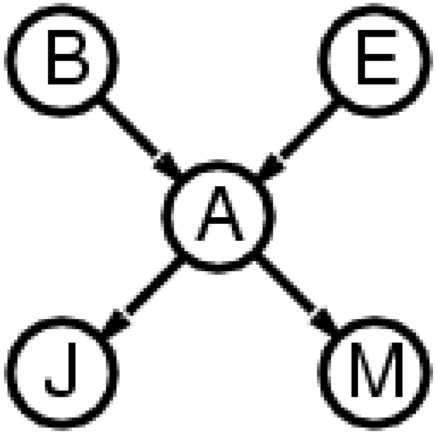
Remaining factor is a function of b, j, m

Eliminate e

Eliminate a

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

Example of VE: P(J)



$P(J)$

$= \Sigma_{M,A,B,E}\ P(J,M,A,B,E)$

$= \Sigma_{M,A,B,E}\ P(J|A)P(M|A)\ P(A|B,E)\ P(B)\ P(E)$

$= \Sigma_A P(J|A)\ \Sigma_M P(M|A)\ \Sigma_B P(B)\ \Sigma_E P(A|B,E)P(E)$

$= \Sigma_A P(J|A)\ \Sigma_M P(M|A)\ \Sigma_B P(B)\ f1(A,B)$

$= \Sigma_A P(J|A)\ \Sigma_M P(M|A)\ f2(A)$

$= \Sigma_A P(J|A)\ f3(A)$

$= f4(J)$

Other Inference Algorithms?

- Direct Sampling:
  - Repeat N times:
    - Use random number generator to generate sample values for each node
    - Start with nodes with no parents
    - Condition on sampled parent values for other nodes
  - Count frequencies of samples to get an approximation to desired distribution
- Other variants: Rejection sampling, likelihood weighting, Gibbs sampling and other MCMC methods (see text)
- Belief Propagation: A "message passing" algorithm for approximating P(X|evidence) for each node variable X
- Variational Methods: Approximate inference using distributions that are more tractable than original ones

(see text for details)

Next Time

- HMMs and Forward Algorithm for Inference
- Particle Filtering
- To Do:
  - Project 4 (last project!)