# Course Logistics

- Happy Election Day!
- We're still working on MP1…
- The MP was intractably difficult
- Instead, HW4 on BNs was posted on the web site yesterday; it is due a week from today.

# BN Construction

- Identify variables
- Order them*
- While there are variables to add
  - Pick the next in the ordering
  - Identify its parents in the net
    - Hold all others constant (in every configuration)
    - If net variable influences it, net var is a parent
  - Draw all arcs and add CPT

* order matters a lot

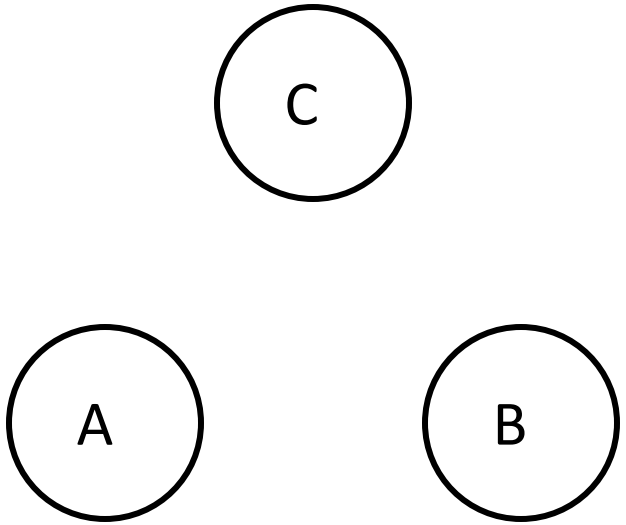
# Dentist Example

3 Boolean Random Variables:

C – Patient has a cavity

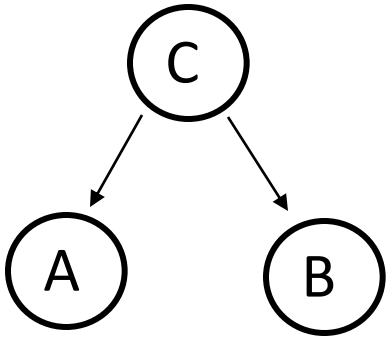
A – Patient reports a toothache

B – Dentist's probe catches on tooth



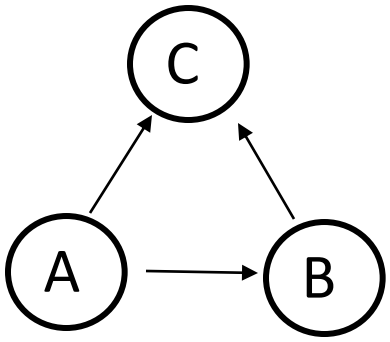
# Effect of Order on BN structure

Order: C A B



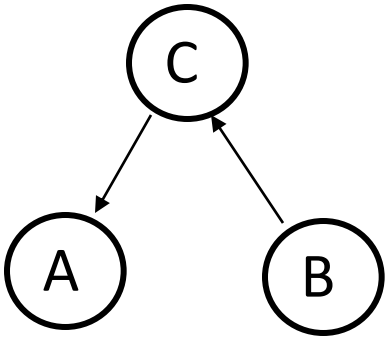

5 parameters

Order: A B C




7 parameters
no savings over
the joint

Order: B C A




5 parameters

# BN Structure

consider ordering M,J,E,B,A

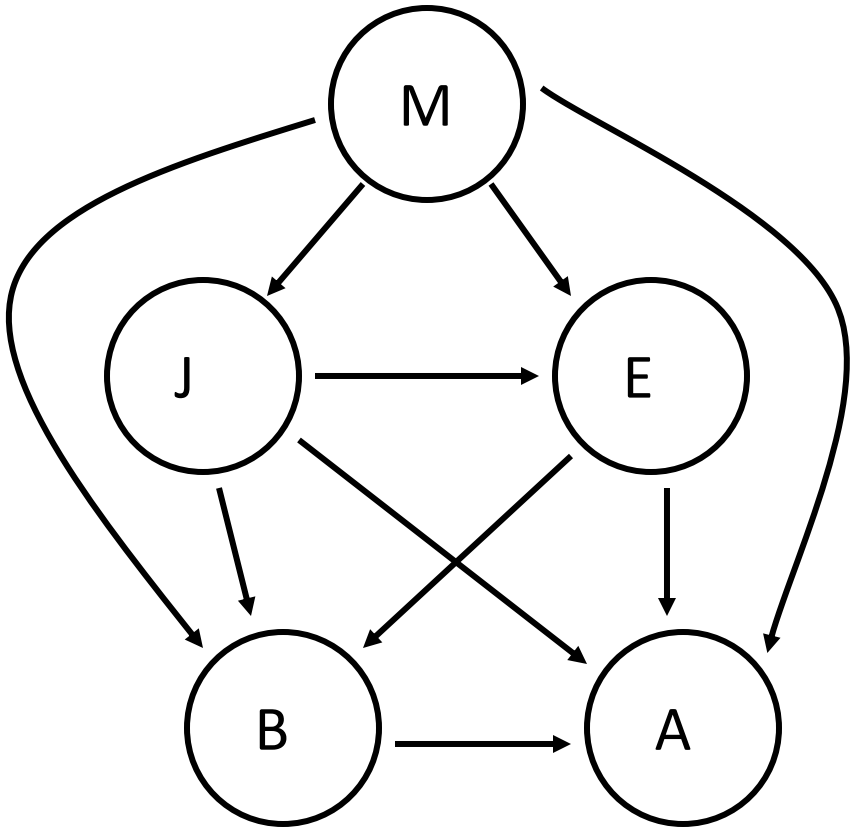
B – a burglary is in progress

E – an earthquake is in progress

A – the alarm is sounding

J – John calls

M – Mary calls




All variables are Boolean

How many numbers?

No savings over the Joint

What conditional independence assumptions does it make?

# Ordering is Important

- Original net B E A J M
  - 10 parameters
  - Saves 21 over the joint
- Ordering M J E B A
  - 31 parameters
  - No savings over the joint
- “Causality” ordering is only heuristic
- Suppose we want to test all orderings of n random variables…

# Proof of Adequacy for BNs

- Recover the joint
- Conditionalize on evidence
- Marginalize across unknowns
- Brute Force calculation of the Answer
- May be intractable for large BNs
- Joint has $a^n$ for n RV's, "a" discrete values each
- Earthquake examples were all much simpler
  - Did not recover the full joint
  - Did not touch irrelevant values

# Space Complexity of Distribution Data Structures

BNs win big if direct interdependencies (fan in) are bounded (limited by k)

Joint complexity:

$O(a^n)$: $2^n-1$ for n Boolean variables

BN w/ at most k parents / node:

$O(n)$: $\leq n \cdot 2^k$ for Boolean variables

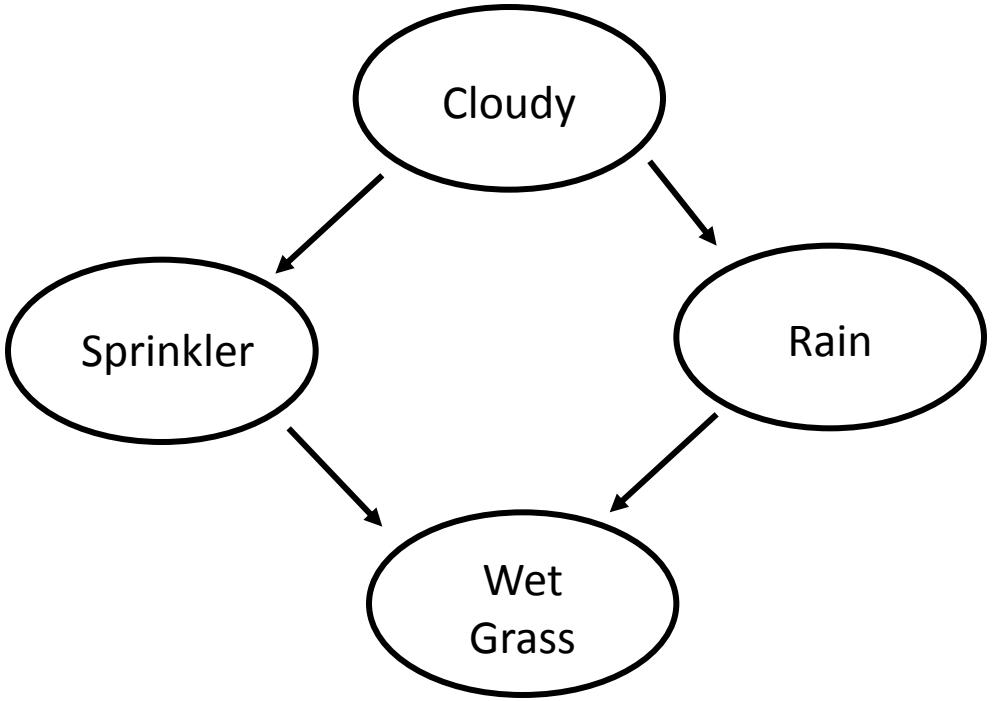
# Time Complexity of Inference

- Efficient exact algorithms for any query (evidence/query pairs) for polytrees*

  * at most one undirected path between any two nodes

- Inference in non-polytrees can be exponential
  - Even approximate inference
  - Often works well in practice
  - Worst case is #P-hard (worse than NP complete)

# Non-Polytrees



# Non-Polytrees

## Exact inference - clustering

- Join tree / junction tree algorithm
- Restore polytree property
- Identify clusters of nodes (super nodes)
- Collect all parents
- Exponential in cluster size
  (context grows as product of contexts)

# Non-Polytrees

## approximate inference

- Loopy belief propagation
  - Pretend it is a poly-tree
  - No guarantee of convergence / correctness
  - Surprisingly effective
- Sampling algorithms
  - Generate examples from the posterior
  - Given the evidence
  - Estimate the distribution by counting

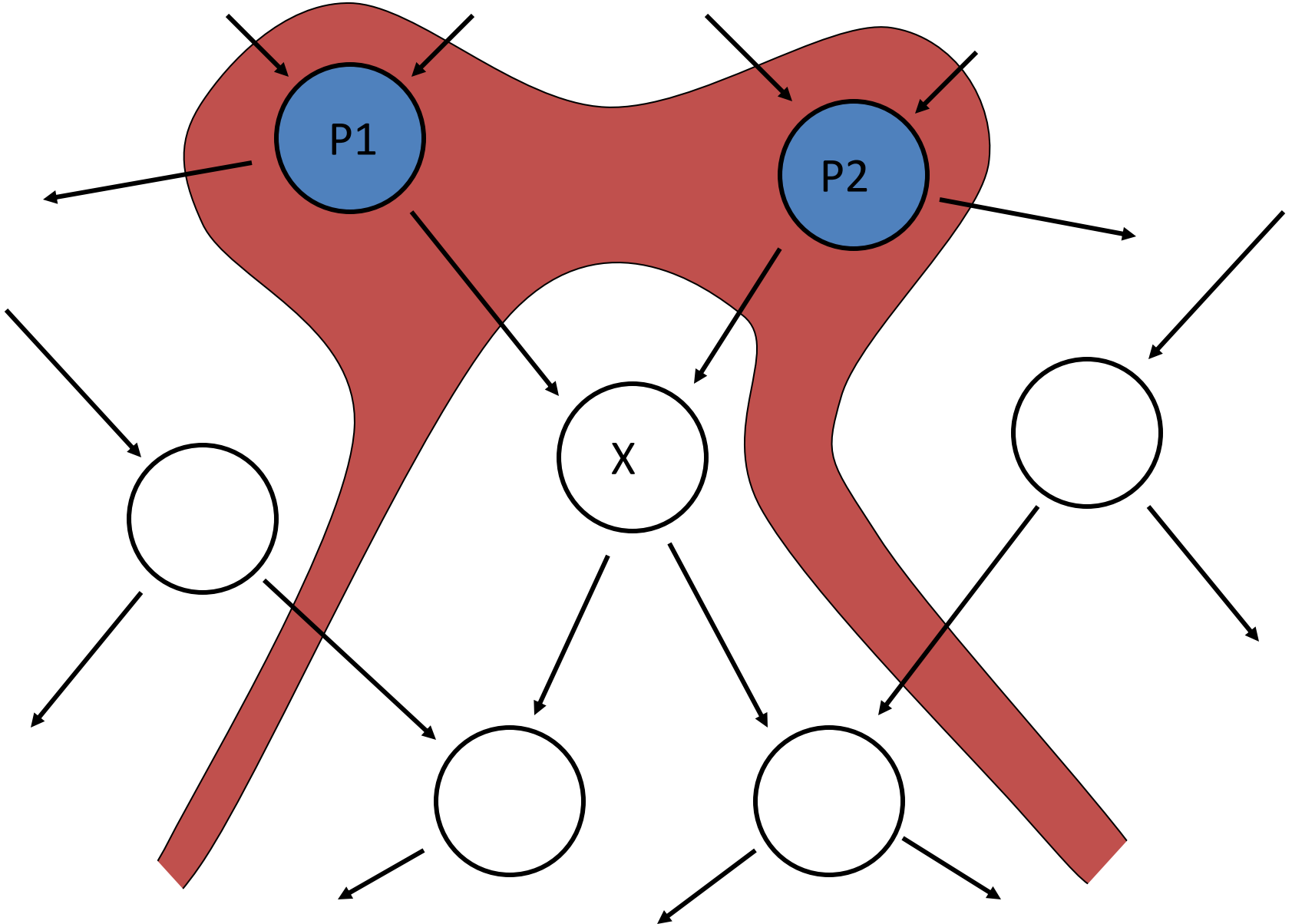
# Non-Polytrees

## Monte Carlo sampling algorithms

### Rejection sampling

- Start w/ all RV's unknown
- Choose a topological ordering
- Get random numbers to assign values
- Descendents are always unknown when assigning an RV
- Generate lots of samples
- Count but ignore samples violating evidence

# X is conditionally independent of its non-descendants given its parents



# Non-Polytrees

## Monte Carlo sampling algorithms

### Likelihood weighting

- Fix evidence variables to their values
- Generate a sample randomly assigning other variables
- Generate lots of samples
- But count each weighted according to Pr(evidence | dice rolls)
- All samples are consistent w/ evidence
- Some count for very little

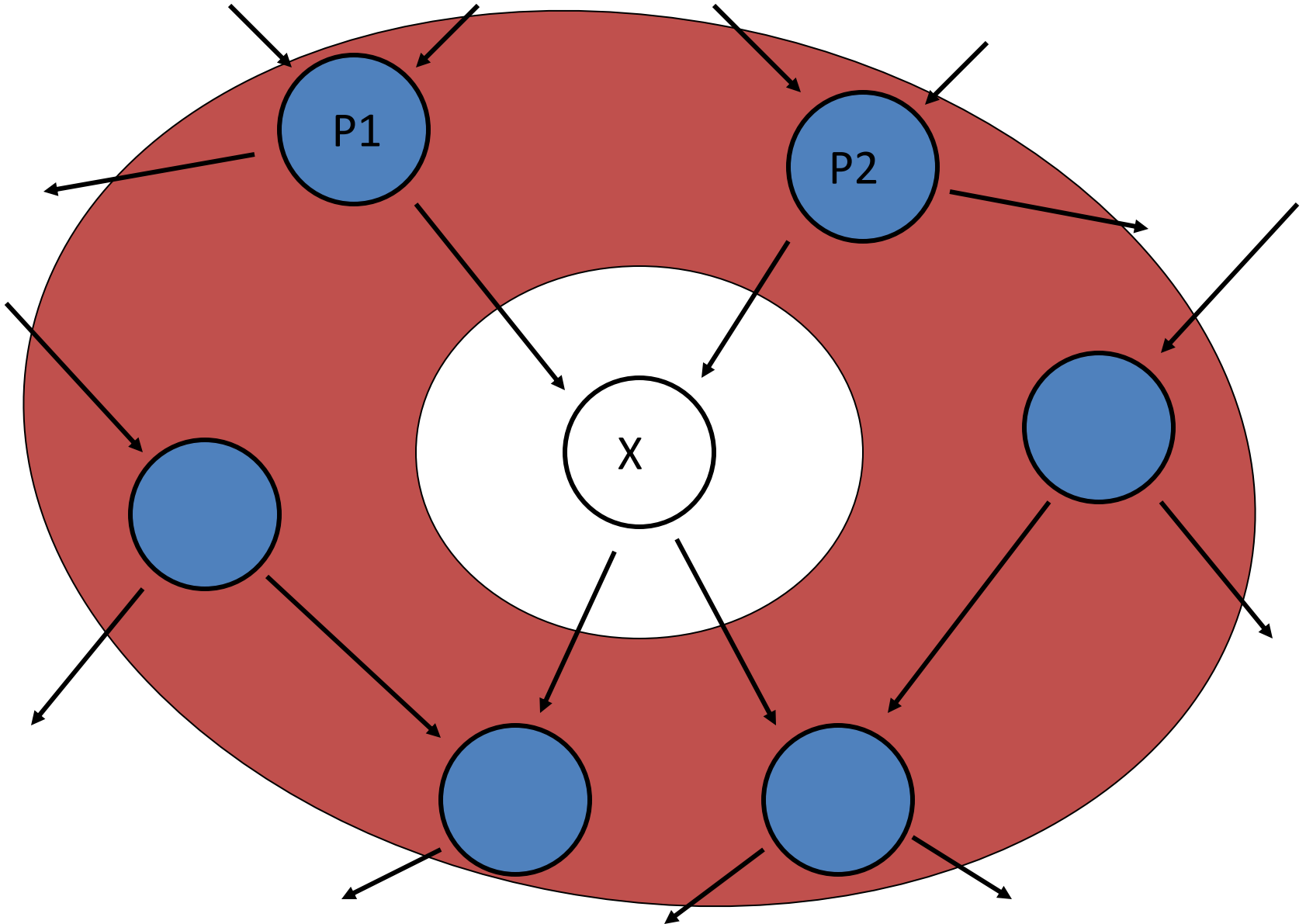
# Non-Polytrees

## Monte Carlo sampling algorithms

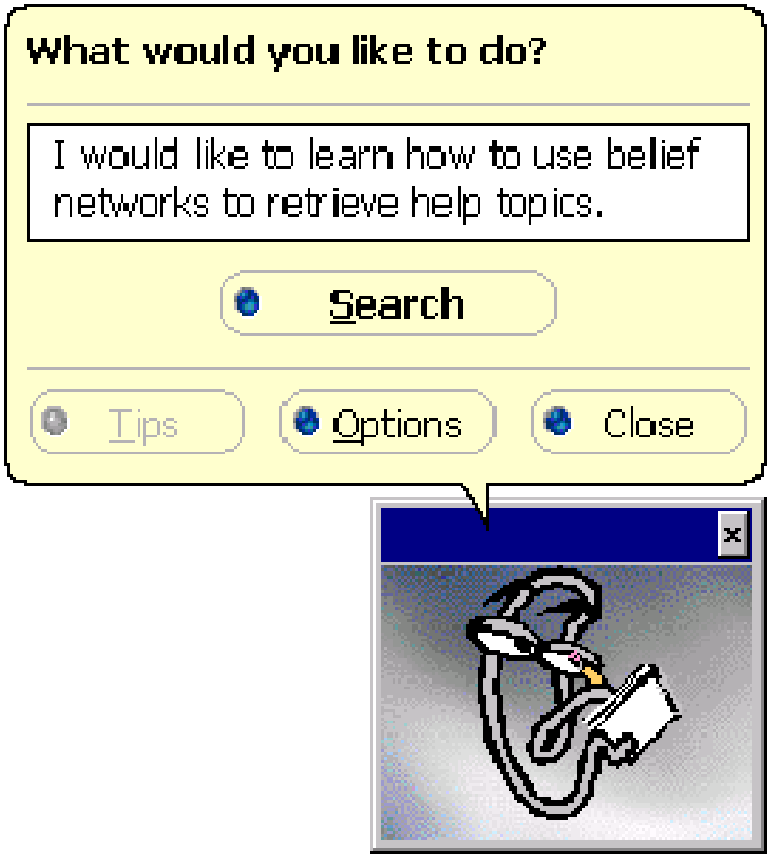
### Gibbs sampling

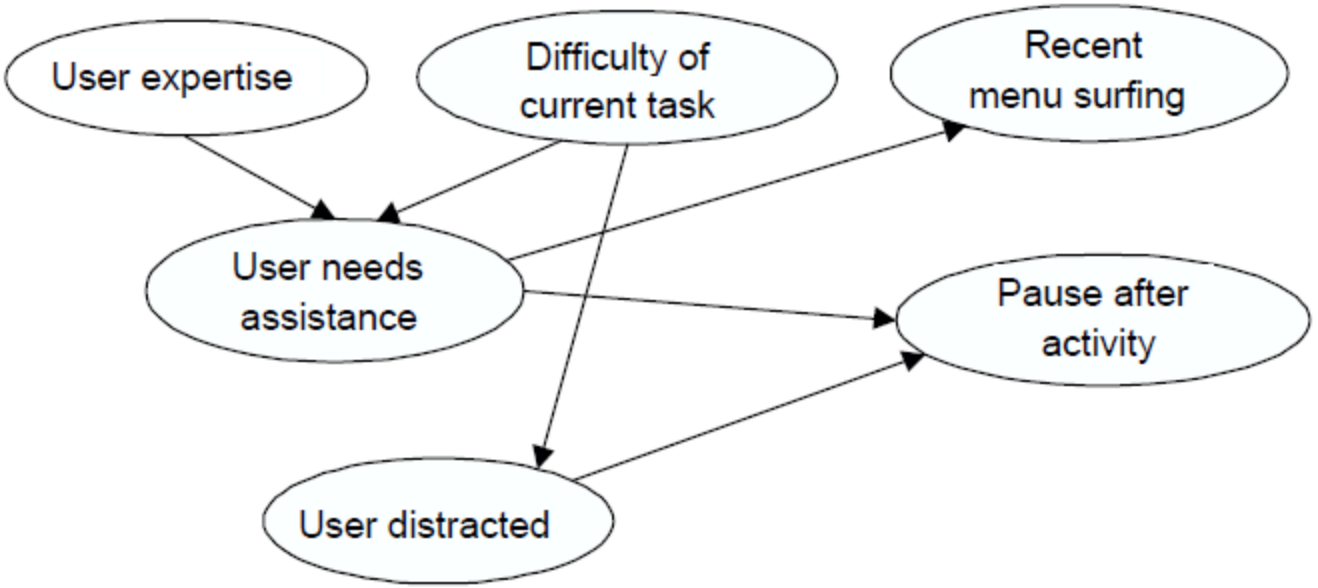
- Markov Chain Monte Carlo (MCMC)
- Fix evidence variables to their values
- Initialize others randomly
- Iterate through nodes
- Resample each given its Markov blanket
- After ‘burn in,’ accumulate each k’th sample
- Count

# X is conditionally independent of everything else given its Markov blanket

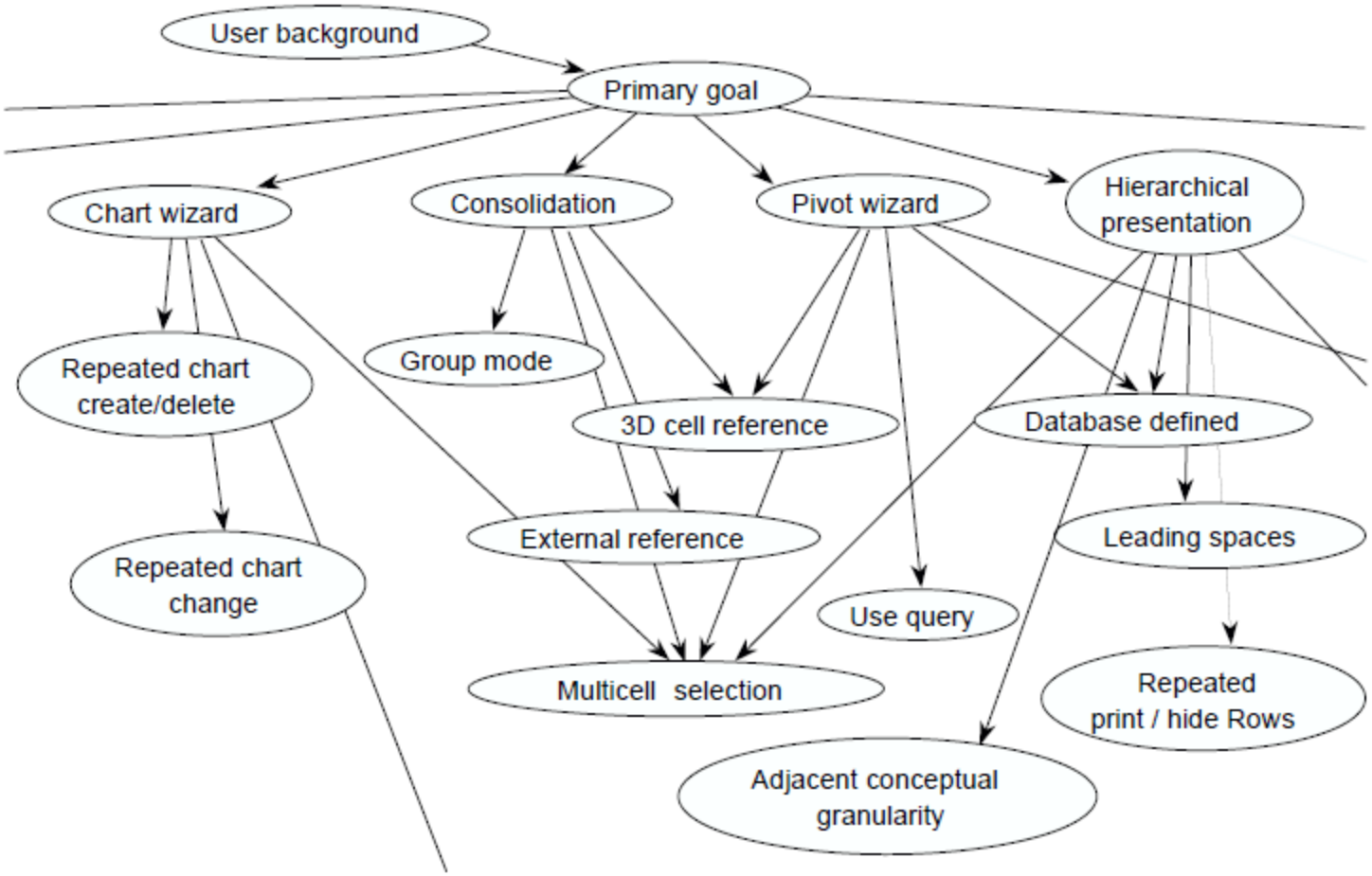


# Example

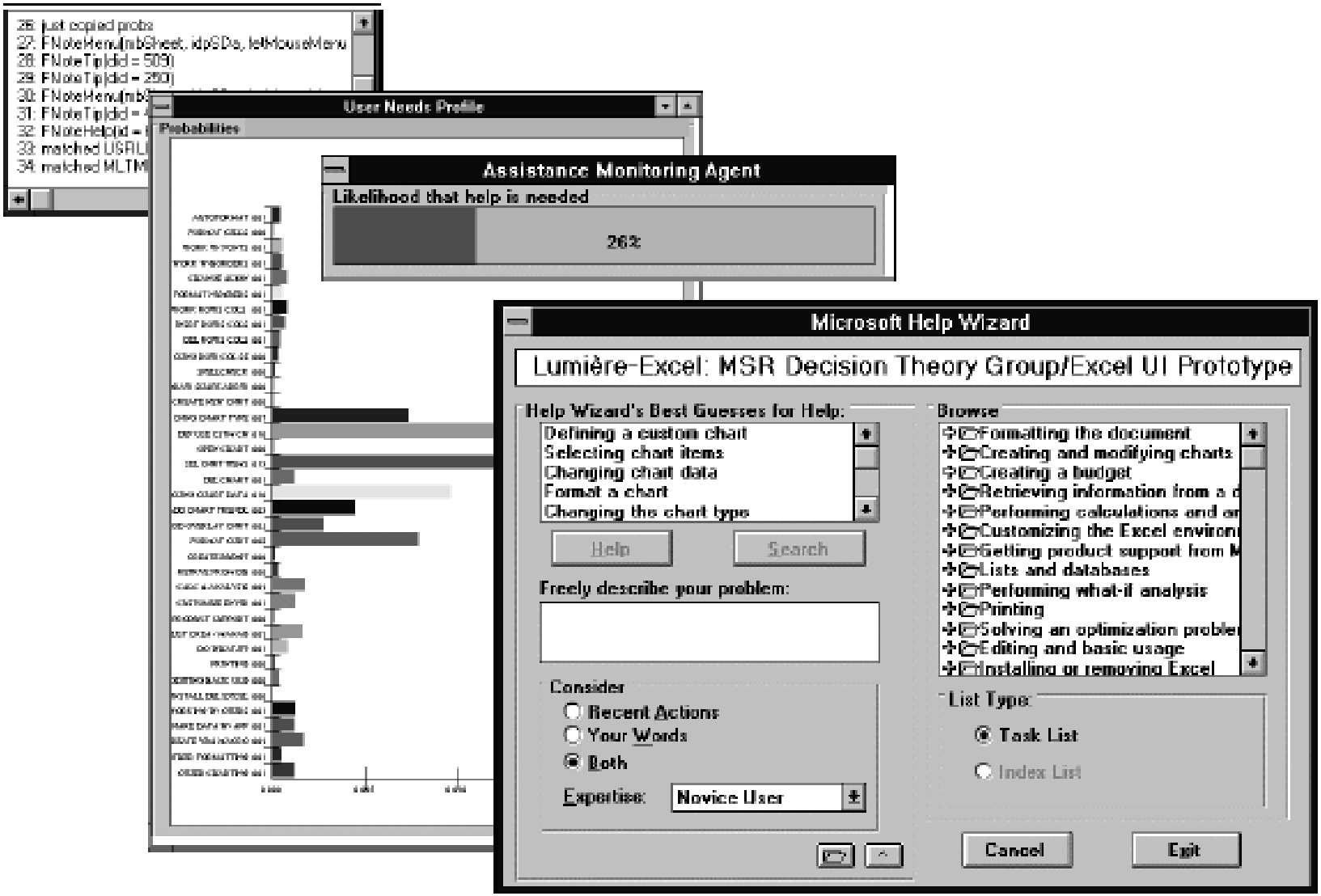

What would you like to do?
I would like to learn how to use belief networks to retrieve help topics.
Search
Tips
Options
Close




Early BN for when Clippy should pop up




Partial BN of Clippy's prior knowledge of MS Excel




Clippy's Bayesian reasoning: Event stream, Current probability distribution of important RVs, Pr(User wants Clippy), Prototype human interface