Artificial Intelligence

Bayesian Networks

Adapted from slides by
Tim Finin and
Marie desJardins.

Some material borrowed
from Lise Getoor.

Outline

- **Bayesian networks**
  - **Network structure**
  - **Conditional probability tables**
  - **Conditional independence**
- **Inference in Bayesian networks**
  - **Exact inference**
  - Approximate inference

Bayesian Belief Networks (BNs)

- Definition: **BN = (DAG, CPD)**
  - **DAG**: directed acyclic graph (BN's **structure**)
    - **Nodes**: random variables (typically binary or discrete, but methods also exist to handle continuous variables)
    - **Arcs**: indicate probabilistic dependencies between nodes (*lack* of link signifies conditional independence)
  - **CPD**: conditional probability distribution (BN's **parameters**)
    - Conditional probabilities at each node, usually stored as a table (conditional probability table, or **CPT**)

      $P(x_i \mid \pi_i)$ where π_i is the set of all parent nodes of x_i

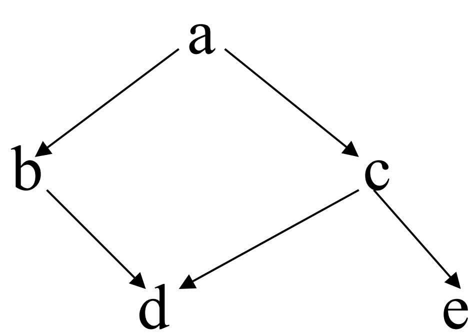
  - Root nodes are a special case – no parents, so just use priors in CPD:

$$\pi_i = \varnothing, \text{so } P(x_i \mid \pi_i) = P(x_i)$$

Example BN

P(A) = 0.001

P(B|A) = 0.3
P(B|¬A) = 0.001



P(C|A) = 0.2
P(C|¬A) = 0.005

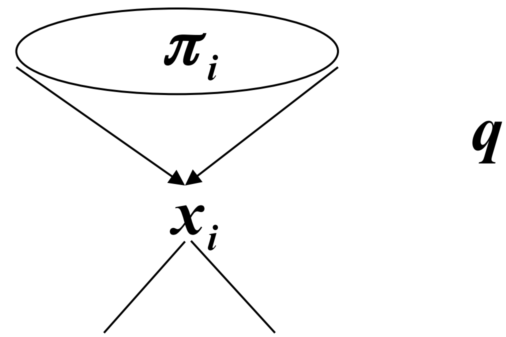
P(D|B,C) = 0.1 P
(D|B,¬C) = 0.01
P(D|¬B,C) = 0.01 P
(D|¬B,¬C) = 0.00001

P(E|C) = 0.4
P(E|¬C) = 0.002

Note that we only specify P(A) etc., not P(¬A), since they have to add to one

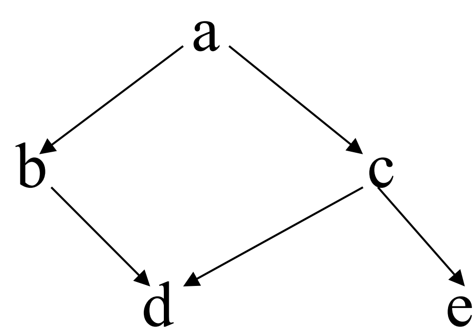
Conditional independence and chaining

- Conditional independence assumption
  - $P(x_i \mid \pi_i, q) = P(x_i \mid \pi_i)$
    where q is any set of variables (nodes) other than x_i and its successors
  - π_i **blocks influence** of other nodes on x_i and its successors (q influences x_i only through variables in π_i)
  - With this assumption, the complete joint probability distribution of all variables in the network can be represented by (recovered from) local CPDs by chaining these CPDs:



$$P(x_1, \ldots, x_n) = \Pi_{i=1}^{n} P(x_i \mid \pi_i)$$

Chaining: Example



Computing the joint probability for all variables is easy:

P(a, b, c, d, e)

= P(e | a, b, ***c,*** d) P(a, b, c, d) by the product rule

= P(e | c) P(a, b, c, d) by cond. indep. assumption

= P(e | c) P(d | a, ***b, c***) P(a, b, c)

= P(e | c) P(d | b, c) P(c | ***a,*** b) P(a, b)

= P(e | c) P(d | b, c) P(c | a) P(b | a) P(a)

Topological semantics

- A node is **conditionally independent** of its **non-descendants** given its **parents**
- A node is **conditionally independent** of all other nodes in the network given its parents, children, and children's parents (also known as its **Markov blanket**)
- The method called **d-separation** can be applied to decide whether a set of nodes X is independent of another set Y, given a third set Z

Inference tasks

- **Simple queries:** Computer posterior marginal $P(X_i \mid E=e)$
  - E.g., P(NoGas | Gauge=empty, Lights=on, Starts=false)
- **Conjunctive queries:**
  - $P(X_i, X_j \mid E=e) = P(X_i \mid e=e)\, P(X_j \mid X_i, E=e)$
- **Optimal decisions:** *Decision networks* include utility information; probabilistic inference is required to find P (outcome | action, evidence)
- **Value of information:** Which evidence should we seek next?
- **Sensitivity analysis:** Which probability values are most critical?
- **Explanation:** Why do I need a new starter motor?

Approaches to inference

- Exact inference
  - **Enumeration**
  - Belief propagation in polytrees
  - Variable elimination
  - Clustering / join tree algorithms
- Approximate inference
  - Stochastic simulation / sampling methods
  - Markov chain Monte Carlo methods
  - Genetic algorithms
  - Neural networks
  - Simulated annealing
  - Mean field theory

Direct inference with BNs

- Instead of computing the joint, suppose we just want the probability for *one* variable
- Exact methods of computation:
  - **Enumeration**
  - *Variable elimination*
- *Join trees: get the probabilities associated with every query variable*

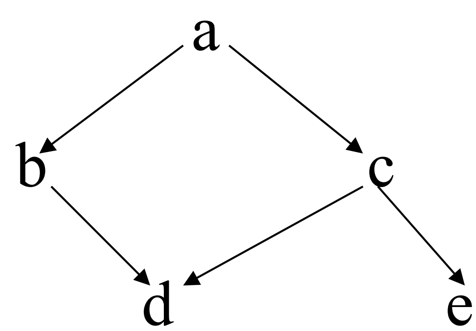
Inference by enumeration

- Add all of the terms (atomic event probabilities) from the full joint distribution
- If **E** are the evidence (observed) variables and **Y** are the other (unobserved) variables, then:

$$P(X|\mathbf{e}) = \alpha\, P(X, \mathbf{E}) = \alpha \sum P(X, \mathbf{E}, \mathbf{Y})$$

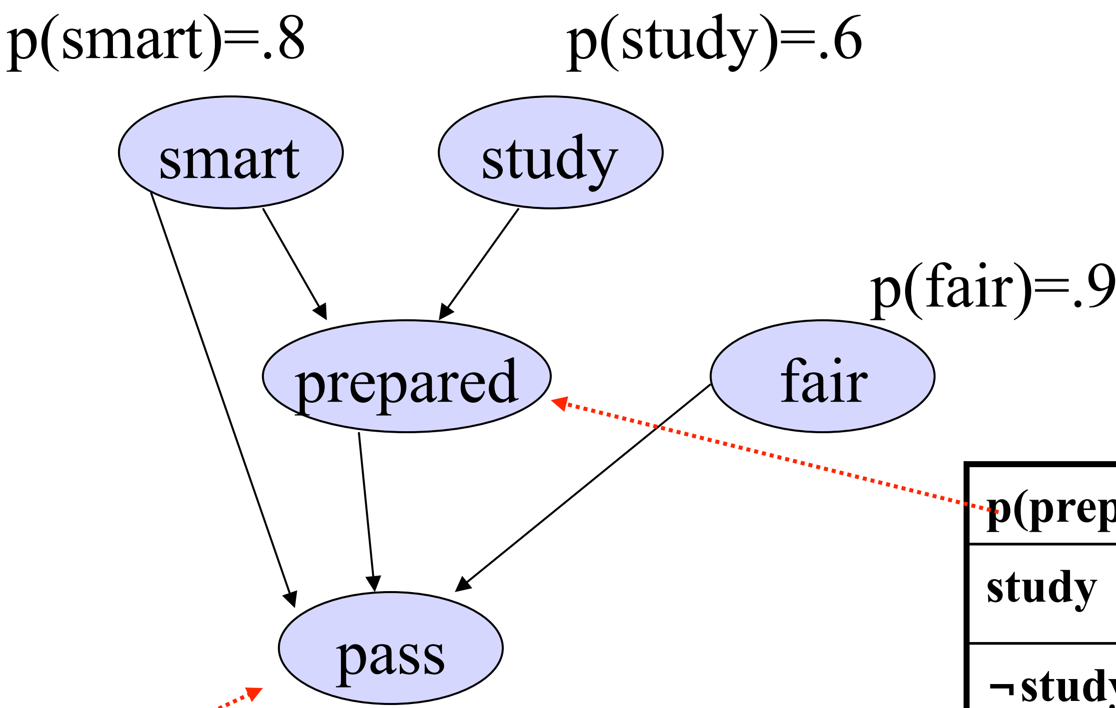
- Each P(X, **E**, **Y**) term can be computed using the chain rule
- Computationally expensive!

Example: Enumeration



- $P(x_i) = \Sigma_{\pi i} P(x_i \mid \pi_i) P(\pi_i)$
- Suppose we want P(D=true), and only the value of E is given as true
- $P(d|e) = \alpha \Sigma_{ABC} P(a, b, c, d, e)$
  $= \alpha \Sigma_{ABC} P(a) P(b|a) P(c|a) P(d|b,c) P(e|c)$
- With simple iteration to compute this expression, there's going to be a lot of repetition (e.g., P(e|c) has to be recomputed every time we iterate over C=true)

Exercise: Enumeration



p(prep|…)	smart	¬smart
study	.9	.7
¬study	.5	.1

p(pass|…)	smart		¬smart	
	prep	¬prep	prep	¬prep
fair	.9	.7	.7	.2
¬fair	.1	.1	.1	.1

Query: What is the probability that a student studied, given that they pass the exam?

Summary

- **Bayes nets**
  - **Structure**
  - **Parameters**
  - **Conditional independence**
  - **Chaining**
- **BN inference**
  - **Enumeration**
  - Variable elimination
  - Sampling methods