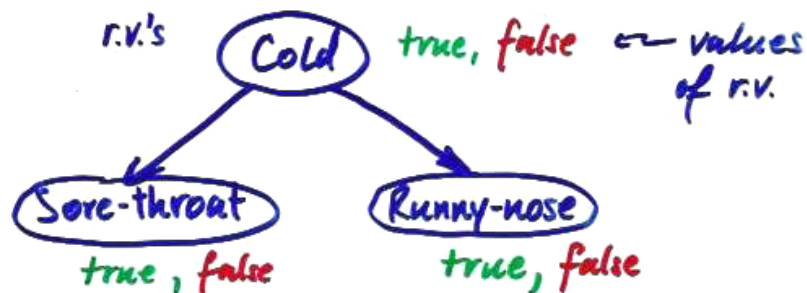


# BAYES NETS

CAUSAL NETS,  
INFLUENCE DIAGRAMS,  
BELIEF NETWORKS

Bayes nets provide a general technique for computing probabilities of causally related random variables, given the values of some of the random var's ("evidence")

E.g., Previous "Cold" example can be viewed as a simple Bayes net:



Joint distribution of the 3 r.v.'s:

$$P(\text{Cold}, \text{Sore-throat}, \text{Runny-nose}) =$$

(vector of 8 prob's., adding up to 1)

## BAYES NET ASSUMPTION

FOR NETS WITH A UNIQUE ROOT :

The joint probability distribution of all the variables in the net equals the probability of the root times the probability of each non-root node, given its parent(s).

Any node is indep. of its non-successor nodes, given its parents.

For the example:

$$P(\text{Cold}, \text{Sore-throat}, \text{Runny-nose}) =$$

RHS prob's assumed to be known

$$P(\text{Cold}) \times P(\text{Sore-throat} | \text{Cold}) \times P(\text{Runny-nose} | \text{Cold})$$

Note: This really stands for 8 equations:

$$P(\text{Cold}=\text{true}, \text{Sore-throat}=\text{true}, \text{Runny-nose}=\text{true})$$

$$= P(\text{Cold}=\text{true}) \times P(\text{Sore-throat}=\text{true} | \text{Cold}=\text{true})$$

$$\times P(\text{Runny-nose}=\text{true} | \text{Cold}=\text{true})$$

⋮

$$P(\text{Cold}=\text{false}, \text{Sore-throat}=\text{false}, \text{Runny-nose}=\text{false})$$

$$= P(\text{Cold}=\text{false}) \times P(\text{Sore-throat}=\text{false} | \text{Cold}=\text{false})$$

$$\times P(\text{Runny-nose}=\text{false} | \text{Cold}=\text{false})$$

For our simple "Cold" example, the Bayes net assumption turns out to be equivalent to the conditional independence assumption we made before:

"Sore-throat" is conditionally independent of "Runny-nose", given "Cold"

$$P(\text{Sore-throat} \mid \text{Cold}, \text{Runny-nose}) \\ = P(\text{Sore-throat} \mid \text{Cold}), \text{ or}$$

$$P(\text{Runny-nose} \mid \text{Cold}, \text{Sore-throat}) \\ = P(\text{Runny-nose} \mid \text{Cold})$$

Proof.  $P(\text{Cold}, \text{Sore-throat}, \text{Runny-nose})$   
 $= P(\text{Cold}) P(\text{Sore-throat} \mid \text{Cold}) \times$   
 $P(\text{Runny-nose} \mid \text{Cold}, \text{Sore-throat})$   
 by product rule  
 $= P(\text{Cold}) P(\text{Sore-throat} \mid \text{Cold}) \times$   
 $P(\text{Runny-nose} \mid \text{Cold})$   
 by conditional independence  
 of the effects (symptoms),  
 given the cause(s)

Q.E.D.

So since the assumptions are the same as before, the math of computing posterior probabilities, given some evidence (node values) is the same as before

E.g.,

$$P(\text{Cold} \mid \text{Sore-throat}) \sim P(\text{Cold} = \text{true} \mid \text{Sore-throat} = \text{true}) \\ = \frac{P(\text{Cold}) P(\text{Sore-throat} \mid \text{Cold})}{P(\text{Cold}) P(\text{Sore-throat} \mid \text{Cold}) + P(\neg \text{Cold}) P(\text{Sore-throat} \mid \neg \text{Cold})}$$

Or, using odds-likelihood formulation:

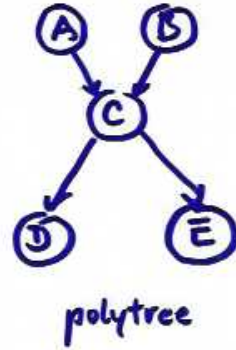
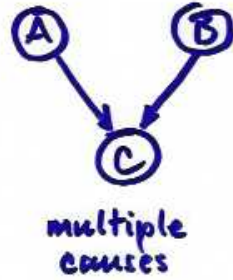
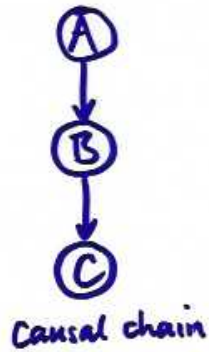
$$O(\text{Cold} \mid \text{Sore-throat}) = O(\text{Cold}) L(\text{Sore-throat} \mid \text{Cold})$$

Similarly, if both symptoms are given:

$$O(\text{Cold} \mid \text{Sore-throat}, \text{Runny-nose}) \\ = O(\text{Cold}) L(\text{Sore-throat} \mid \text{Cold}) L(\text{Runny-nose} \mid \text{Cold})$$

as before.

Other simple examples:



**BAYES NET ASSUMPTION**

**IN GENERAL:**

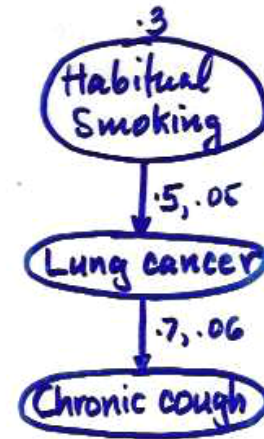
The joint probability distribution of all the variables in the net equals the product of root probabilities times the probability of each non-root node, given its parents

$$P(x_1, \dots, x_n) = \prod_{i=1}^n P(x_i | \text{Parents}(x_i))$$

For root  $x_i$ ,  $P(x_i | \text{Parents}(x_i)) = P(x_i)$

Any node is indep<sup>t</sup> of its non-successor nodes, given its parents.

E.g.,



$P(S) = .3$

$P(L|S) = .5, P(L|\neg S) = .05$

$P(C|L) = .7, P(C|\neg L) = .06$

$P(S, L, C) = P(S)P(L|S)P(C|L)$

We might want, e.g.,  $P(L|C), P(S|C), P(L|S, C)$

**General (but inefficient) way of computing any (un)conditional prob.:**

1. Express the (un)conditional probability in terms of joint prob's for all the nodes
2. Use the Bayes net assumption to evaluate the joint probabilities

involved in the query & those ancestors

- Can often simplify (1) by computing odds instead of probabilities
- General algorithms exist that compute conditional prob's in Bayes nets more efficiently

Let's compute  $P(L|C) = P(\text{Lung-canc.} | \text{Chronic-cough})$   
for the example:

$$\text{Find odds } O(L|C) = \frac{P(L|C)}{P(\neg L|C)} = \frac{P(L \wedge C) / \cancel{P(C)}}{P(\neg L \wedge C) / \cancel{P(C)}}$$

$$\begin{aligned} P(L \wedge C) &= P(S \wedge L \wedge C) + P(\neg S \wedge L \wedge C) \\ &= P(S)P(L|S)P(C|L) + P(\neg S)P(L|\neg S)P(C|L) \\ &= .3(.5)(.7) + .7(.05)(.7) = .1295 \end{aligned}$$

$$\begin{aligned} P(\neg L \wedge C) &= P(S)P(\neg L|S)P(C|\neg L) + P(\neg S)P(\neg L|\neg S)P(C|\neg L) \\ &= .3(.5)(.06) + .7(.95)(.06) = .0489 \end{aligned}$$

$$\therefore O(L|C) = \frac{.1295}{.0489} = 2.648$$

$$\therefore P(L|C) = \frac{O(L|C)}{1 + O(L|C)} = \boxed{.726} \quad 73\% \text{ chance}$$

Compute  $P(S|C) = P(\text{Smoker} | \text{Chronic-cough})$

$$\begin{aligned} O(S|C) &= \frac{P(S|C)}{P(\neg S|C)} = \frac{P(S \wedge C) / \cancel{P(C)}}{P(\neg S \wedge C) / \cancel{P(C)}} \\ &= \frac{P(S \wedge L \wedge C) + P(S \wedge \neg L \wedge C)}{P(\neg S \wedge L \wedge C) + P(\neg S \wedge \neg L \wedge C)} \\ &= \frac{.3(.5)(.7) + .3(.5)(.06)}{.7(.05)(.7) + .7(.95)(.06)} = 1.77 \end{aligned}$$

$$\therefore P(S|C) = 1.77 / 2.77 = \boxed{.639} \quad 64\% \text{ chance of being smoker}$$

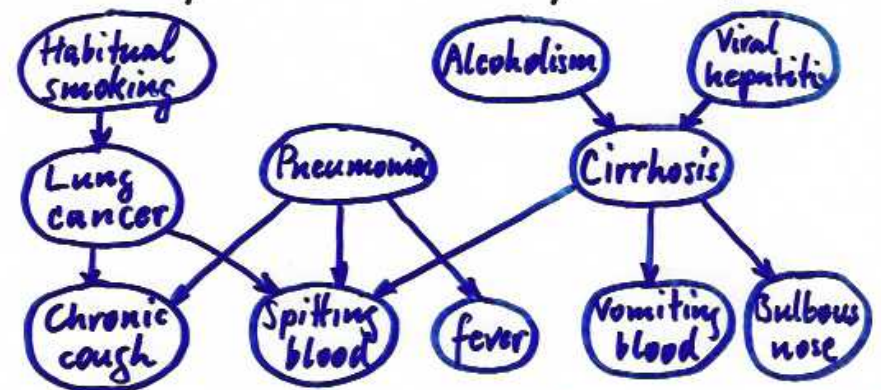
Finally, let's compute  $P(L|S \wedge C) =$   
 $P(\text{Lung-cancer} | \text{Smoker} \wedge \text{Chronic-cough})$

$$\begin{aligned} O(L|S \wedge C) &= \frac{P(L|S \wedge C)}{P(\neg L|S \wedge C)} = \frac{P(L \wedge S \wedge C) / \cancel{P(S \wedge C)}}{P(\neg L \wedge S \wedge C) / \cancel{P(S \wedge C)}} \\ &= \frac{P(S)P(L|S)P(C|L)}{P(S)P(\neg L|S)P(C|\neg L)} = \frac{.3(.5)(.7)}{.3(.5)(.06)} = 11.67 \end{aligned}$$

$$P(L|S \wedge C) = 11.67 / 12.67 = \boxed{.921} \quad 92\% \text{ chance}$$

\*\*\*

In real applications, the networks can have hundreds of nodes, with lots of "overlap"



etc.

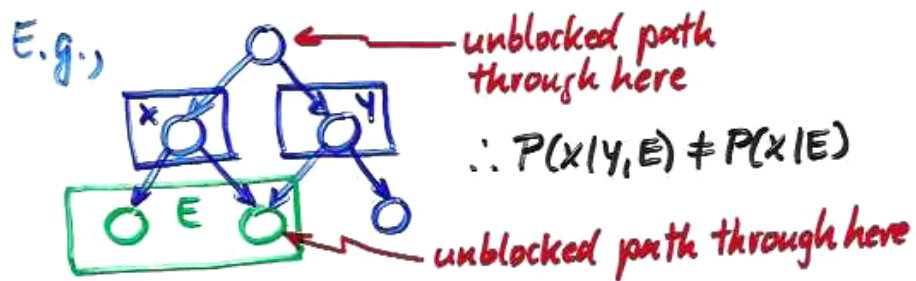
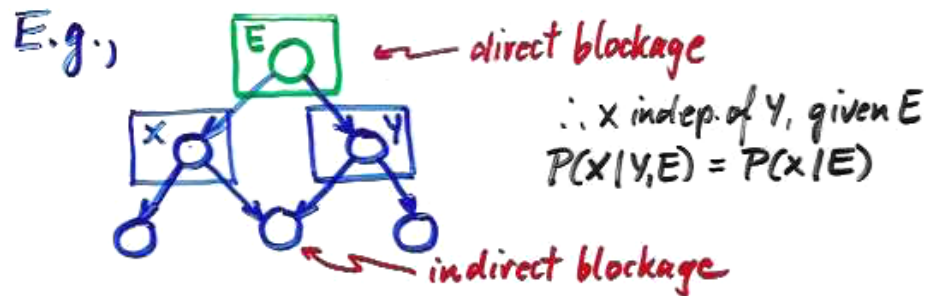
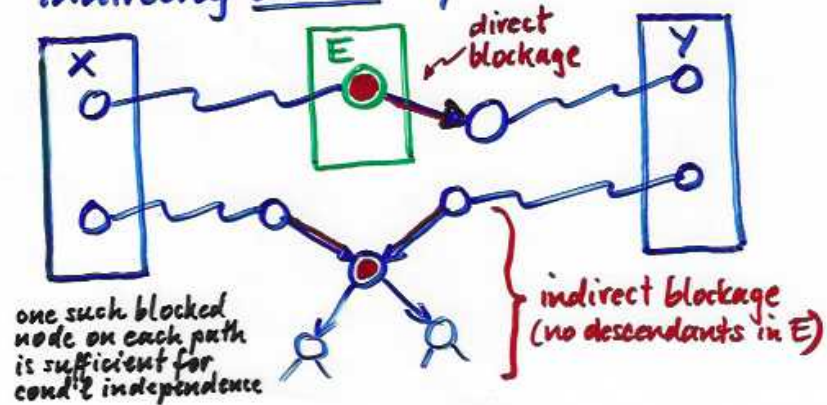
Inference exponentially complex (in # nodes) in worst case.

# INFERENCE IN BAYES NETS

- Already saw a general (but inefficient) method
- Better methods depend on systematic use of the independence assumptions implicit in the Bayes net assumption:

A set of nodes  $X$  is independent of a set of nodes  $Y$  given nodes  $E$  iff (they are "d-separated")

every undirected path connecting a node in  $X$  with a node in  $Y$  is directly or indirectly blocked by  $E$

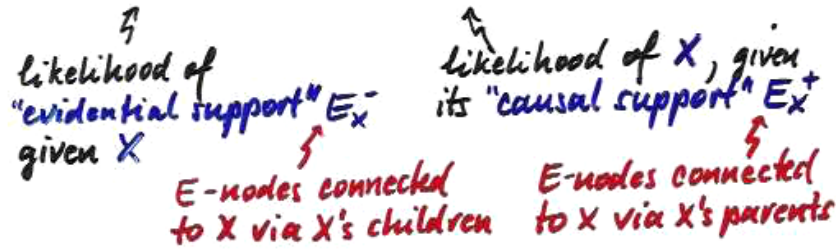


## Inference in singly-connected networks (polytrees)

- Only one undirected path between any two nodes
- R&N develop a recursive formula for  $P(X|E)$  for arbitrary disjoint sets of nodes  $X, E$  in polytree

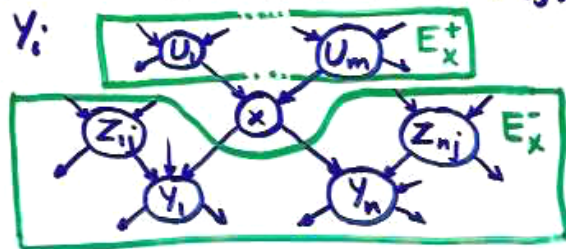
They do this in 3 stages:

- Express  $P(X|E)$  in terms of  $P(E_x^-|X)$  and  $P(X|E_x^+)$



- Express  $P(X|E_x^+)$  recursively in terms of  $P(U_i|E_x^+)$  where the  $U_i$  are  $X$ 's parents (also use the given network prob's  $P(X|U_1, U_2, \dots)$ )

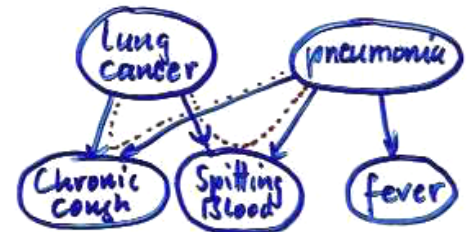
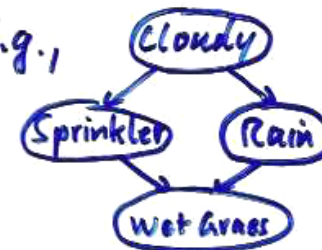
- Express  $P(E_x^-|X)$  recursively in terms of  $P(E_{y_i}^-|Y_i)$  where the  $Y_i$  are  $X$ 's children, and in terms of  $P(Z_{ij}|E_{Z_{ij}} \setminus Y_i)$ , where the  $Z_{ij}$  are the parents of  $X$ 's child  $Y_i$ , and  $E_{Z_{ij}} \setminus Y_i$  are the  $E$ -nodes connected to  $Z_{ij}$ , except via  $Y_i$ :



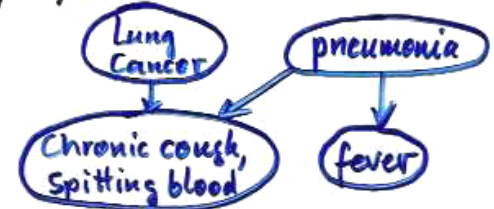
- Details - see p447-451 (no need to study in detail)
- Algorithm based on these recursions: p452
- Pearl (1982) and Kim & Pearl (1983) devised a "message-passing" algorithm for polytrees, with propagation of messages starting at evidence nodes. Has strong similarity to neural nets.

Inference in Multiply-Connected Networks

E.g.1



- Clustering to get polytree



- Cutset conditioning - splitting net into polytrees (omit)
- STOCHASTIC SIMULATION

### One simulation:

- Choose values for the root nodes randomly, using the given root probabilities
- Repeat until all nodes have values: select a node all of whose parents have values, but which has no value yet; choose a value for it randomly, using the given conditional probabilities.

Do many simulations;

$$P(\underline{X} = \underline{x} \mid \underline{E} = \underline{e}) \approx$$

$$\frac{\# \text{simulations for which } \underline{X} = \underline{x} \ \& \ \underline{E} = \underline{e}}{\# \text{simulations for which } \underline{E} = \underline{e}}$$

- Better: simulation with LIKELIHOOD WEIGHTING:  
When assigning a value to an E-node, just give it the known value (nonrandomly). But multiply the weight of the simulation (initially 1) by the corresponding probability.

## EXTENSIONS, APPLICATIONS

- Often extended to probability intervals using Dempster-Shafer belief functions.  
That's what "belief networks" most commonly refer to.
- Extension to continuous variables (with Gaussian distributions). Mixing with discrete variables not fully understood.

Many successful applications in medicine, geological prospecting, insurance underwriting, web browsing (big Microsoft investment in B.N.s)

E.g., PATHFINDER II (Stanford, late 80's) outperforms leading experts in lymph node pathology. (One of 32 major areas of pathology; 60 diseases, 100 symptoms/test results) 14,000 prob's

E.g., CPSC project (Pradhan et al '94) - internal medicine (hundreds of diseases, thousands of conditional prob. values)  
35 min for stoch. sim. with likelihood weighting

# DECISION MAKING USING INFLUENCE DIAGRAMS

Ch.16  
p 471-2  
16.5

Simple example of decision problem: sky-diving

Friend invites you to go parachute jumping with him. Do you go?



Jump	Result		Prob	Utility
	Survive	Experience		
T	Thrill		.75	1000
T	Misery		.2	-1000
T	Blah		.049	100
F	Thrill		.0001	-99,999,000
F	Misery		.00005	-100,000,000
F	Blah		.00005	-100,000,000

¬Jump	Result		Prob	Utility (as above)
	Survive	Experience		
T	Thrill		.002	1000
T	Misery		.002	-1000
T	Blah		.996	100
F	Thrill		.00000001	-99,999,000
F	Misery		.000000005	-100,000,000
F	Blah		.000000002	-100,000,000

## Expected Utility of an action:

$EU(\text{Jump})$  = average utility we would get for arbitrarily many repetitions of the action (always starting in the same state!)

= sum of utilities for all possible results, with each utility weighted by its probability

$$= \sum_i P(\text{Result}_i | \text{Do}(\text{Jump})) U(\text{Result}_i)$$

$$= .75(1000) + \dots + .00005(-100,000,000)$$

$$= \boxed{-144,445.85}$$

$$EU(\neg\text{Jump}) = .002(1000) + \dots + .000000002(-100,000,000)$$

$$= \boxed{98.80}$$

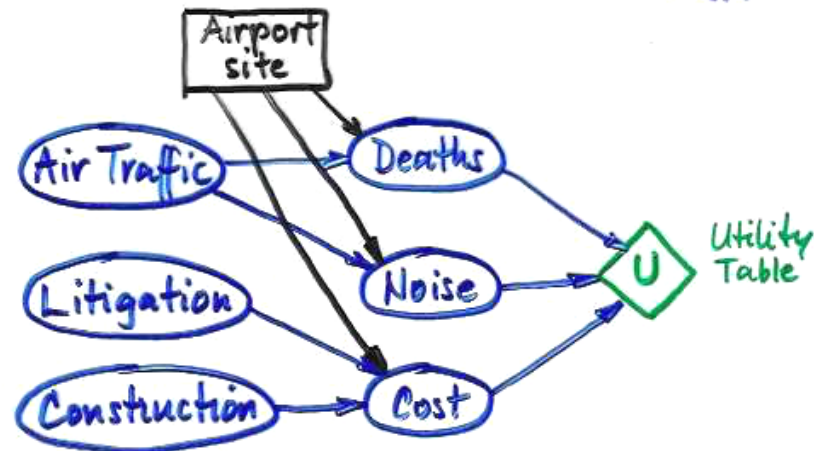
∴ DON'T JUMP!

In general, there may be factors other than the decision(s) which influence the possible results

$$EU(A|E) = \sum_i P(\text{Result}_i | \text{Do}(A), E) U(\text{Result}_i)$$



E.g., Siting an airport (sec 16.5, text)  
R&N



Influence Diagram (Decision Network)

 decision node

 chance node (Bayes net node)

 utility node

We now need

- prob's of chance nodes, given the immediate causes and decisions that influence them
- utility tables

For best decision, evaluate network & expected utility for each possible decision (allowing for evidence).