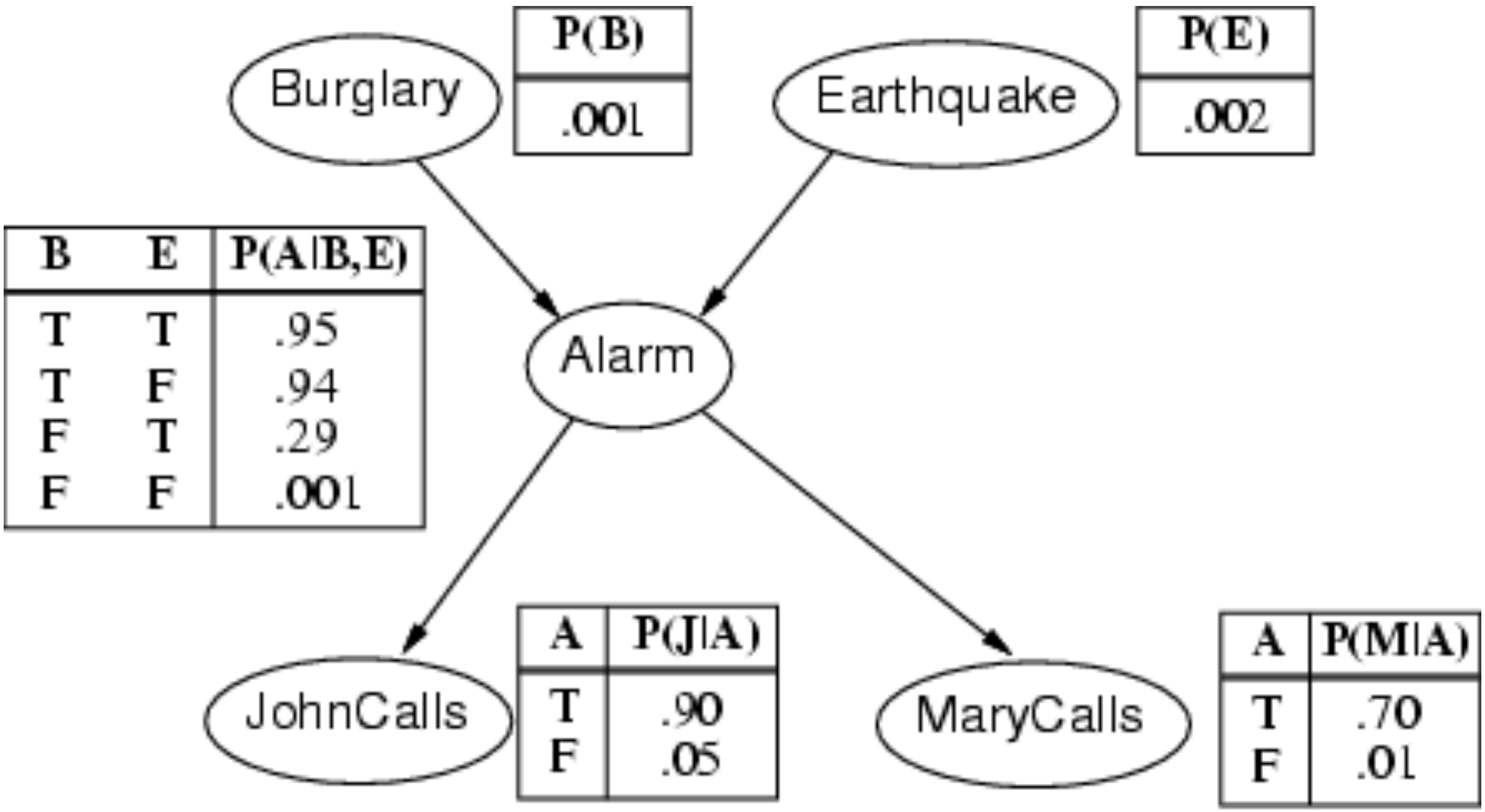
Example

- **I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?**

- **Variables: *Burglary*, *Earthquake*, *Alarm*, *JohnCalls*, *MaryCalls***

- **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**

Example contd.



P(B)
.001

P(E)
.002

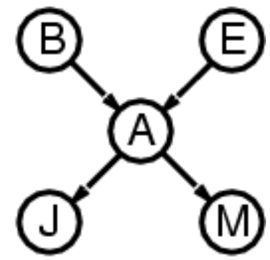
B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

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

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

Compactness

- A 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 number for X_i = *false* is just *1-p*)

- If each variable has no more than k parents, the complete network requires $O(n \cdot 2^k)$ numbers

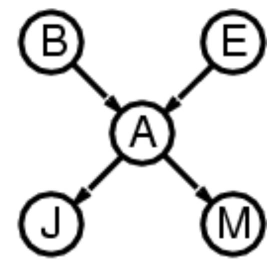
- I.e., grows linearly with n, vs. $O(2^n)$ for the full joint distribution

- For burglary net, 1 + 1 + 4 + 2 + 2 = 10 numbers (vs. 2^5-1 = 31)

Semantics

The full joint distribution is defined as the product of the local conditional distributions:

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



e.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

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

Constructing Bayesian networks

- **1. Choose an ordering of variables $X_1, \dots, X_n$**
- **2. For $i = 1$ to n**
  - **add X_i to the network**
  - **select parents from $X_1, \dots, X_{i-1}$ such that**

$$P(X_i \mid Parents(X_i)) = P(X_i \mid X_1, \dots X_{i-1})$$

This choice of parents guarantees:

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

(chain rule)

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

(by construction)

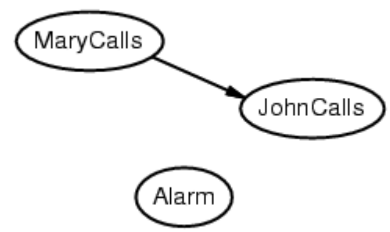
Example

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

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

Example

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



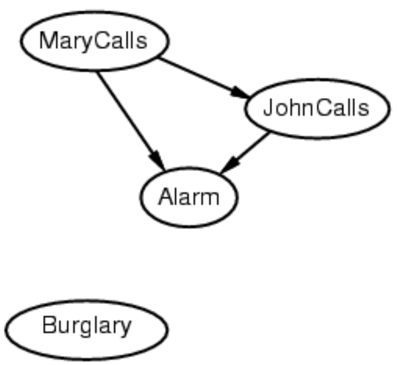
$P(J \mid M) = P(J)?$

No

$P(A \mid J, M) = P(A \mid J)?$ $P(A \mid J, M) = P(A)?$

Example

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



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

No

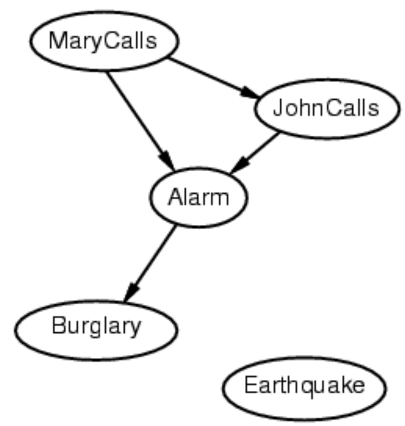
***P(A | J, M) = P(A | J)? P(A | J, M) = P(A)?* No**

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

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

Example

- **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**

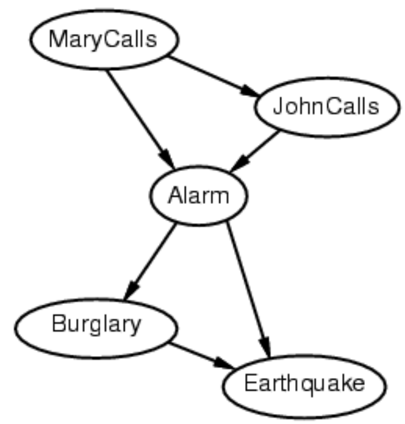
***P(B | A, J, M) = P(B)?* No**

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

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

Example

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



$P(J \mid M) = P(J)$?

No

$P(A \mid J, M) = P(A \mid J)$? $P(A \mid J, M) = P(A)$? **No**

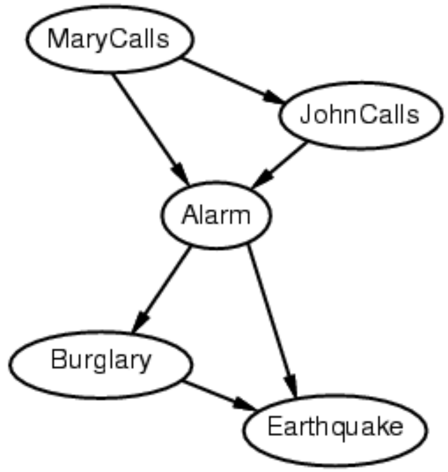
$P(B \mid A, J, M) = P(B \mid A)$? **Yes**

$P(B \mid A, J, M) = P(B)$? **No**

$P(E \mid B, A ,J, M) = P(E \mid A)$? **No**

$P(E \mid B, A, J, M) = P(E \mid A, B)$? **Yes**

Example contd.



- **Deciding conditional** **noncausal directions**
- **(Causal models and conditional independence seem hardwired for humans!)**
- **Network is less compact: 1 + 2 + 4 + 2 + 4 = 13 numbers needed**

Summary

- **Bayesian networks provide a natural representation for (causally induced) conditional independence**
- **Topology + CPTs = compact representation of joint distribution**
- **Generally easy for domain experts to construct**

Inference Using Bayes Theorem

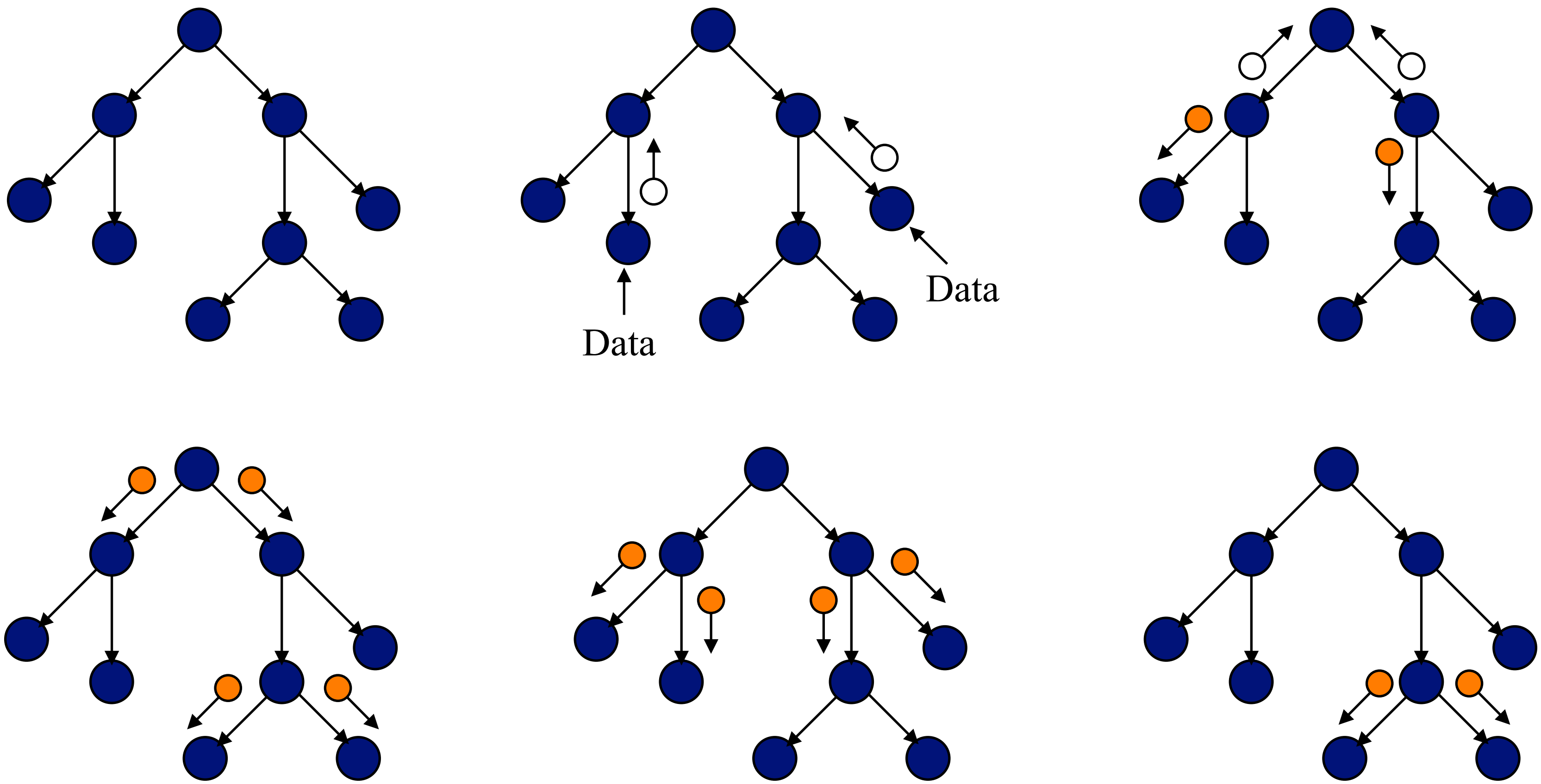
- The general probabilistic inference problem is to find the probability of an event given a set of evidence;

- This can be done in Bayesian nets with sequential applications of Bayes Theorem;

- In 1986 Judea Pearl published an innovative algorithm for performing inference in Bayesian nets.

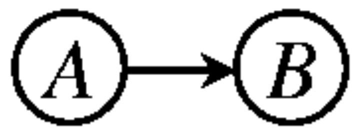
Propagation Example

"The impact of each new piece of evidence is viewed as a perturbation that propagates through the network via message-passing between neighboring variables . . ." (Pearl, 1988, p 143`



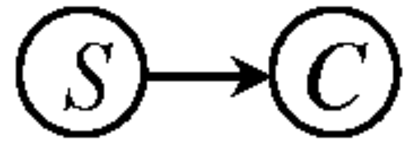
- **The example above requires five time periods to reach equilibrium after the introduction of data**

Basic Inference



$$P(b) = ?$$

Product Rule



- $P(C,S) = P(C|S)\,P(S)$

$S\Downarrow$ $C\Rightarrow$	*none*	*benign*	*malignant*
no	0.768	0.024	0.008
light	0.132	0.012	0.006
heavy	0.035	0.010	0.005

Marginalization

$S\Downarrow \quad C\Rightarrow$	*none*	*benign*	*malig*	total
no	0.768	0.024	0.008	.80
light	0.132	0.012	0.006	.15
heavy	0.035	0.010	0.005	.05
total	0.935	0.046	0.019	

P(Smoke)

P(Cancer)

Basic Inference

$$A \to B \to C$$

$$\underbrace{P(b)} = \sum_a P(a, b) = \sum_a P(b \mid a)\, P(a)$$

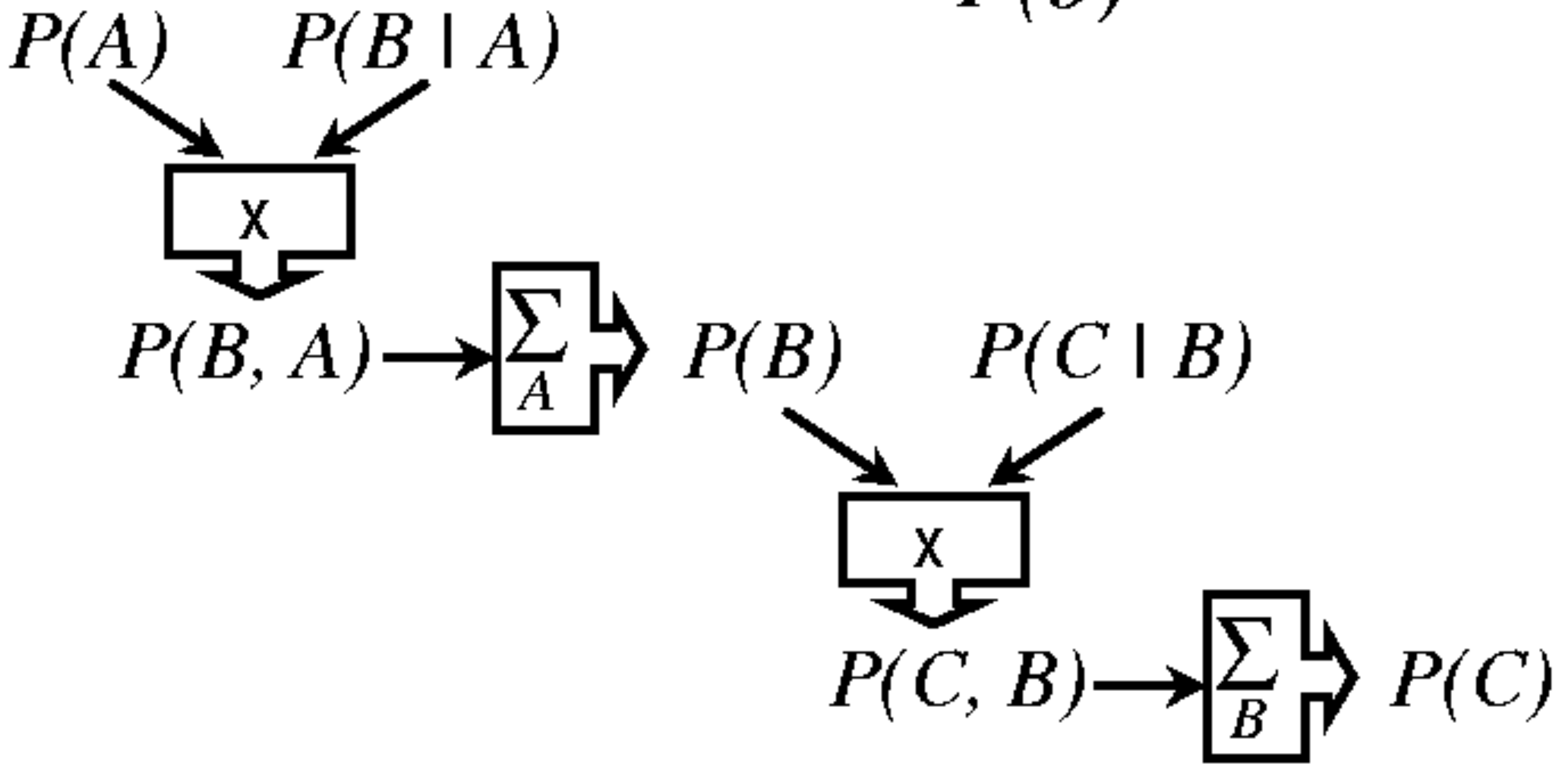
$$P(c) = \sum_b P(c \mid b)\, \overbrace{P(b)}$$

$$P(c) = \sum_{b,a} P(a, b, c) = \sum_{b,a} P(c \mid b)\, P(b \mid a)\, P(a)$$

$$= \sum_b P(c \mid b) \underbrace{\sum_a P(b \mid a)\, P(a)}_{P(b)}$$

Variable elimination

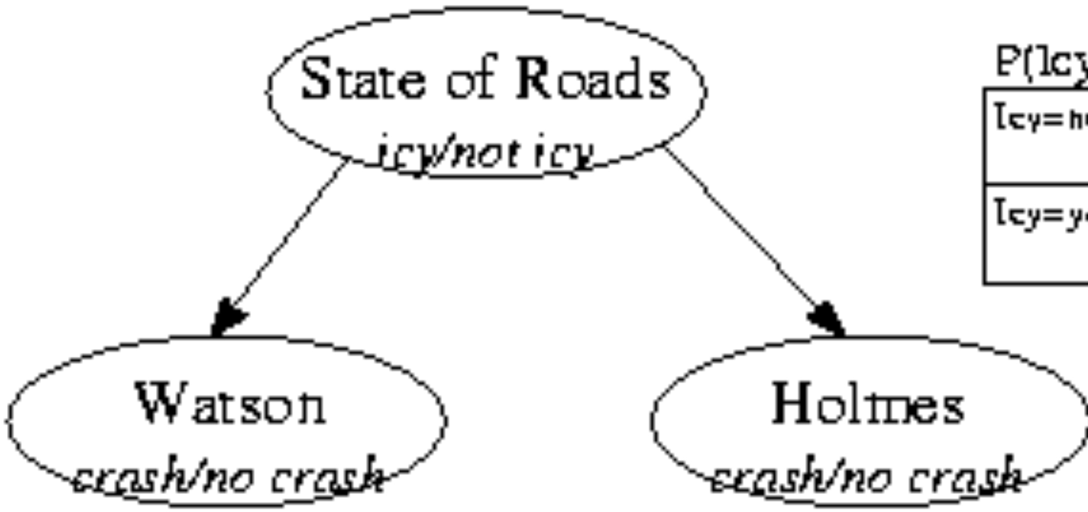
$$P(c) = \sum_b P(c \mid b) \underbrace{\sum_a P(b \mid a)\, P(a)}_{P(b)}$$



"Icy roads" example

- Inspector Smith is waiting for Holmes and Watson who are both late for an appointment.
- Smith is worried that if the roads are icy one or both of them may have crashed his car.
- Suddenly Smith learns that Watson has crashed.
- Smith thinks: *If Watson has crashed, probably the roads are icy, then Holmes has probably crashed too!*
- Smith then learns it is warm outside and roads are salted
- Smith thinks: *Watson was unlucky; Holmes should still make it.*

Bayes net for "Icy roads" example



P(Icy)

Icy=no	0.3
Icy=yes	0.7

P(Watson | Icy)	Icy = yes	Icy = no
Watson Crash = yes	0.8	0.1
Watson Crash = no	0.2	0.9

P(Holmes | Icy)	Icy = yes	Icy = no
Holmes Crash = yes	0.8	0.1
Holmes Crash = no	0.2	0.9

Extracting marginals

To find P(Holmes Crash) we first compute P(Holmes Crash, Icy) using the fundamental rule:

e.g. P(H Crash = yes, Icy = yes)

= P(H Crash=yes | Icy =yes)P(Icy=yes)

P(Holmes,Icy)	Icy = yes	Icy = no	P(H Crash)
Holmes Crash = yes	0.8 x0,7=0,56	0.1 x0,3=0,03	0.56+0.03=0.59
Holmes Crash = no	0.2 x0,7=0,14	0.9 x0,3=0,27	0.14+0.27=0.41

Then summing each row gives us the requred probabilities.

By symmetry P (W Crash) is the same.

Updating with Bayes rule (given evidence "Watson has crashed")

After we discover that Watson has crashed we can compute P(Icy | W Crash =y) using Bayes rule:

$$P(\text{Icy} \mid \text{W Crash}=y) = \frac{P(\text{W Crash} = y \mid \text{Icy})P(\text{Icy})}{P(\text{W Crash} = y)}$$

$$= (0.8\text{x}0.7, 0.1\text{x}0.3)/0.59$$

$$= (0.95, 0.05)$$

Extracting the marginal

- To calculate P(H Crash | W Crash = y) we first calculate P(H Crash, Icy | W Crash)

P(H | W=y, Icy)	Icy = yes	Icy = no	
Holmes Crash = yes	0.8 x0.95=0,76	0.1 x0,05=0,005	0.765
Holmes Crash = no	0.2 x0.95=0,19	0.9 x0,05=0,045	0.235

Again, summing gives us P(H Crash | W Crash = yes)

P(H Crash | W Crash, Icy=no) = P(H Crash | Icy=no)
= (0.1,0.9)

Alternative perspective

Icy= no	0.3
Icy= yes	0.7

	Icy = yes	Icy = no
Watson Crash = yes	0.56	0.03
Watson Crash = no	0.14	0.27

	Icy = yes	Icy = no
Holmes Crash = yes	0.56	0.03
Holmes Crash = no	0.14	0.27

We represent the model as two joint tables, P(Watson, Icy) and P(Holmes, Icy) with a table for the overlap P(Icy).

Alternative perspective

Icy=no	0.3
Icy=yes	0.7

	Icy = yes	Icy = no
Watson Crash = yes	0.56/0.59 = 0.95	0.03/0.59= 0.05
Watson Crash = no	0	0

	Icy = yes	Icy = no
Holmes Crash = yes	0.56	0.03
Holmes Crash = no	0.14	0.27

If evidence on Watson arrives of the form $P^{New}(\text{W Crash}) = (1,0)$ then

$$P^{New}(\text{W Crash, Icy})$$

$$= P(\text{Icy} \mid \text{W Crash})\, P^{New}(\text{W Crash}) = \frac{P(\text{W Crash, Icy})}{P(\text{W Crash})}\, P^{New}(\text{W Crash})$$

Alternative perspective

Icy=no	0.95
Icy=yes	0.05

	Icy = yes	Icy = no
Watson Crash = yes	0.56/0.59 = 0.95	0.03/0.59= 0.05
Watson Crash = no	0	0

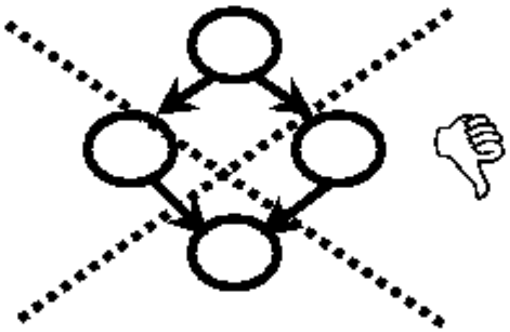
	Icy = yes	Icy = no
Holmes Crash = yes	0.56*0.95/0.7= 0.76	0.03*0.05/0.3= 0.005
Holmes Crash = no	0.14*0.95/0.7= 0.19	0.27*0.05/0.3= 0.045

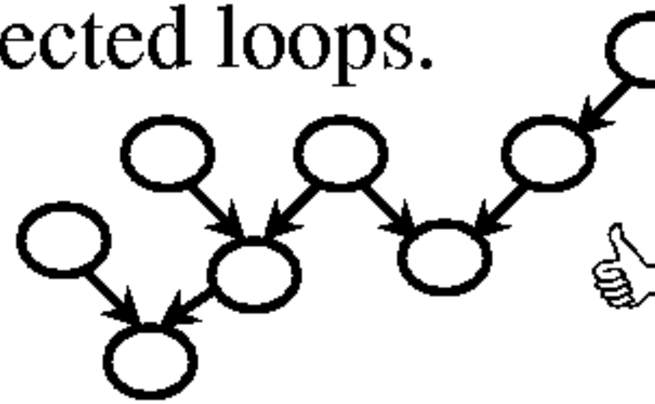
The table for Icy can then be updated by marginalizing the table for Watson. The table for Holmes can then be updated using the same rule:

$$P^{New}(H\ Crash, Icy) = \frac{P(H\ Crash, Icy)}{P(Icy)} P^{New}(Icy)$$

Polytrees

- A network is *singly connected* (a *polytree*) if it contains no undirected loops.

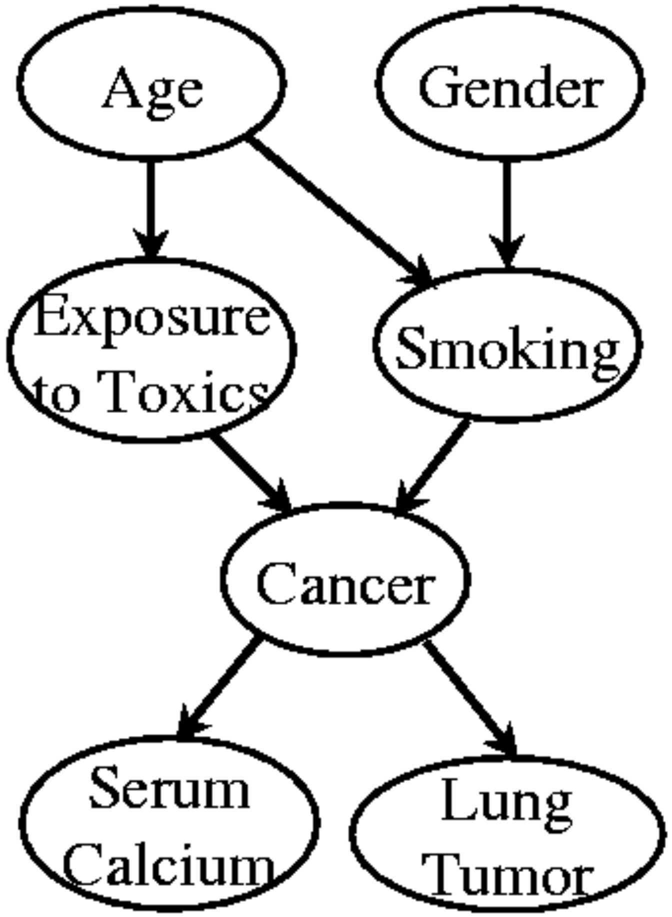


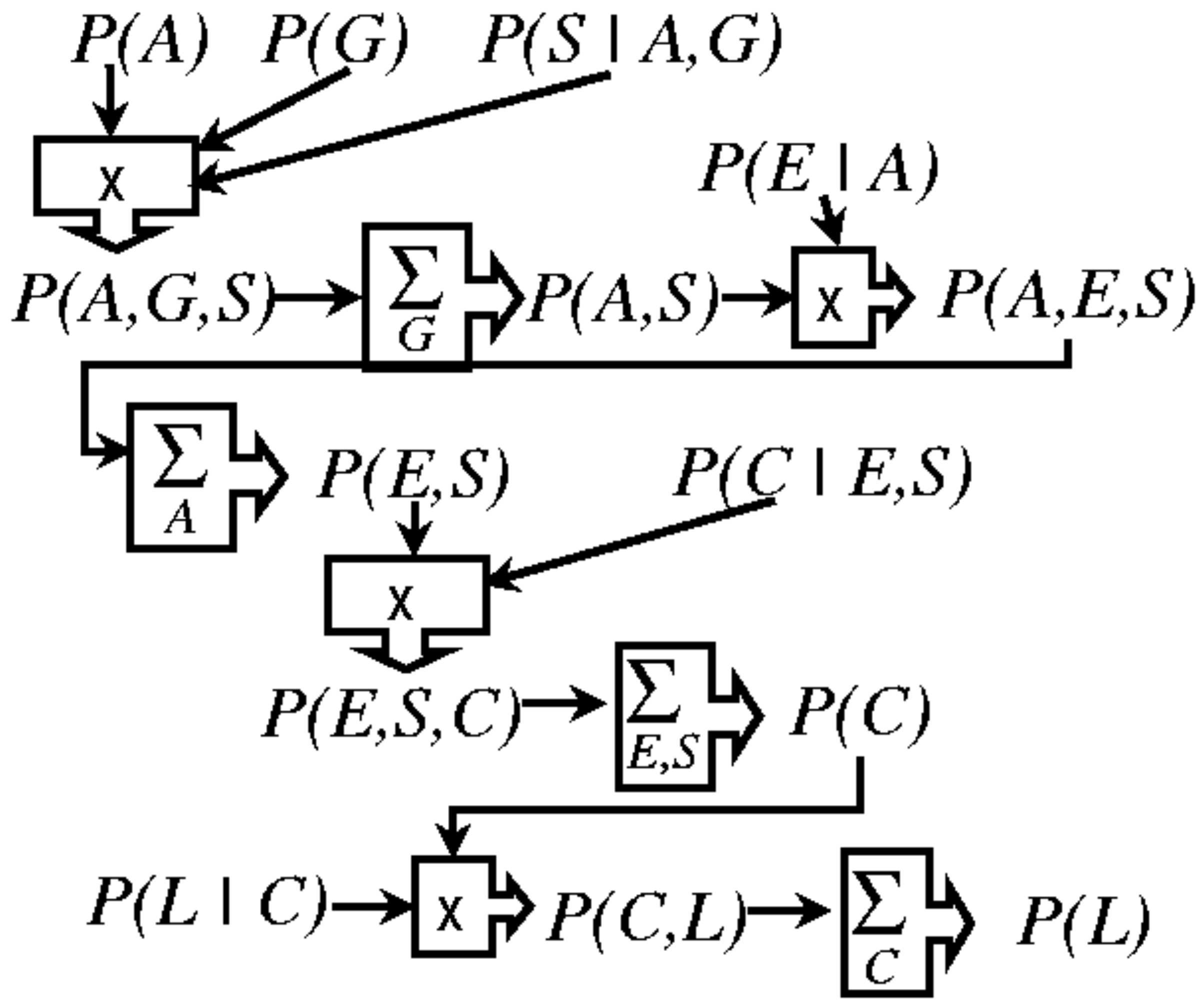


Theorem: Inference in a singly connected network can be done in linear time*.

Main idea: in variable elimination, need only maintain distributions over single nodes.

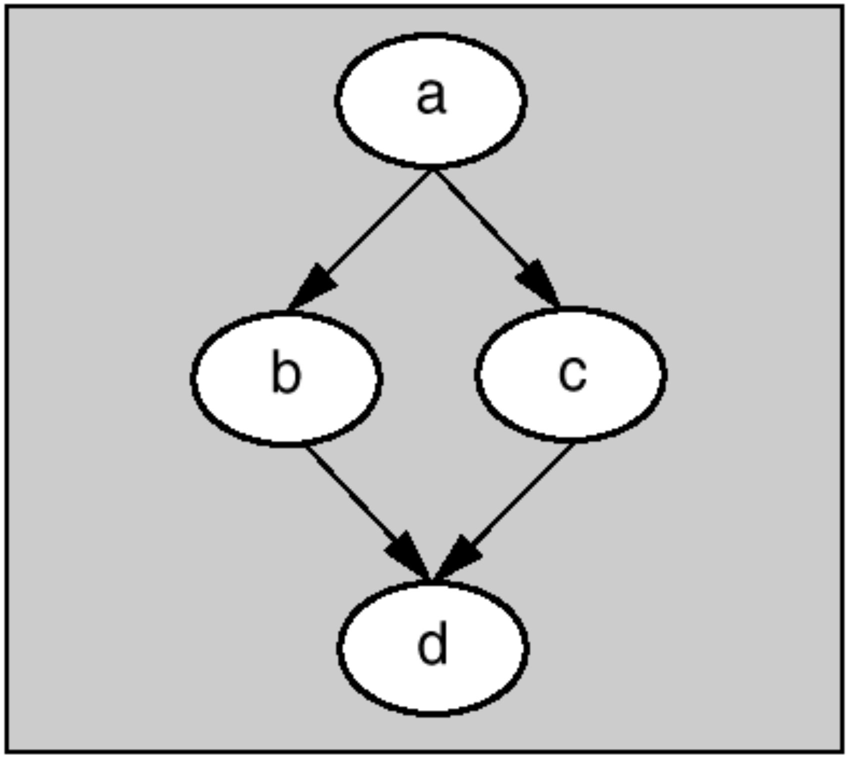
Variable Elimination with loops





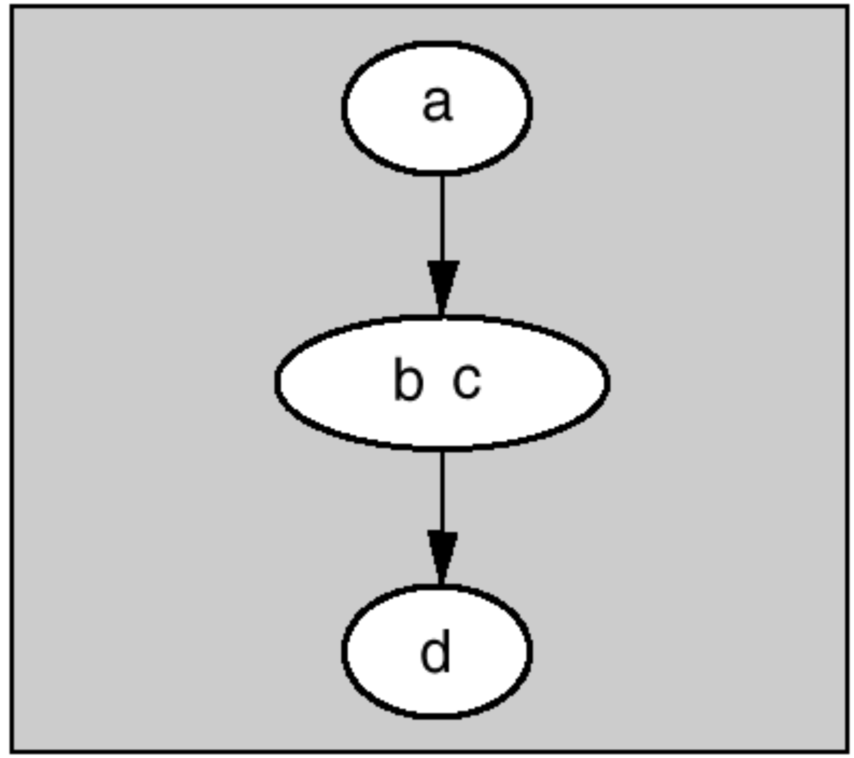
Complexity is exponential in the size of the factors

Join Trees



A Multiply Connected Network.

There are two paths between node a *and node* d.



A Clustered, Multiply Connected Network.

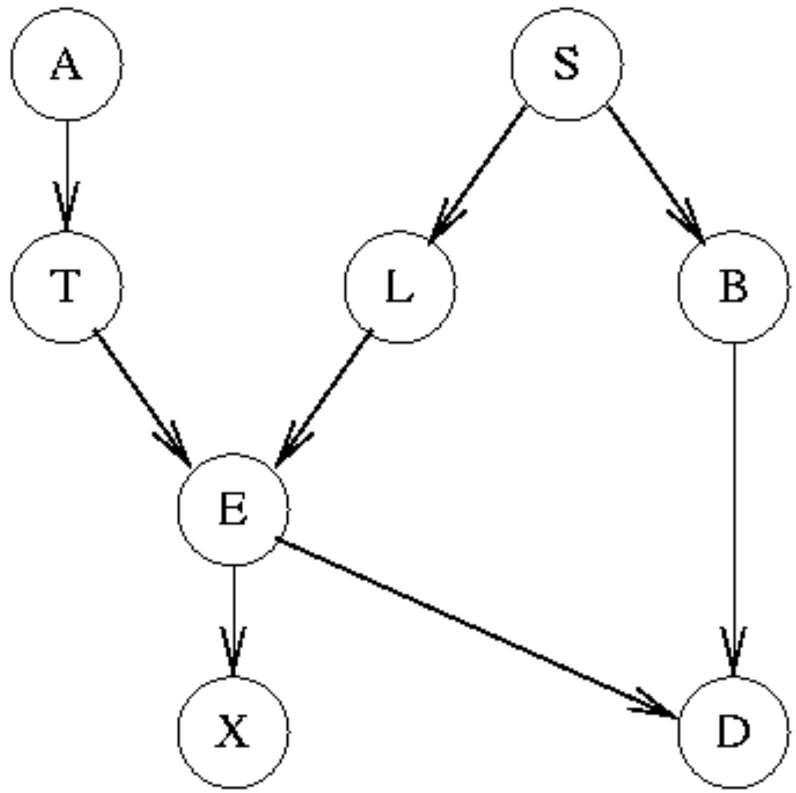
By clustering nodes b *and* c, *we turned the graph into a singly connected network.*

Graphical Method of Building the Junction Tree

The Junction Tree can be constructed through a series of graph operations

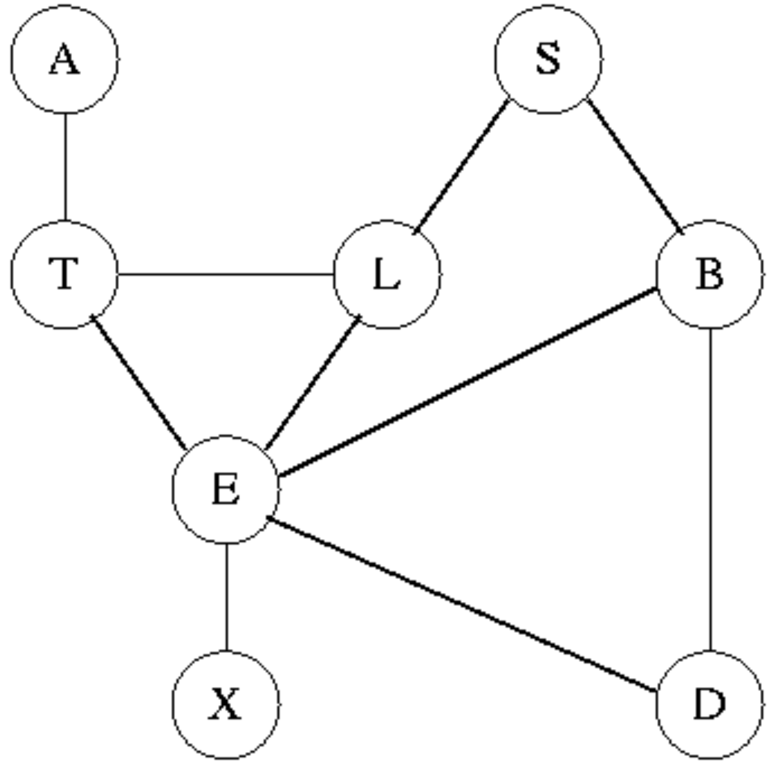
- **Marry the Parents** ("moralize the graph"): Add an undirected edge between every pair of parents of a node (unless they are already connected.
- **Make All Arrows Undirected**
- **Triangulate the Graph:** Add edges so that every cycle of length 4 or more contains a chord.
- **Identify the maximal Cliques:** A clique is a complete graph. A maximal clique is a maximal complete subgraph.
- **Form Junction Graph:** Create a cluster node for each clique and label it with the variables in the clique.
  Create an edge between any pair of cluster nodes that share variables.
  Place a separator node on the edge labeled with the set of variables shared by the cluster nodes it joins.
- **Form the junction tree:** Compute a maximum weighted spanning tree of the junction graph where the weight on each edge is the number of variables in the separator of the edge.

Example



$$P(U) = P(A)P(S)P(T|A)P(L|S)P(B|S)P(E|L,T)P(D|B,E)P(X|E)$$

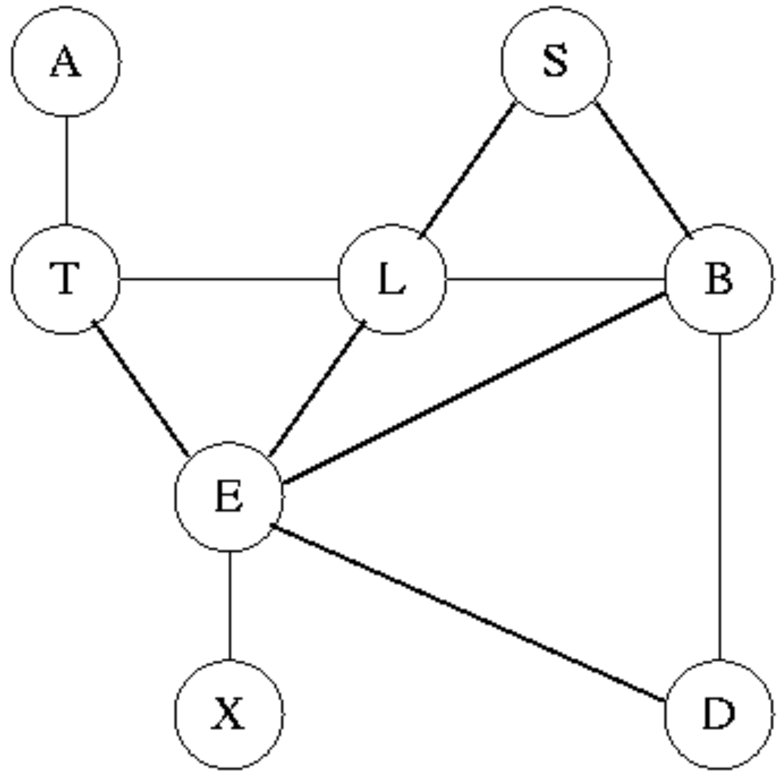
Step 1: Moralize the Graph



We join **T** and **L** because they are parents of **E**.

We join **E** and **B** because they are parents of **D**.

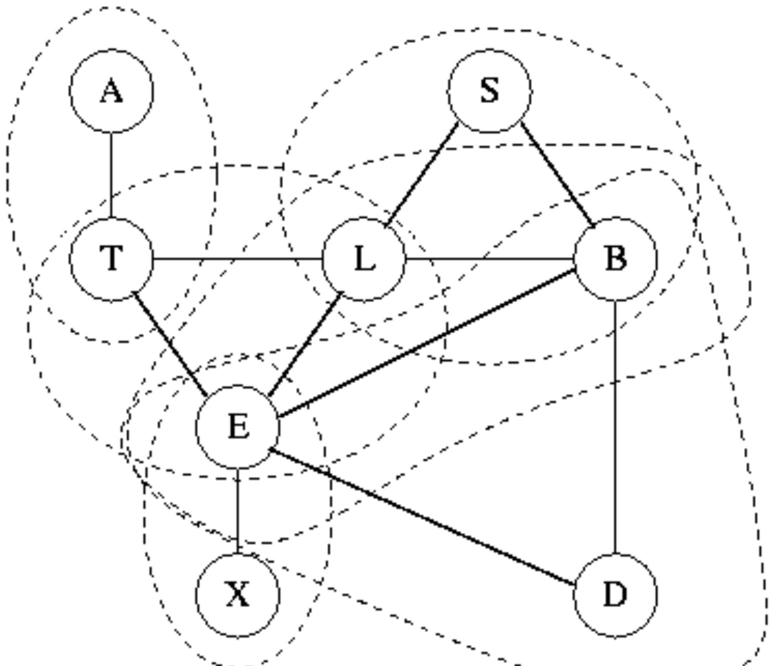
Step 2: Triangulate the Moral Graph

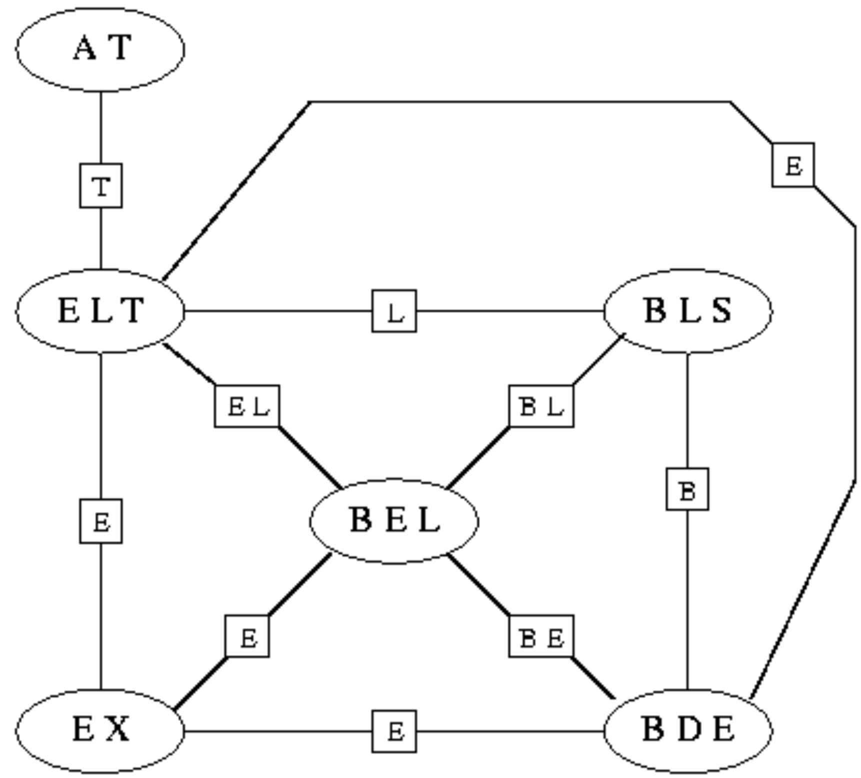


There is a cycle of length four with no shortcuts: **E**, **L**, **S**, **B**.

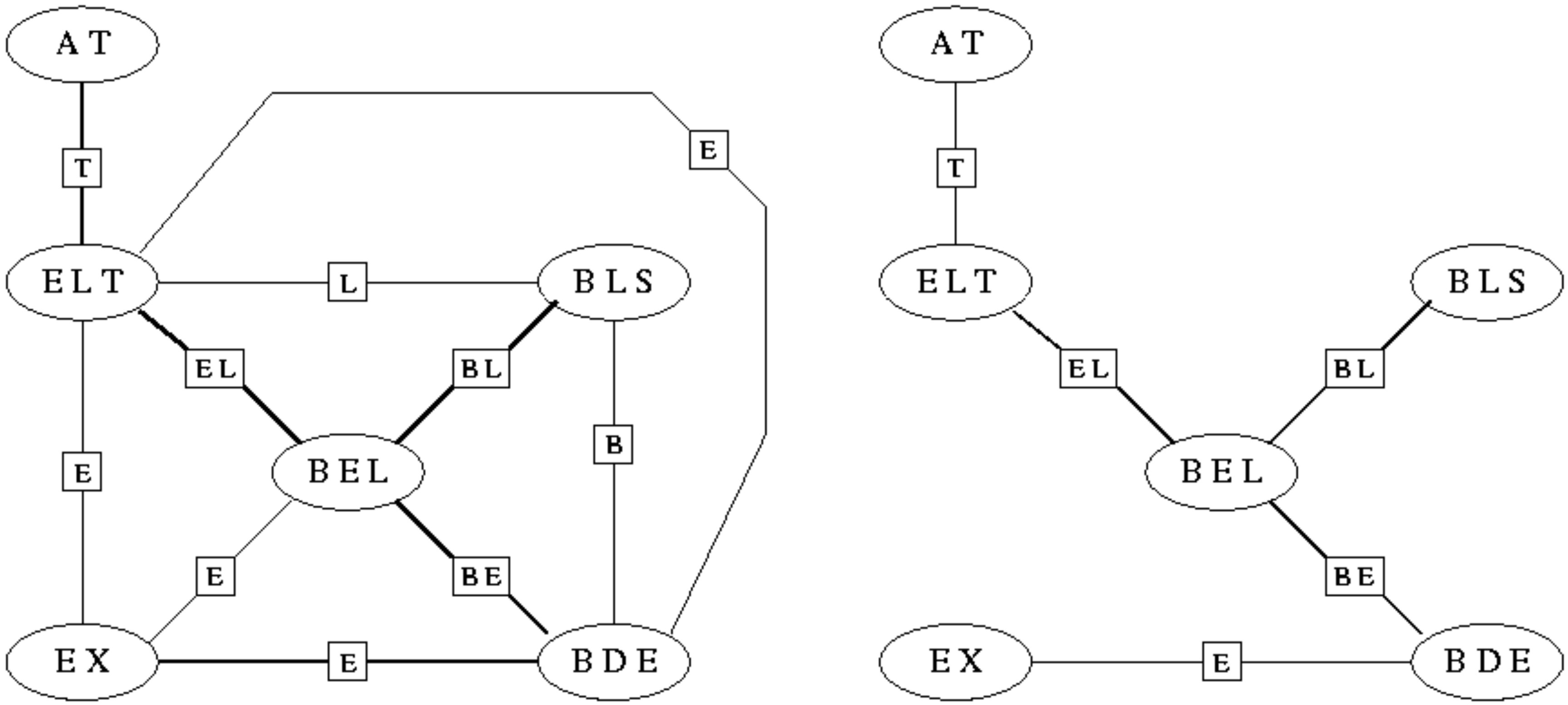
We have a choice of where to add the shortcut. Either **LB** or **SE** would work.

Step 3: Cliques and Junction Graph





Step 4: Junction Tree



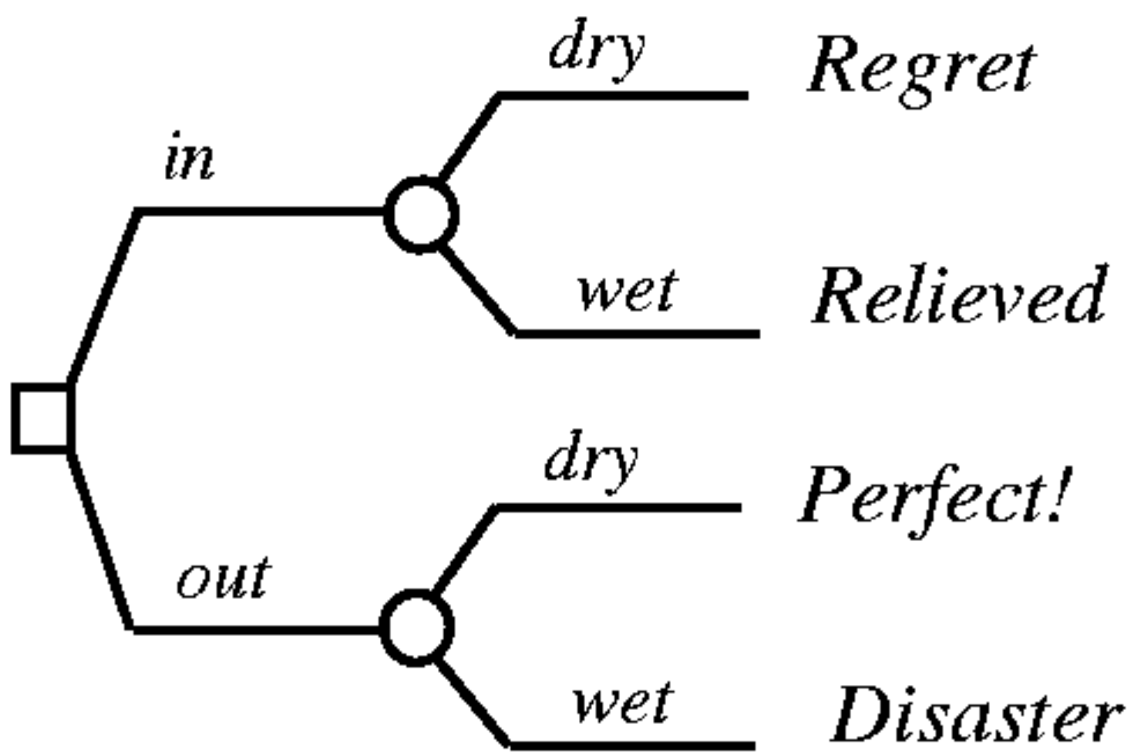
Notice that the running intersection property holds (this is guaranteed by the maximum weight spanning tree and the moralizing and triangulating edges).

Decision making

- Decision - an irrevocable allocation of domain resources
- Decision should be made so as to maximize expected utility.
- View decision making in terms of
  - Beliefs/Uncertainties
  - Alternatives/Decisions
  - Objectives/Utilities

A Decision Problem

Should I have my party inside or outside?

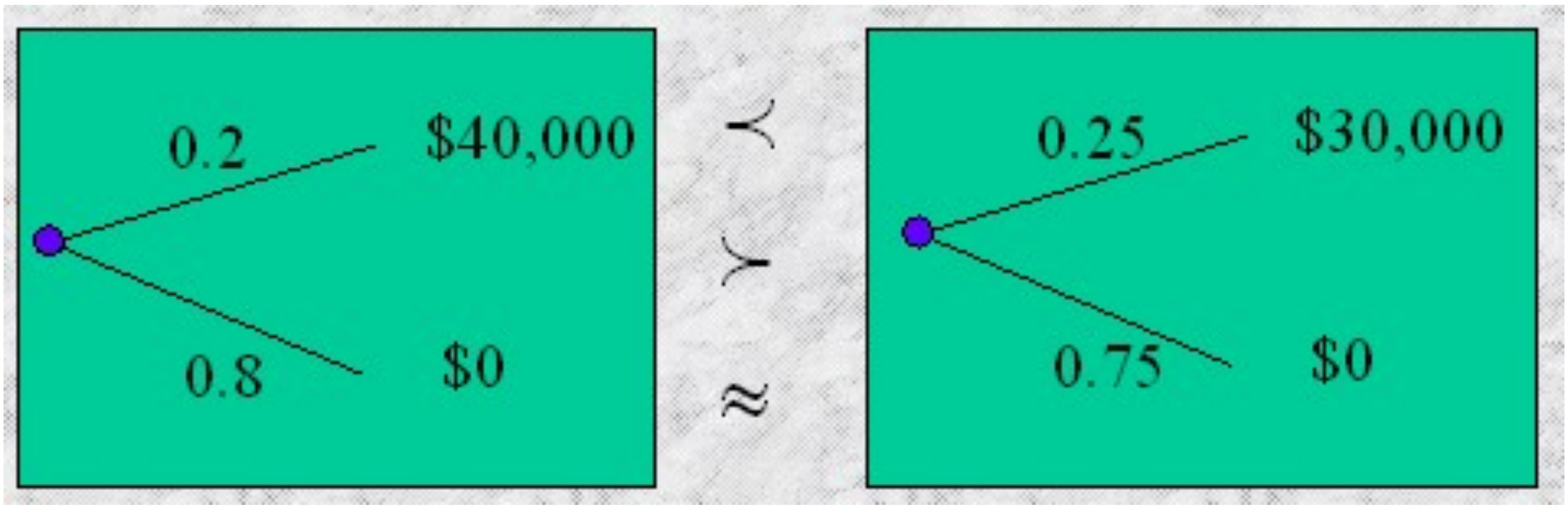


Value Function

- A numerical score over all possible states of the world.

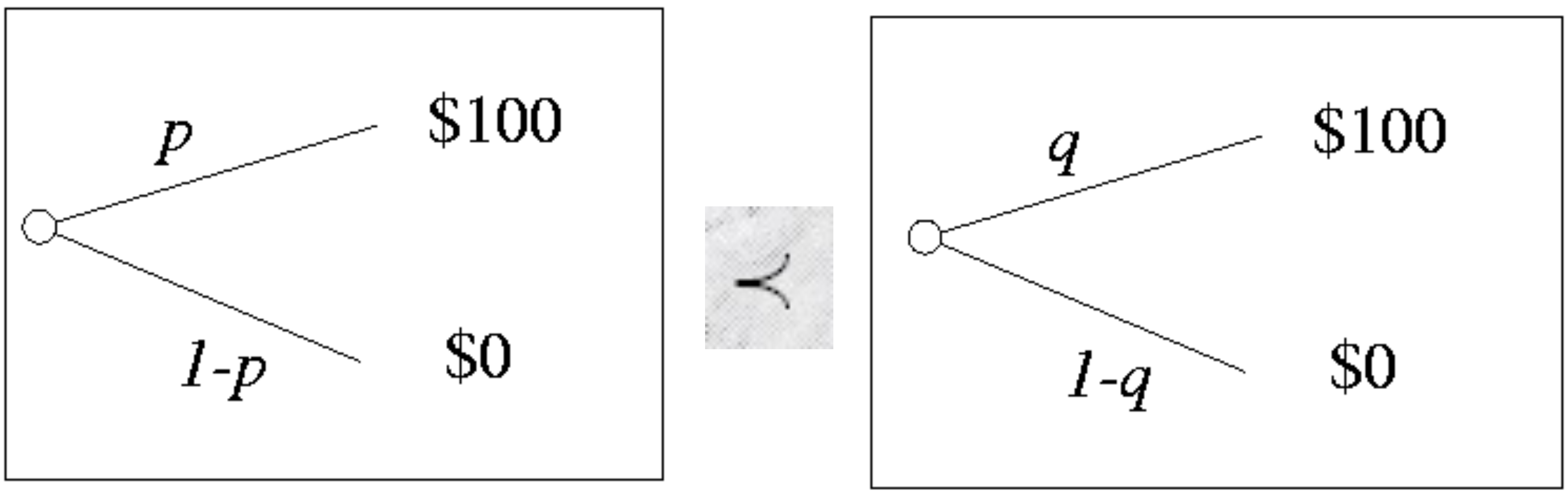
Location?	Weather?	Value
in	dry	$50
in	wet	$60
out	dry	$100
out	wet	$0

Preference for Lotteries



Desired Properties for Preferences over Lotteries

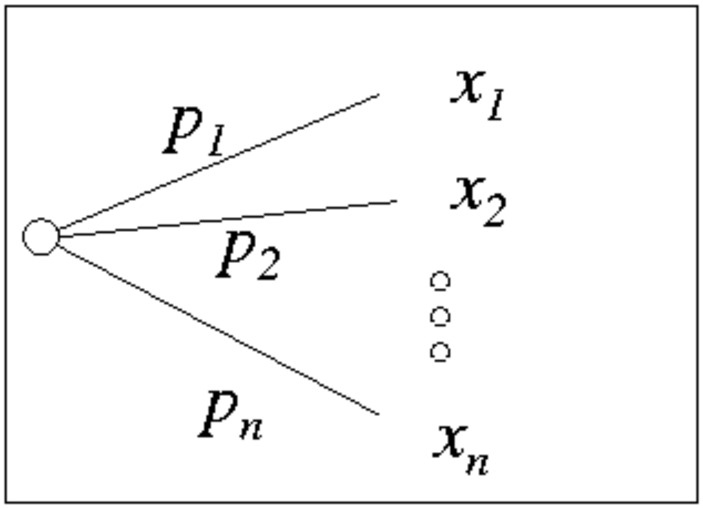
If you prefer \$100 to \$0 and $p < q$ then



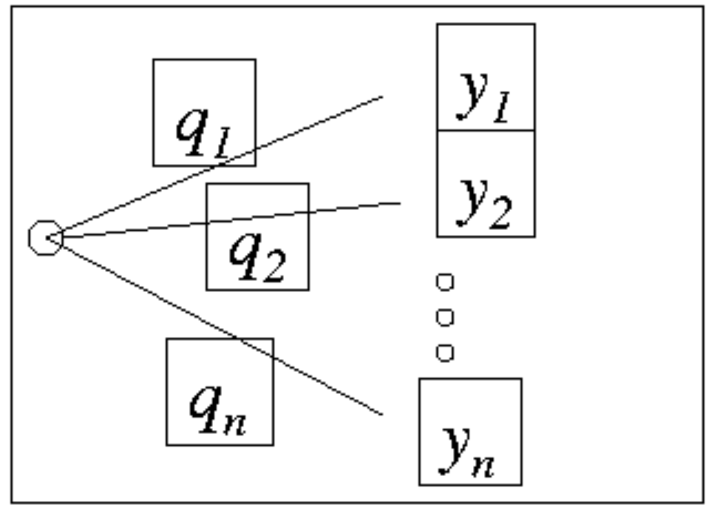
(always)

Expected Utility

Properties of preference $\Rightarrow$
existence of function U, that satisfies:



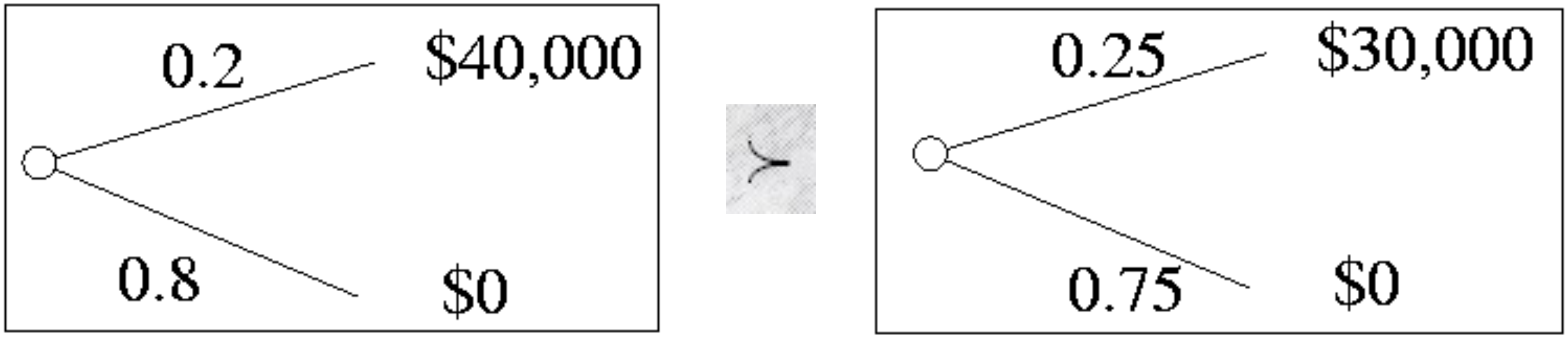
$\prec$



iff

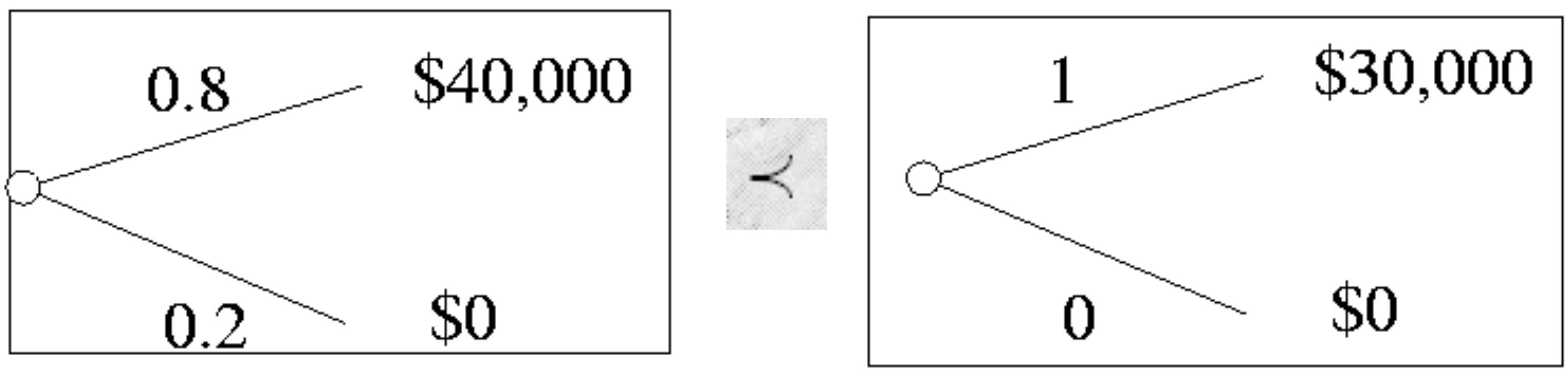
$$\Sigma_i\, p_i\, U(x_i) \quad < \quad \Sigma_i\, q_i\, U(y_i)$$

Are people rational?



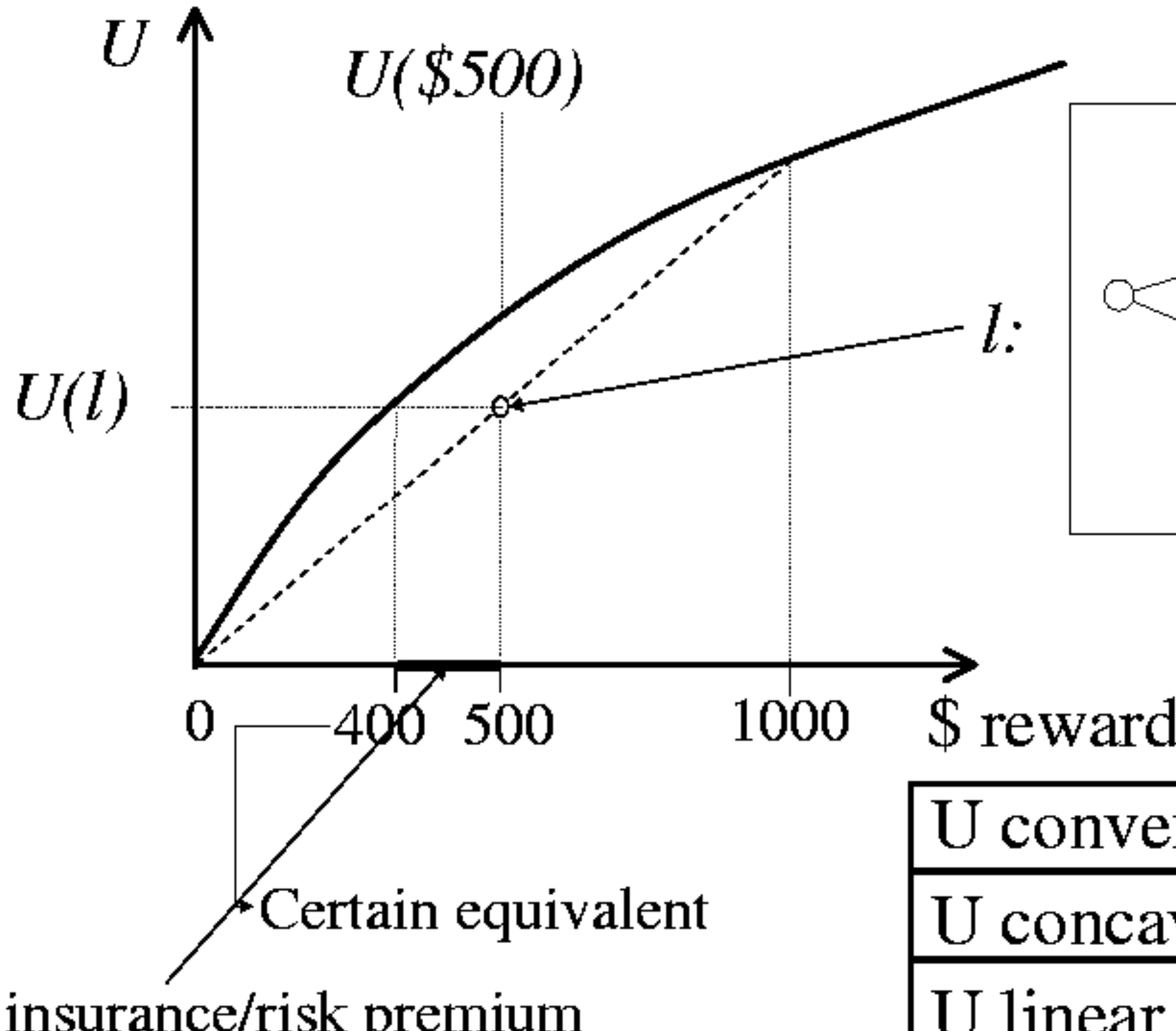
$$0.2 \bullet U(\$40k) \quad > \quad 0.25 \bullet U(\$30k)$$

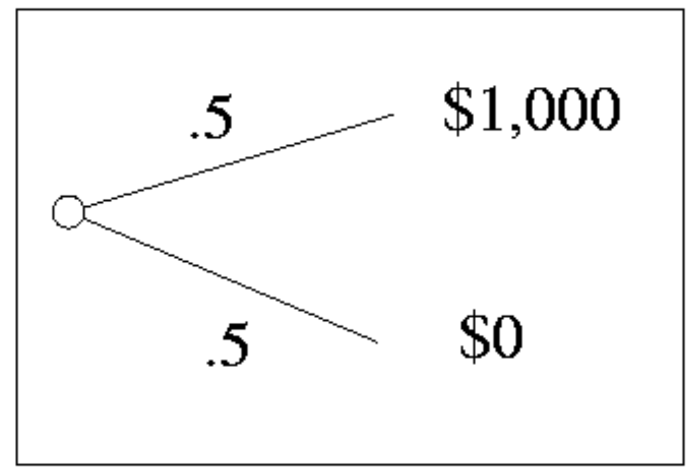
$$0.8 \bullet U(\$40k) \quad > \quad U(\$30k)$$



$$0.8 \bullet U(\$40k) \quad < \quad U(\$30k)$$

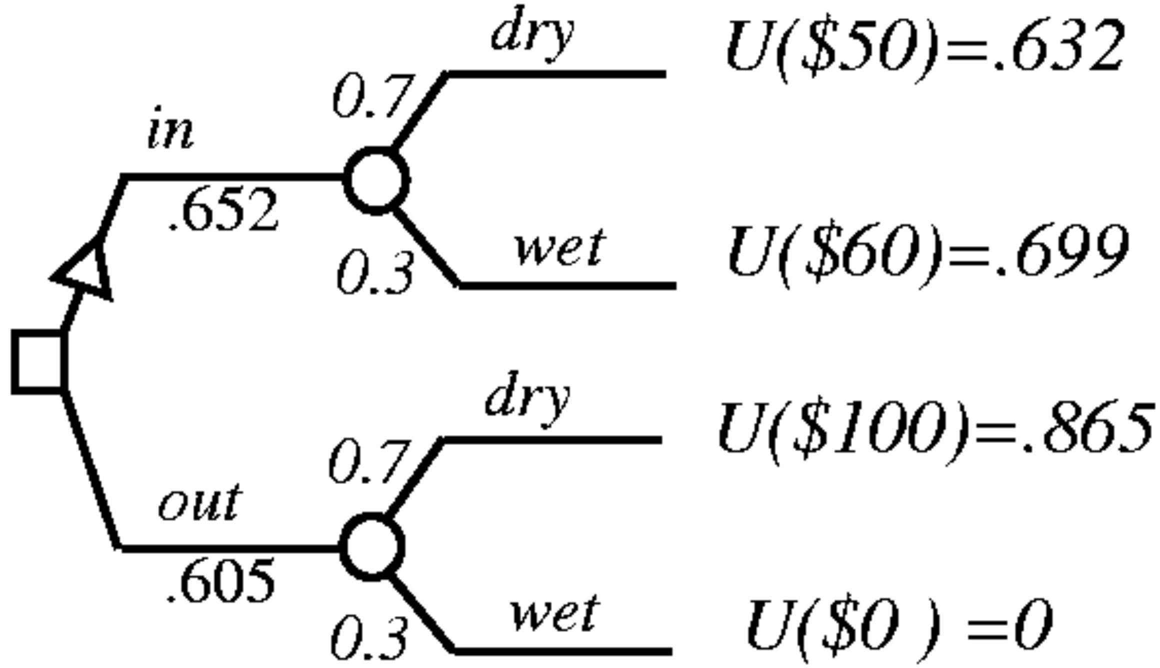
Attitudes towards risk





U convex	risk averse
U concave	risk seeking
U linear	risk neutral

Maximizing Expected Utility



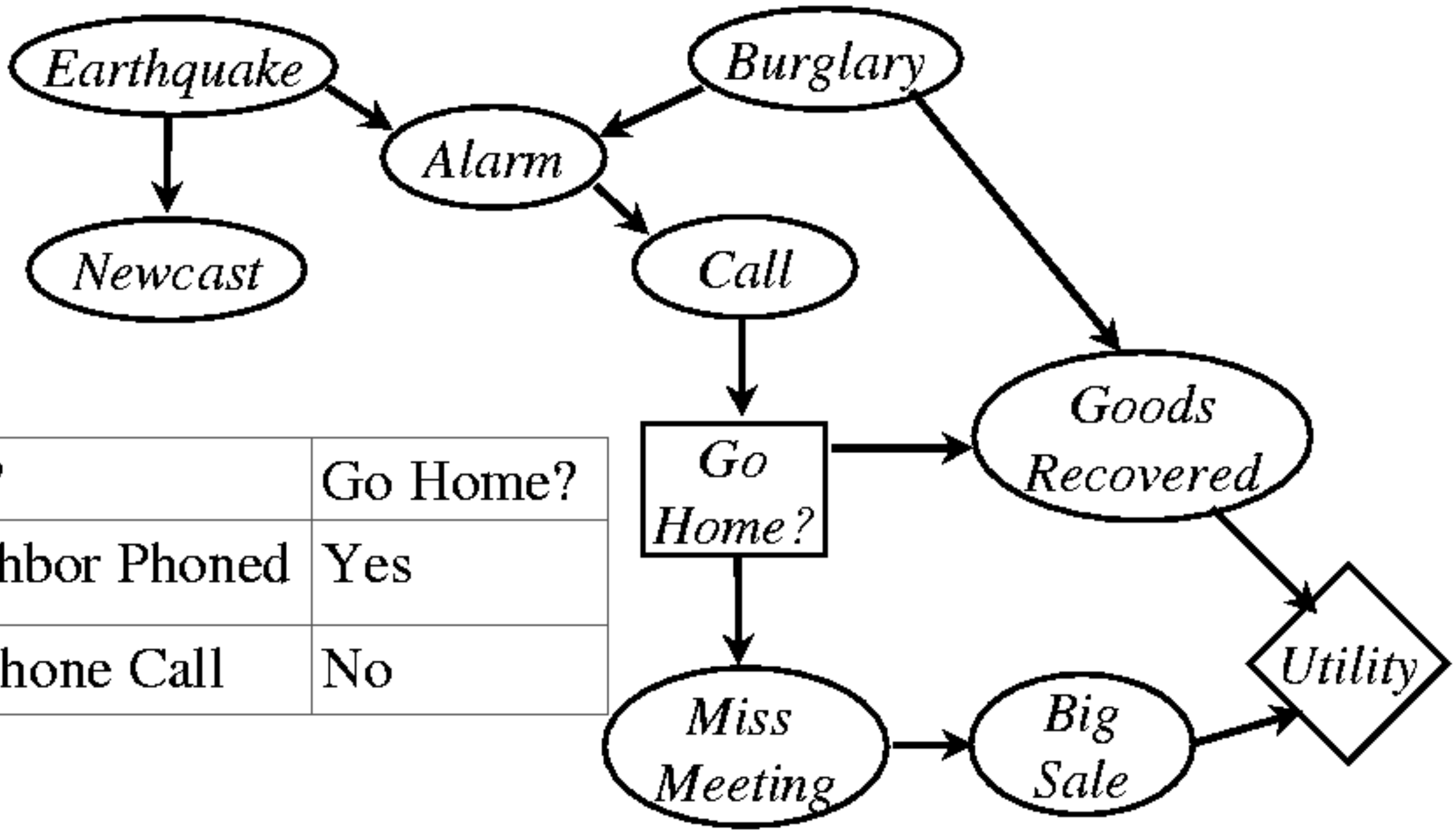
choose the action that maximizes expected utility

$$EU(in) = 0.7 \cdot .632 + 0.3 \cdot .699 = .652$$

$\Rightarrow$ Choose *in*

$$EU(out) = 0.7 \cdot .865 + 0.3 \cdot 0 = .605$$

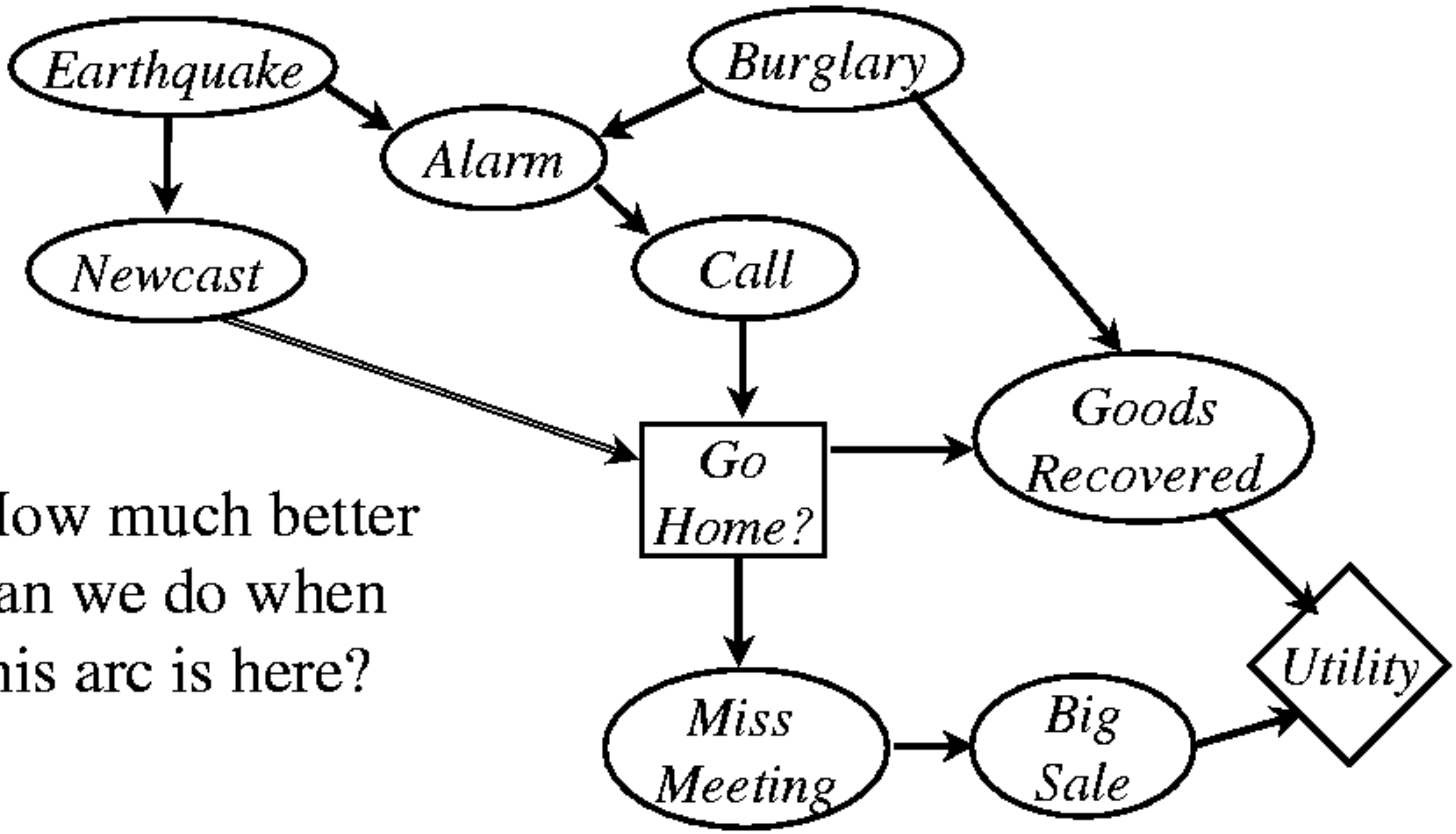
Decision Making with Influence Diagrams



Call?	Go Home?
Neighbor Phoned	Yes
No Phone Call	No

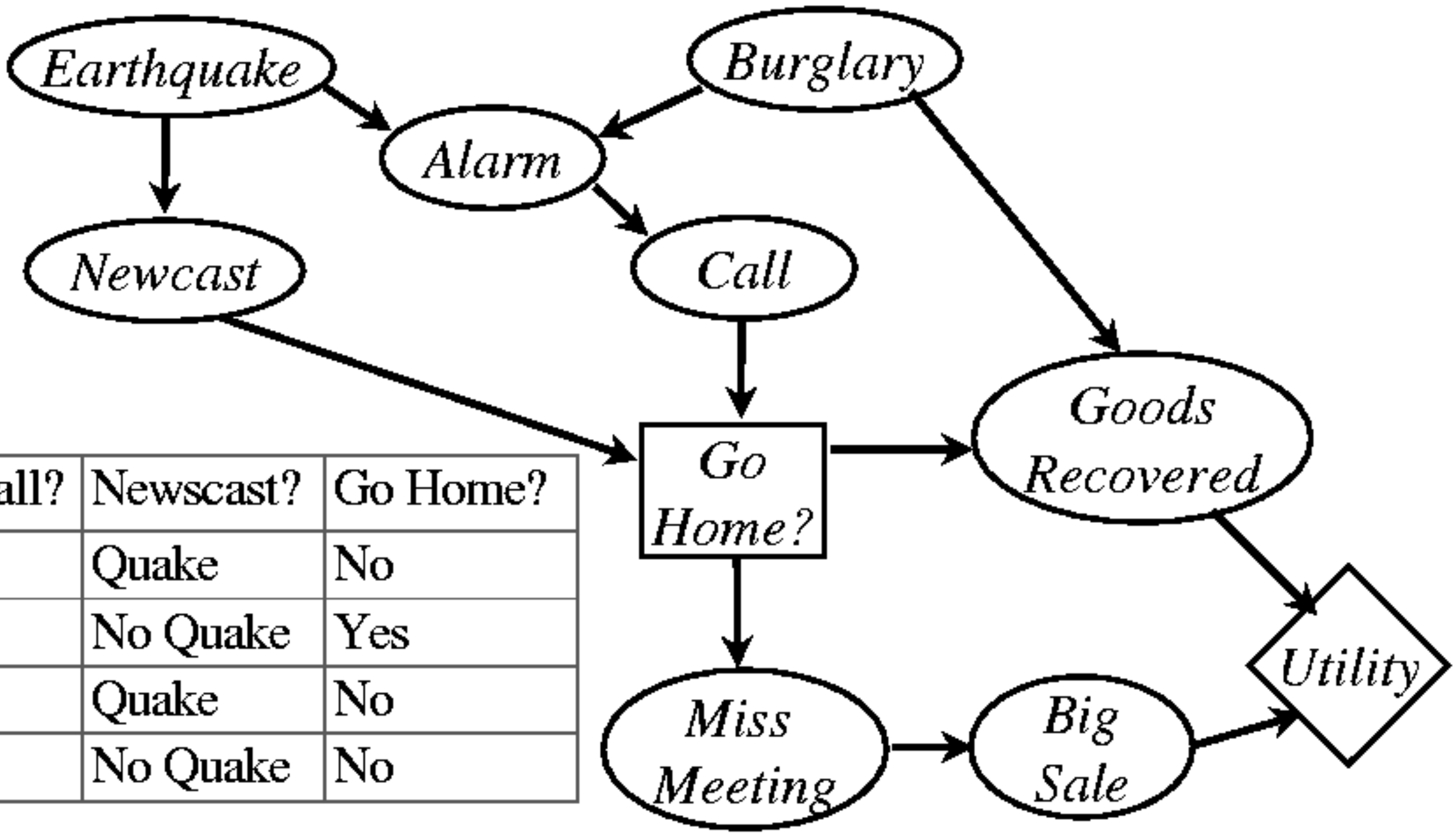
Expected Utility of this policy is 100

Value-of-Information in an Influence Diagram



How much better can we do when this arc is here?

Value-of-Information is the increase in Expected Utility

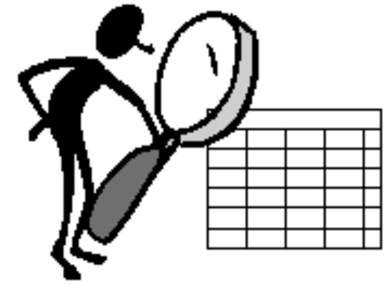


Phonecall?	Newscast?	Go Home?
Yes	Quake	No
Yes	No Quake	Yes
No	Quake	No
No	No Quake	No

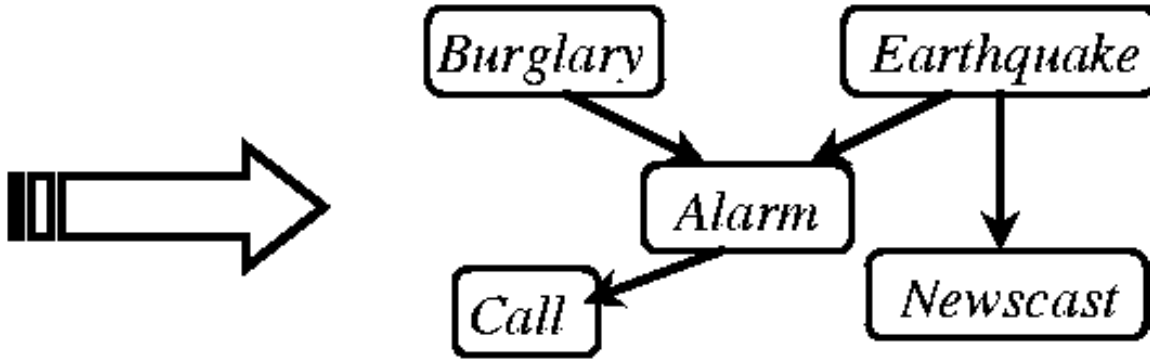
Expected Utility of this policy is 112.5

LEARNING BAYES NETS

The learning task



B	*E*	*A*	*C*	*N*
$\bar{b}$	e	a	c	$\bar{n}$
b	$\bar{e}$	$\bar{a}$	$\bar{c}$	n
		⋮		

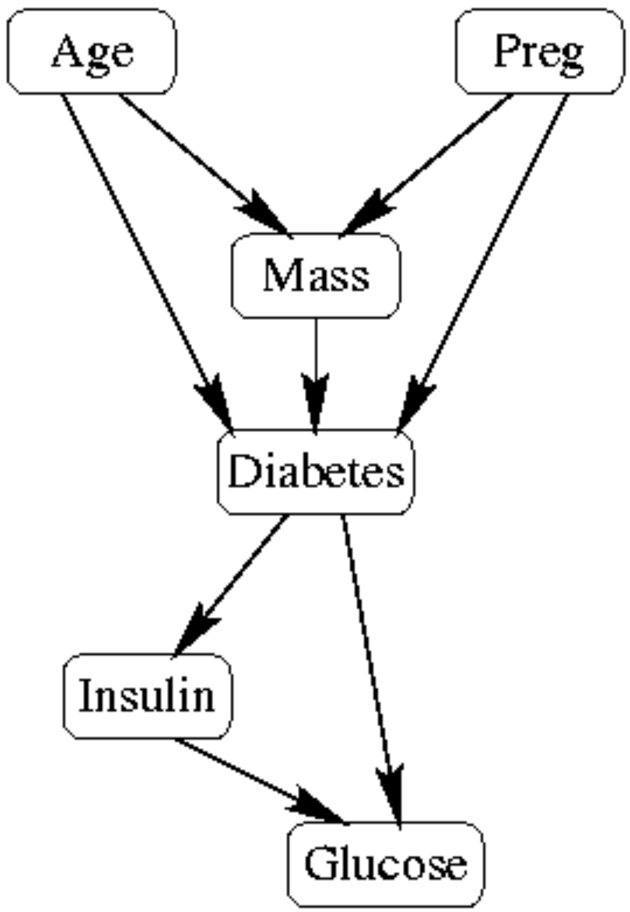


Input: training data

Output: BN modeling data

- Input: fully or partially observable data cases?
- Output: parameters or also structure?

Known structure, Fully Observable



Preg	Glucose	Insulin	Mass	Age	Diabetes
5	121	112	26.2	30	0
10	101	180	32.9	63	0
7	137	0	32.0	39	0
12	100	105	30.0	46	0
9	140	0	32.7	45	1
1	102	0	39.5	42	1
2	99	160	36.6	21	0
2	174	120	44.5	24	1
1	111	0	32.8	45	0
5	117	105	39.1	42	0

Learning Process

- **Discretize the Data**

  Glucose $< 100 \Rightarrow 0$

  $100 \leq$ Glucose $< 120 \Rightarrow 1$

  $120 \leq$ Glucose $< 140 \Rightarrow 2$

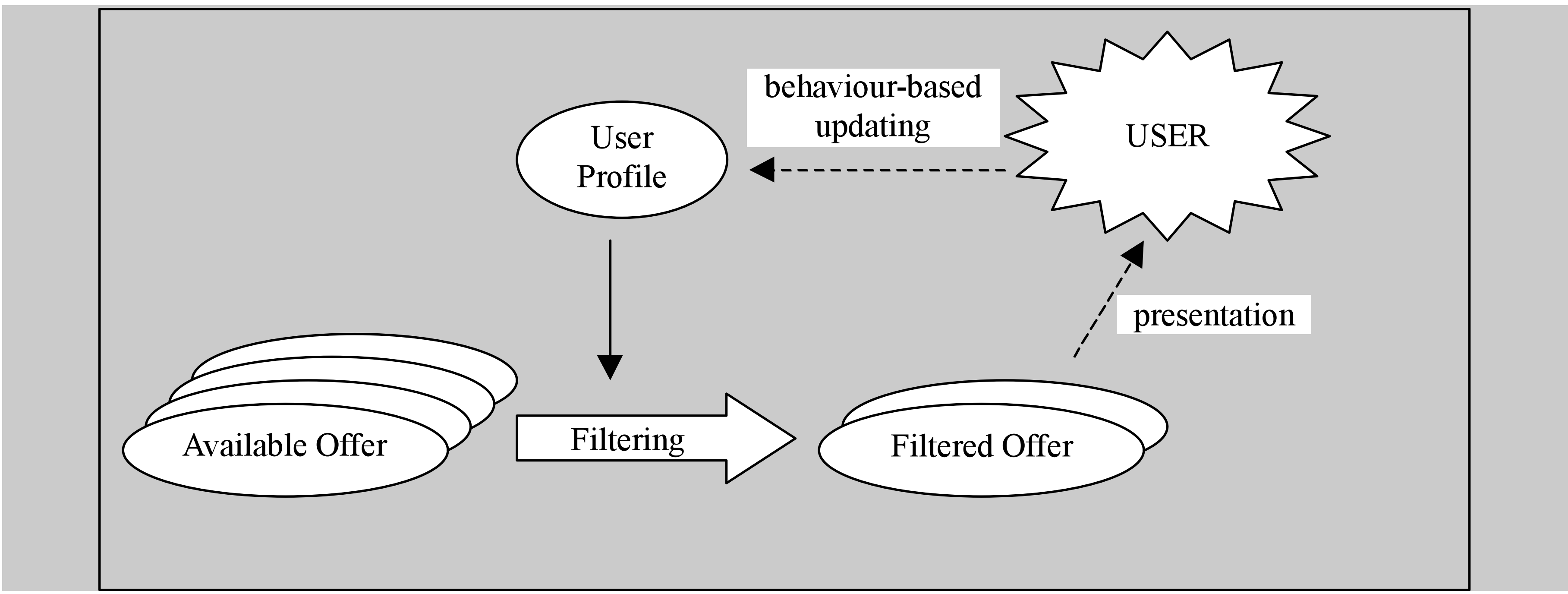
  $140 \leq$ Glucose $\Rightarrow 3$

- **Count Cases**

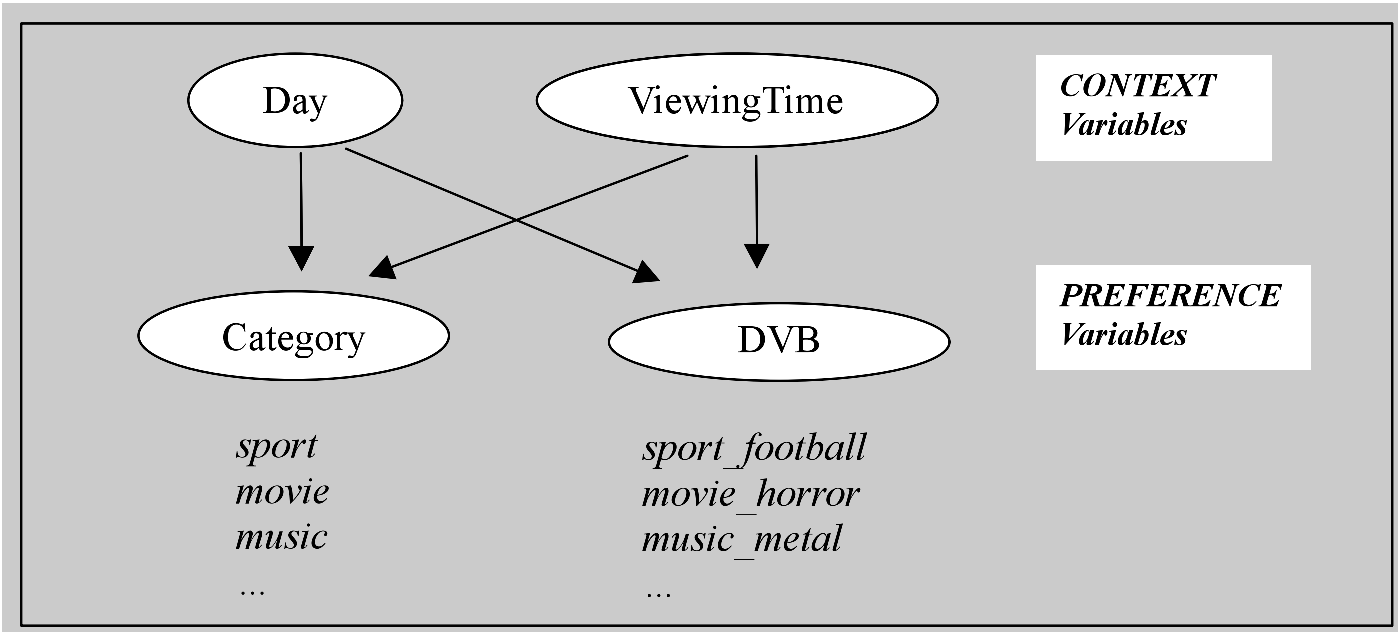
$$P(Mass = 0 | Preg = 1, Age = 2) = \frac{N(Mass = 0, Preg = 1, Age = 2)}{N(Preg = 1, Age = 2)}$$

Read more about Learning BN in: http://http.cs.berkeley.edu/~murphyk/Bayes/learn.html

Profiling with Bayes Nets
User Profiling: the problem



The BBN encoding the user preference



- Preference Variables:
  what kind of TV programmes does the user prefer and how much?
- Context Variables:
  in which (temporal) conditions does the user prefer …?

BBN based filtering

1) From each item of the input offer extract:

- **the classification**
- **the possible (empty) context**

2) For each item compute

Prob **(<*classification*> | <*context*>)**

3) Items with highest probabilities are the output of the filtering

Example of filtering

The input offer is a set of 3 items:

1. **a concert of classical music on Thursday afternoon**
2. **a football match on Wednesday night**
3. **a subscription for 10 movies on evening**

The probabilities to be computed are:

1. **P (MUS = CLASSIC_MUS | Day = Thursday, ViewingTime = afternoon)**

2. **P (SPO = FOOTBAL_SPO | Day = Wednesday, ViewingTime = night)**

3. **P (CATEGORY = MOV | ViewingTime = evening)**

BBN based updating

- **The BBN of a new user is initialised with uniform distributions**
- **The distributions are updated using a Bayesian learning technique on the basis of user's actual behaviour**
- **Different user's behaviours -> different *learning weights*:**
  1) the user declares their preference
  2) the user watches a specific TV programme
  3) the user searches for specific kind of programmes