Bayesian Reasoning and Bayesian Networks

Kees van Deemter
Adapting slides from *Rosen & Rusconi* at UCL; *Weng-Keen Wong* at Oregon State University; *Chris Mellish* at Aberdeen; and many others

Bayesian reasoning and networks

Possibly the most significant bit of AI so far

I. Bayesian reasoning (statistics; psychology)

II. Bayesian networks (computing)

III. The Noisy Channel model (e.g. NLP)

I. Bayesian Reasoning

First:

Basics of probability
for Bayesian reasoning

Bayesian probability is a measure of the plausibility of a proposition. One interpretation: the degree of belief that a rational agent should attach to the proposition

Counting and experiments can underpin Bayesian probabilities, but they are not necessary.

Probability: one type of Bayesian Definition

- Suppose a rational, profit-maximizing agent R is offered a choice between two rewards:
  - Winning \$1 if and only if the event E occurs.
  - Receiving p dollars (where $p \in [0,1]$) unconditionally.
- If R can honestly state that he is indifferent between these two rewards, then we say that R's probability for E is p, that is, $\Pr_R[E] :\equiv p$.
- **Problem:** A subjective definition; depends on R, and his knowledge, beliefs, & rationality.
  - The version above additionally assumes that the utility of money is linear

Calculations in Bayesian probability often work in the same way as in frequentist probability ...

... only the probabilities of atomic events may have a different source (estimation, experimentation, gut feeling, etc.)

Some special terminology: prior and posterior probability; marginalisation

Probabilities

- Probability distribution $P(X|\xi)$
  - X is a random variable
    - Discrete
    - Continuous
  - ξ is background state of information

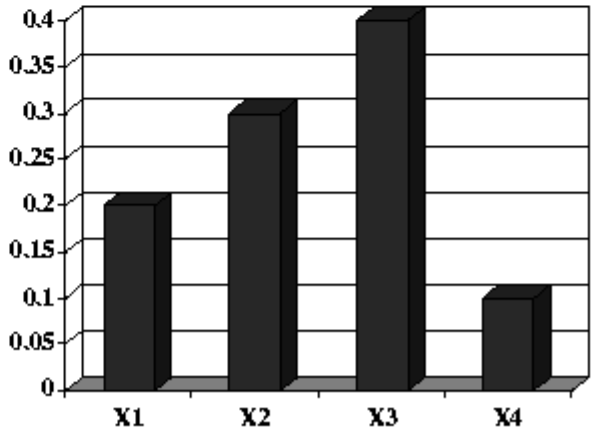
Discrete Random Variables

- Finite set of possible outcomes

$$X \in \{x_1, x_2, x_3, \ldots, x_n\}$$

$$P(x_i) \geq 0$$

$$\sum_{i=1}^{n} P(x_i) = 1$$



X binary: $P(x) + P(\overline{x}) = 1$

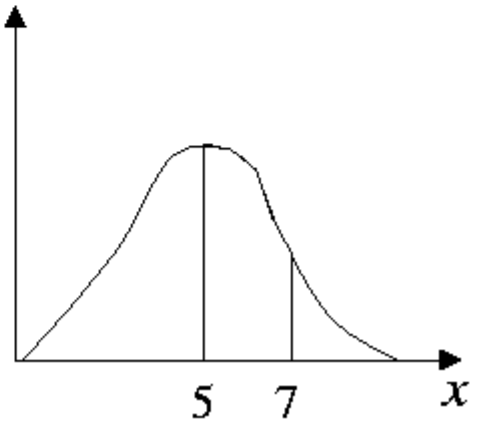
Continuous Random Variable

- Probability distribution (density function) over continuous values

$$X \in [0,10] \qquad P(x) \geq 0$$

$$\int_0^{10} P(x)\,dx = 1$$

$$P(5 \leq x \leq 7) = \int_5^7 P(x)\,dx$$



More Probabilities

- Conditional

$$P(x \mid y) \equiv P(X = x \mid Y = y)$$

  - Probability that $X=x$ given we know that $Y=y$

- Joint

$$P(x, y) \equiv P(X = x \wedge Y = y)$$

  - Probability that both $X=x$ and Y=y

Conditional independence

- Two events A and B are said to be (probabilistically) independent if:

$$P(A,B) = P(A)P(B)$$

equivalently $P(A \mid B) = P(A)$ $\quad (P(B) > 0)$

- Two events A and B are said to be (probabilistically) independent given C if:

$$P(A,B \mid C) = P(A \mid C)P(B \mid C)$$

equivalently $P(A \mid B,C) = P(A \mid C)$ $\quad (P(B) > 0)$

Rules of Probability

- Product Rule

$$P(X,Y) = P(X \mid Y)P(Y) = P(Y \mid X)P(X)$$

- Marginalization

$$P(Y) = \sum_{i=1}^{n} P(Y, x_i)$$

X binary: $P(Y) = P(Y,x) + P(Y,\bar{x})$

Prove: $P(Y) = P(Y \wedge A) + P(Y \wedge \neg A)$

Prove: $P(Y) = P(Y \wedge A) + P(Y \wedge \neg A)$

$Y \Leftrightarrow (Y \wedge A) \vee (Y \wedge \neg A)$, therefore

$P(Y) = P((Y \wedge A) \vee (Y \wedge \neg A))$

These disjuncts are mutually exclusive, so

$P(Y) = P(Y \wedge A) + P(Y \wedge \neg A)$ $\square$

Bayes Law (repeated with proof)

$P(E_1 \text{ and } E_2) = P(E_1) * P(E_2 \mid E_1)$

$P(E_2 \text{ and } E_1) = P(E_2) * P(E_1 \mid E_2)$. Therefore,

$P(E_2) * P(E_1 \mid E_2) = P(E_1) * P(E_2 \mid E_1)$. So,

$$P(E_1 \mid E_2) = \frac{P(E_1) * P(E_2 \mid E_1)}{P(E_2)}$$

Bayesian Reasoning

People often struggle with conditional probabilities

Much studied: interpreting results of medical tests (e.g., G.Gigerenzer 2002, "Reckoning with Risk")

Other examples in D.Kahneman's "Thinking Fast, Thinking Slow"

Bayesian Reasoning example

ASSUMPTIONS

1% of women aged forty who participate in a routine screening have breast cancer

80% of women with breast cancer will get positive tests

9.6% of women without breast cancer will also get positive tests

EVIDENCE

A woman in this age group had a positive test in a routine screening

PROBLEM

What's the probability that she has breast cancer?

Compact Formulation

C = cancer present, T = positive test

PRIOR PROBABILITY

$p(C) = 1\%$

CONDITIONAL PROBABILITIES

$p(T|C) = 80\%$

$p(T|\sim C) = 9.6\%$

POSTERIOR PROBABILITY

$p(C|T) = ?$

Bayes' theorem in this situation

$$p(C|T) = \frac{p(T|C)*p(C)}{p(T)}$$

How do we find p(T)?

A variant of Bayes' theorem

$$p(C|T) = \frac{p(T|C)*p(C)}{p(T|C)*p(C) + p(T|\sim C)*p(\sim C)}$$

A

A + C

Bayesian Reasoning

Prior Probabilities:

0.01 = p(C) 0.99 = p(~C)

Conditional Probabilities:

A = 0.8 * 0.01 = p(T|C)*p(C) = 0.008

C = 0.096 * 0.99 = p(T|~C)*p(~C) = 0.095

Rate of cancer patients with positive results, within the group of ALL patients with positive results:

A/(A+C) = 0.008/(0.008+0.095) = 0.008/0.103 = 0.078 = 7.8%

Formula with these numbers

$$A = 0.8 * 0.01 = 0.008$$

$$p(C|T) = \frac{p(T|C)*p(C)}{p(T|C)*p(C) + p(T|\sim C)*p(\sim C)}$$

$$A + C$$

$$0.008 + 0.095 = 0.103$$

Outcome: 0.078

Revised: p(C) = 0.1

How does this affect p(C|T)?

Revised: p(C) = 0.1

Prior Probabilities:

0.1 = p(C) (was 0.01) 0.9 = p(~C) (was 0.09)

Conditional Probabilities:

A = 0.8 * 0.1 = p(T|C)*p(C) = 0.08 (was 0.008)

C = 0.096 * 0.9 = p(T|~C)*p(~C) = 0.0864 (was 0.095)

Rate of cancer patients with positive results, within the group of ALL patients with positive results:

A/(A+C) = 0.08/(0.08+0.0864) = 0.08/0.1664 = 0.48 = 48%

Formula with revised numbers

$A = 0.8 * 0.1 = 0.08$

$$p(C|T) = \frac{p(T|C)*p(C)}{p(T|C)*p(C) + p(T|\sim C)*p(\sim C)}$$

$$A + C$$

$$0.08 + 0.0864 = 0.1664$$

Outcome: 0.48

Comments

Common mistake: to ignore the prior probability

The conditional probability slides the posterior probability in its direction but doesn't replace the prior probability

Worth reading in popular science: Nate Silver (2012), "The Signal and the Noise". Even better, Daniel Kahneman (2011), "Thinking Fast, Thinking Slow"

II. Bayesian Networks

A computational implementation
of Bayesian reasoning

First the basic idea,
then the clever implementation

Summary based on Tutorials and Presentations by

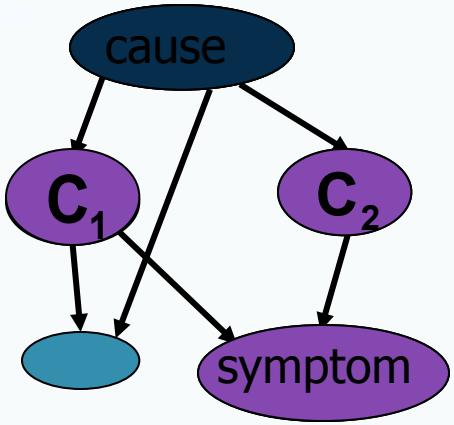
(1) Dennis M. Buede Joseph A. Tatman, Terry A. Bresnick;
(2) Jack Breese and Daphne Koller;
(3) Scott Davies and Andrew Moore;
(4) Thomas Richardson
(5) Roldano Cattoni
(6) Irina Rich

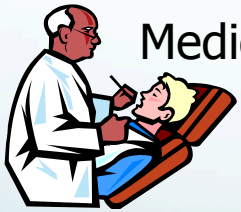
What are Bayesian nets?

- A graph-based framework for representing and analyzing uncertain information
- Uncertainty is handled in a mathematically rigorous yet efficient and simple way
- Different from other probabilistic analysis tools because of graphs, use of Bayesian statistics, and the synergy between these

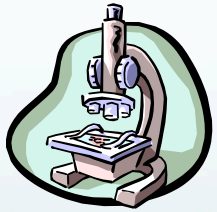
What Bayesian Networks are good for

- Diagnosis: P(cause|symptom)=?
- Prediction: P(symptom|cause)=?
- Classification: $\max_{class} P(class|data)$
- Decision-making (given a cost function)





Medicine



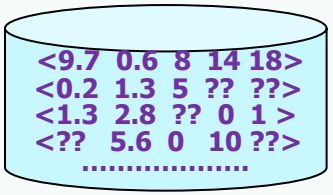
Bio-informatics

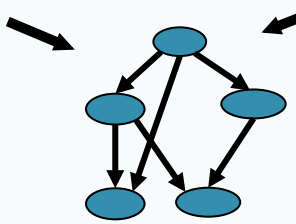
Stock market

Computer troubleshooting

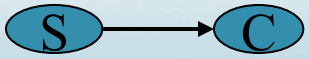
Why learn Bayesian networks?

- Combining domain expert knowledge with data





- Efficient representation and inference
- Incremental learning
- Handling missing data: **<1.3 2.8 ?? 0 1 >**
- Learning causal relationships:



Definition of a Bayesian Network

Knowledge structure:

- variables are nodes
- arcs represent probabilistic dependence between variables
- conditional probabilities encode the strength of the dependencies

Computational architecture:

- computes posterior probabilities given evidence about some nodes
- assumes probabilistic independence for efficient computation

These slides focus on **discrete variables**

We start with a **non-Boolean** example:

- A tiny network with just two nodes
- Two variables, each of which have 3 values

Bayesian Networks

P(S)

$P(S=no)$	0.80
$P(S=light)$	0.15
$P(S=heavy)$	0.05

$C \in \{none, benign, malignant\}$

P(C|S)

$Smoking=$	no	$light$	$heavy$
$P(C=none)$	0.96	0.88	0.60
$P(C=benign)$	0.03	0.08	0.25
$P(C=malig)$	0.01	0.04	0.15

Note: the table for **P(C|S)** gives information about ALL levels of C and S

If C and S were Boolean variables, then this would give us both **P(C|S)** and **P(C | not S)**.

How were P(S) and P(C|S) obtained?
-- Numbers from a medical database? (Automatic or by hand.)
-- An expert's opinion? Not directly based on numbers (i.e., not frequentist).

Given were **P(S)** and **P(C|S)**.
The two tables above allow us to compute **P(S|C)**, after computing **P(C)** first.

The main trick is *marginalisation*
We start by calculating **P(C,S)**.

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

$S\Downarrow$ $C\Rightarrow$	*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)

We now know:

P(C=none) = 0.935
P(C=benign) = 0.046
P(C=malig) = 0.019

We know P(C), P(C|S), and P(S).

From this, we compute **P(S|C)** using Bayes' Rule:

Bayes Rule Revisited

$$P(S \mid C) = \frac{P(C \mid S)P(S)}{P(C)} = \frac{P(C,S)}{P(C)}$$

$S\Downarrow \quad C\Rightarrow$	*none*	*benign*	*malig*
no	0.768/.935	0.024/.046	0.008/.019
light	0.132/.935	0.012/.046	0.006/.019
heavy	0.030/.935	0.015/.046	0.005/.019

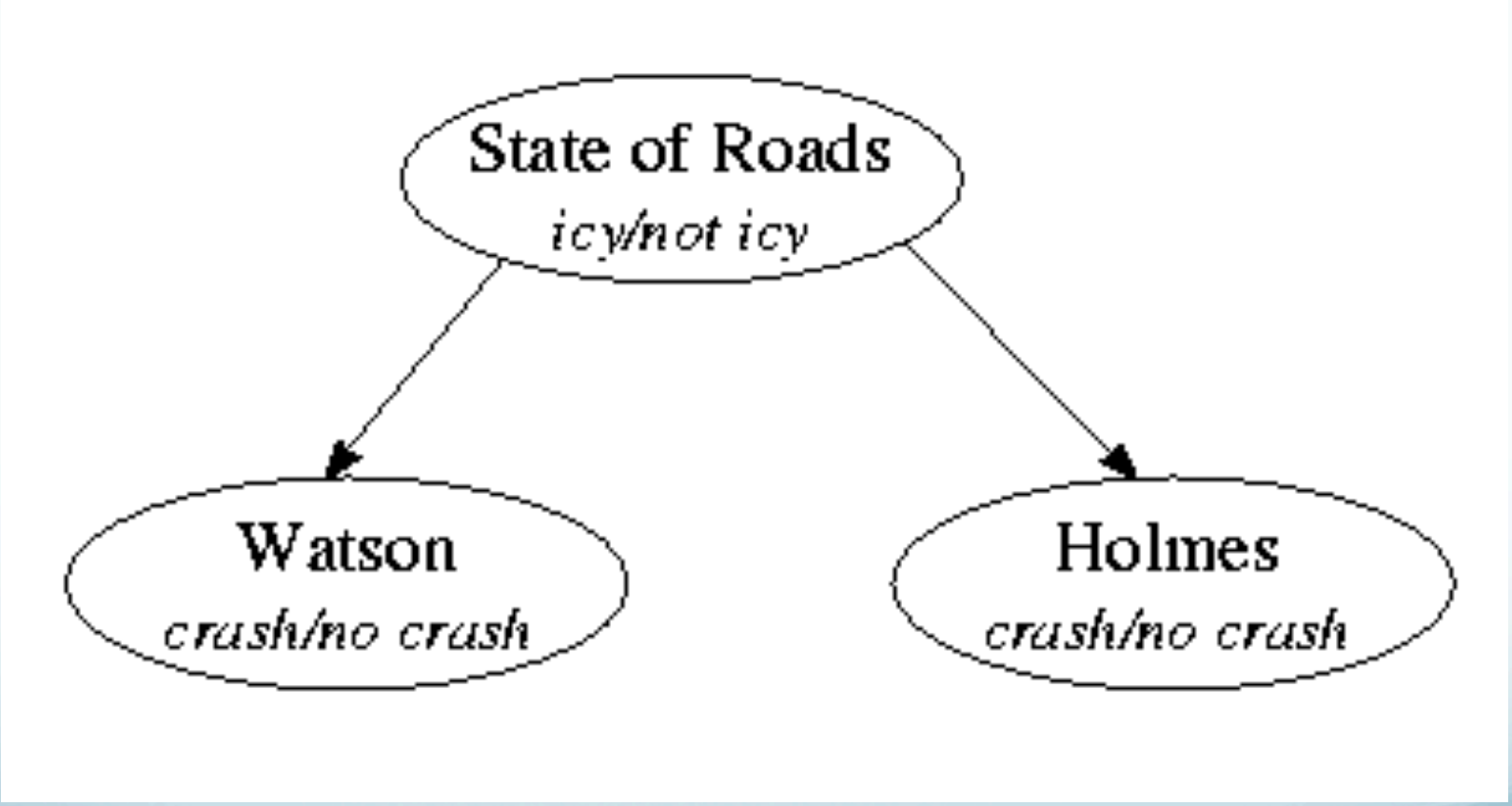
Cancer=	*none*	*benign*	*malignant*
P(S=no)	0.821	0.522	0.421
P(S=light)	0.141	0.261	0.316
P(S=heavy)	0.037	0.217	0.263

Let’s look at more complex examples, tracking how probabilities change by new information

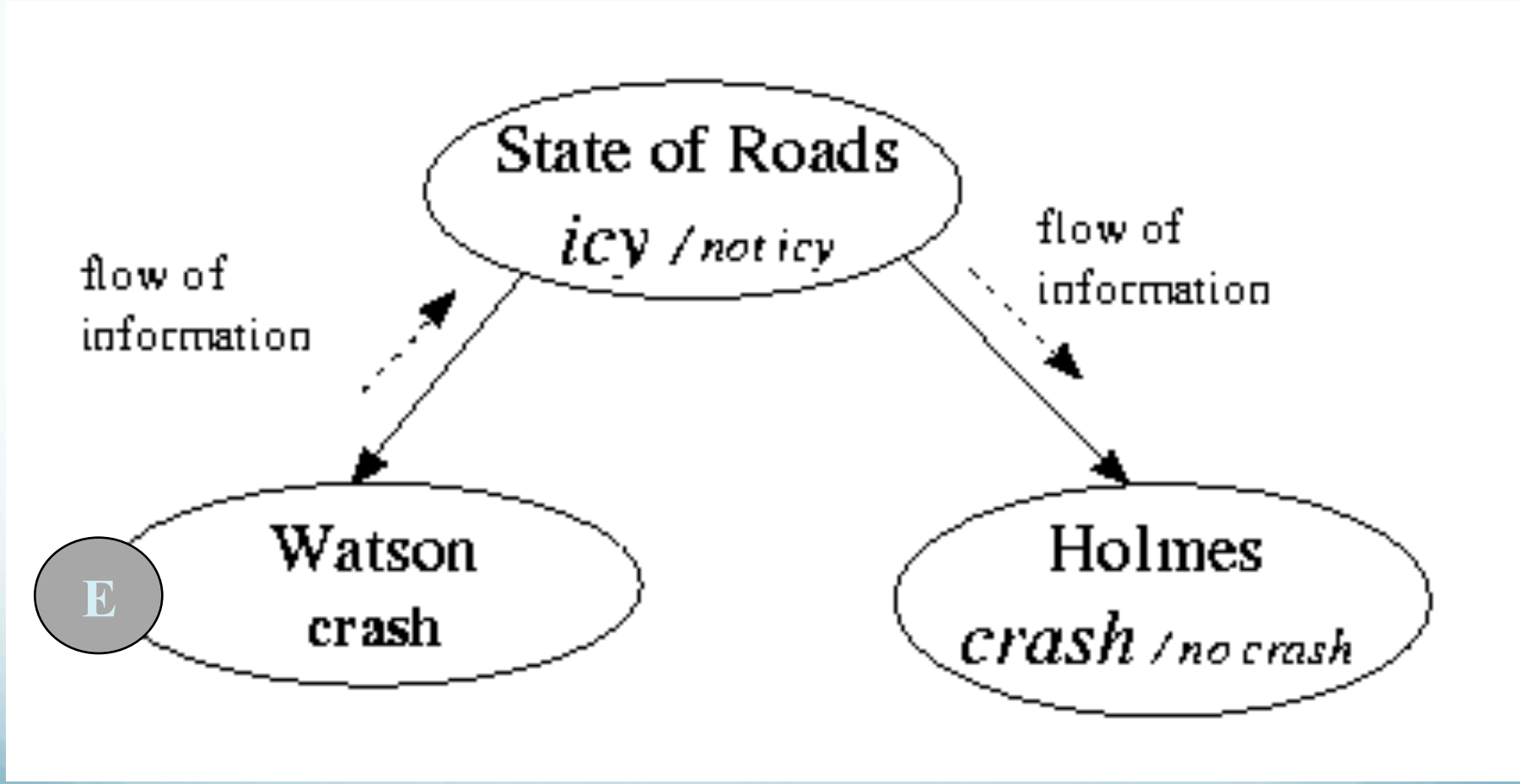
Example 1: "Icy roads"

- 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.*

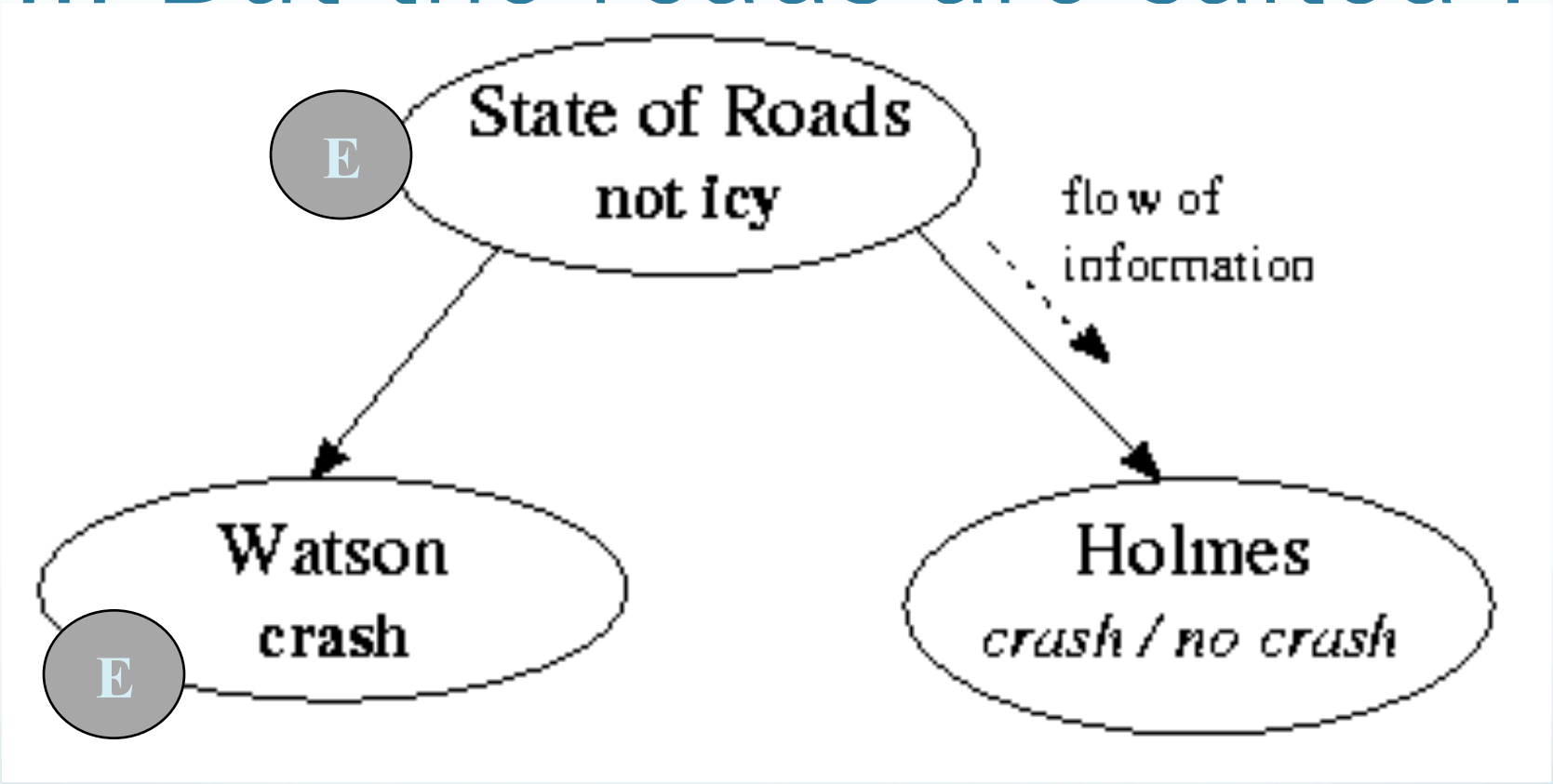
Causal relationships



Watson has crashed !



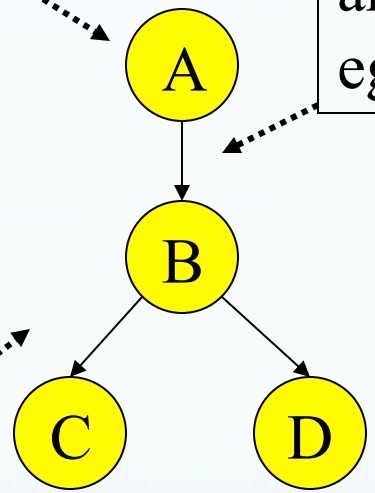
… But the roads are salted !



Interpretation of graphs

Each node in the graph is a random variable

A node *X* is called a *parent* of another node *Y* if there is an arrow from node *X* to node *Y* eg. *A* is a parent of *B*



Informally, an arrow from node *X* to node *Y* means *X* has a direct influence on *Y*

Why this type of graphs

Bayesian Networks use directed acyclic graphs

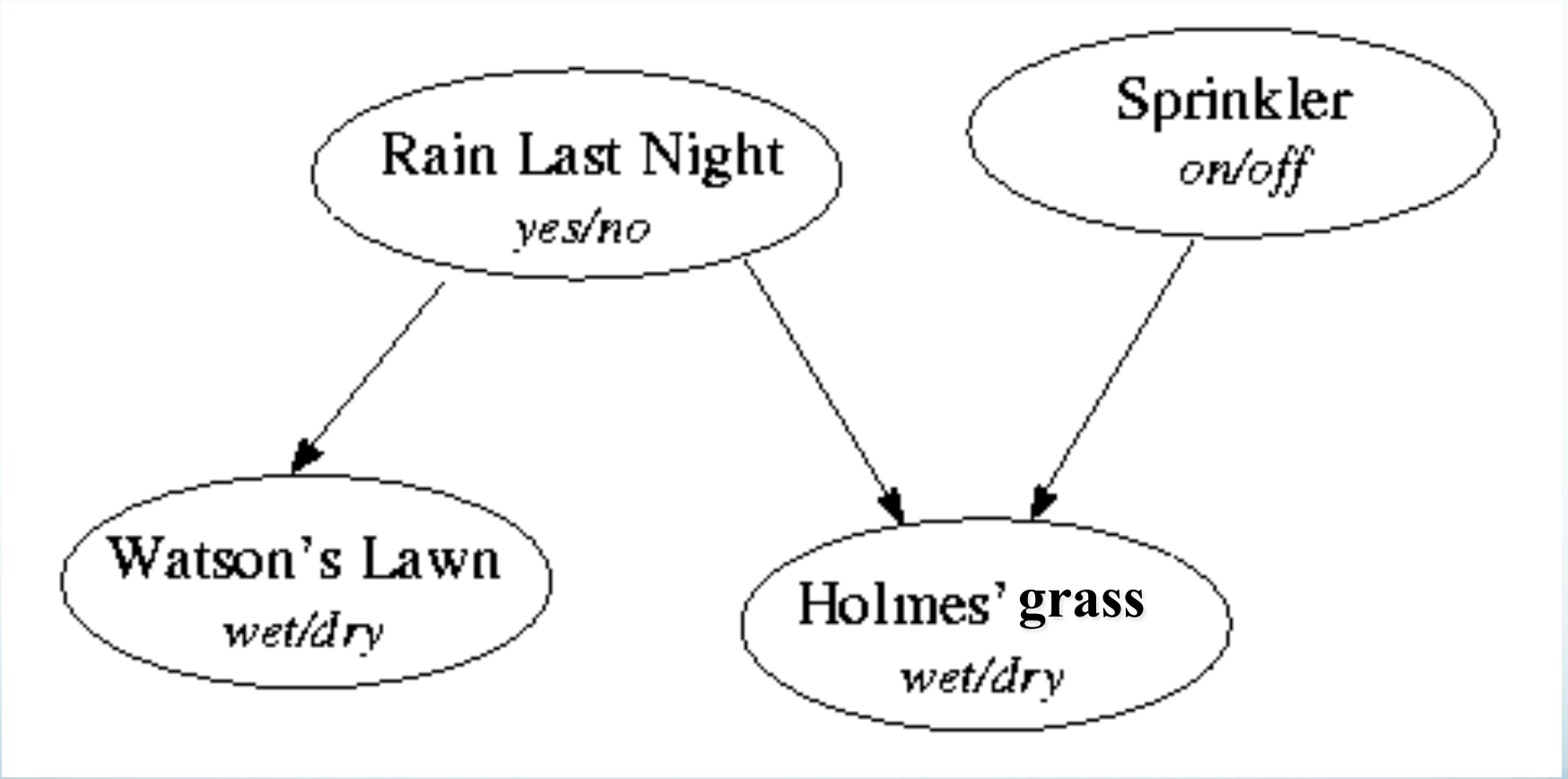
Directed because p(X|Y) and p(Y|X) may well be different. Represented using lines with arrows. (Dotted lines with arrows show indirect flow of information.)

Acyclic: no directed path from X to X.
I.e., no cycles (of any length).

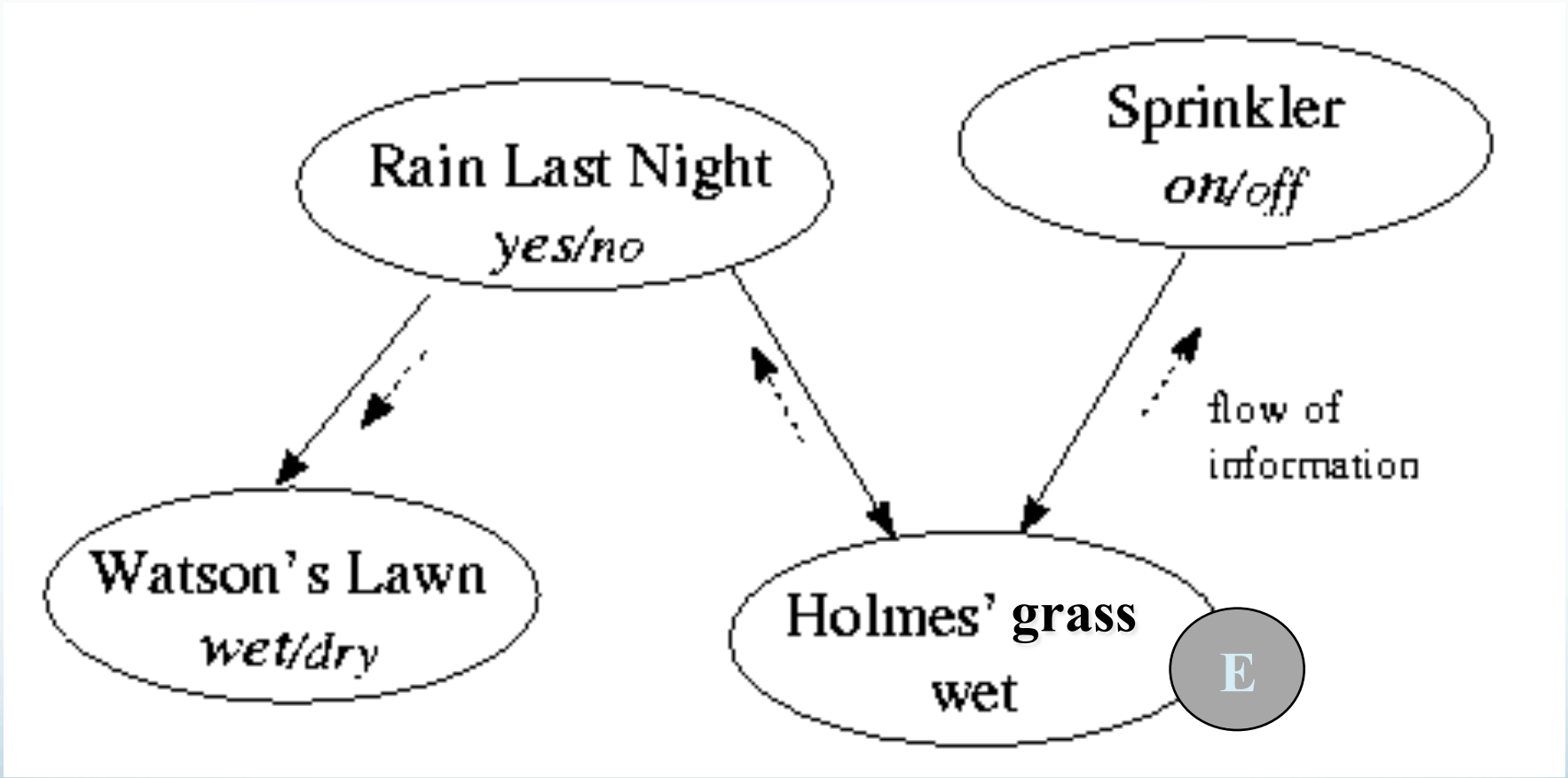
Example 2: "Wet grass"

- One morning as Holmes leaves for work, he notices that his grass is wet. He wonders whether he has left his sprinkler on, or it has rained.
- Glancing over to Watson's lawn he notices that it is also wet.
- Holmes thinks: *Since Watson's lawn is wet, it probably rained last night.*
- He then thinks: *If it rained then that explains why my **grass** is wet, so probably the sprinkler is off.*

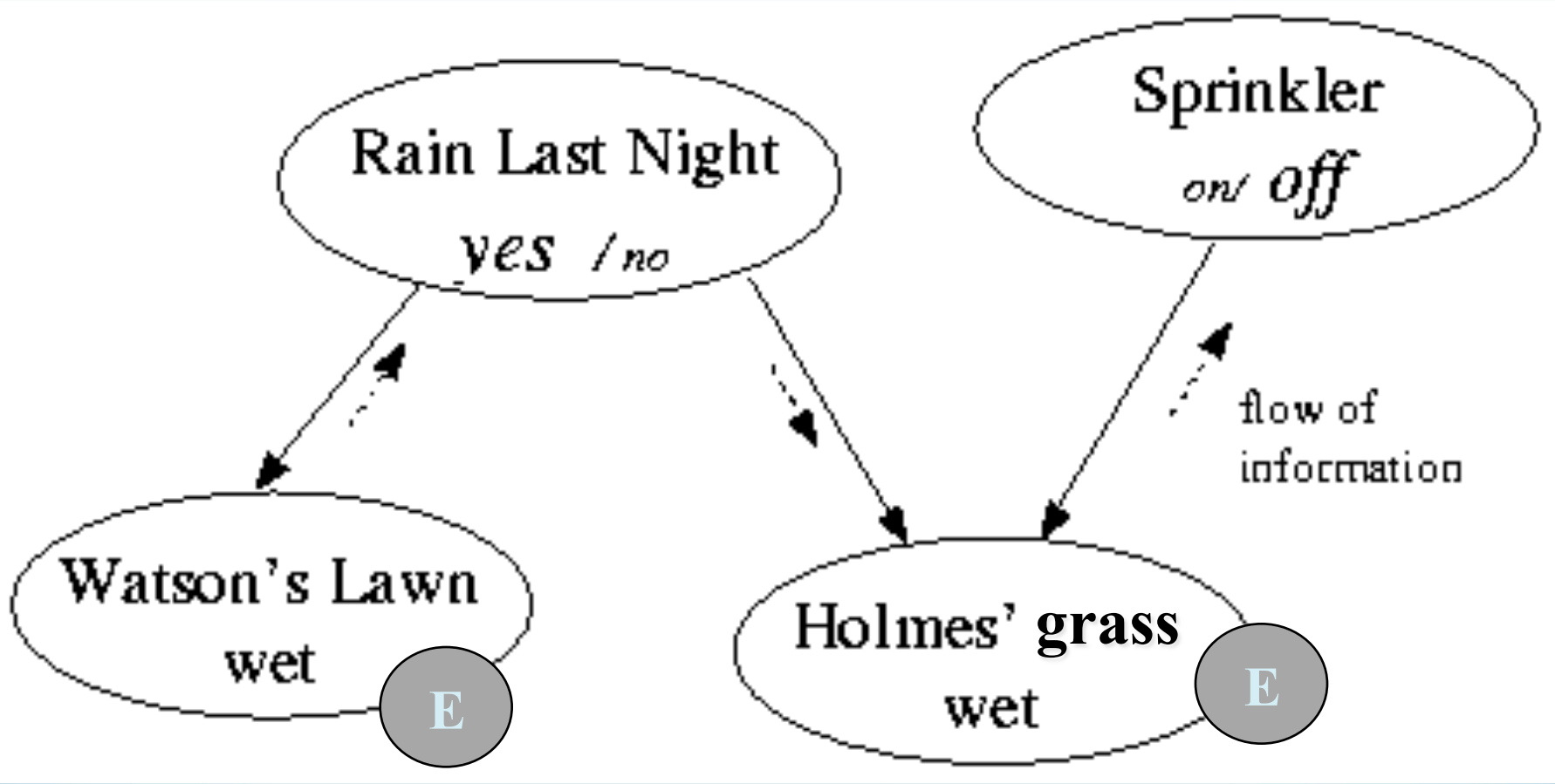
Causal relationships



Holmes' grass is wet !



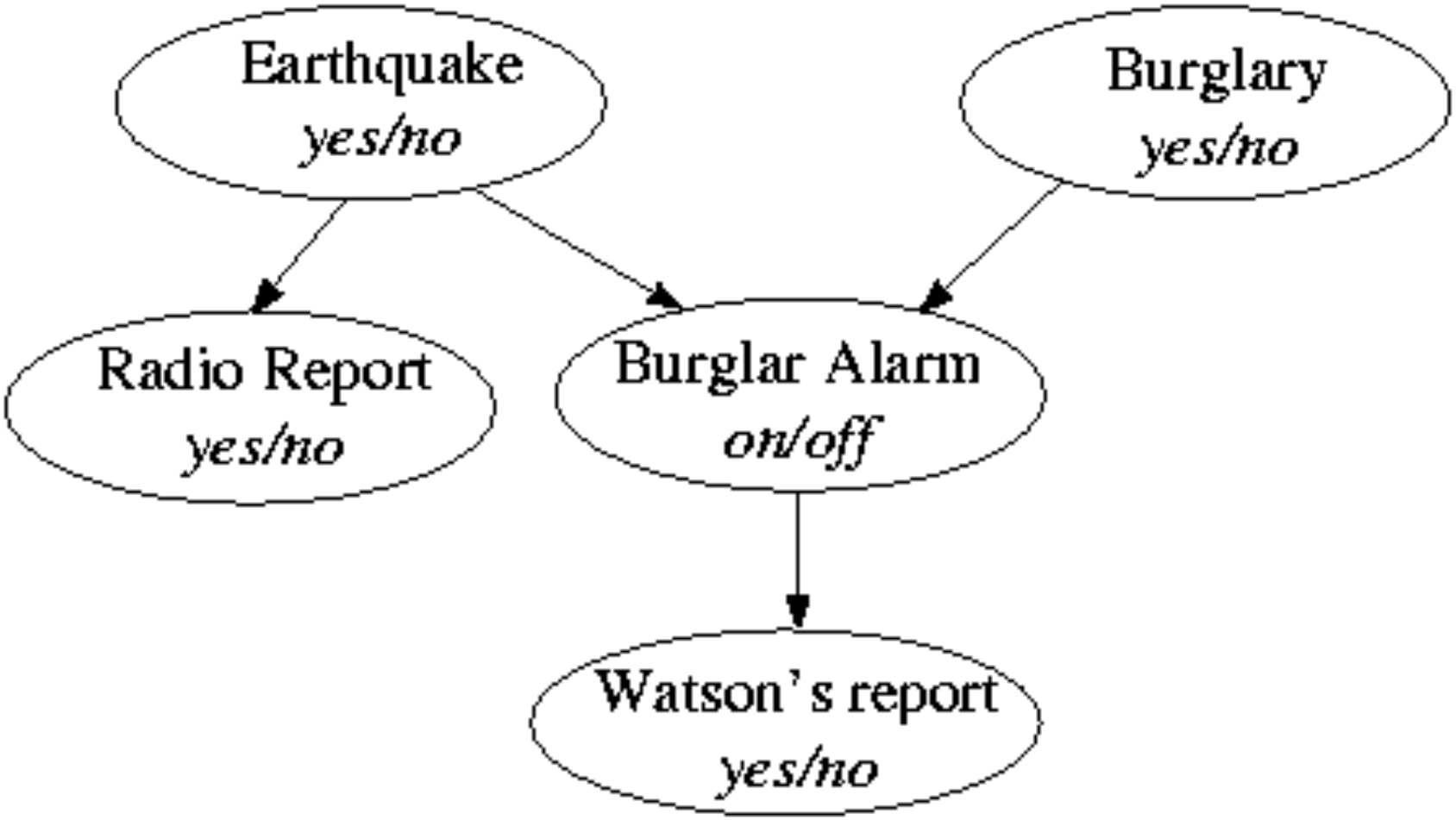
Watson's lawn is also wet !



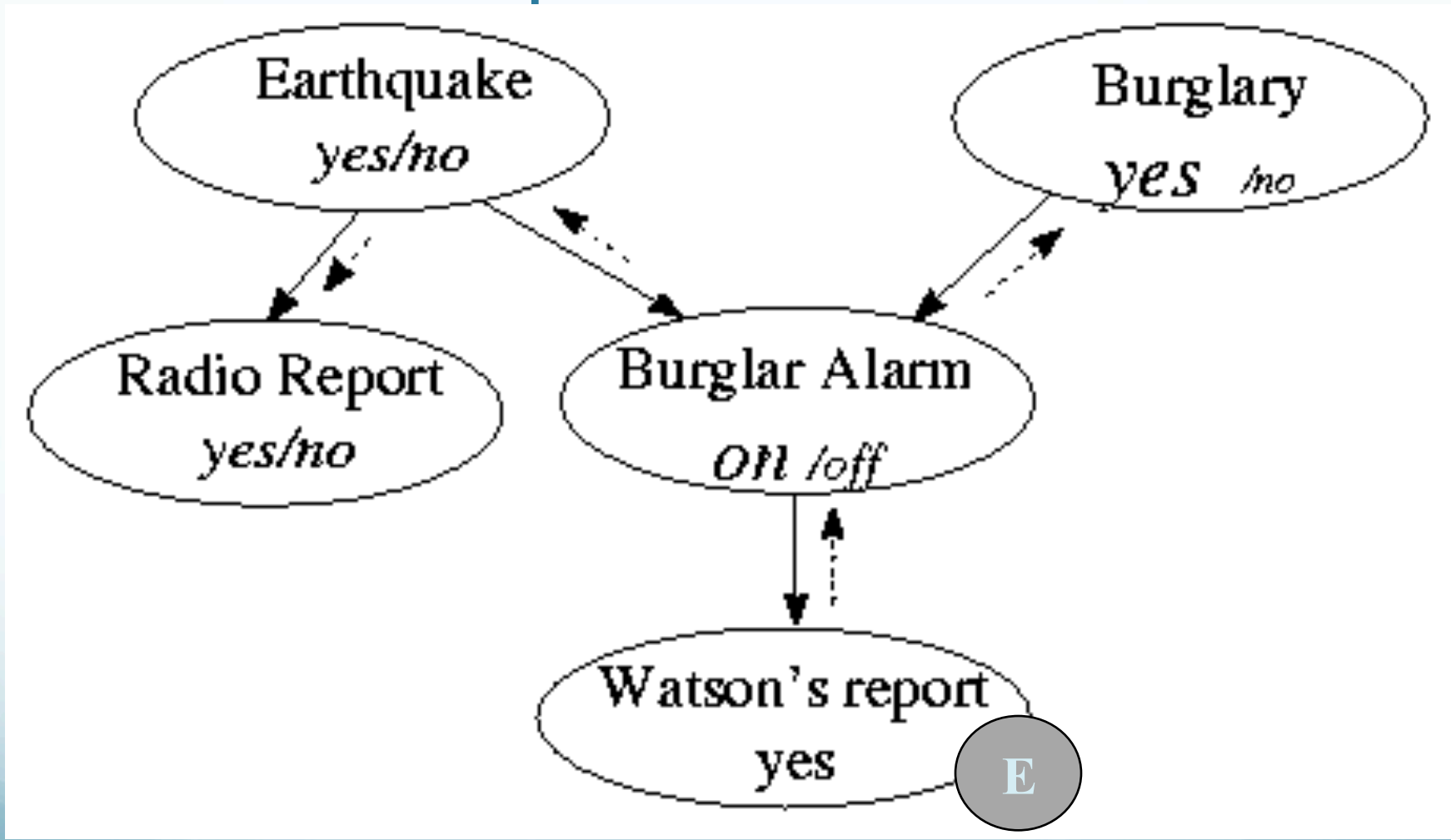
Example 3: "Burglar alarm"

- Holmes is at work when he receives a call from Watson, informing him that his alarm has gone off.
- Holmes thinks it is likely that the alarm really went off, although Watson sometimes play practical jokes.
- Holmes is on his way home when he hears a report on the radio, that there was an earthquake in the vicinity.
- Since the burglar alarm has been known to go off when there is an earthquake, Holmes reckons that a burglary is unlikely.
- Holmes goes back to work. (Leaving the noise for Watson)

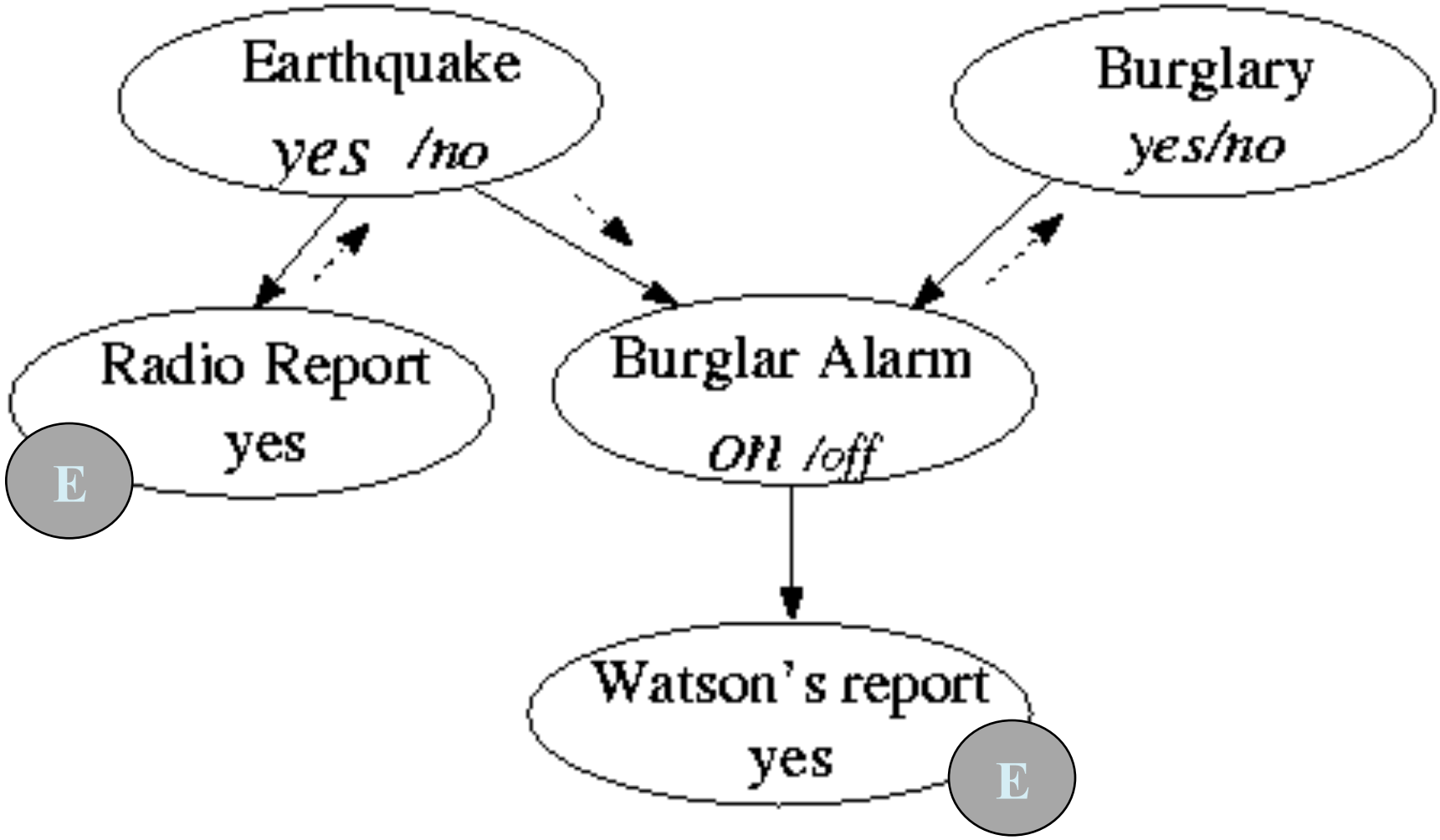
Causal relationships



Watson reports about alarm



Radio reports about earthquake



Inference Using Bayes Theorem

- Many problems have this shape: find the probability of an event given some evidence

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

- In 1986/88 Judea Pearl published an innovative algorithm for performing inference in Bayesian nets.

How to keep networks manageable?

A few assumptions allow us to compute the probability of each combination of nodes/ variables (at each value of the variable) easily:

-- No cycles (as we have seen)

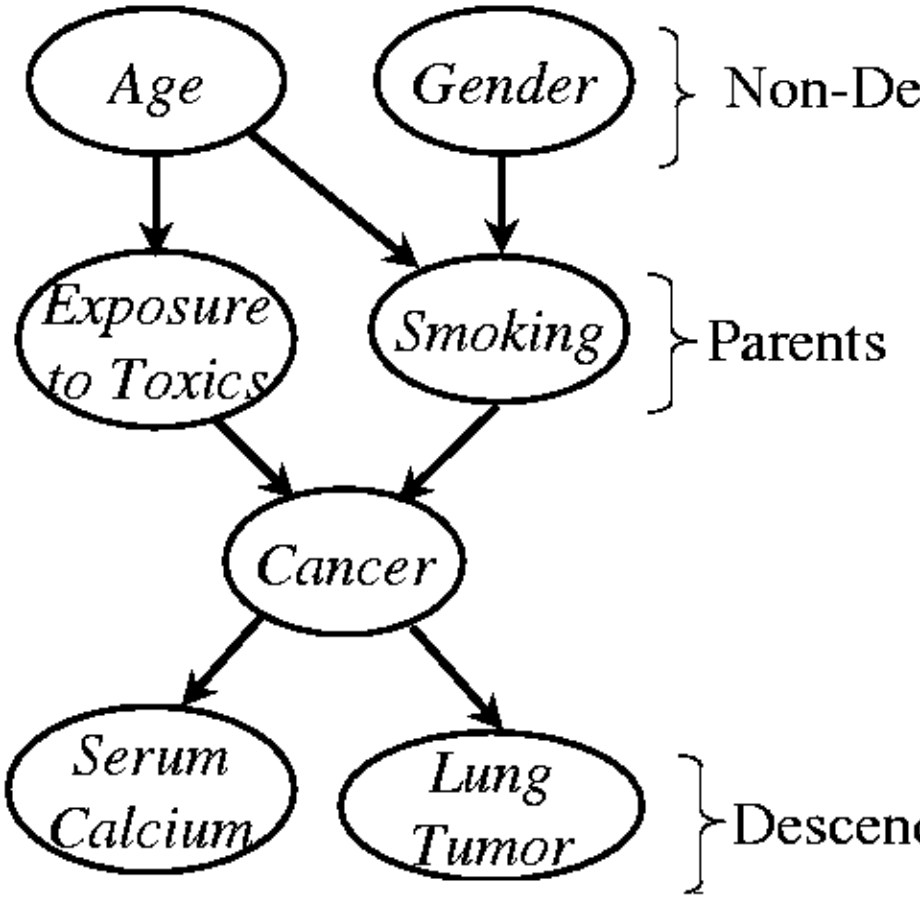
-- An independence assumption:

the Markov condition

NB These assumptions are not always realistic

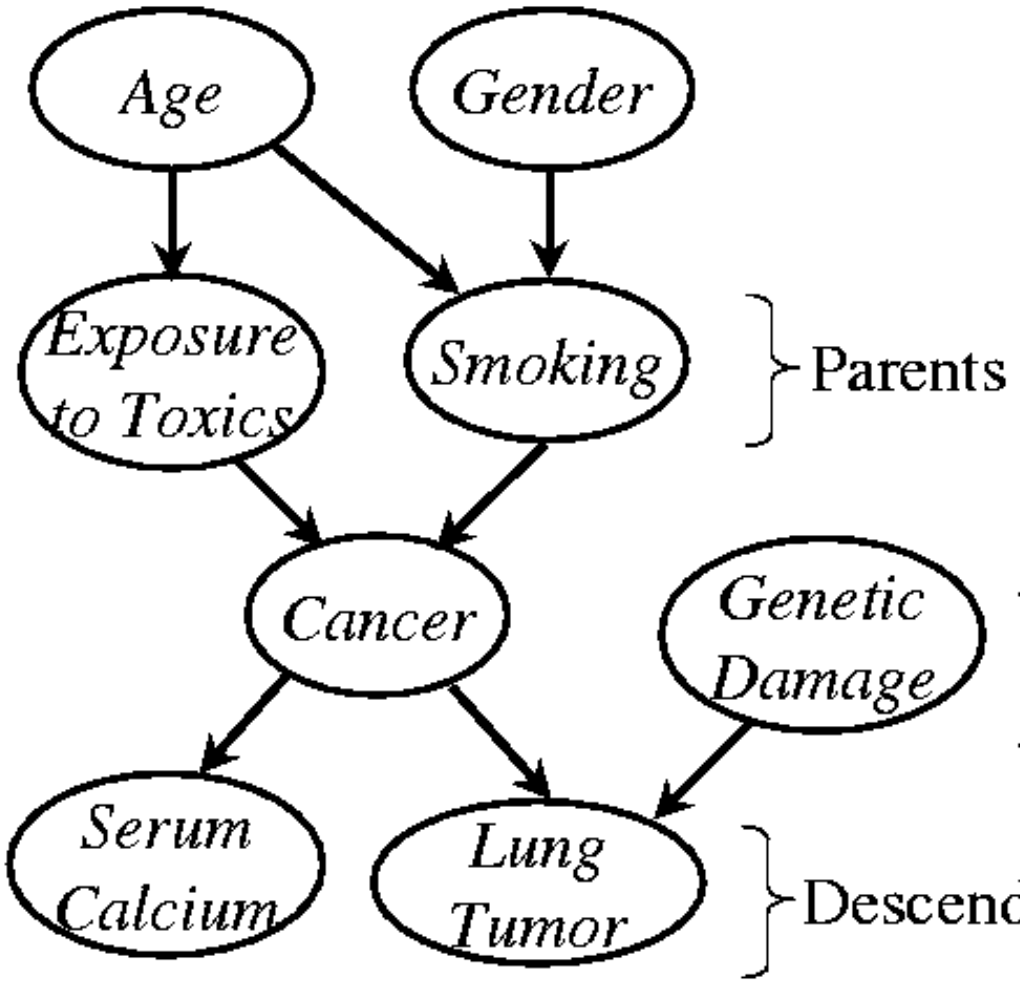
Conditional Independence

A variable (node) is conditionally independent of its non-descendants given its parents.



Cancer is independent of *Age* and *Gender* given *Exposure to Toxics* and *Smoking*.

Another non-descendant



Cancer is independent of *Genetic Damage* given *Exposure to Toxics* and *Smoking*.

The Markov condition

The Markov condition: given its parents, every node is conditionally independent of its non-descendants

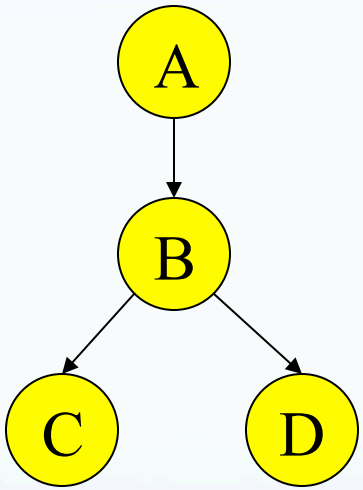
For networks that satisfy the Markov condition, the General Product Chain rule can be used for computing the probability of all the variables in the network:

General Product (Chain) Rule for Bayesian Networks

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid \boldsymbol{Pa}_i)$$

$$\boldsymbol{Pa}_i = parents(X_i)$$

Let's populate an earlier network example with probabilities



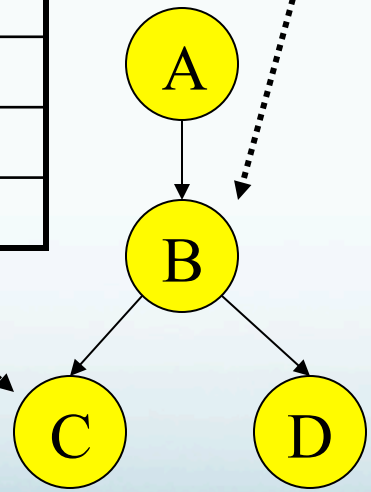
We take these 4 variables to be Boolean

Read: "Given A=false then [P(B=false) is 0.01"] *(Etc.)*

A	P(A)
false	0.6
true	0.4

A	B	P(B|A)
false	false	0.01
false	true	0.99
true	false	0.7
true	true	0.3

B	C	P(C|B)
false	false	0.4
false	true	0.6
true	false	0.9
true	true	0.1



B	D	P(D|B)
false	false	0.02
false	true	0.98
true	false	0.05
true	true	0.95

As before: Each node X_i has a conditional probability distribution $P(X_i \mid \text{Parents}(X_i))$ that quantifies the effect of the parents on the node

Using a Bayesian Network Example

Using the network, suppose you want to calculate:

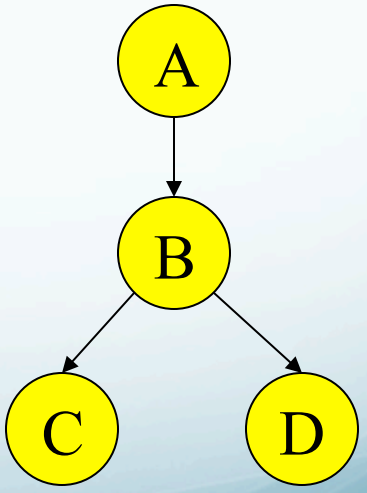
P(A = **true**, B = **true**, C = **true**, D = **true**)

= P(A = true | Parents(A)) *

P(B = true | Parents(B)) *

P(C = true | Parents(C)) *

P(D = true | Parents(D))



Using a Bayesian Network Example

Using the network, suppose you want to calculate:

P(A = true, B = true, C = true, D = true)

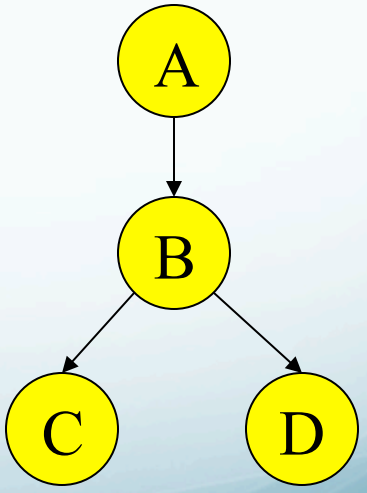
= P(A = true) *

P(B = true | A = true) *

P(C = true | B = true) *

P(D = true | B = true)

= (0.4)*(0.3)*(0.1)*(0.95)



Using a Bayesian Network Example

Using the network in the example, suppose you want to calculate:

P(A = true, B = true, C = true, D = true)

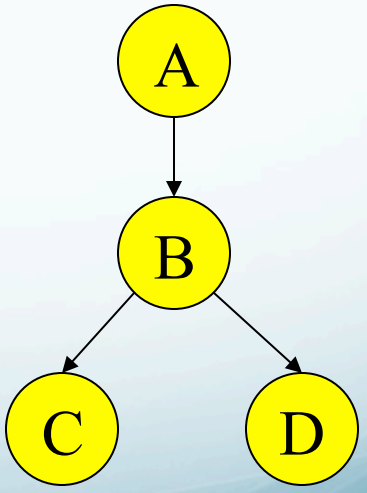
= P(A = true) * P(B = true | A = true) *

P(C = true | B = true) P(D = true | B = true)

= (0.4)*(0.3)*(0.1)*(0.95)

This is from the graph structure

These numbers are from the conditional probability tables



Using Bayesian Networks

Let's look at the first example involving Holmes and Watson ("Icy Roads")

"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.*

Did Holmes Crash?

Let's make some numerical assumptions and put these into a Bayesian network.

We variously write (with apol's for confusion)

-- Holmes Crash, H Crash, H

-- Watson Crash, W Crash, W

-- Icy, I

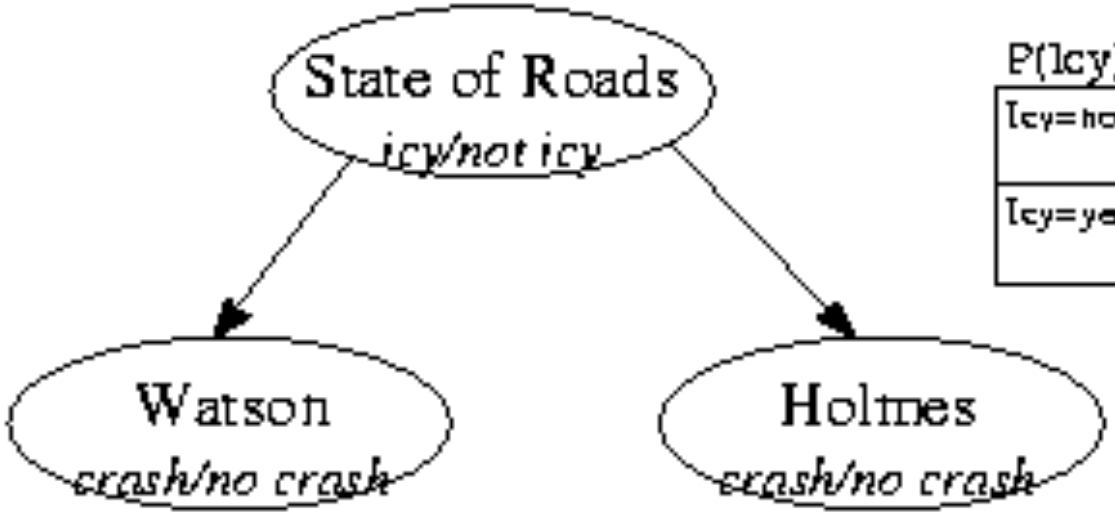
-- yes, y

Note the difference between e.g.

-- H

-- H=yes

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 required probabilities.

By symmetry P (W Crash) is the same.

Current state of play

We had been given:

P(Icy)

P(Holmes|Icy)

P(Watson|Icy)

We have computed: P(Holmes) (and P(Watson))

Now we learn that Watson has crashed!

We want to compute P(Icy) in light of this new info:

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:

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

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

$$= (0.95, 0.05)$$

Current state of play

We had been given:

P(Icy) (0.8, initial info!)

P(Holmes|Icy)

P(Watson|Icy)

We have computed: P(Holmes) (and P(Watson))

We have computed: P(Icy|Watson=yes) and P(Icy) (0.95, updated info given that Watson crashed)

Computing P(Holmes)

Having learned that Watson crashed,
we want to know P(Holmes)

To compute P(Holmes|Watson=yes), we need to go via Icy (the only connection between Holmes and Watson in the network):

-- Calculate P(Icy=yes,Holmes | Watson=yes)
-- Calculate P(Icy=no,Holmes | Watson=yes)
The sum of the two is P(Holmes | Watson=yes)

Calculating P(H|W=y)

First calculate P(I,H | W=y)

P(I,H|W=y)	I=y	I=n
H=y	0.95*0.8	0.05*0.1
H=n	0.95*0.2	0.05*0.9

For example, P(I=y,H=y | W=y) =
P(I=y | W=y) * P(H=y | I=y) = 0.95*0.8=0.76 :

Calculating P(H|W=y)

Doing the sums:

P(I,H|W=y)	I=y	I=n	
H=y	0.76	0.005	tot = 0.765
H=n	0.19	0.045	tot = 0.235

It follows that P(H | W=y) = (0.765 , 0.235)

End of calculation:

We had been given:

P(Icy) (0.8, initial info)

P(Holmes|Icy)

P(Watson|Icy)

P(Holmes) (0.59, initial info)

P(Watson) (0.59, initial info)

We learned that Watson=yes and concluded

P(Icy) (0.95, updated info)

P(Holmes) (0.765, updated info)

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).

Distribute values for the row W=no over the row W=yes (divide by P(W=yes))

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 $\mathbf{P^{New}(W\ Crash) = (1,0)}$

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

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

Update the table for I, then the table for H

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:

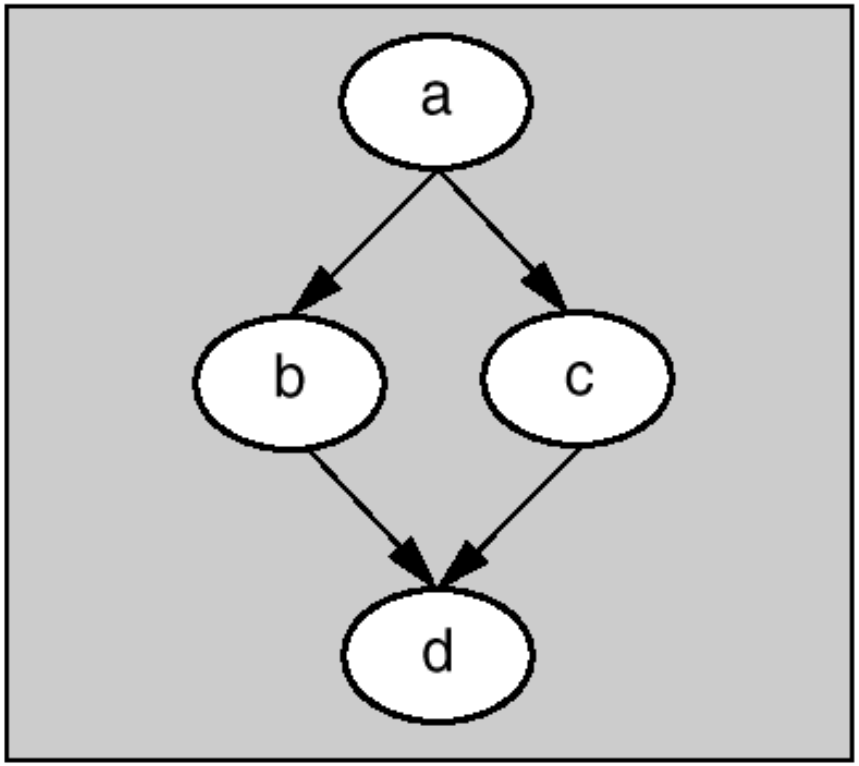
$$P^{New}(H\ Crash, Icy) = \frac{P(H\ Crash, Icy)}{P(Icy)} P^{New}(Icy)$$

Sophisticated techniques exist for converting complex networks into simpler trees that express (approximately) the same information.

Example: trees that contain undirected loops (these are called multiply connected trees) are converted into ones that do not. The resulting trees are called singly connected.

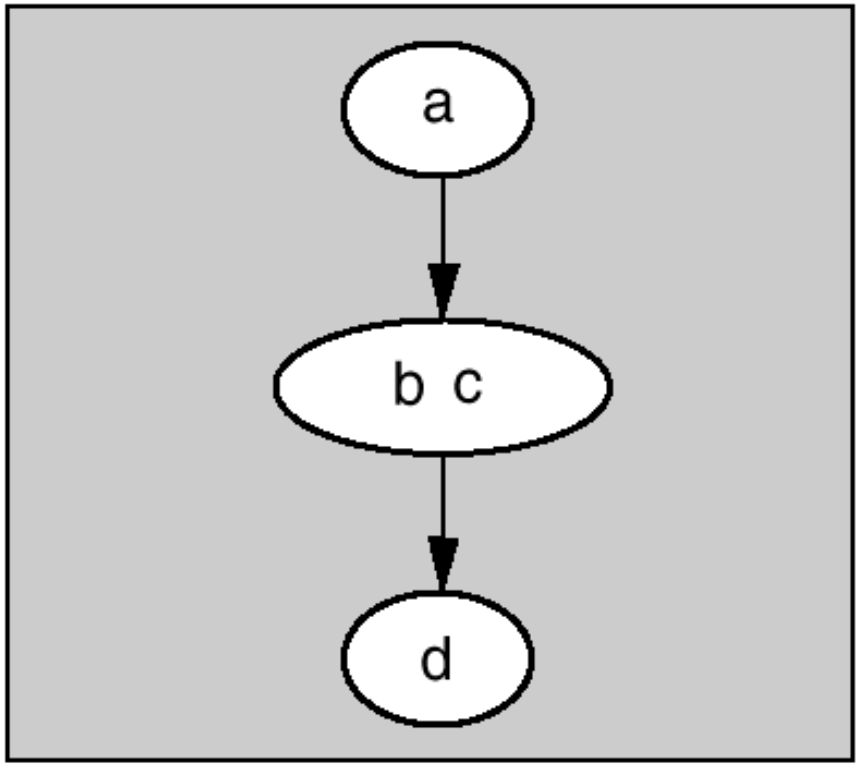
Theorem: inference with singly connected trees can be done in linear time.

Converting a tree into a polytree (example)



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.*

A conceivable example

Example of conversion:
Physical exercise (a) tends to
(b) strengthen legs and
(c) strengthen arms

Strong arms make a good swimmer (d), and
Strong legs make a good swimmer (d).

It would be natural to combine (b) and (c) into "strong arms and legs"

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

For example: "If Holmes had an accident
then Smith should ask for reinforcement"

Modeled in Decision Theory, where each action (given a situation) is associated with a number that reflects its utility, e.g.,

utility(CallReinf=yes & Holmes=yes) = 10
utility(CallReinf=yes & Holmes=no) = -1
utility(CallReinf=no & Holmes=yes) = -10
utility(CallReinf=no & Holmes=no) = 1

utility(CallReinf=yes & Holmes=yes) = 10
utility(CallReinf=yes & Holmes=no) = -1
utility(CallReinf=no & Holmes=yes) = -10
utility(CallReinf=no & Holmes=no) = 1

Combined with network-derived probabilities, we can calculate expected utility of CallReinf:
CallReinf=Yes: 0.765*10 + 0.235*-1 = 7.415
CallReinf=No: 0.765*-10 + 0.235*1 = -7.415

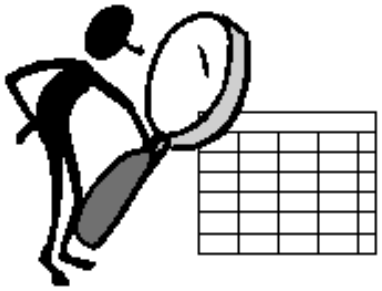
This suggests you'd better call reinforcements …

… but Decision Theory is not the topic of this course

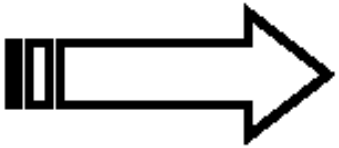
Bayesian Networks and learning

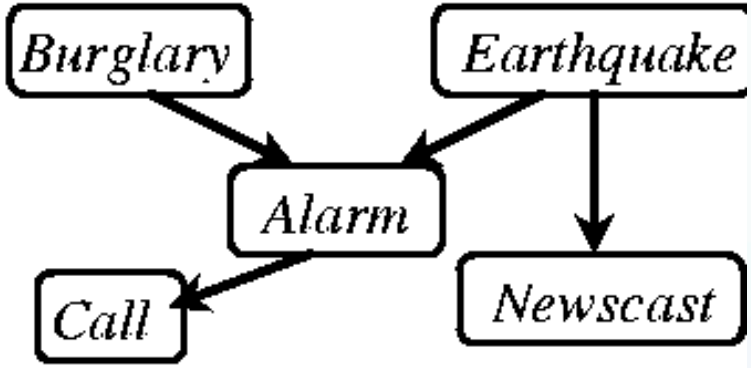
1. How might one learn a Bayesian Network?

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?

Bayesian Networks and learning

2. How can a Bayesian Network support learning?
   Example: learning to classify an item.

   e.g.: Is a mushroom likely to be edible, given attributes like colour, height, stem type, etc?

Simple Bayesian prediction

- P (H|E) = probability that some hypothesis, H, is true, given evidence E.
- A set of hypotheses $H_1 \ldots H_n$.
- For each H_i $$P(H_i \mid E) = \frac{P(E \mid H_i) \cdot P(H_i)}{P(E)}$$
- Given E, a learner finds the most likely explanation by finding the hypothesis that has the highest posterior probability.

Simple Bayesian prediction

- This can be simplified.
- Since P(E) is independent of H_i it will have the same value for each hypothesis.
- Hence, P(E) can be ignored, and we can find the hypothesis with the highest value of:

$$P(E \mid H_i) \cdot P(H_i)$$

- (If all the hypotheses are equally likely, we can simplify this further by simply seeking the hypothesis H_i with the highest $P(E|H_i)$.)

A conceivable example

Holmes's Alarm went off

H1: Earthquake
H2: Burglers

Suppose

$P(Alarm|H1) < P(Alarm|H2)$
$P(H1) \sim P(H2)$

Then H2 is more plausible than H1

Bayes' Optimal Classifier

- We have a piece of n data about y
- We seek the best hypothesis from H_1 ... H_m, each of which assigns a classification to y.
- The probability that y should be classified as c_j is:

$$P(c_j \mid x_1 \ldots x_n) = \sum_{i=1}^{m} P(c_j \mid h_i) \cdot P(h_i \mid x_1 \ldots x_n)$$

- In practice this is often too difficult to calculate

The Naïve Bayes Classifier (1)

- A vector of *n* data is classified: $P(c_i | d_1, \ldots, d_n)$
- The classification c_i with the highest posterior probability is chosen.
- Bayes' theorem is used to find the posterior probability:

$$\frac{P(d_1,\ldots,d_n \mid c_i)\cdot P(c_i)}{P(d_1,\ldots,d_n)}$$

- Now we reason as before:

- Since $P(d_1, \ldots, d_n)$ is constant, we can eliminate it, and simply aim to find the classification c_i, for which the following is maximised:

$$P(d_1,\ldots,d_n \mid c_i) \cdot P(c_i)$$

- We now assume that all the attributes $d_1, \ldots, d_n$ are independent
- So $P(d_1, \ldots, d_n|c_i)$ can be rewritten, as in:

$$P(c_i) \cdot \prod_{j=1}^{n} P(d_j \mid c_i)$$

- The classification for which this is highest is chosen to classify the data.

Classifier Example

Training Data

x	y	z	Classification
2	3	2	A
4	1	4	B
1	3	2	A
2	4	3	A
4	2	4	B
2	1	3	C
1	2	4	A
2	3	3	B
2	2	4	A
3	3	3	C
3	2	1	A
1	2	1	B
2	1	4	A
4	3	4	C
2	2	4	A

- New piece of data to classify
  - (x = 2, y = 3, z =4)
- Want $P(c_i | x=2,y=3,z=4)$
- P(A) * P(x=2|A) * P(y=3|A) * P(z=4|A)

$$\frac{8}{15}\cdot\frac{5}{8}\cdot\frac{2}{8}\cdot\frac{4}{8} = 0.0417$$

- P(B) * P(x=2|B) * P(y=3|B) * P(z=4|B)
- etc.

- Called Naïve Bayes because of the independence assumption
- More sophisticated versions of the classifier exist, which use dependencies between attributes
- But Naïve Bayes often works surprisingly well.

Ask yourself:

- What does Bayes' Theorem contribute to the Naïve Bayes Classifier
- Could you have used the formula $P(c_i | d_1, \ldots, d_n)$directly?

$P(c_i | d_1, \ldots, d_n)$

1. You could not have used this formula directly to classify an unseen mushroom, because its combination of features $d_1, \ldots, d_n$ may not have been seen before.

2. You could have tried to simplify:
   $P(c_i | d_1, \ldots, d_n)$ =?=
   $P(c_i | d_1) + \ldots + P(c_i | d_n)$
   but this would have been wrong because the d_i overlap!

Key Events in Bayesian Nets & related

- 1763 Bayes Theorem presented by Rev Thomas Bayes (posthumously) in the Philosophical Transactions of the Royal Society of London
- 1950s Naïve Bayes Classifier invented & starting to be explored
- 1976 Influence diagrams in SRI technical report for DARPA as technique for improving efficiency of analyzing large decision trees
- 1980s Several software packages are developed in the academic environment for the solution of influence diagrams
- 1986 "Fusion, Propagation, and Structuring in Belief Networks" by Judea Pearl appears in the journal *Artificial Intelligence*
- *After that*: faster algorithms, commercial implementations, etc.

Software

Examples

- **Netica**
  - **www.norsys.com**
  - **Very easy to use**
  - **Implements learning of probabilities**
- **Hugin**
  - **www.hugin.dk**
  - **Good user interface**
  - **Implements continuous variables**

A small detour (optional material)

D-Separation of variables

- **There is a relatively simple algorithm for determining whether two variables in a Bayesian network are conditionally independent: *d-separation*.**

- **Definition: *X* and *Z* are *d-separated* by a set of evidence variables *E* iff every undirected path from *X* to *Z* is "blocked".**

- **A path is "blocked" iff one or more of the following conditions is true: ...**

A path is blocked when:

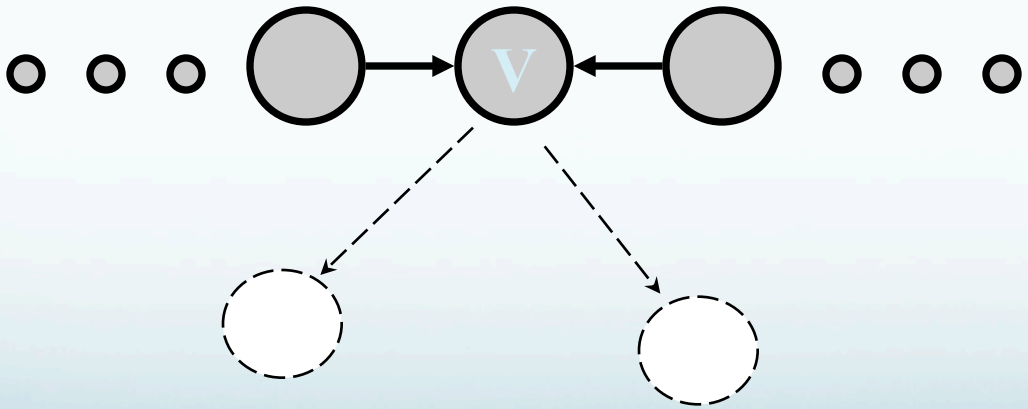
- There exists a variable V on the path such that
  - it is in the evidence set E
  - the arcs putting V in the path are "tail-to-tail"

- Or, there exists a variable V on the path such that
  - it is in the evidence set E
  - the arcs putting V in the path are "tail-to-head"

- Or, ...

… a path is blocked when:

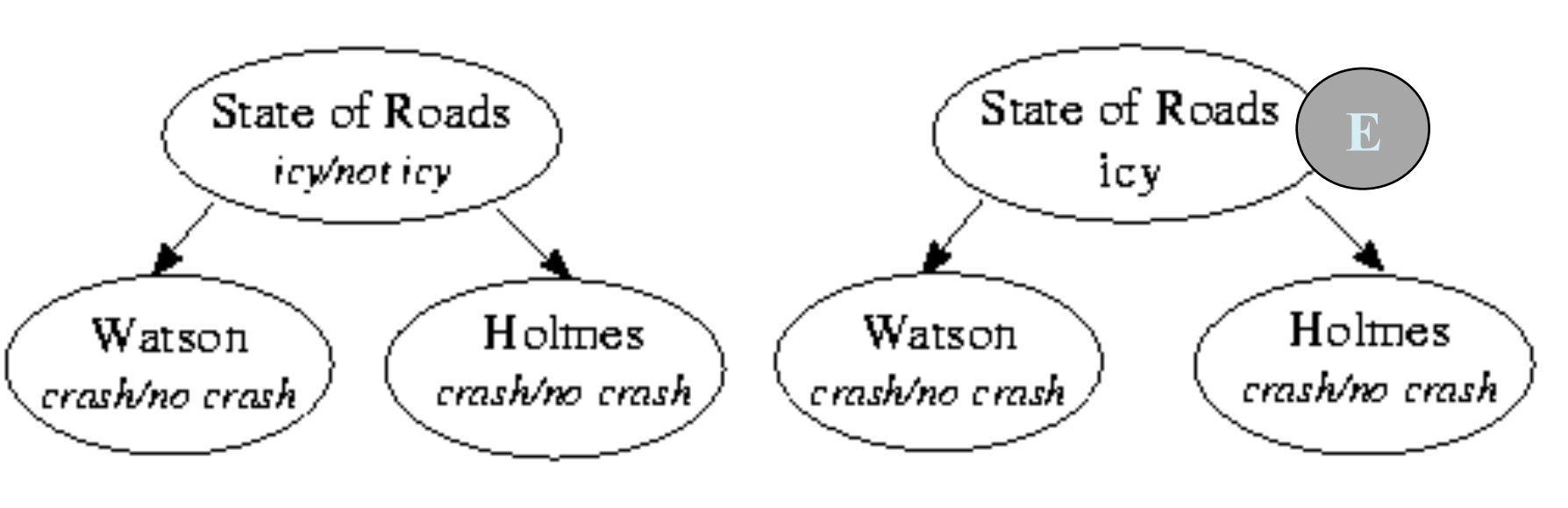
- **… Or, there exists a variable *V* on the path such that**
  - **it** is NOT **in the evidence set *E***
  - neither are any of its descendants
  - **the arcs putting *V* on the path are "head-to-head"**



D-Separation and independence

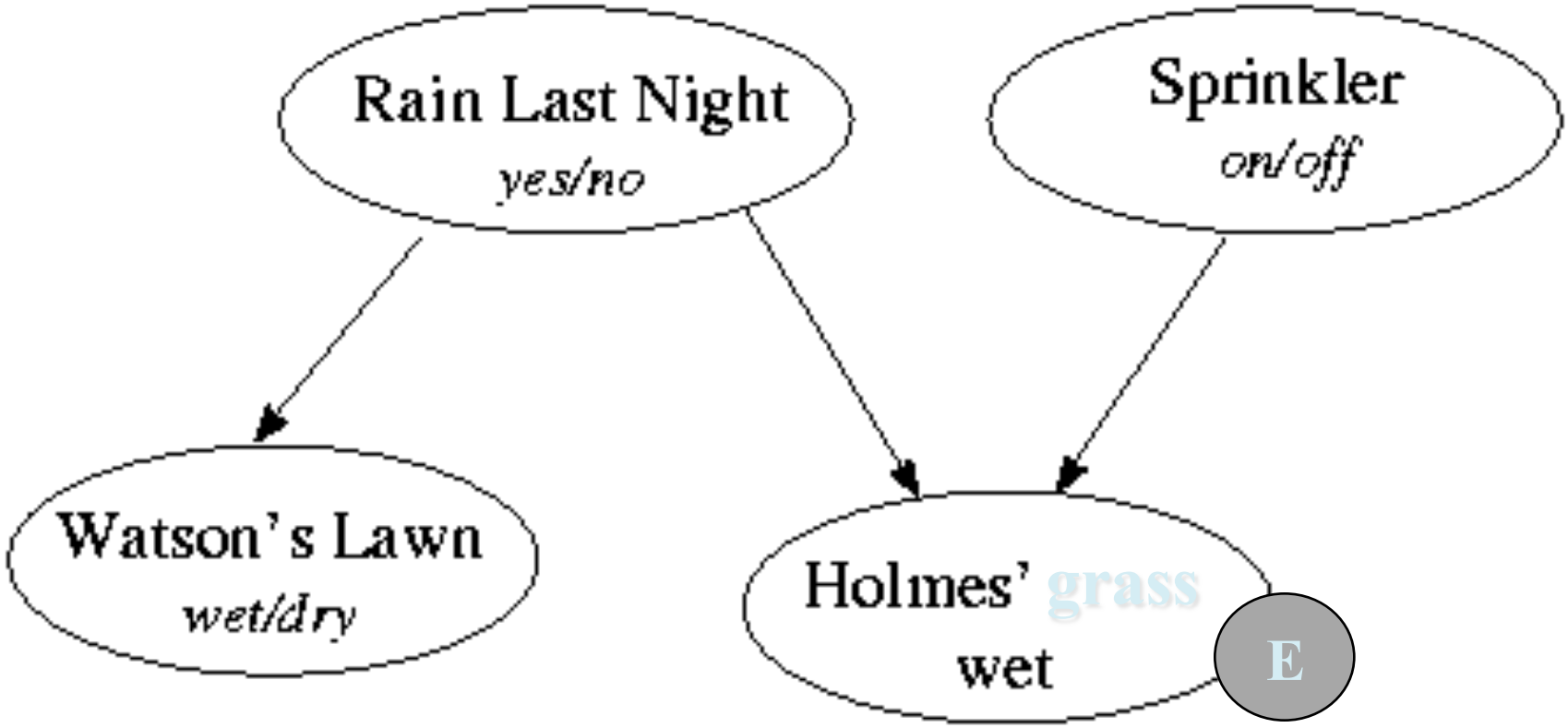
- **Theorem [Verma & Pearl, 1998]:**
  - **If a set of evidence variables *E* d-separates *X* and *Z* in a Bayesian network's graph, then *X* and *Z* will be independent.**
- ***d*-separation can be computed in linear time.**
- **Thus we now have a fast algorithm for automatically inferring whether learning the value of one variable might give us any additional hints about some other variable, given what we already know.**

“Icy roads” example



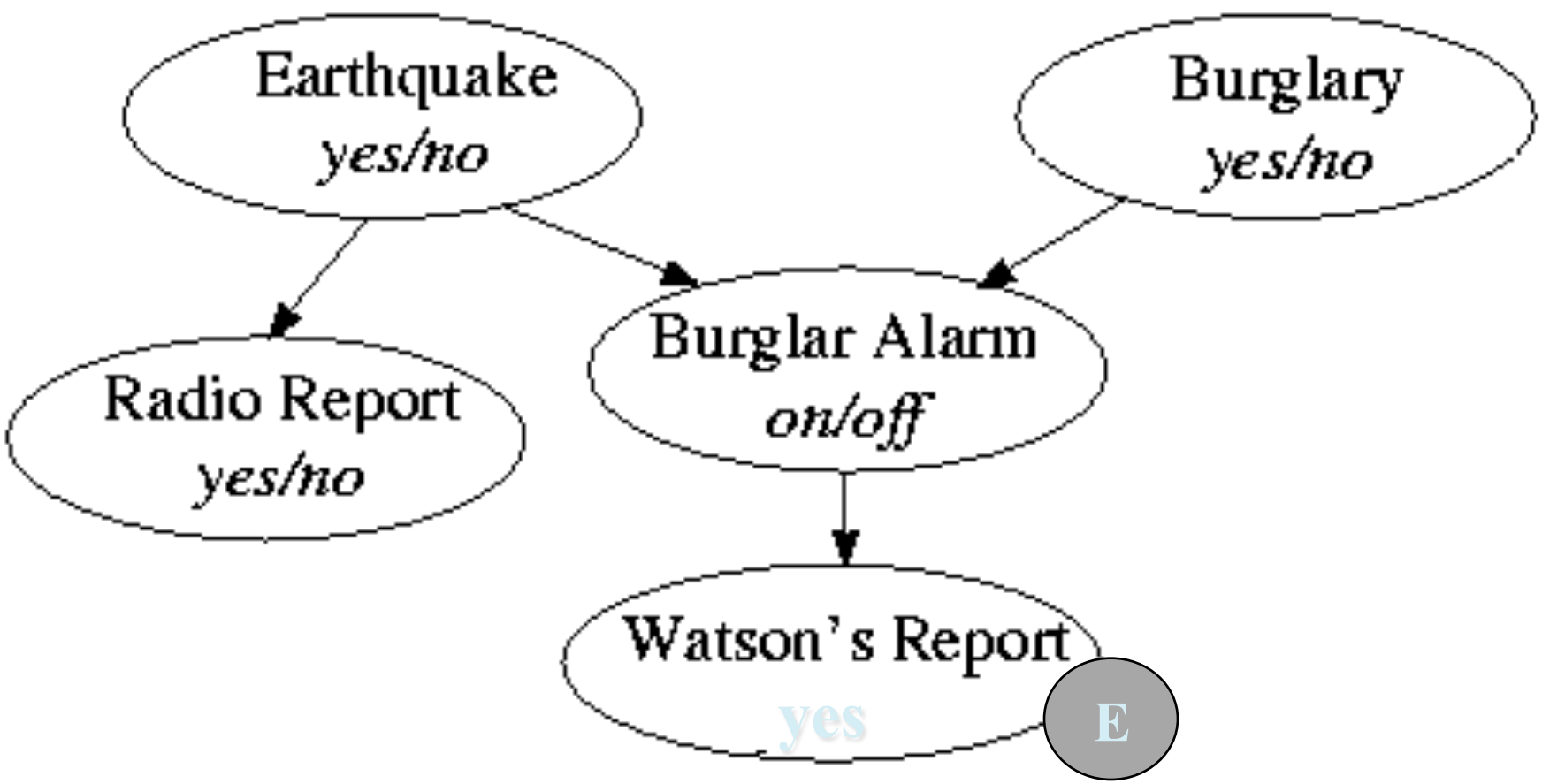
Watson and Holmes are d-connected given no evidence, but
Watson and Holmes are d-separated given State of Roads

“Wet grass” example



Watson’s Lawn and Sprinkler are d-connected given Holmes’ grass

“Burglar alarm” example



Radio Report and Burglary are d-connected given Watson’s Report

A small detour (end)