Bayesian (Belief) Networks - Motivation

- Two problems with using full joint distribution tables as our probabilistic models:
  - Unless there are only a few variables, the joint is WAY too big to represent explicitly
  - Hard to learn (estimate) anything empirically about more than a few variables at a time

- Bayesian network: a technique for describing complex joint distributions (models) using simple, local distributions (conditional probabilities)
  - More properly called graphical models
  - We describe how variables locally interact
  - Local interactions chain together to give global, indirect interactions

Bayesian Networks

•A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions

•Syntax:

–a set of nodes, one per variable –

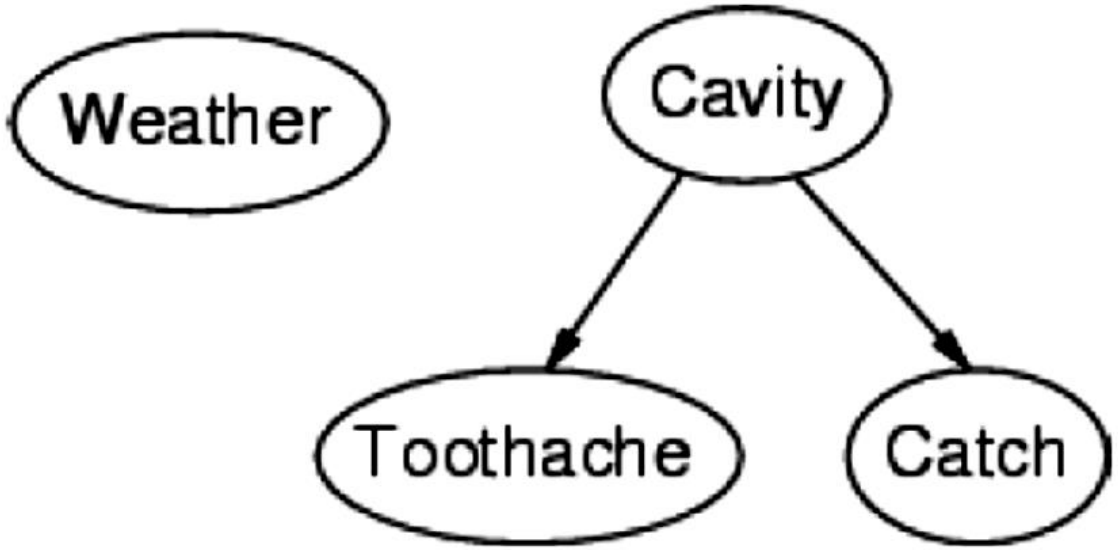
–a directed, acyclic graph (link ≈ "directly influences")

–a conditional distribution for each node given its parents: **P** (Xi | Parents(Xi))

•In the simplest case, conditional distribution represented as a conditional probability table (CPT) giving the distribution over *Xi* for each combination of parent values

Example

- Topology of network encodes conditional independence assertions:



- *Weather* is independent of the other variables
- *Toothache* and *Catch* are conditionally independent given *Cavity*

Example: Coin Flips

- N independent coin flips

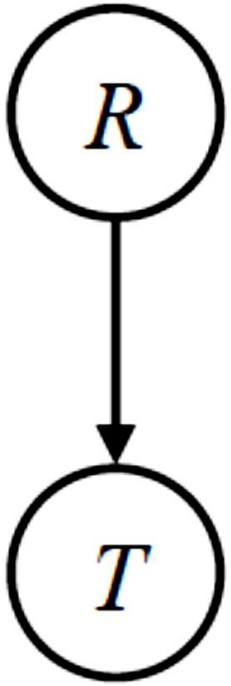
- No interactions between variables: absolute independence

Example: Traffic

- Variables:
  - R: It rains
  - T: There is traffic

- Model 1: independence
- Model 2: rain causes traffic
- Why is an agent using model 2 better?

Example: Traffic



$P(R)$

+r	1/4
¬r	3/4

$P(T|R)$

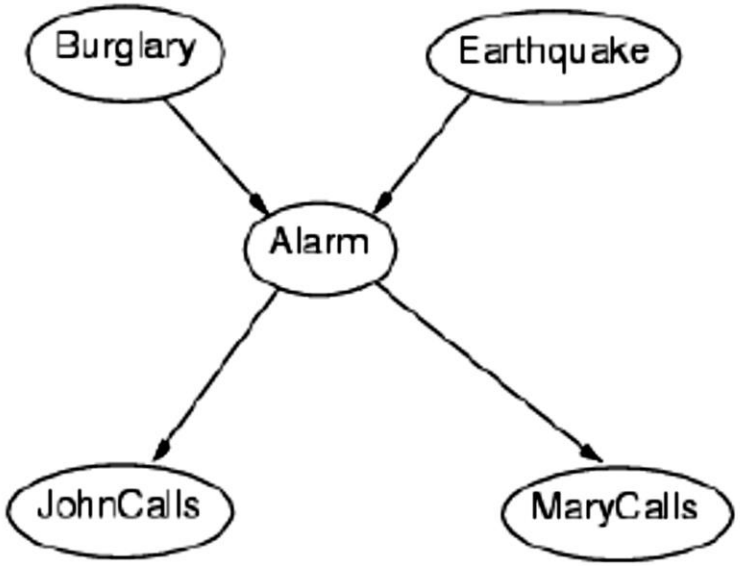
+r →	+t	3/4
	¬t	1/4
¬r →	+t	1/2
	¬t	1/2

Alarm Example

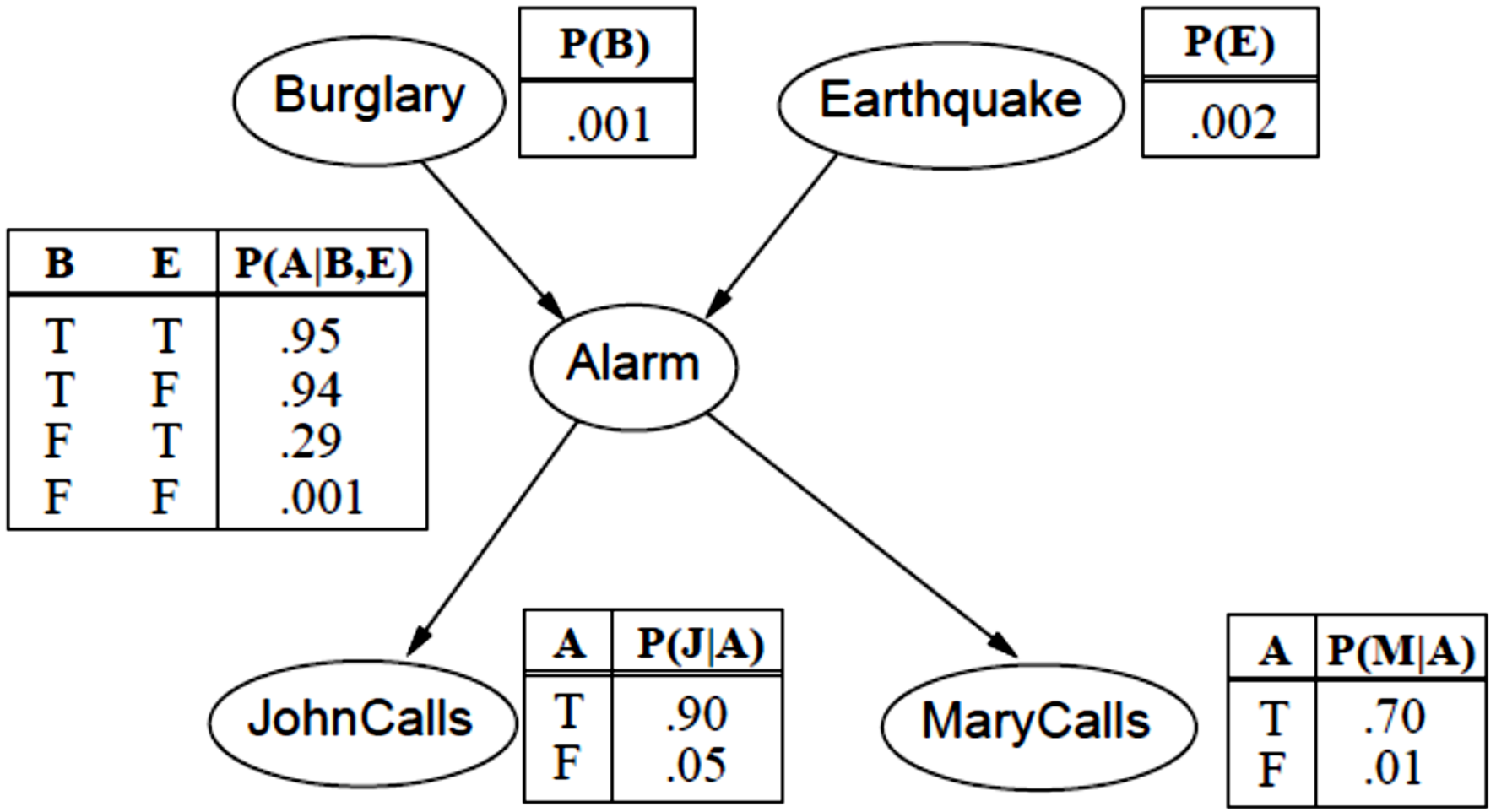
- I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?

- 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

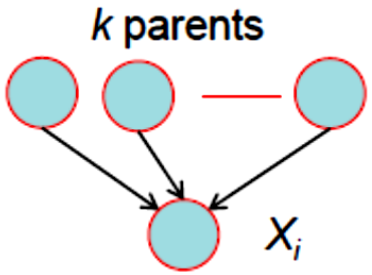


Bayesian Net for Alarm Example



Compact Representation of Probabilities in Bayesian Networks

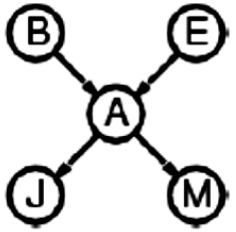
- A CPT for Boolean X_i with k Boolean parents has 2^k rows for the combinations of parent values



- Each row requires one number p for X_i = *true* (the other number for X_i = *false* is just *1-p*)

- If each variable has no more than k parents, an n-variable network requires $O(n \cdot 2^k)$ numbers
  - This grows linearly with n vs. $O(2^n)$ for full joint distribution

- For burglar network, 1+1+4+2+2 = 10 numbers (vs. 2^5-1 = 31 numbers) for full joint distribution



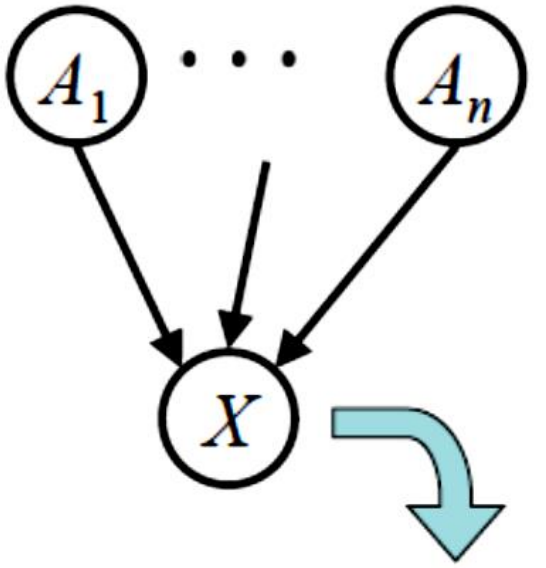
BNs: Huge space savings

Bayesian Net Semantics

- Let's formalize the semantics of a Bayes' net
- A set of nodes, one per variable X
- A directed, acyclic graph
- A conditional distribution for each node
  - A collection of distributions over X, one for each combination of parents' values

$$P(X|a_1 \ldots a_n)$$

  - CPT: conditional probability table



$$P(X|A_1 \ldots A_n)$$

A Bayesian Net = Topology (graph) + Local Conditional Probabilities

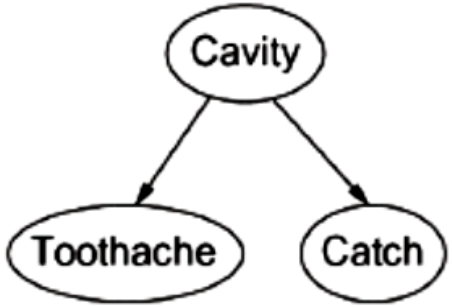
Bayesian Net Semantics

Bayesian Nets implicitly **encode Joint Distributions**

- As a product of local conditional distributions
- To see what probability a BN gives to a full assignment, multiply all the relevant conditionals together:

$$P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

- Example:



$P(+cavity, +catch, -toothache)$

$P(+cavity) * P(-toothache|+cavity) * P(+catch|$+cavity$)$

Encoding Joint Distribution in BNs

- Why are we guaranteed that setting

$$P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

results in a proper joint distribution?

- Chain rule (valid for all distributions): $P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i | x_1 \ldots x_{i-1})$
- Assume conditional independences: $P(x_i | x_1, \ldots x_{i-1}) = P(x_i | parents(X_i))$

→ Consequence: $P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$

- Not every BN can represent every joint distribution
  - The topology enforces certain conditional independencies

More generally it holds for Distributions:

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

Semantics

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

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

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

$$= P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e)$$
$$= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998$$
$$\approx 0.00063$$

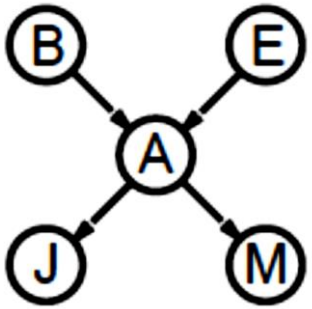
P(B)
.001

P(E)
.002

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

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

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

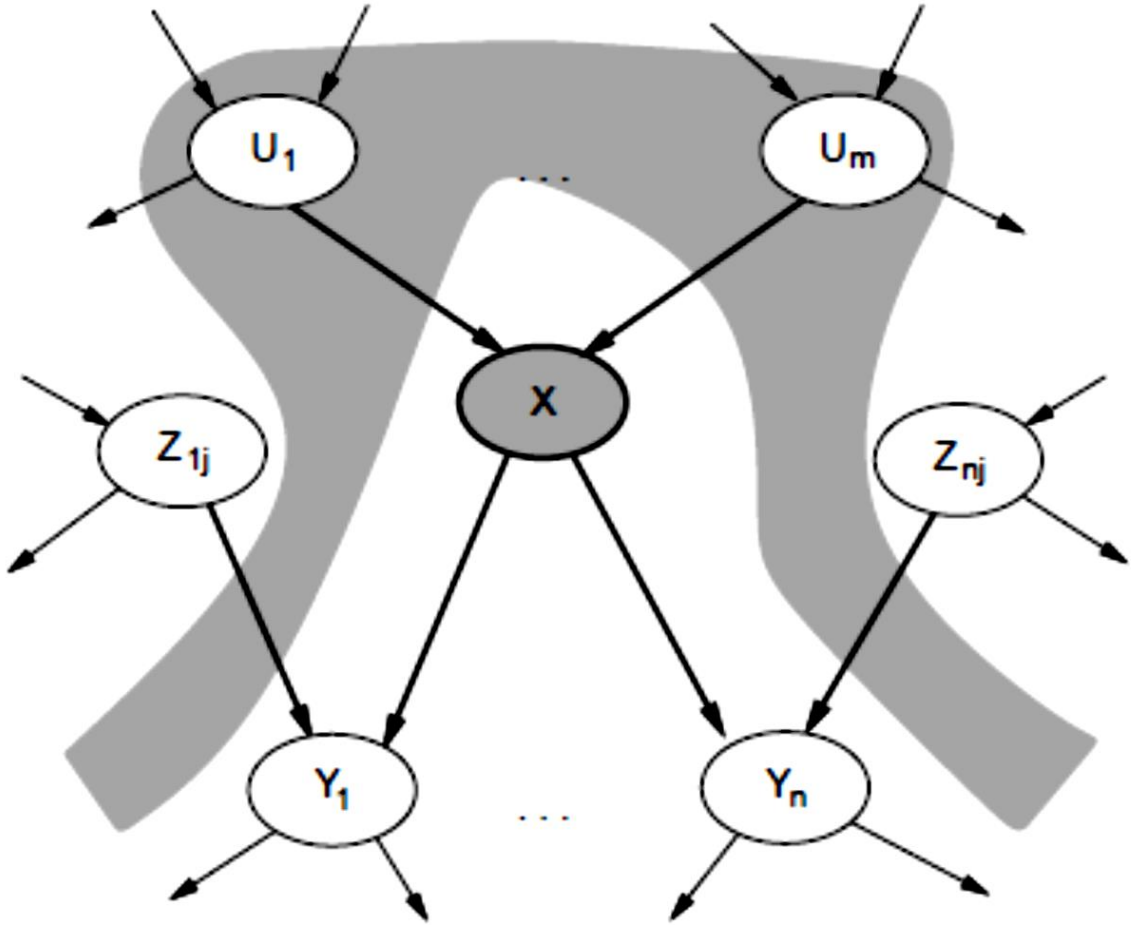


Parents(X_i) denotes the values of *Parents*(X_i) that appears in $x_1, x_2, .., x_n$

To emphasize: every BN over a domain implicitly defines a joint distribution over that domain, specified by local probabilities and graph structure

Bayesian Net Property

Each node is conditionally independent of its non-descendants given its parents.

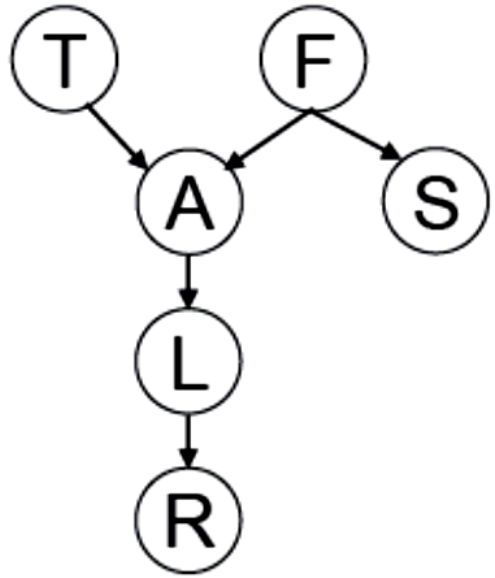


Constructing Bayesian Nets – An Example

Variables and domains:

- *Tampering* = *true* when there is tampering with the alarm.
- *Fire* = *true* when there is a fire.
- *Alarm* = *true* when the alarm sounds.
- *Smoke* = *true* when there is smoke.
- *Leaving* = *true* if there is an exodus of people.
- *Report* = *true* if there is a report given by someone of people leaving.

Are there any independence relationships
between these variables?



Consider the variables in the order of causality:

- ***Fire*** is independent of ***Tampering***.
- ***Alarm*** depends on both *Fire* and *Tampering*.
- ***Smoke*** depends only on *Fire*. It is conditionally independent of *Tampering* and *Alarm* given whether there is a *Fire*.
- ***Leaving*** only depends on *Alarm* and not directly on *Fire* or *Tampering* or *Smoke*.
- ***Report*** depends directly only on *Leaving*.

The network topology expresses the conditional independencies above.

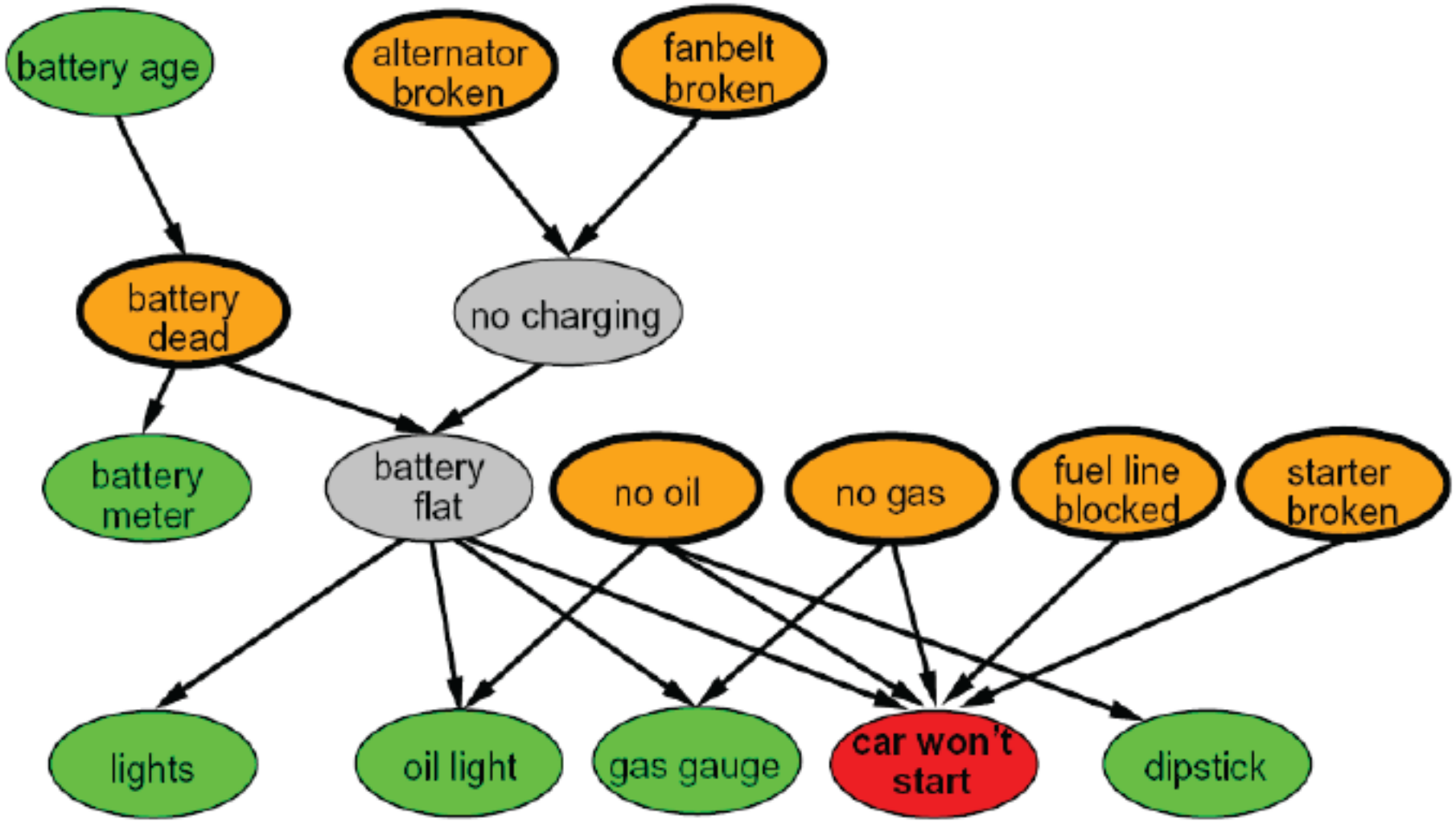
Causality?

- When Bayes' nets reflect the true causal patterns:
  - Often simpler (nodes have fewer parents)
  - Often easier to think about
  - Often easier to elicit from experts
- BNs need not actually be causal
  - Sometimes no causal net exists over the domain
  - End up with arrows that reflect correlation, not causation
- What do the arrows really mean?
  - Topology may happen to encode causal structure
  - Topology only guaranteed to encode conditional independence

Summarizing:

- 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

Example


battery age
alternator broken
fanbelt broken
battery dead
no charging
battery meter
battery flat
no oil
no gas
fuel line blocked
starter broken
lights
oil light
gas gauge
car won't start
dipstick

Inferencing with BN

- Inference: calculating some useful quantity from a joint probability distribution

- Examples:
  - Posterior probability

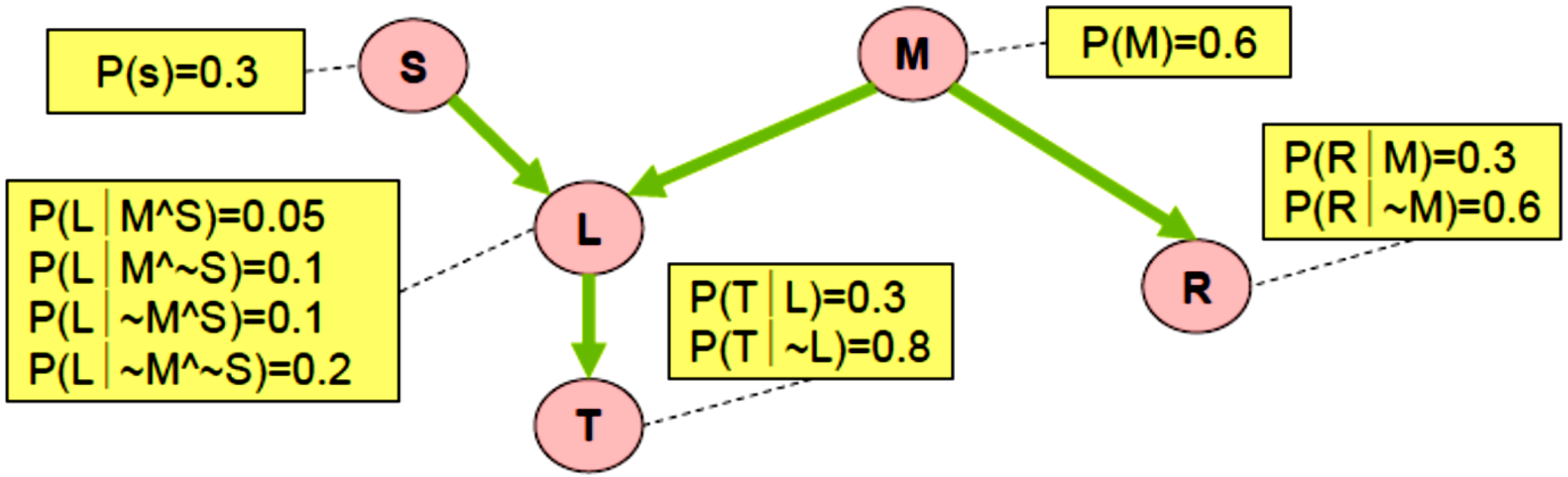
$$P(Q|E_1 = e_1, \ldots E_k = e_k)$$

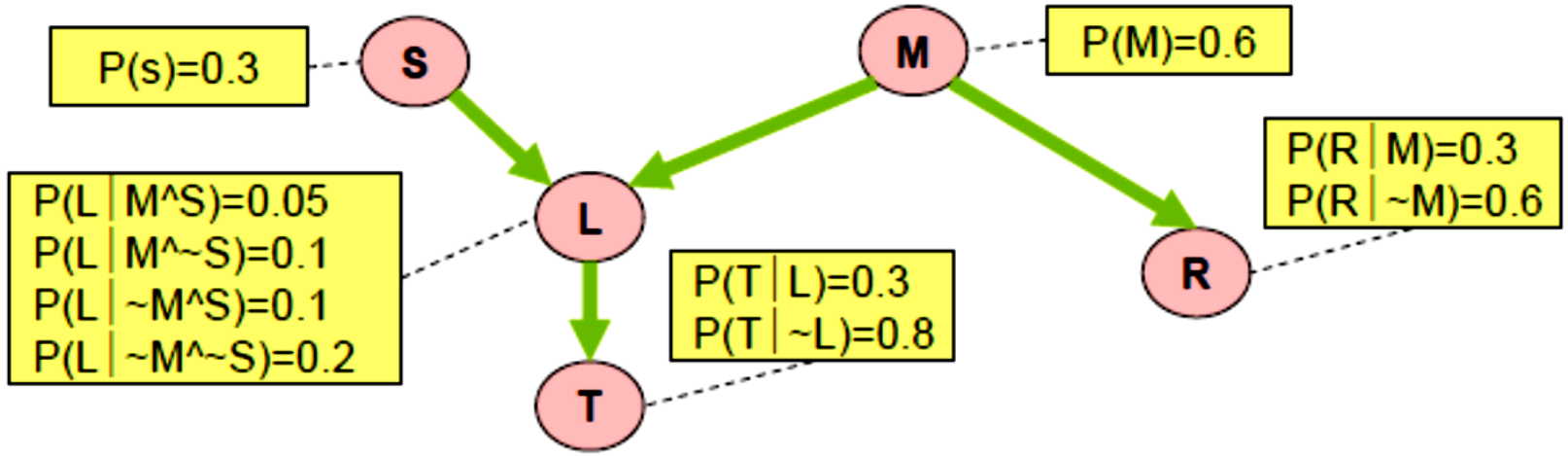
- The graphical independence representation yields efficient inference schemes
- We generally want to compute
  - $P(X|E)$ where E is evidence from sensory measurements etc. (known values for variables)
  - Sometimes, may want to compute just $P(X)$

Computing a Joint Entry

How to compute an entry in a joint distribution?

E.G: What is P(S ^ ~M ^ L ~R ^ T)?

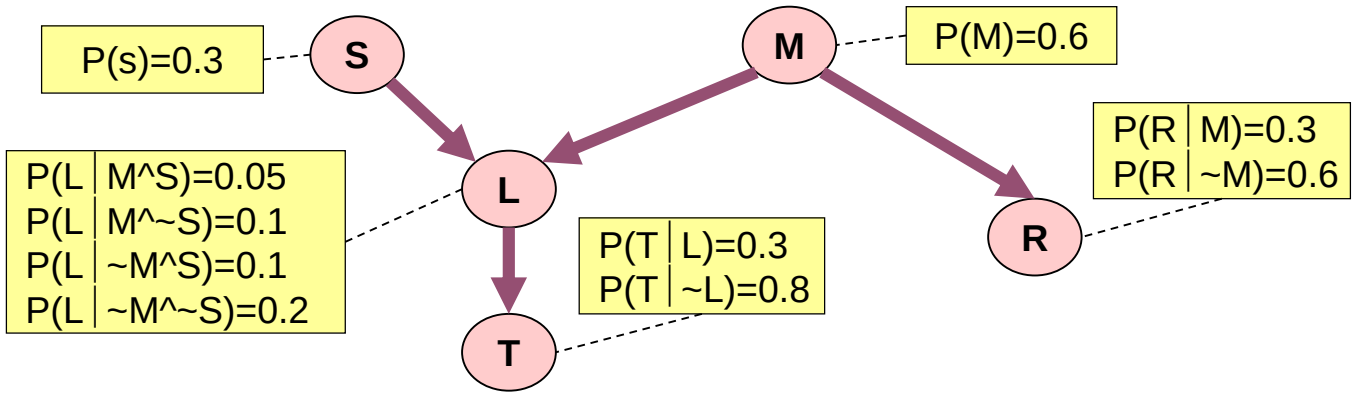




$P(T \wedge \neg R \wedge L \wedge \neg M \wedge S) =$

$P(T \mid \neg R \wedge L \wedge \neg M \wedge S) * P(\neg R \wedge L \wedge \neg M \wedge S) =$
$P(T \mid L) * P(\neg R \wedge L \wedge \neg M \wedge S) =$
$P(T \mid L) * P(\neg R \mid L \wedge \neg M \wedge S) * P(L \wedge \neg M \wedge S) =$
$P(T \mid L) * P(\neg R \mid \neg M) * P(L \wedge \neg M \wedge S) =$
$P(T \mid L) * P(\neg R \mid \neg M) * P(L \mid \neg M \wedge S) * P(\neg M \wedge S) =$
$P(T \mid L) * P(\neg R \mid \neg M) * P(L \mid \neg M \wedge S) * P(\neg M \mid S) * P(S) =$

$P(T \mid L) * P(\neg R \mid \neg M) * P(L \mid \neg M \wedge S) * P(\neg M) * P(S).$

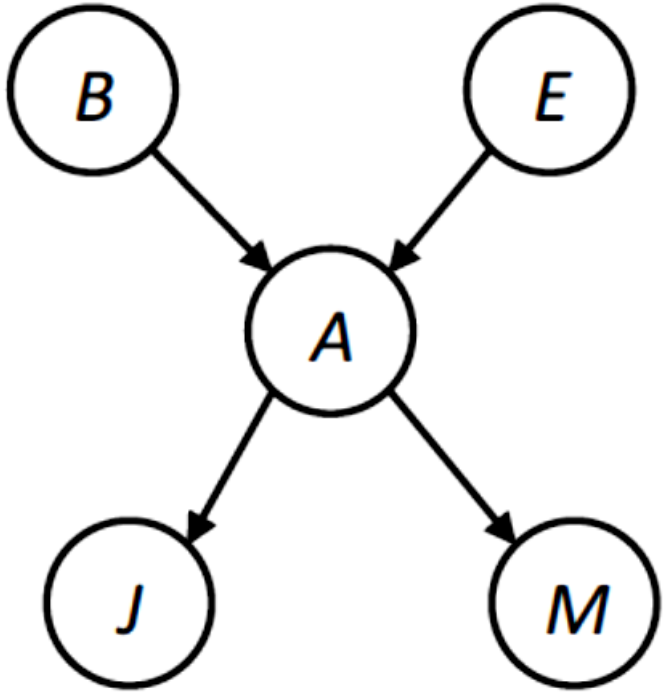


E.G. What could we do to compute P(R | T,~S)?

$$\frac{P(R \wedge T \wedge \neg S)}{P(R \wedge T \wedge \neg S) + P(\neg R \wedge T \wedge \neg S)}$$

Inference by Enumeration

- Given unlimited time, inference in BNs is easy
- Recipe:
  - State the marginal probabilities you need
  - Figure out ALL atomic probabilities you need
  - Calculate and combine them
- Example: $P(+b|+j,+m) = \frac{P(+b,+j,+m)}{P(+j,+m)}$



$$P(+b,+j,+m) = P(+b)P(+e)P(+a|+b,+e)P(+j|+a)P(+m|+a)+$$
$$P(+b)P(+e)P(-a|+b,+e)P(+j|-a)P(+m|-a)+$$
$$P(+b)P(-e)P(+a|+b,-e)P(+j|+a)P(+m|+a)+$$
$$P(+b)P(-e)P(-a|+b,-e)P(+j|-a)P(+m|-a)$$

Complexity of Enumeration

Assume n Boolean variables in a Bayesian Network.
Then worst case running time is :

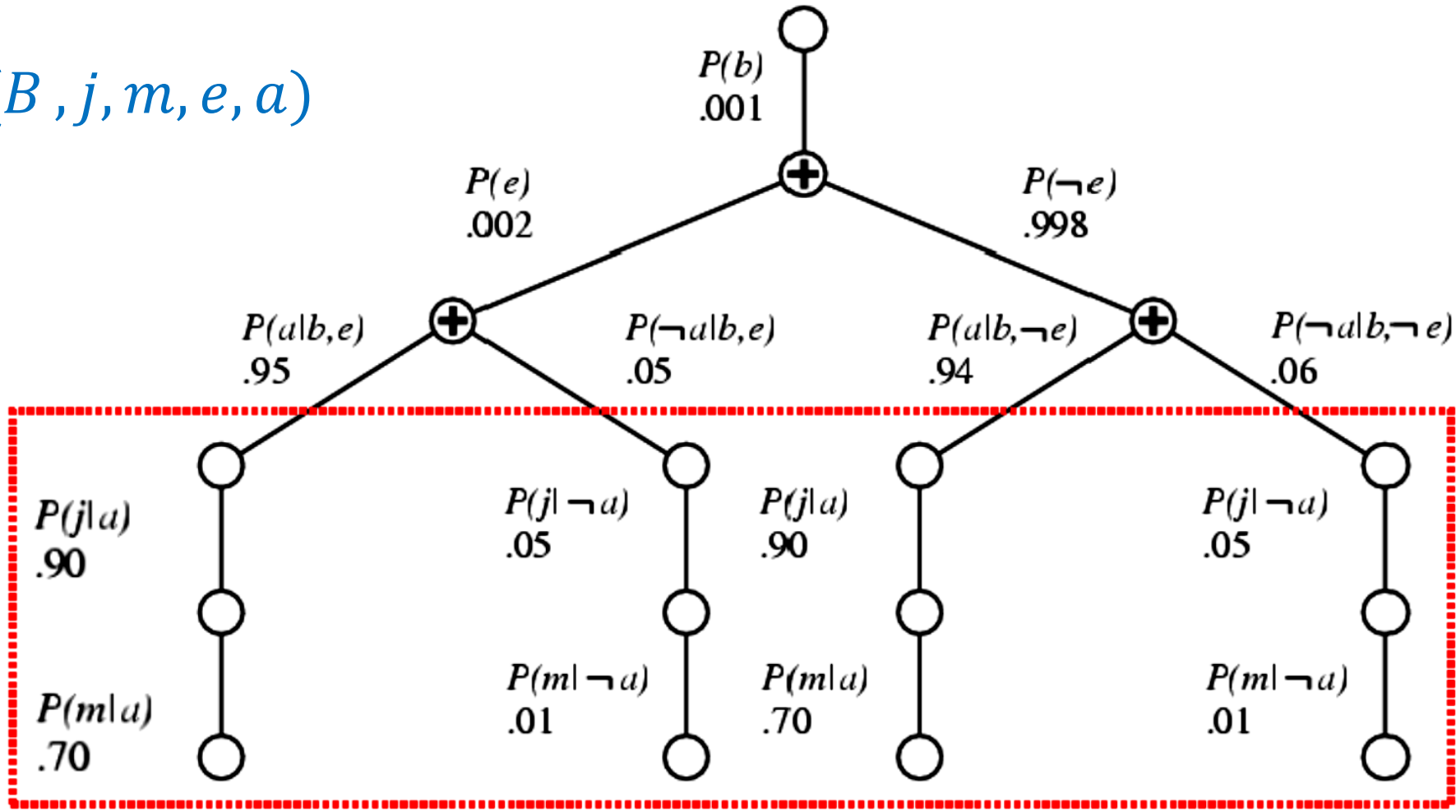
$$O(n2^n)$$

O(n) possible value combinations and O(n) multiplications per term. Each value combination is a term.

In general exact inference in Bayesian Nets is NP-hard:
Proved by encoding 3-SAT as a Bayesian Net

Structure of Computation

$$P(B \mid j,m) = \alpha \sum_e \sum_a P(B, j, m, e, a)$$



$P(b|j,m) = \alpha\, P(b) \Sigma_e P(e) \Sigma_a P(a|b,e) P(j|a) P(m|a)$

Repeated computations ⇒ use dynamic programming

12 multiplications vs. naive: 16 mults

Variable Elimination to the Rescue! (Heuristic)