Bayesian Networks

Chris Amato
Northeastern University

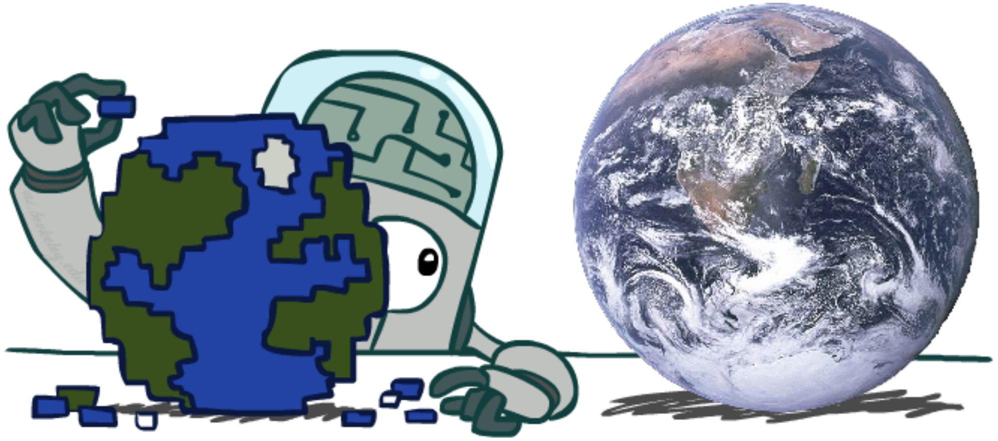
Some images and slides are used from: Rob Platt, CS188 UC Berkeley, AIMA, Mykel Kochenderfer

Probabilistic models

Models describe how (a portion of) the world works

Models are always simplifications

- May not account for every variable
- May not account for all interactions between variables
- "All models are wrong; but some are useful."
  – George E. P. Box



What do we do with probabilistic models?

- We (or our agents) need to reason about unknown variables, given evidence
- Example: explanation (diagnostic reasoning)
- Example: prediction (causal reasoning)
- Example: value of information

Bayes' nets: Big picture

Two problems with using full joint distribution tables as our probabilistic models:

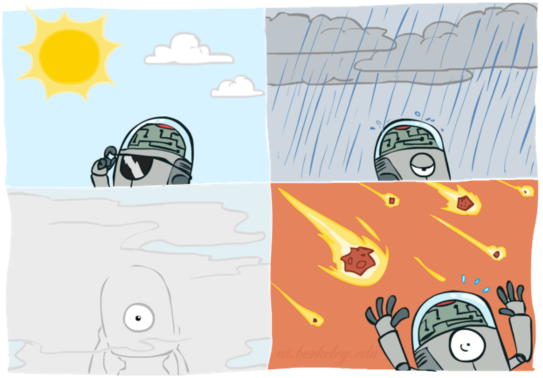
- Unless there are only a few variables, the joint is WAY too big to represent explicitly
- Hard to learn (estimate) anything empirically about more than a few variables at a time

Bayes' nets: a technique for describing complex joint distributions (models) using simple, local distributions (conditional probabilities)

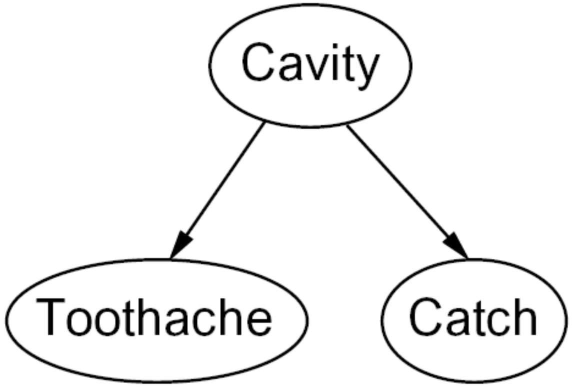
- More properly called graphical models
- We describe how variables locally interact
- Local interactions chain together to give global, indirect interactions

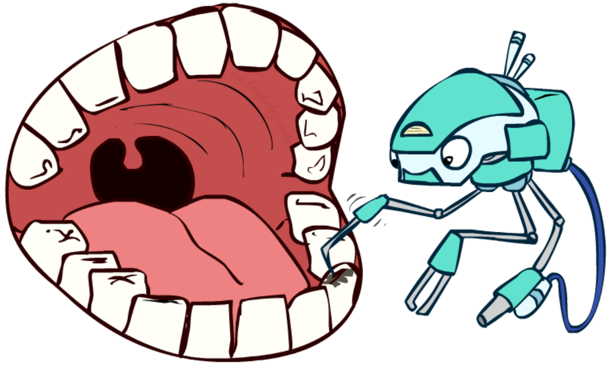
Graphical model notation

- Nodes: variables (with domains)
  - Can be assigned (observed) or unassigned (unobserved)



- Arcs: interactions
  - Similar to CSP constraints
  - Indicate "direct influence" between variables
  - Formally: encode conditional independence (more later)

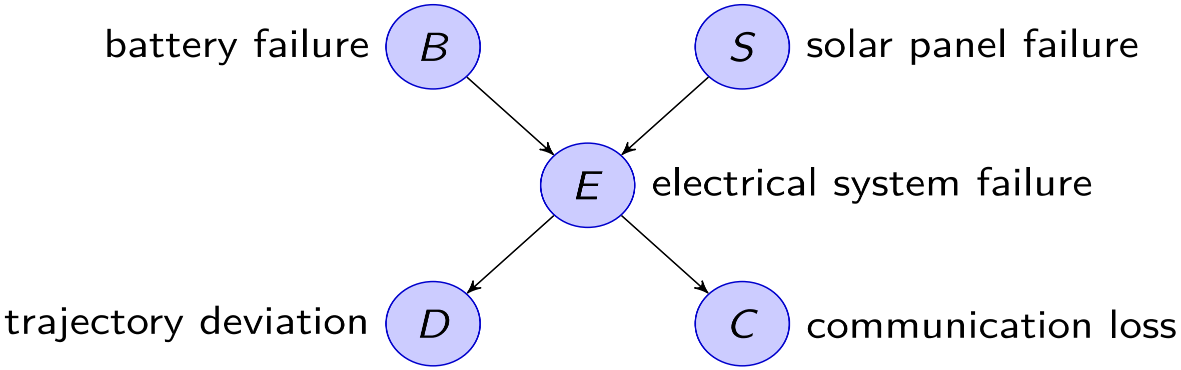




Bayes' net semantics

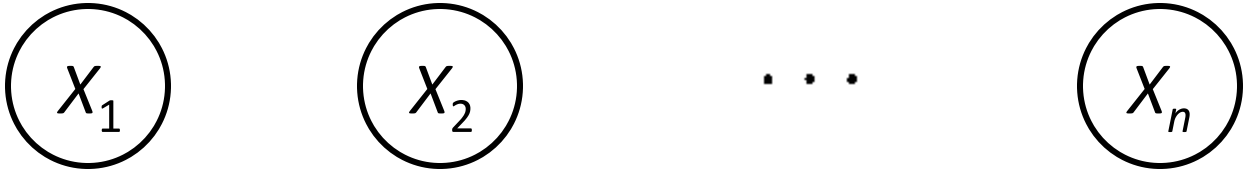
- A directed, acyclic graph, one node per random variable
- A conditional probability table (CPT) for each node
  - A collection of distributions over X, one for each combination of parents' values $P(X|a_1 \ldots a_n)$
- Bayes' nets implicitly encode joint distributions
  - As a product of local conditional distributions
  - To see what probability a BN gives to a full assignment, multiply all the relevant conditionals together:

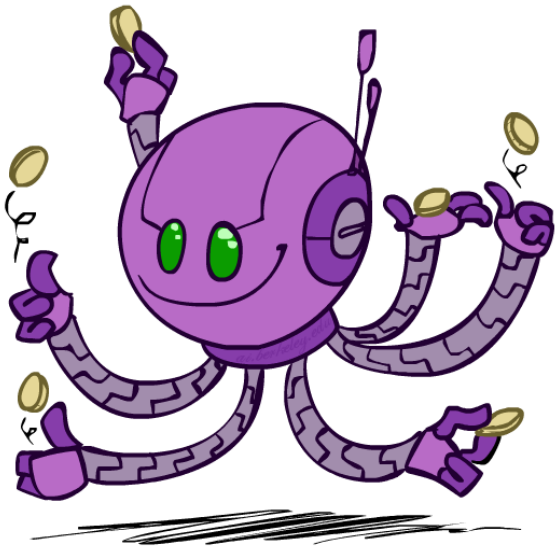
$$P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i|\mathit{parents}(X_i))$$



Example: Coin flips

N independent coin flips





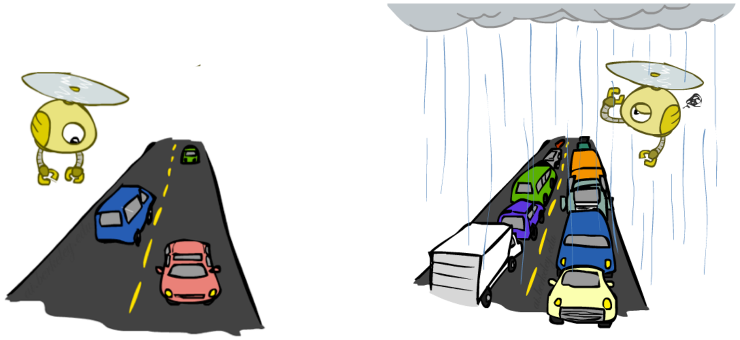
No interactions between variables: absolute independence

Example: Traffic

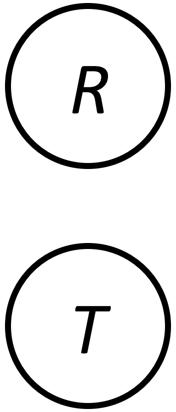
Variables:

R: It rains

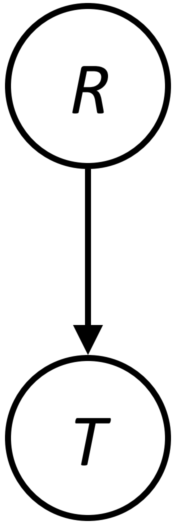
T: There is traffic



Model 1: independence



- Model 2: rain causes traffic



Why is an agent using model 2 better?

Probabilities in Bayes' nets

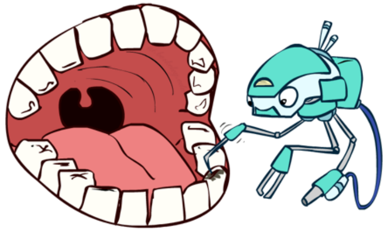
Bayes' nets implicitly encode joint distributions

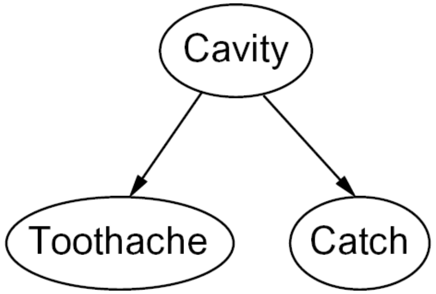
As a product of local conditional distributions

To see what probability a BN gives to a full assignment, multiply all the relevant conditionals together:

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

Example:





$$P(+\mathit{cavity},\ +\mathit{catch},\ \text{-}\mathit{toothache})$$

Probabilities in Bayes' nets

Why are we guaranteed that setting

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

results in a proper joint distribution?

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

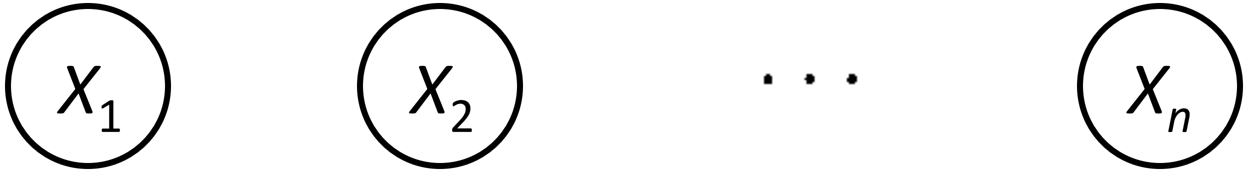
<u>Assume</u> conditional independences: $P(x_i | x_1, \ldots x_{i-1}) = P(x_i | \mathit{parents}(X_i))$

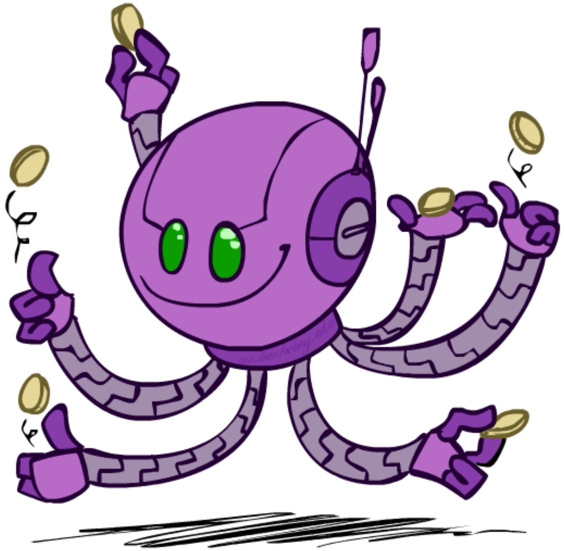
→ Consequence: $P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i | \mathit{parents}(X_i))$

Not every BN can represent every joint distribution

The topology enforces certain conditional independencies

Example: Coin flips





$P(X_1)$

h	0.5
t	0.5

$P(X_2)$

h	0.5
t	0.5

. . .

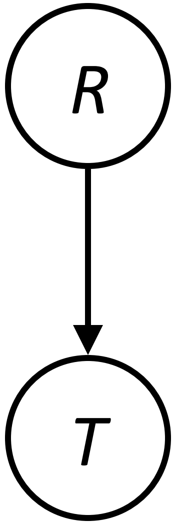
$P(X_n)$

h	0.5
t	0.5

$$P(h, h, t, h) =$$

Only distributions whose variables are absolutely independent can be represented by a Bayes' net with no arcs.

Example: Traffic



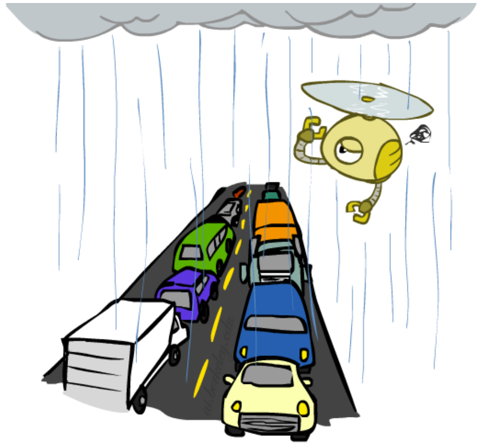
$$P(R)$$

+r	1/4
-r	3/4

$$P(T|R)$$

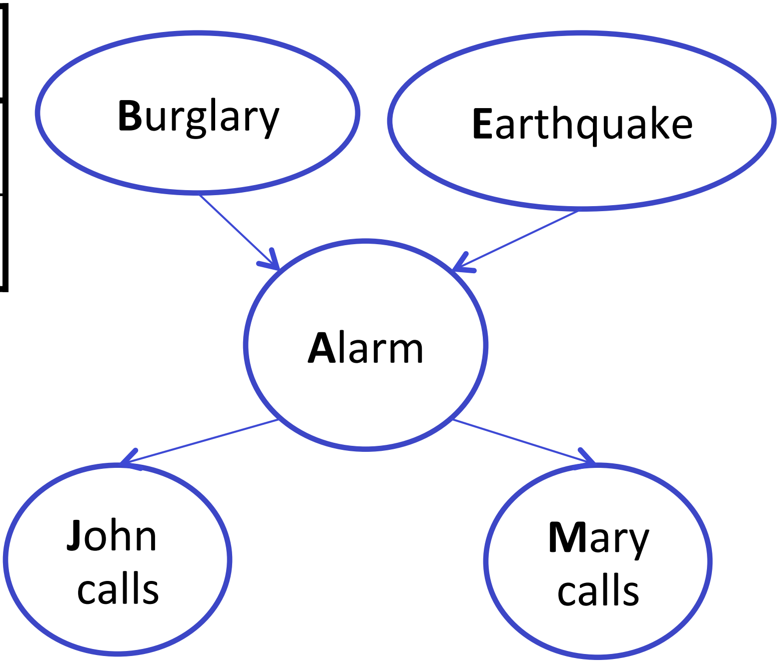
+r	+t	3/4
	-t	1/4
-r	+t	1/2
	-t	1/2

$$P(+r, -t) =$$

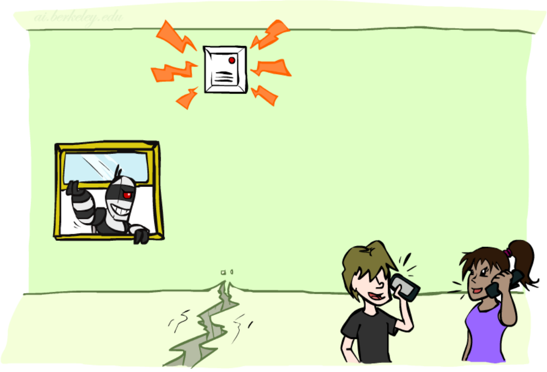


Example: Alarm network

B	P(B)
+b	0.001
-b	0.999



E	P(E)
+e	0.002
-e	0.998

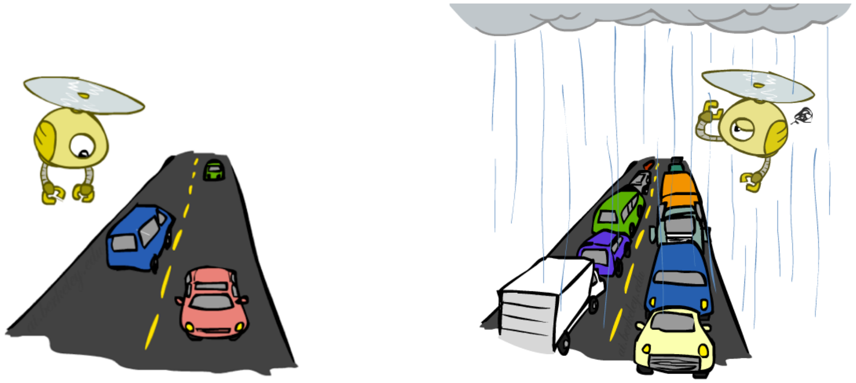


A	J	P(J|A)
+a	+j	0.9
+a	-j	0.1
-a	+j	0.05
-a	-j	0.95

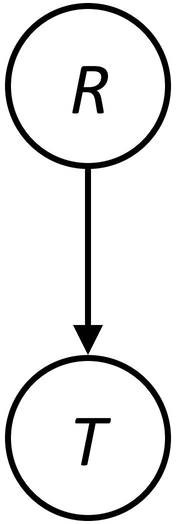
A	M	P(M|A)
+a	+m	0.7
+a	-m	0.3
-a	+m	0.01
-a	-m	0.99

B	E	A	P(A|B,E)
+b	+e	+a	0.95
+b	+e	-a	0.05
+b	-e	+a	0.94
+b	-e	-a	0.06
-b	+e	+a	0.29
-b	+e	-a	0.71
-b	-e	+a	0.001
-b	-e	-a	0.999

Example: Traffic



Causal direction



$P(R)$

+r	1/4
-r	3/4

$P(T|R)$

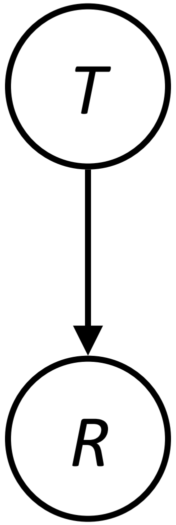
+r	+t	3/4
	-t	1/4
-r	+t	1/2
	-t	1/2

$P(T, R)$

+r	+t	3/16
+r	-t	1/16
-r	+t	6/16
-r	-t	6/16

Example: Reverse traffic

Reverse causality?



$P(T)$

+t	9/16
-t	7/16

$P(R|T)$

+t	+r	1/3
	-r	2/3

-t	+r	1/7
	-r	6/7

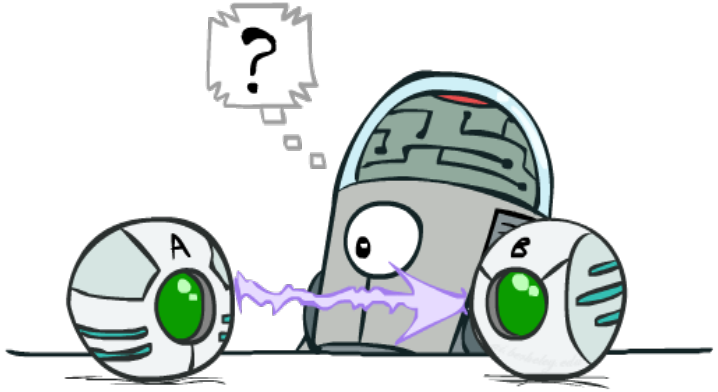
$P(T,R)$

+r	+t	3/16
+r	-t	1/16
-r	+t	6/16
-r	-t	6/16

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 (especially if variables are missing)
- E.g. consider the variable *Traffic*
- End up with arrows that reflect correlation, not causation

What do the arrows really mean?

- Topology may happen to encode causal structure
- Topology really encodes conditional independence

$$P(x_i|x_1, \ldots x_{i-1}) = P(x_i|\mathit{parents}(X_i))$$

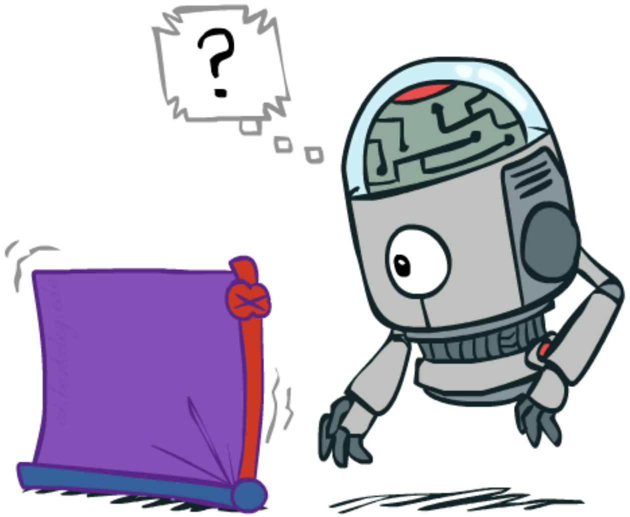
Size of a Bayes' net

- How big is a joint distribution over N Boolean variables?

  2^N

- How big is an N-node net if nodes have up to k parents?

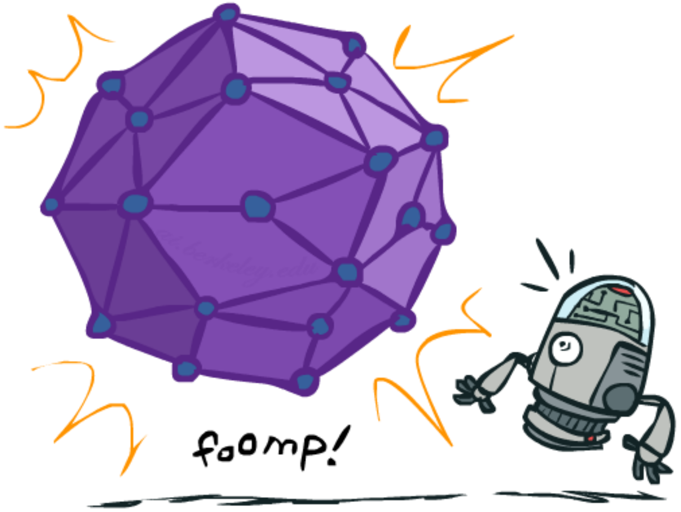
  $O(N * 2^{k+1})$



- Both give you the power to calculate

$$P(X_1, X_2, \dots X_n)$$

- BNs: Huge space savings!
- Also easier to elicit local CPTs
- Also faster to answer queries (coming)

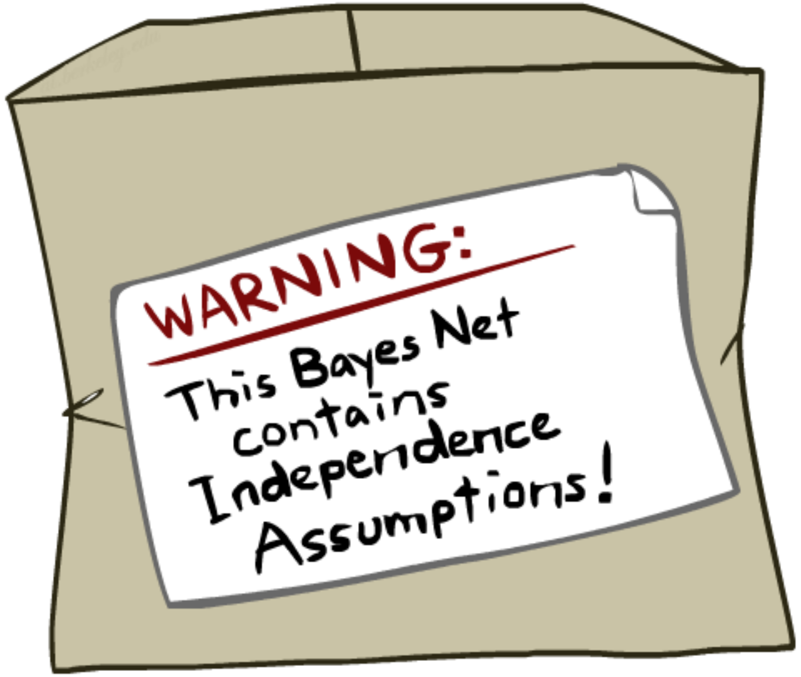


Bayes' nets: Assumptions

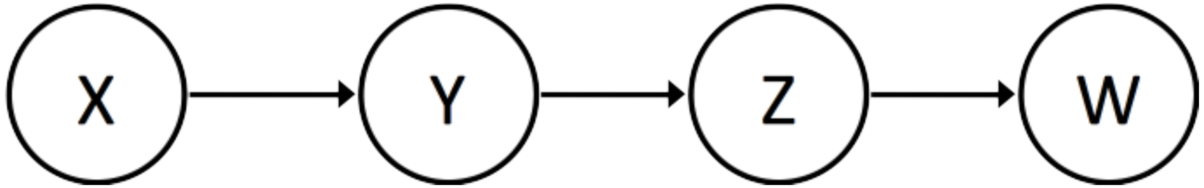
- Assumptions we are required to make to define the Bayes net when given the graph:

$$P(x_i|x_1 \cdots x_{i-1}) = P(x_i|parents(X_i))$$

- Beyond above "chain rule → Bayes net" conditional independence assumptions
  - Often additional conditional independences
  - They can be read off the graph
- Important for modeling: understand assumptions made when choosing a Bayes net graph



Example

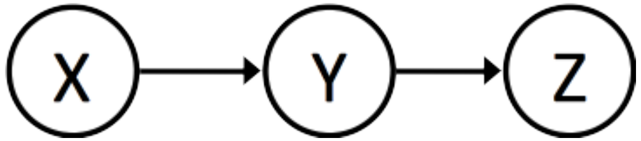


- Conditional independence assumptions directly from simplifications in chain rule:

- Additional implied conditional independence assumptions?

Independence in a Bayes' net

- Important question about a BN:
  - Are two nodes independent given certain evidence?
  - If yes, can prove using algebra (tedious in general)
  - If no, can prove with a counter example
  - Example:



  - Question: are X and Z necessarily independent?
    - Answer: no. Example: low pressure causes rain, which causes traffic.
    - X can influence Z, Z can influence X (via Y)
    - Addendum: they *could* be independent: how?

D-separation: Outline

- Study independence properties for triples
- Analyze complex cases in terms of member triples
- D-separation: a condition / algorithm for answering such queries

Causal chains

- This configuration is a "causal chain"

X: Low pressure Y: Rain Z: Traffic

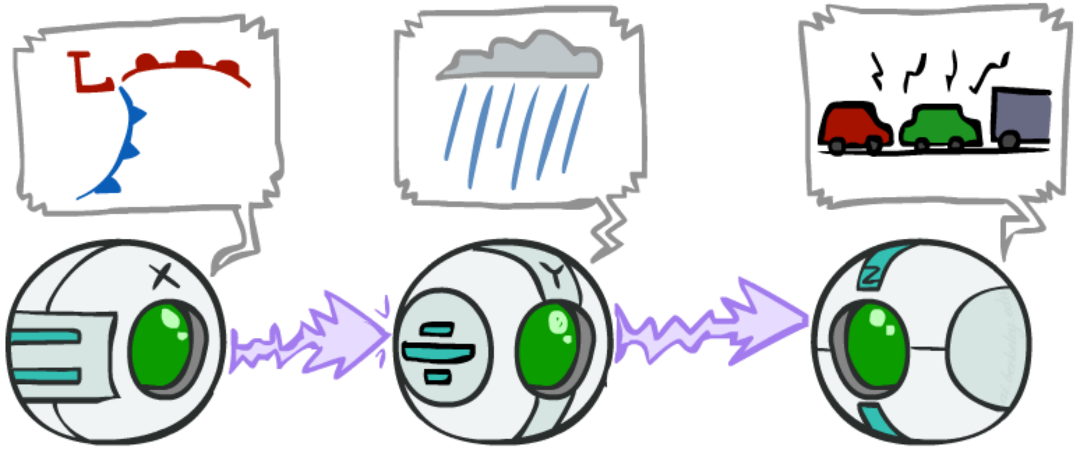
$$P(x, y, z) = P(x)P(y|x)P(z|y)$$

- Guaranteed X independent of Z ?
- *No!*
  - One example set of CPTs for which X is not independent of Z is sufficient to show this independence is not guaranteed.
  - Example:
    - Low pressure causes rain causes traffic, high pressure causes no rain causes no traffic
    - In numbers:

      P(+y | +x) = 1, P(-y | - x) = 1,
      P(+z | +y) = 1, P(-z | -y) = 1

Causal chains

- This configuration is a "causal chain"



X: Low pressure Y: Rain Z: Traffic

$$P(x, y, z) = P(x)P(y|x)P(z|y)$$

- Guaranteed X independent of Z given Y?

$$P(z|x, y) = \frac{P(x, y, z)}{P(x, y)}$$

$$= \frac{P(x)P(y|x)P(z|y)}{P(x)P(y|x)}$$

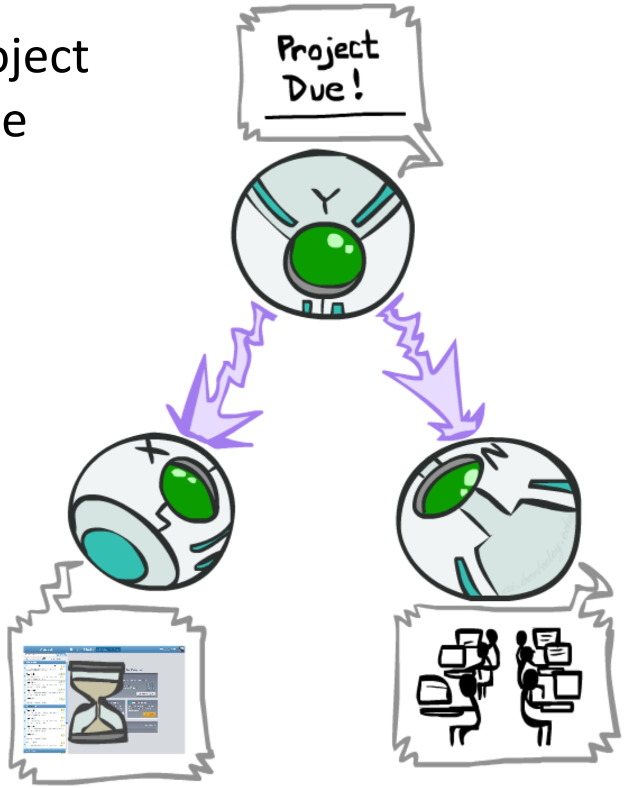
$$= P(z|y)$$

Yes!

- Evidence along the chain "blocks" the influence

Common cause

- This configuration is a “common cause”



Y: Project due

X: Forums busy

Z: Lab full

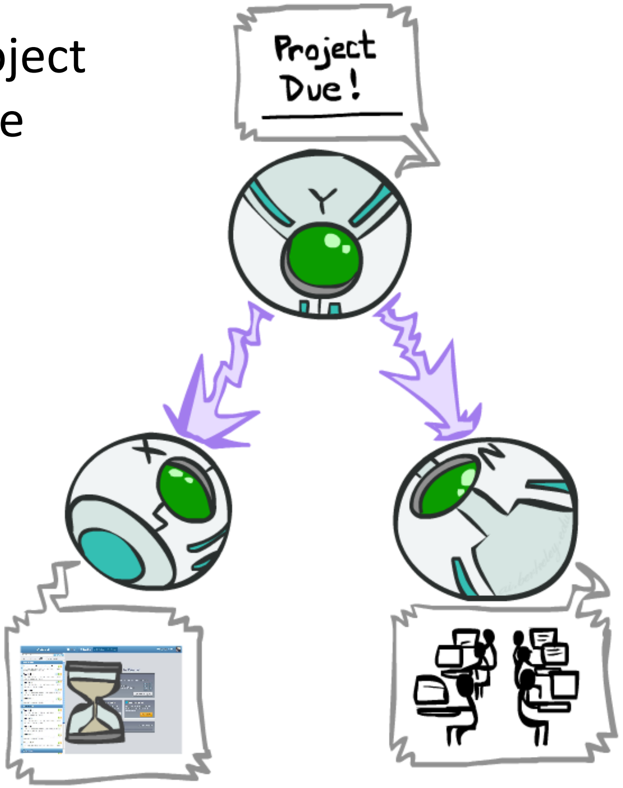
$$P(x,y,z) = P(y)P(x|y)P(z|y)$$

- Guaranteed X independent of Z ?
- *No!*
    - One example set of CPTs for which X is not independent of Z is sufficient to show this independence is not guaranteed.
    - Example:
        - Project due causes both forums busy and lab full
        - In numbers:

          P(+x | +y) = 1, P(-x | -y) = 1,
          P(+z | +y) = 1, P(-z | -y) = 1

Common cause

- This configuration is a "common cause"



Y: Project due

X: Forums busy

Z: Lab full

$$P(x, y, z) = P(y)P(x|y)P(z|y)$$

- Guaranteed X and Z independent given Y?

$$P(z|x, y) = \frac{P(x, y, z)}{P(x, y)}$$

$$= \frac{P(y)P(x|y)P(z|y)}{P(y)P(x|y)}$$

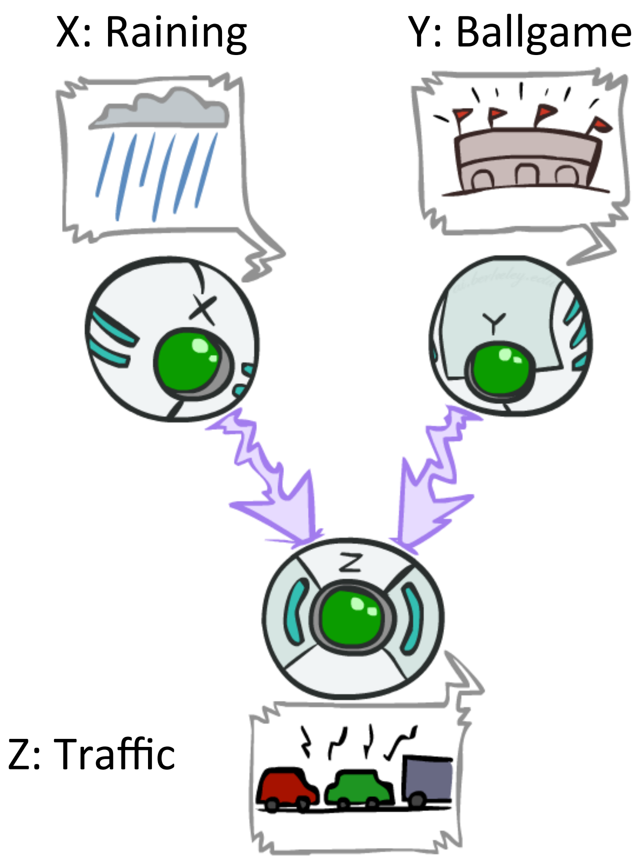
$$= P(z|y)$$

Yes!

- Observing the cause blocks influence between effects.

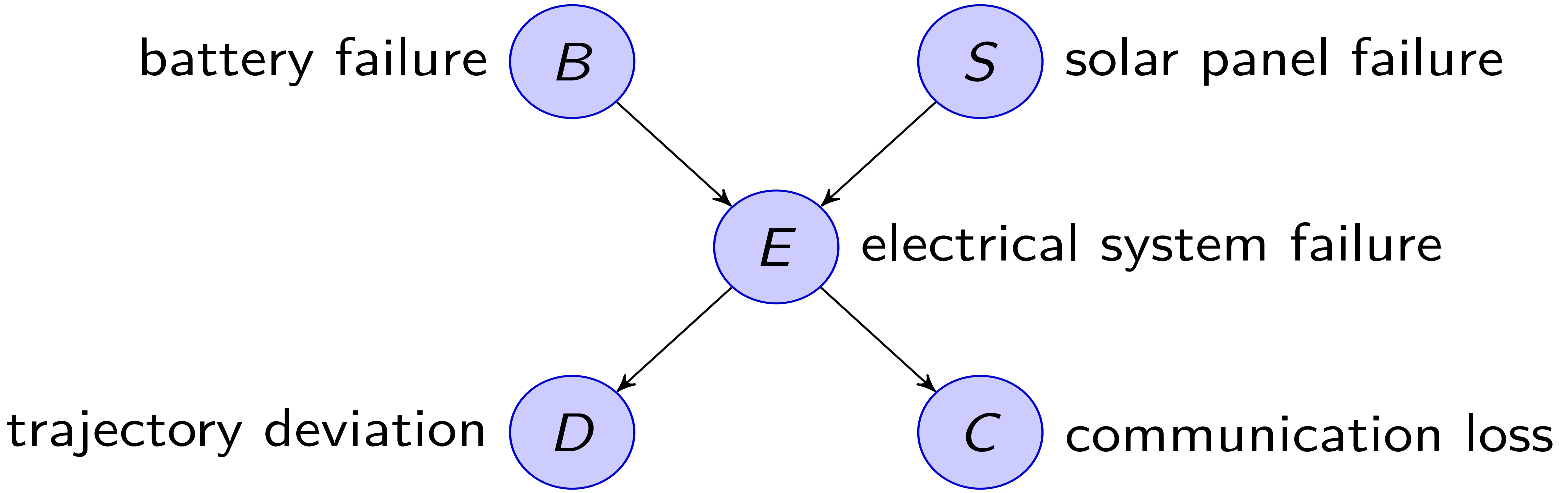
Common effect

- Last configuration: two causes of one effect (v-structures)

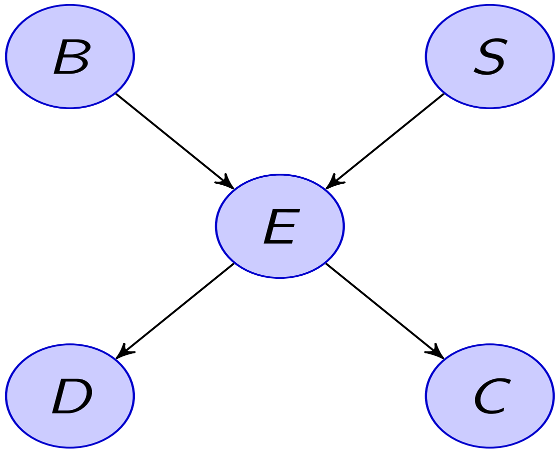


- Are X and Y independent?
  - *Yes*: the ballgame and the rain cause traffic, but they are not correlated
  - Still need to prove they must be (try it!)
- Are X and Y independent given Z?
  - *No*: seeing traffic puts the rain and the ballgame in competition as explanation.
- This is backwards from the other cases
  - Observing an effect activates influence between possible causes.

Conditional independence: Battery example



Bayes nets: conditional independence example



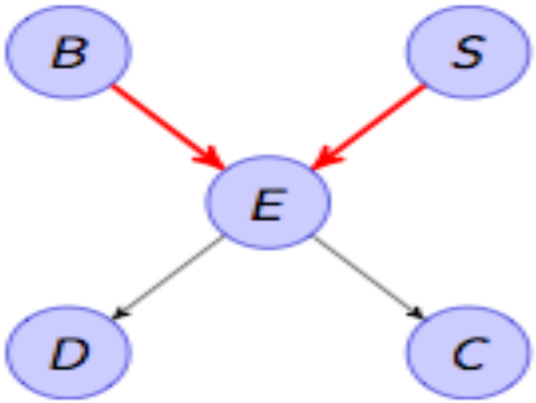
C is independent of B given E

Knowing that you have a battery failure does not affect your belief that there is a communication loss if you know that there has been an electrical system failure

D is independent of S given E

If you know there is an electrical failure, observing a trajectory deviation does not affect your belief that there has been a solar panel failure

Conditional independence in v-structures



B is independent of S

Knowing there is a battery failure does not affect your belief about whether there has been a solar panel failure

B is not independent of S given E

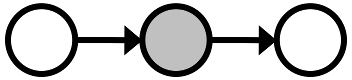
If you know there has been an electrical failure and there has not been a solar panel failure, then it is more likely there was a battery failure

Influence only flows through $B \rightarrow E \leftarrow S$ (a v-structure) when E (or one of its descendants) is known

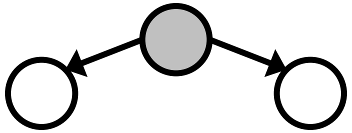
Independence concepts: General case

Two sets of nodes, A and B, are conditionally independent given node set C are called d-separated in the Bayes net if for any path between A and B:

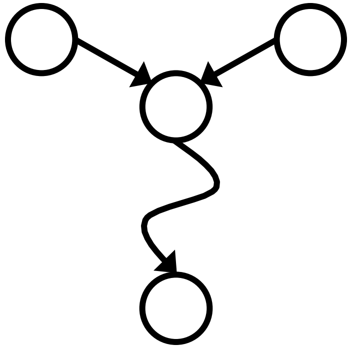
- The path contains a chain of nodes, $A \rightarrow X \rightarrow B$, such that X is in C



- The path contains a fork, $A \leftarrow X \rightarrow B$, such that X is in C



- The path contains a v-structure, $A \rightarrow X \leftarrow B$, such that X is *not* in C and no descendant of X is in C



Markov blanket: the parents, the children and the parents of the children

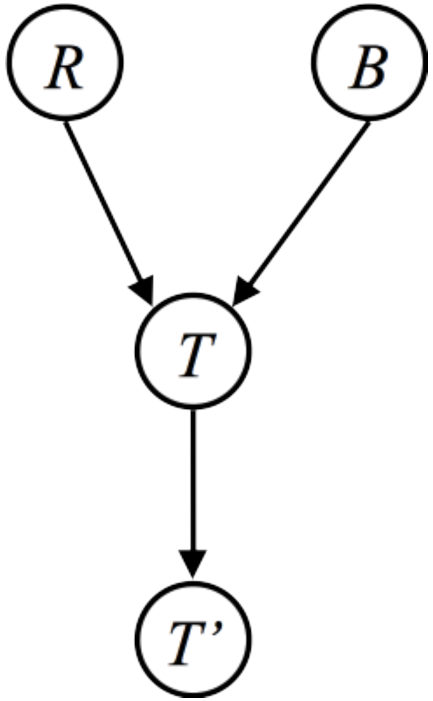
- A node is independent of all other nodes in the graph given its Markov blanket

Example

$R \perp\!\!\!\perp B$ Yes

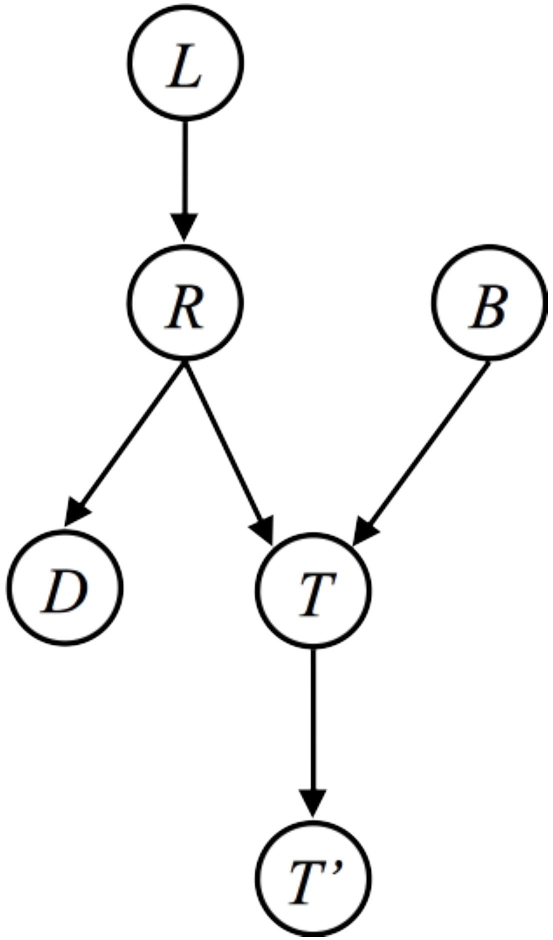
$R \perp\!\!\!\perp B | T$

$R \perp\!\!\!\perp B | T'$



Example

$L \perp\!\!\!\perp T' \mid T$	*Yes*
$L \perp\!\!\!\perp B$	*Yes*
$L \perp\!\!\!\perp B \mid T$	
$L \perp\!\!\!\perp B \mid T'$	
$L \perp\!\!\!\perp B \mid T, R$	*Yes*



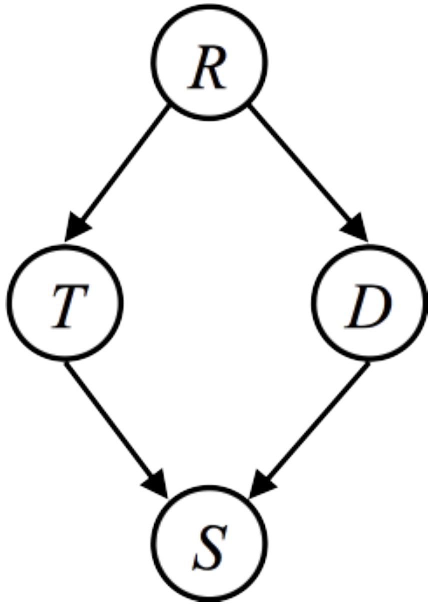
Example

- Variables:
  - R: Raining
  - T: Traffic
  - D: Roof drips
  - S: I'm sad
- Questions:

$$T \perp\!\!\!\perp D$$

$$T \perp\!\!\!\perp D | R$$ *Yes*

$$T \perp\!\!\!\perp D | R, S$$

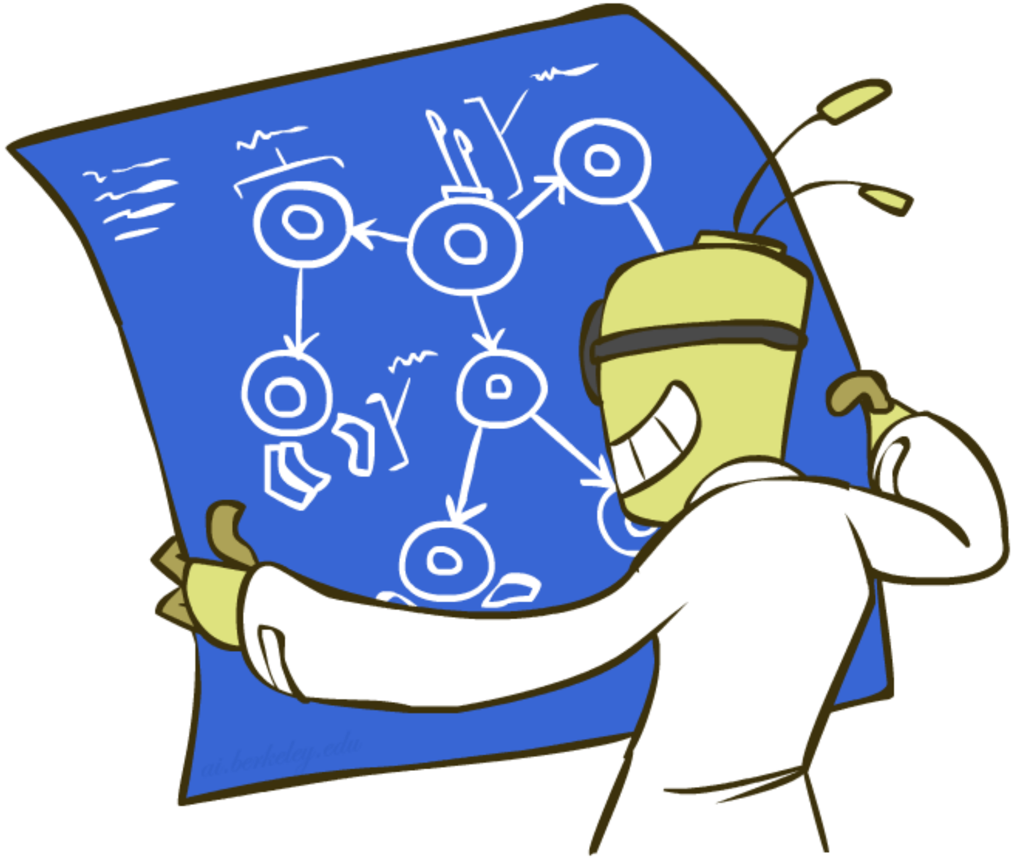


Structure implications

- Given a Bayes net structure, can run d-separation algorithm to build a complete list of conditional independences that are necessarily true of the form

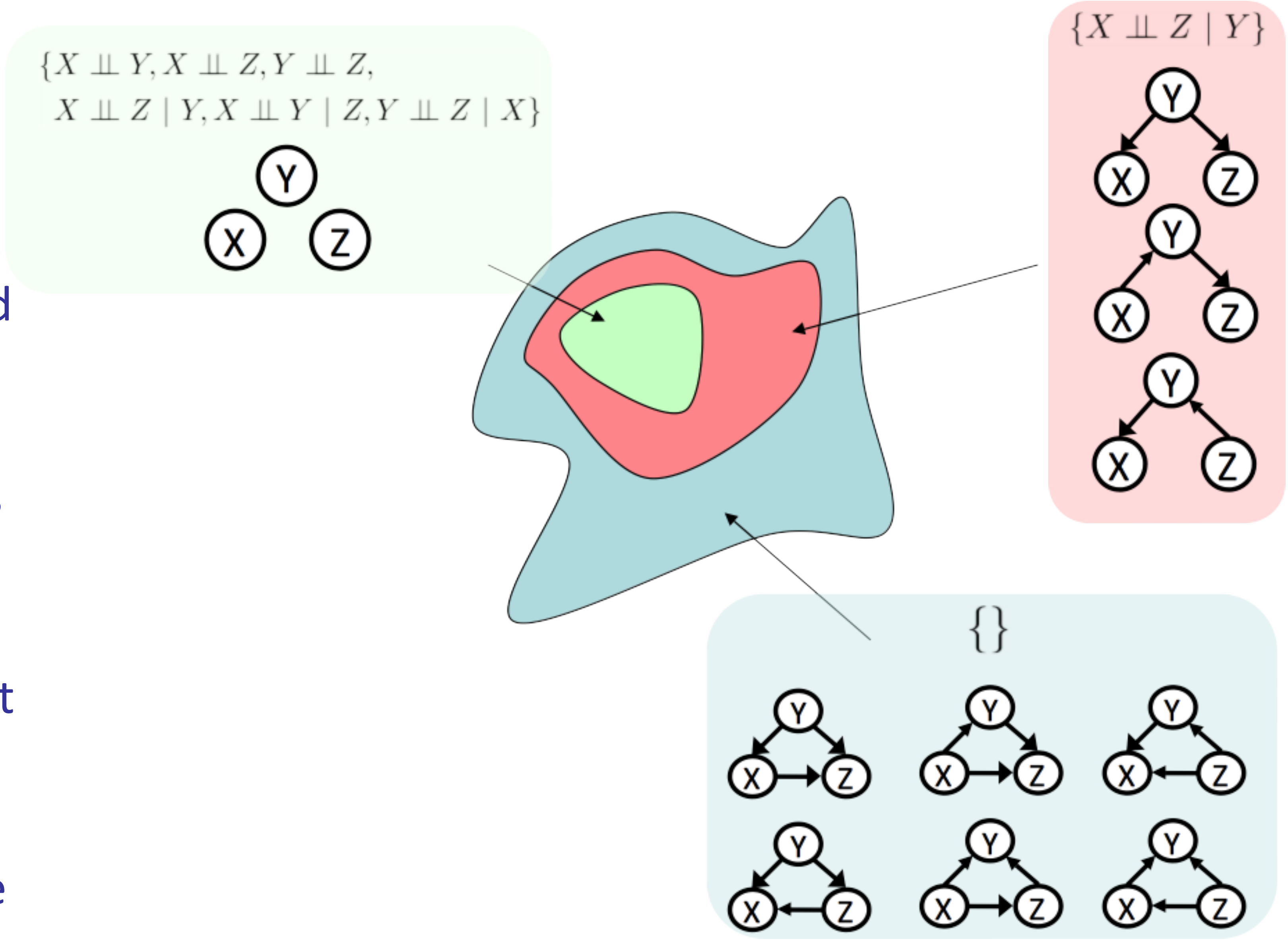
$$X_i \perp\!\!\!\perp X_j | \{X_{k_1}, ..., X_{k_n}\}$$

- This list determines the set of probability distributions that can be represented



Topology limits distributions

- Given some graph topology G, only certain joint distributions can be encoded
- The graph structure guarantees certain (conditional) independences
- (There might be more independence)
- Adding arcs increases the set of distributions, but has several costs
- Full conditioning can encode any distribution



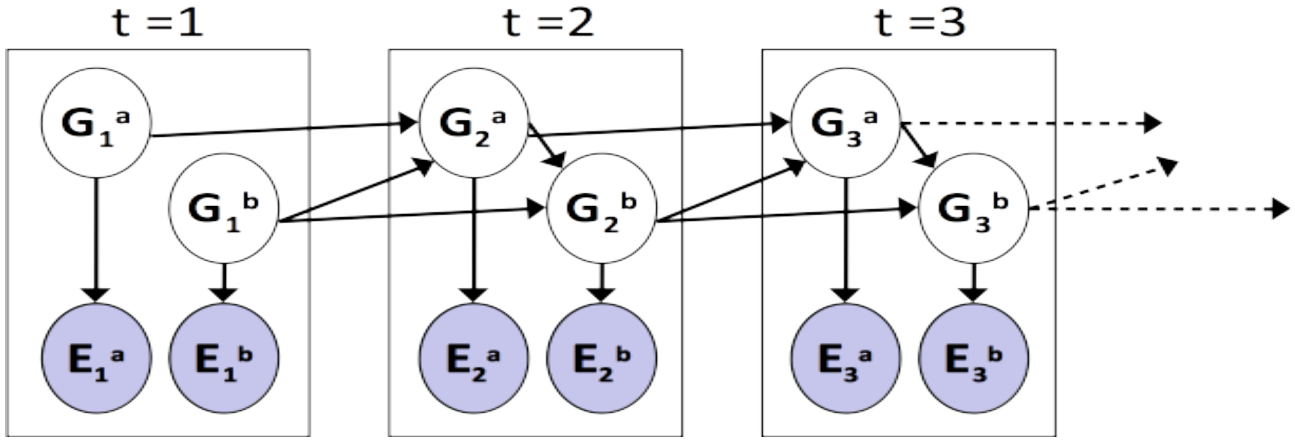
$\{X \perp\!\!\!\perp Y, X \perp\!\!\!\perp Z, Y \perp\!\!\!\perp Z,$
$X \perp\!\!\!\perp Z \mid Y, X \perp\!\!\!\perp Y \mid Z, Y \perp\!\!\!\perp Z \mid X\}$

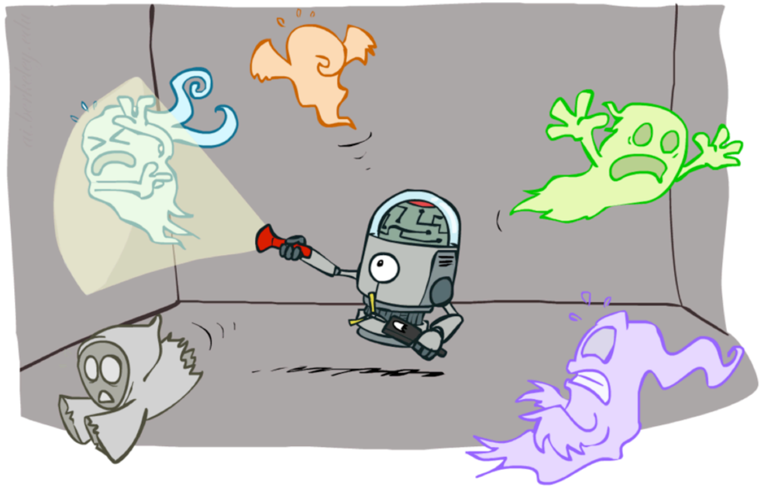
$\{X \perp\!\!\!\perp Z \mid Y\}$

$\{\}$

Dynamic Bayes Nets (DBNs)

- We want to track multiple variables over time, using multiple sources of evidence
- Idea: Repeat a fixed Bayes net structure at each time
- Variables from time *t* can condition on those from *t-1*





- Dynamic Bayes nets are a generalization of HMMs

DBN particle filters

- A particle is a complete sample for a time step
- **Initialize**: Generate prior samples for the t=1 Bayes net
  - Example particle: $\mathbf{G_1^a}$ = (3,3) $\mathbf{G_1^b}$ = (5,3)
- **Elapse time**: Sample a successor for each particle
  - Example successor: $\mathbf{G_2^a}$ = (2,3) $\mathbf{G_2^b}$ = (6,3)
- **Observe**: Weight each *entire* sample by the likelihood of the evidence conditioned on the sample
  - Likelihood: $P(\mathbf{E_1^a} \mid \mathbf{G_1^a}) * P(\mathbf{E_1^b} \mid \mathbf{G_1^b})$
- **Resample:** Select prior samples (tuples of values) in proportion to their likelihood

Video: Pacman sonar -- Ghost DBN

Inference

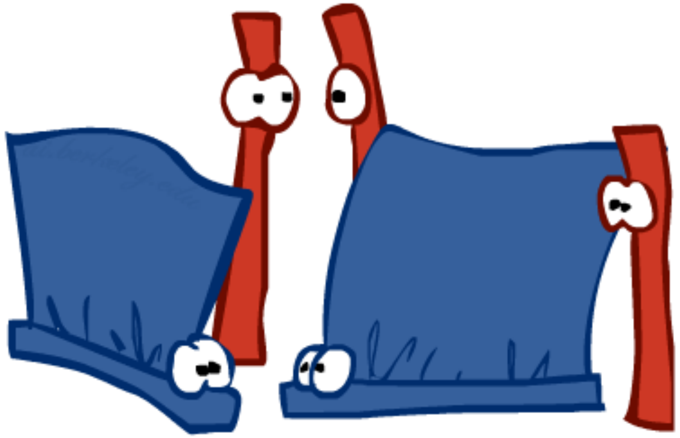
- Inference: calculating some useful quantity from a joint probability distribution

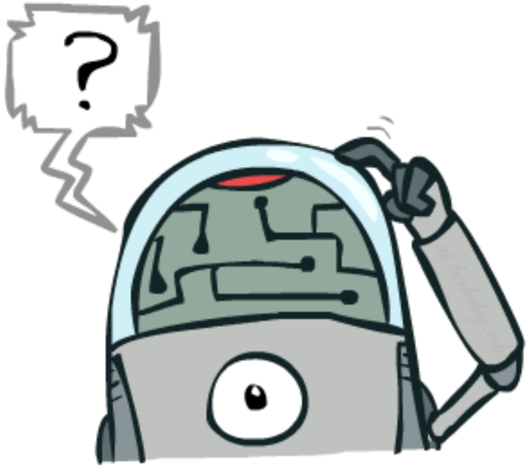
- Examples:
  - Posterior probability

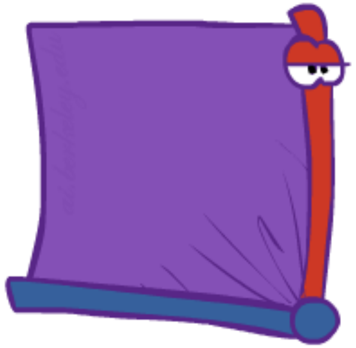
    $$P(Q|E_1 = e_1, \ldots E_k = e_k)$$

  - Most likely explanation:

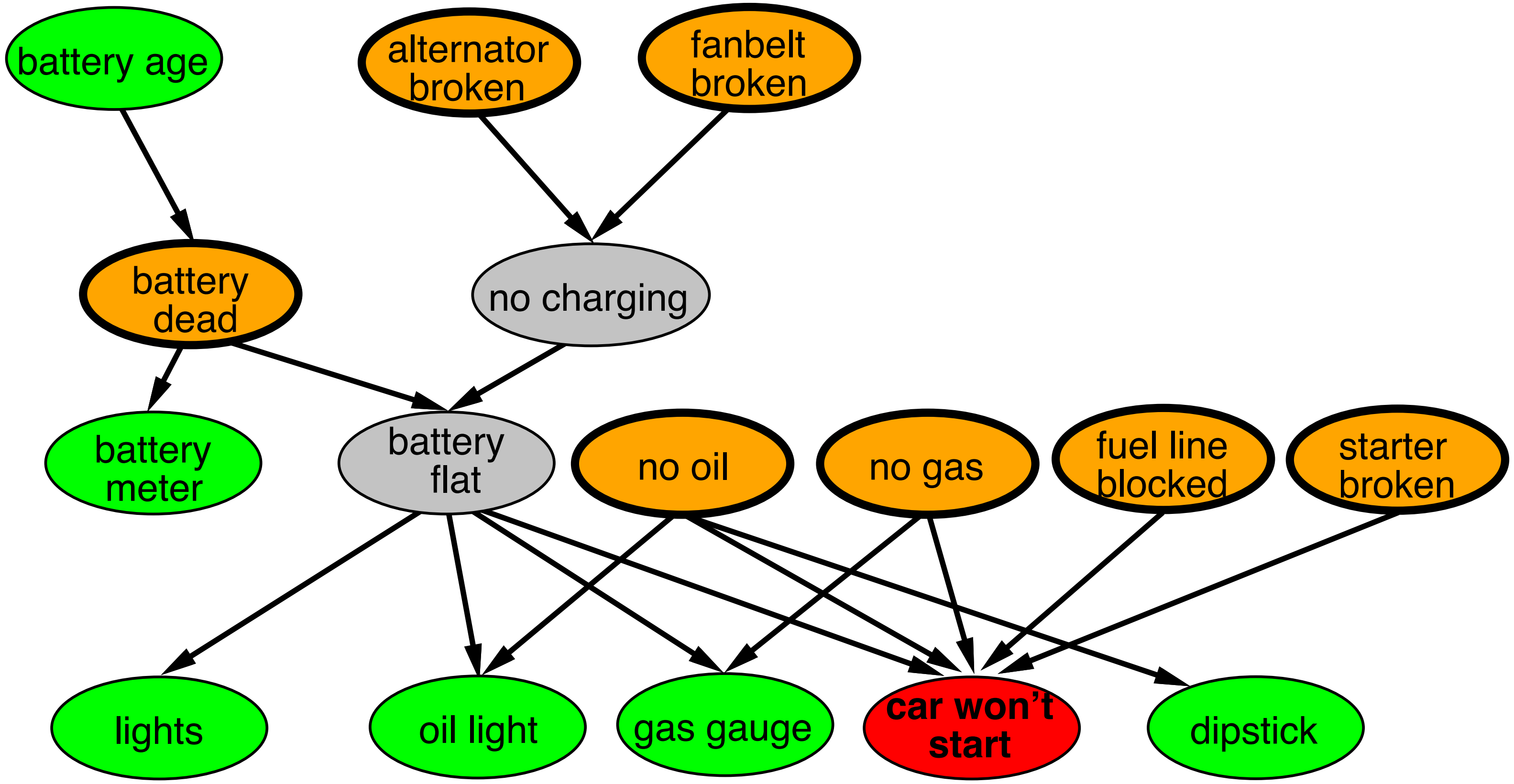
    $$\text{argmax}_q \; P(Q = q|E_1 = e_1 \ldots)$$







Bayes net for car diagnosis



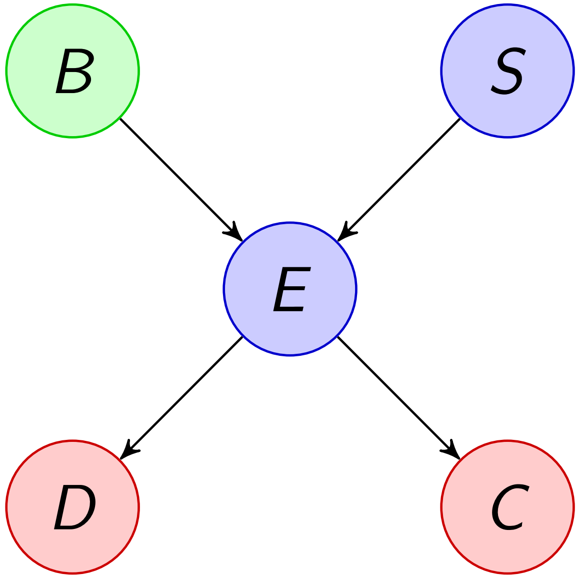
Initial evidence: car won’t start

Testable variables (green), “broken, so fix it” variables (orange)

Hidden variables (gray) ensure sparse structure, reduce parameters

Inference

Suppose we want to infer the distribution $P(B=true \mid D=true, C=true)$

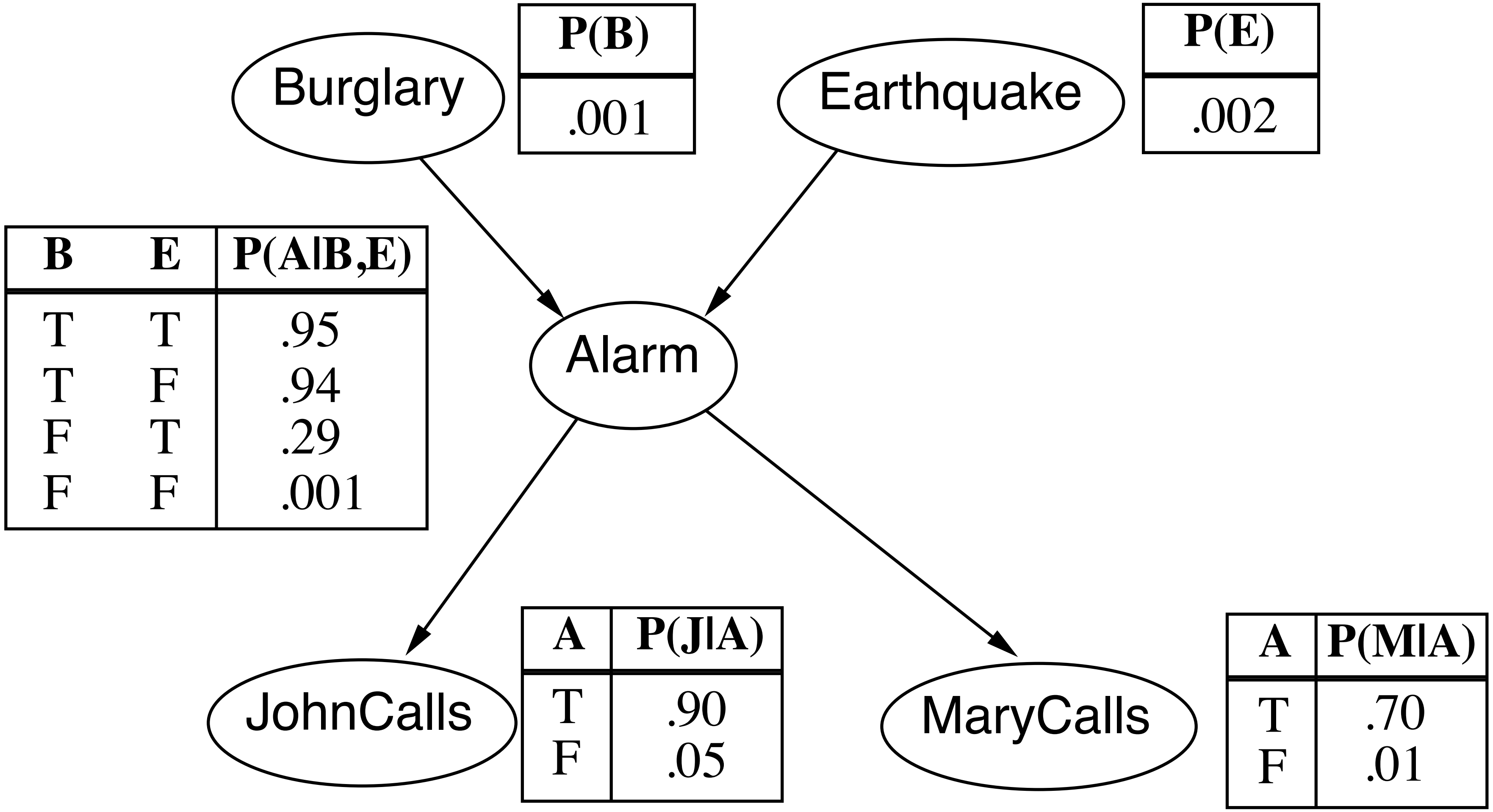


Query variables: B

Evidence variables: D, C

Hidden variables: S, E

Burglar alarm example



P(B)
.001

P(E)
.002

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

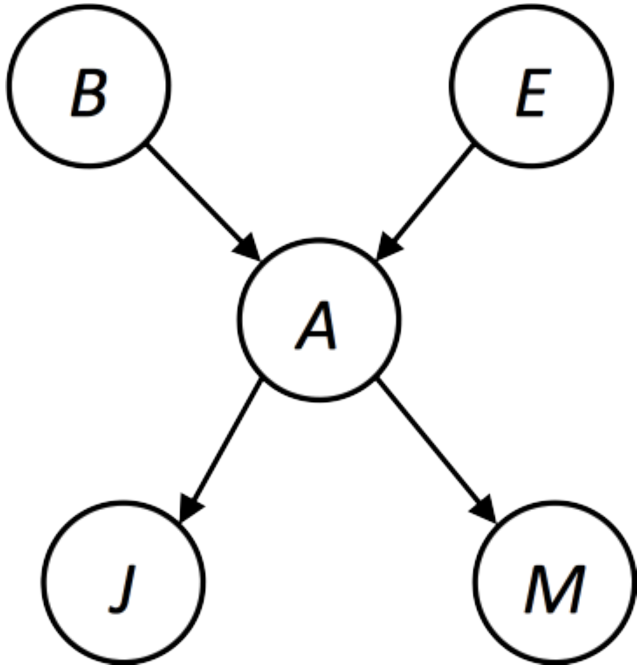
A	P(J|A)
T	.90
F	.05

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

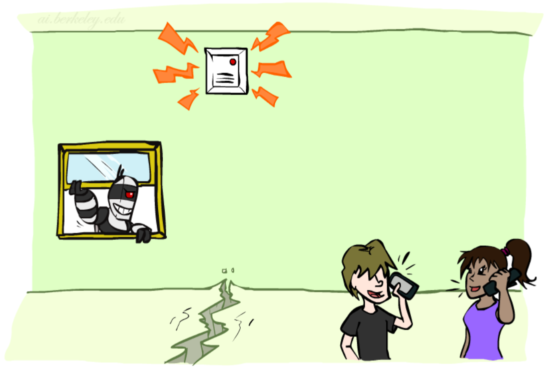
$$P(+b, -e, +a, -j, +m) =$$

Burglar alarm example

B	P(B)
+b	0.001
-b	0.999



E	P(E)
+e	0.002
-e	0.998



A	J	P(J|A)
+a	+j	0.9
+a	-j	0.1
-a	+j	0.05
-a	-j	0.95

A	M	P(M|A)
+a	+m	0.7
+a	-m	0.3
-a	+m	0.01
-a	-m	0.99

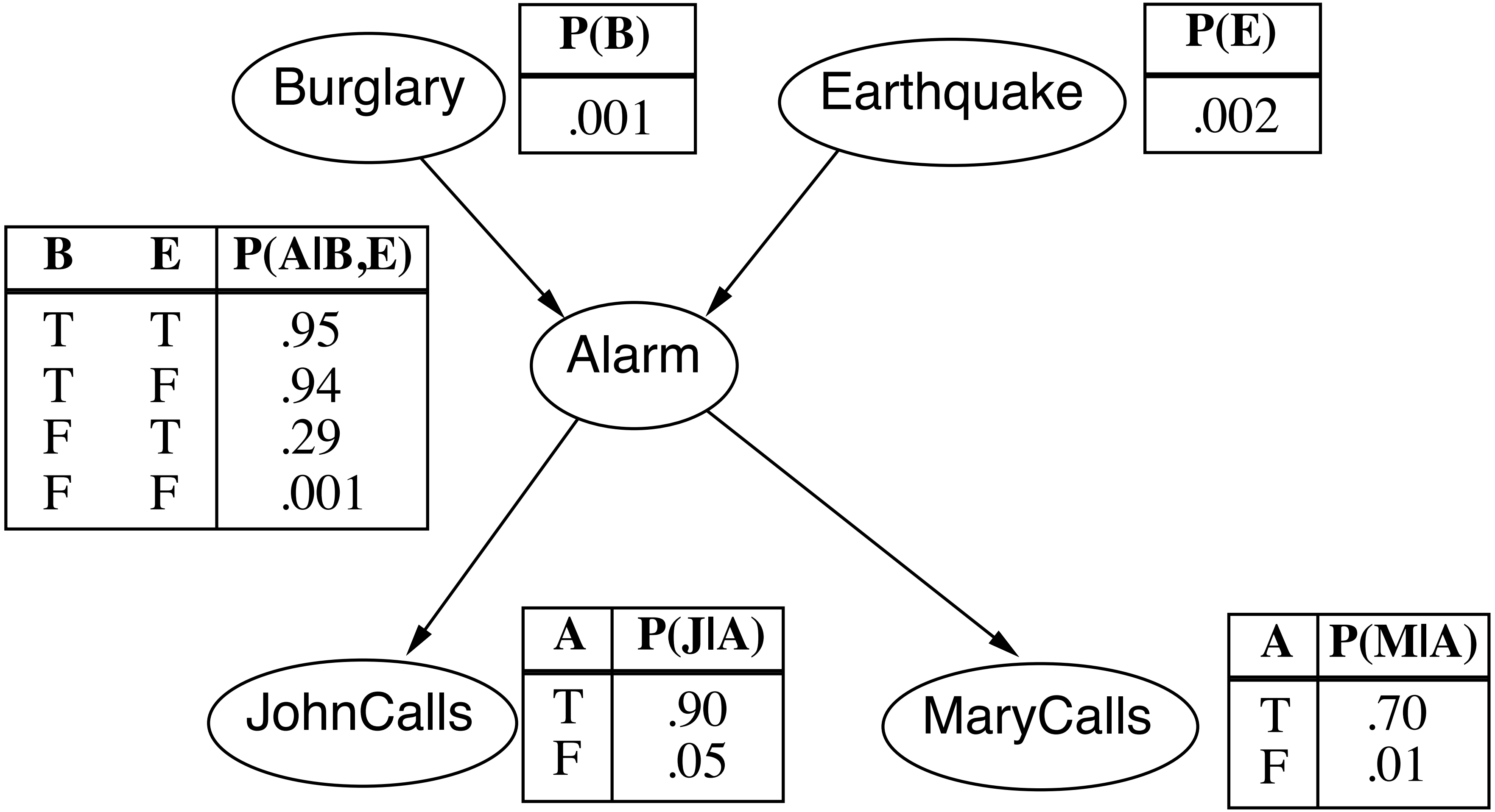
B	E	A	P(A|B,E)
+b	+e	+a	0.95
+b	+e	-a	0.05
+b	-e	+a	0.94
+b	-e	-a	0.06
-b	+e	+a	0.29
-b	+e	-a	0.71
-b	-e	+a	0.001
-b	-e	-a	0.999

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

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

$$0.001 \times 0.998 \times 0.94 \times 0.1 \times 0.7$$

Burglar alarm example



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

What if I want to calculate $P(B|j,m)$?

Inference by enumeration

General case:

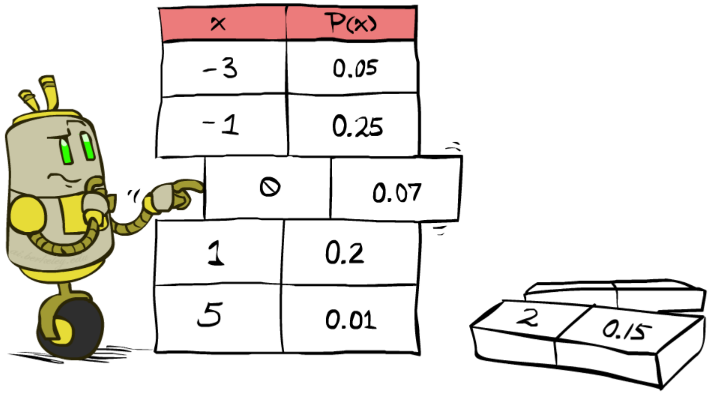
$$\left.\begin{array}{ll} \text{Evidence variables:} & E_1 \dots E_k = e_1 \dots e_k \\ \text{Query* variable:} & Q \\ \text{Hidden variables:} & H_1 \dots H_r \end{array}\right\} \begin{array}{l} X_1, X_2, \dots X_n \\ \textit{All variables} \end{array}$$

- We want:

$$P(Q|e_1 \dots e_k)$$

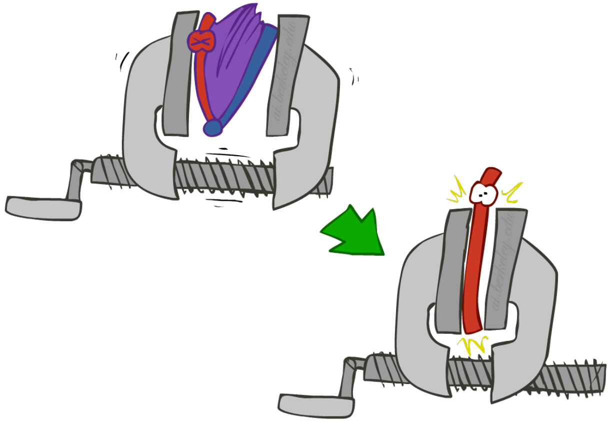
** Works fine with multiple query variables, too*

- Step 1: Select the entries consistent with the evidence



x	P(x)
-3	0.05
-1	0.25
0	0.07
1	0.2
5	0.01
2	0.15

- Step 2: Sum out H to get joint of query and evidence



$$P(Q, e_1 \dots e_k) = \sum_{h_1 \dots h_r} \underbrace{P(Q, h_1 \dots h_r, e_1 \dots e_k)}_{X_1, X_2, \dots X_n}$$

- Step 3: Normalize

$$\times \frac{1}{Z}$$

$$Z = \sum_q P(Q, e_1 \cdots e_k)$$

$$P(Q|e_1 \cdots e_k) = \frac{1}{Z} P(Q, e_1 \cdots e_k)$$

Inference by enumeration

P(W)?

P(W | winter)?

P(W | winter, hot)?

S	T	W	P
summer	hot	sun	0.30
summer	hot	rain	0.05
summer	cold	sun	0.10
summer	cold	rain	0.05
winter	hot	sun	0.10
winter	hot	rain	0.05
winter	cold	sun	0.15
winter	cold	rain	0.20

Inference by enumeration

Slightly intelligent way to sum out variables from the joint without actually constructing its explicit representation

$P(B|j,m)$

$= P(B,j,m)/P(j,m)$

$= \alpha\ P(B,j,m)$

$$= \alpha \sum_{e} \sum_{a} P(B,e,a,j,m)$$

Inference by enumeration

Slightly intelligent way to sum out variables from the joint without actually constructing its explicit representation

Rewrite full joint entries using product of CPT entries:

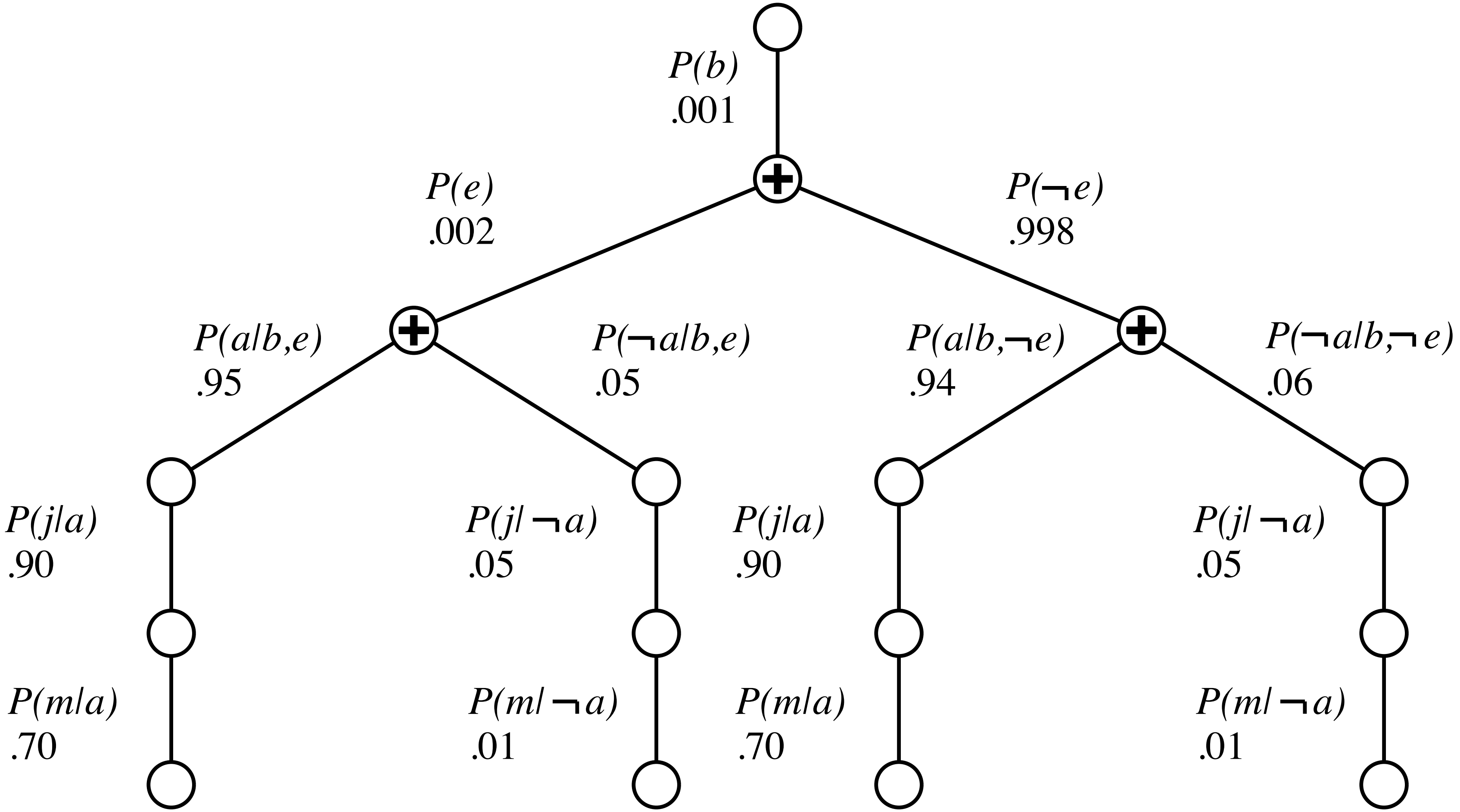
$P(B|j,m)$

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

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

Recursive depth-first enumeration: $O(n)$ space, $O(d^n)$ time

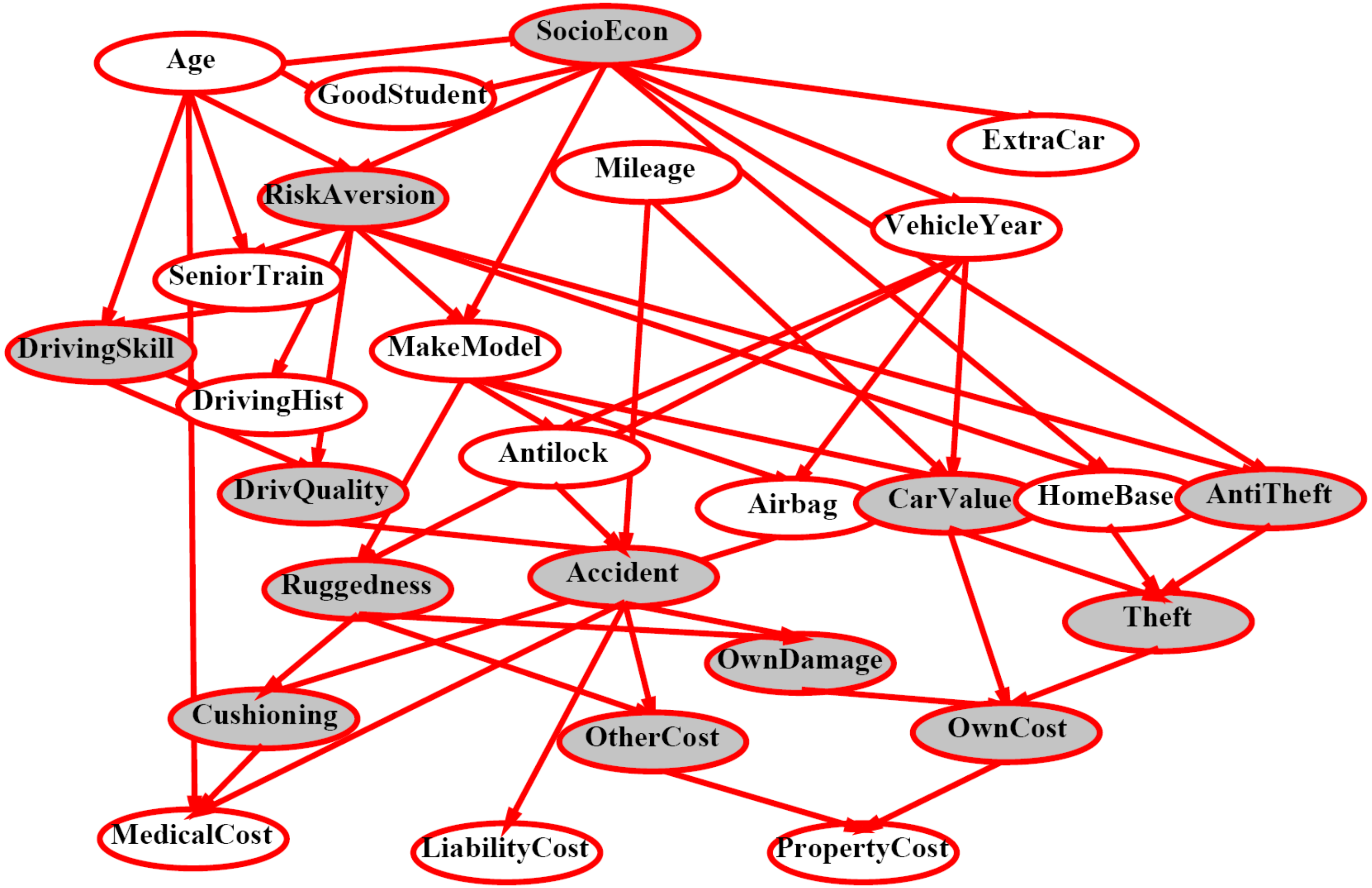
Evaluation tree



Enumeration is inefficient: repeated computation

e.g., computes $P(j|a)P(m|a)$ for each value of e

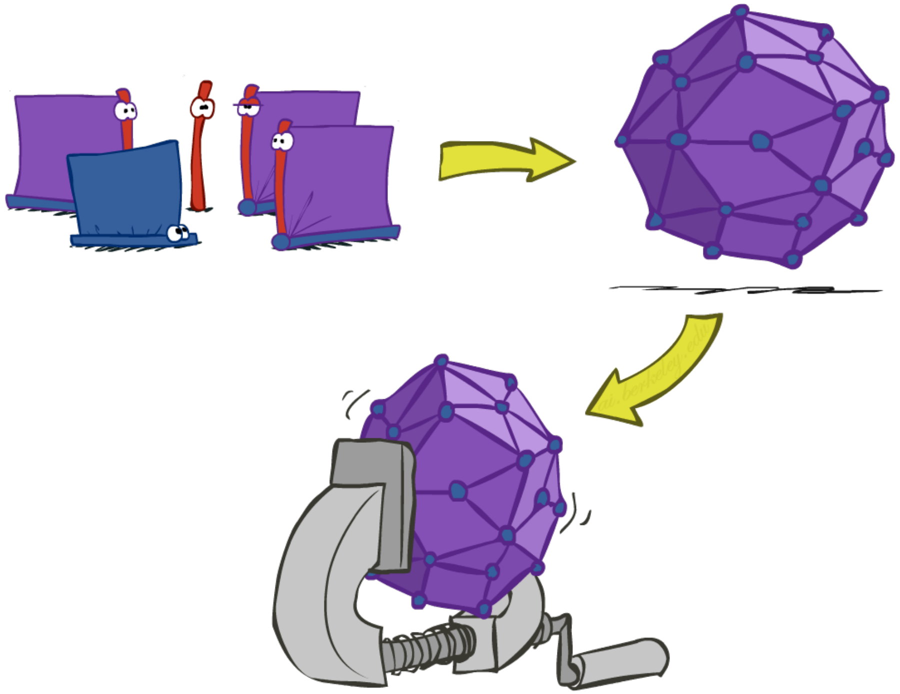
Inference by enumeration?



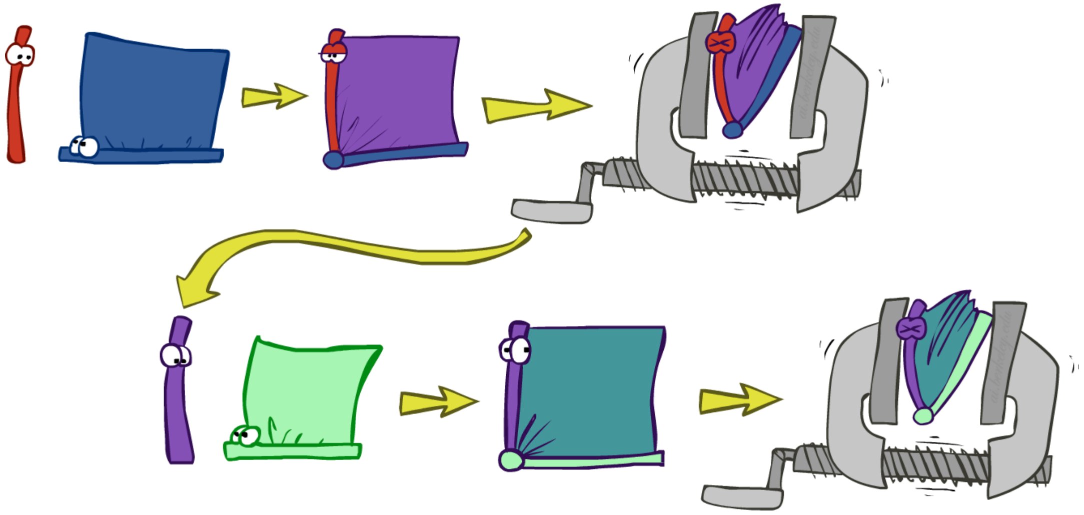
$$P(Antilock|observed\ variables) = ?$$

Inference by enumeration vs. variable elimination

- Why is inference by enumeration so slow?
  - You join up the whole joint distribution before you sum out the hidden variables



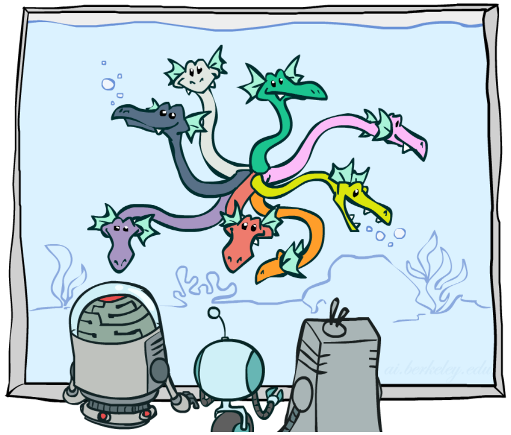
- Idea: interleave joining and marginalizing!
  - Called "Variable Elimination"
  - Sum right-to-left, storing intermediate results (*factors*) to avoid recomputation
  - Still NP-hard, but usually much faster than inference

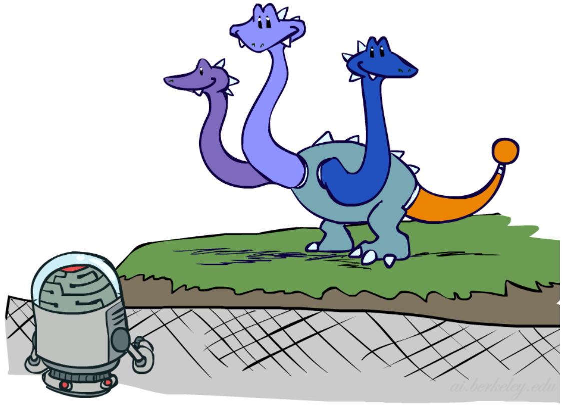


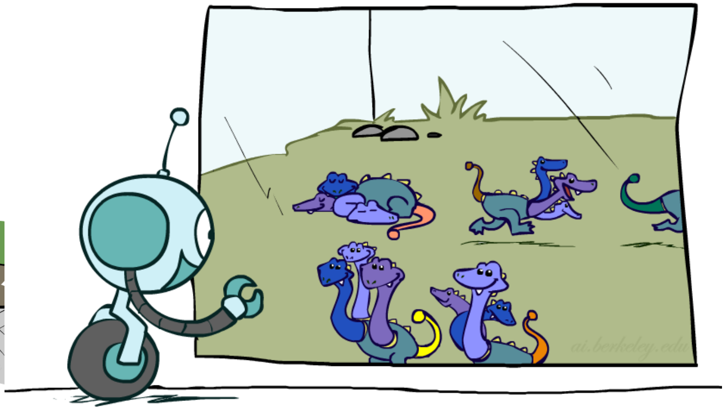
  - First we'll need some new notation: factors

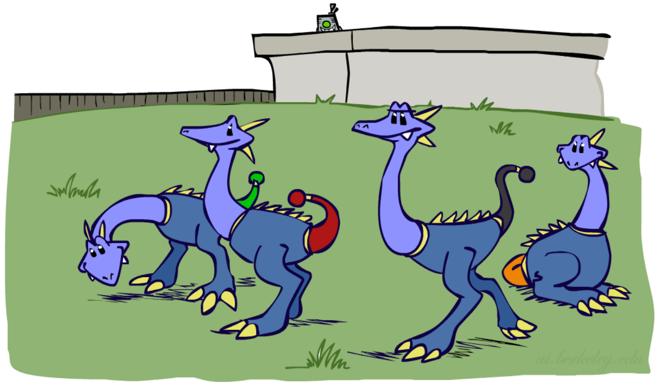
Factors

- In general, when we write $P(Y_1 \ldots Y_N \mid X_1 \ldots X_M)$
  - It is a “factor,” a multi-dimensional array
  - Its values are $P(y_1 \ldots y_N \mid x_1 \ldots x_M)$
  - Any assigned (=lower-case) X or Y is a dimension missing (selected) from the array



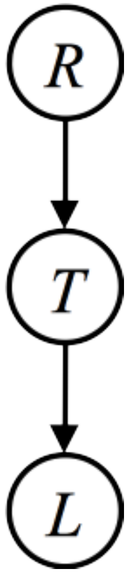






Example: Traffic domain

- Random Variables
  - R: Raining
  - T: Traffic
  - L: Late for class!



$$P(R)$$

+r	0.1
-r	0.9

$$P(T|R)$$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$$P(L|T)$$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

$$P(L) = ?$$

$$= \sum_{r,t} P(r, t, L)$$

$$= \sum_{r,t} P(r)P(t|r)P(L|t)$$

Inference by enumeration: Procedural outline

- Track objects called factors
- Initial factors are local CPTs (one per node)

$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

- Any known values are selected
  - E.g. if we know $L = +\ell$, the initial factors are

$P(R)$

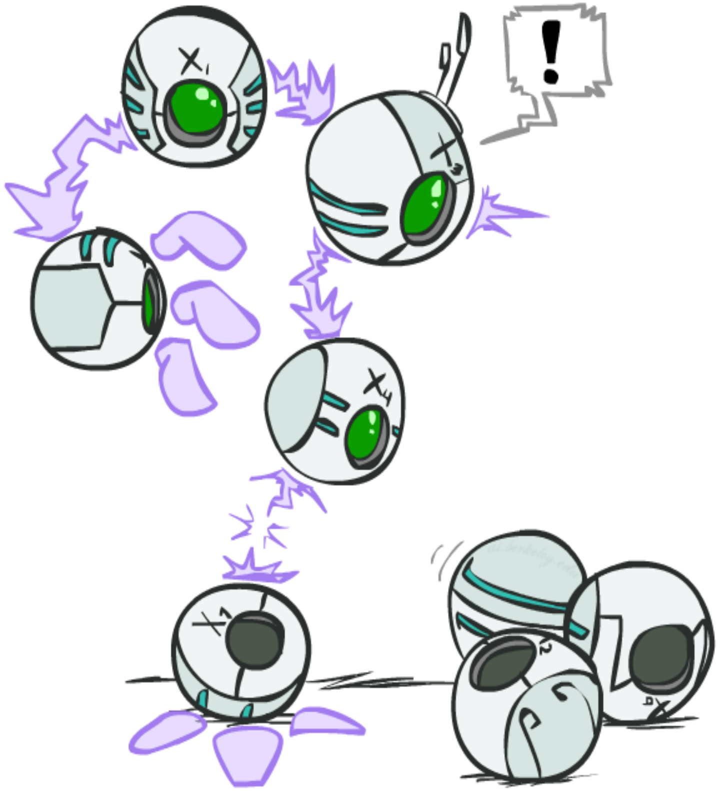
+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(+\ell|T)$

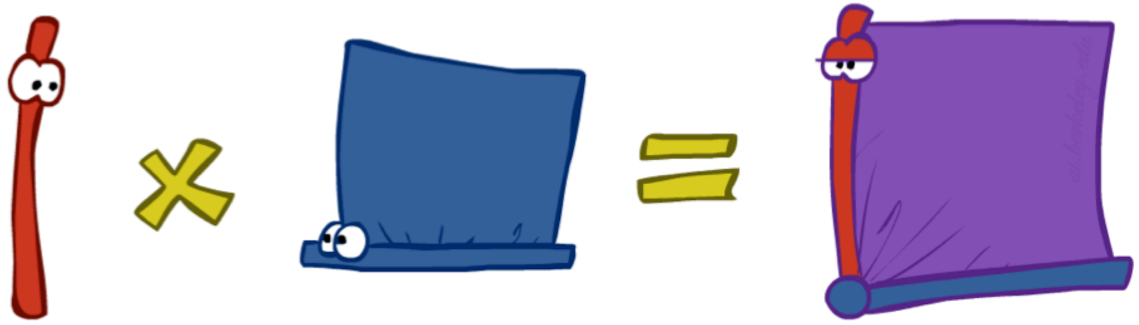
+t	+l	0.3
-t	+l	0.1



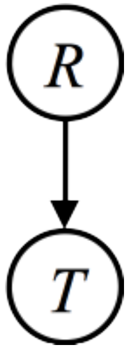
- Procedure: Join all factors, then eliminate all hidden variables

Operation 1: Join factors

- First basic operation: joining factors
- Combining factors:
  - Just like a database join
  - Get all factors over the joining variable
  - Build a new factor over the union of the variables involved



- Example: Join on R



$$P(R) \times P(T|R) \longrightarrow P(R,T)$$

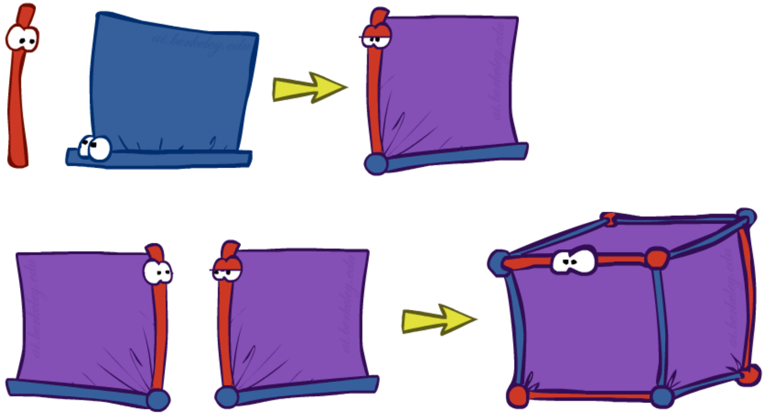
+r	0.1
-r	0.9

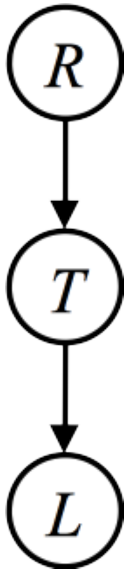
+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

- Computation for each entry: pointwise products $\forall r, t : \quad P(r,t) = P(r) \cdot P(t|r)$

Example: Multiple joins





$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

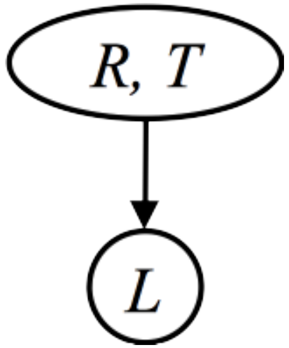
Join R

$P(R,T)$

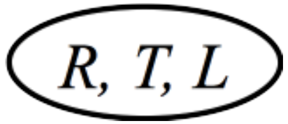
+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9



Join T



$P(R,T,L)$

+r	+t	+l	0.024
+r	+t	-l	0.056
+r	-t	+l	0.002
+r	-t	-l	0.018
-r	+t	+l	0.027
-r	+t	-l	0.063
-r	-t	+l	0.081
-r	-t	-l	0.729

Operation 2: Eliminate

- Second basic operation: marginalization
- Take a factor and sum out a variable
  - Shrinks a factor to a smaller one
  - A projection operation
- Example:

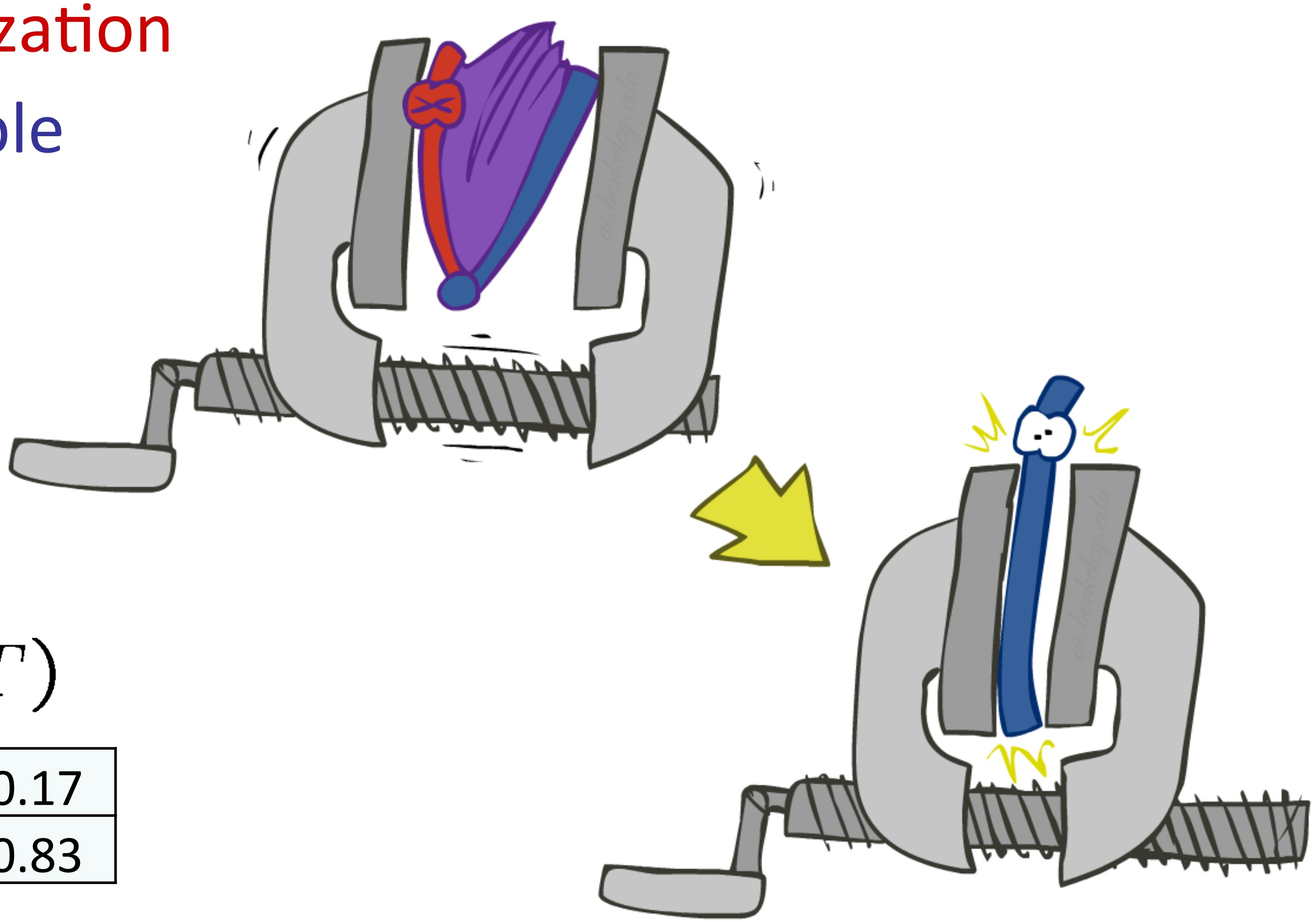
$P(R,T)$

+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

sum R →

$P(T)$

+t	0.17
-t	0.83



Multiple elimination

R, T, L

$P(R, T, L)$

+r	+t	+l	0.024
+r	+t	-l	0.056
+r	-t	+l	0.002
+r	-t	-l	0.018
-r	+t	+l	0.027
-r	+t	-l	0.063
-r	-t	+l	0.081
-r	-t	-l	0.729

Sum out R

T, L

$P(T, L)$

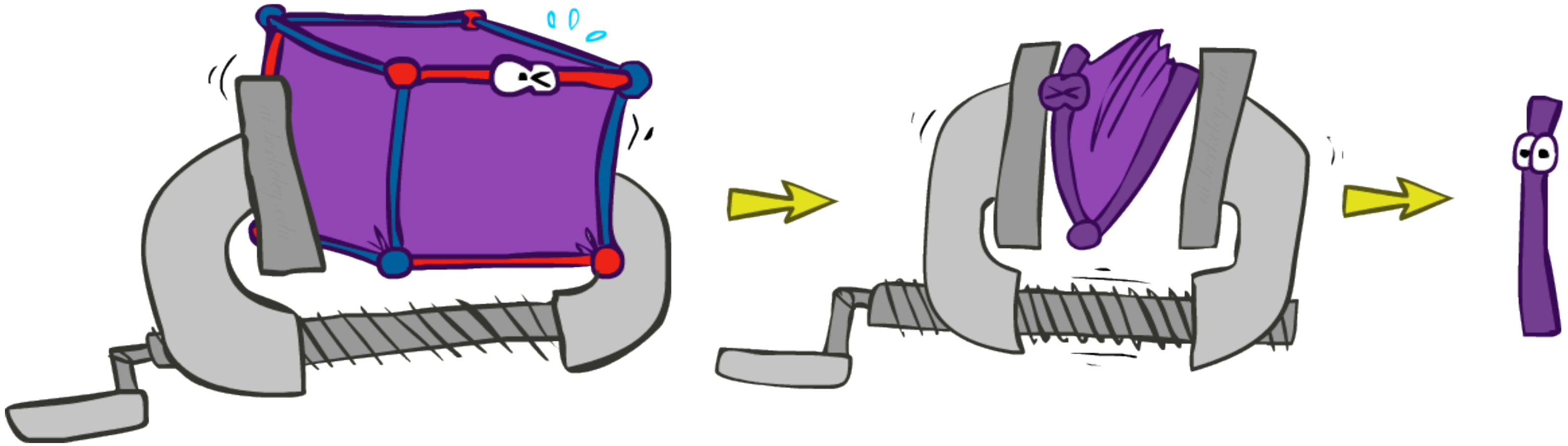
+t	+l	0.051
+t	-l	0.119
-t	+l	0.083
-t	-l	0.747

Sum out T

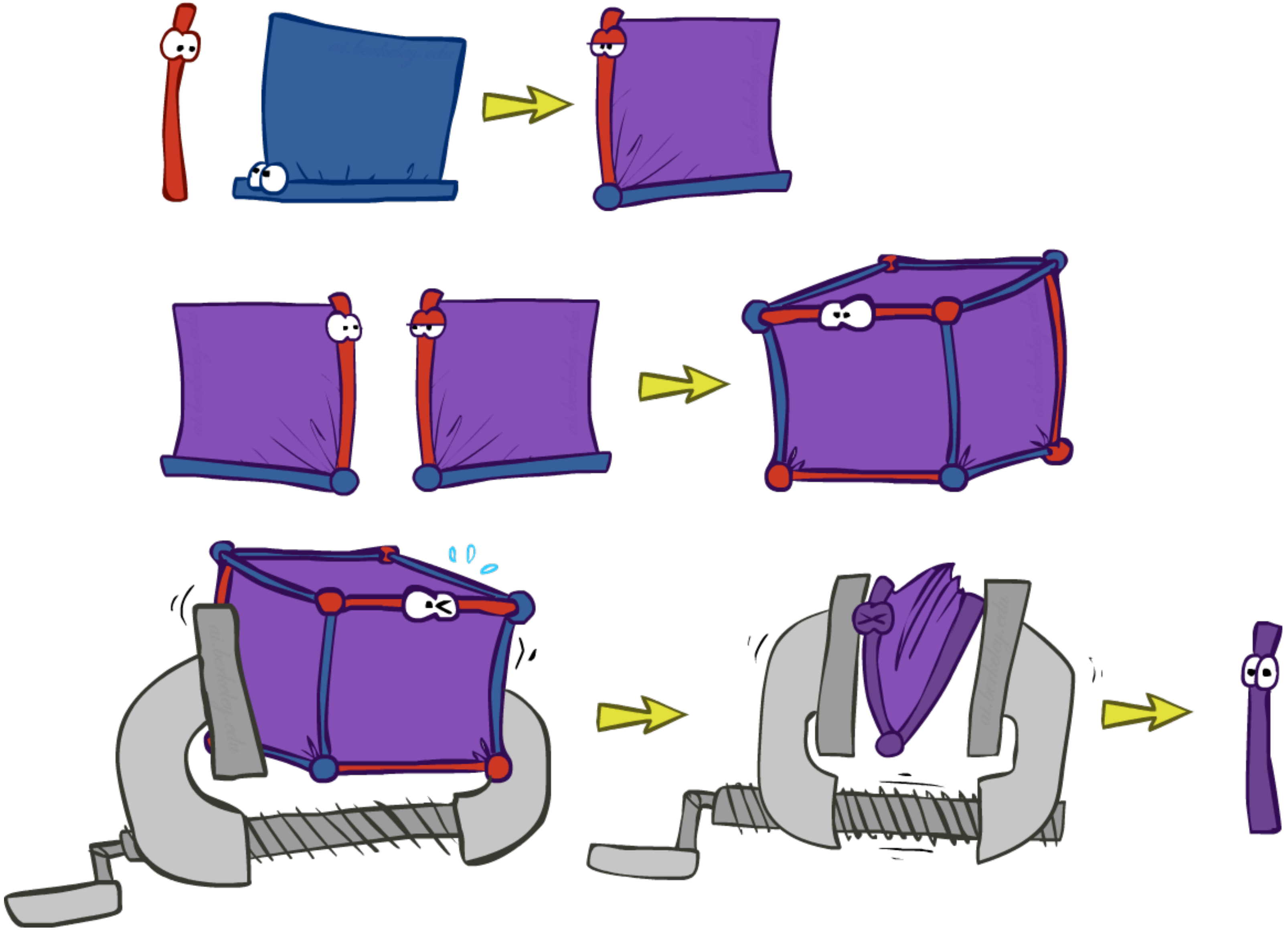
L

$P(L)$

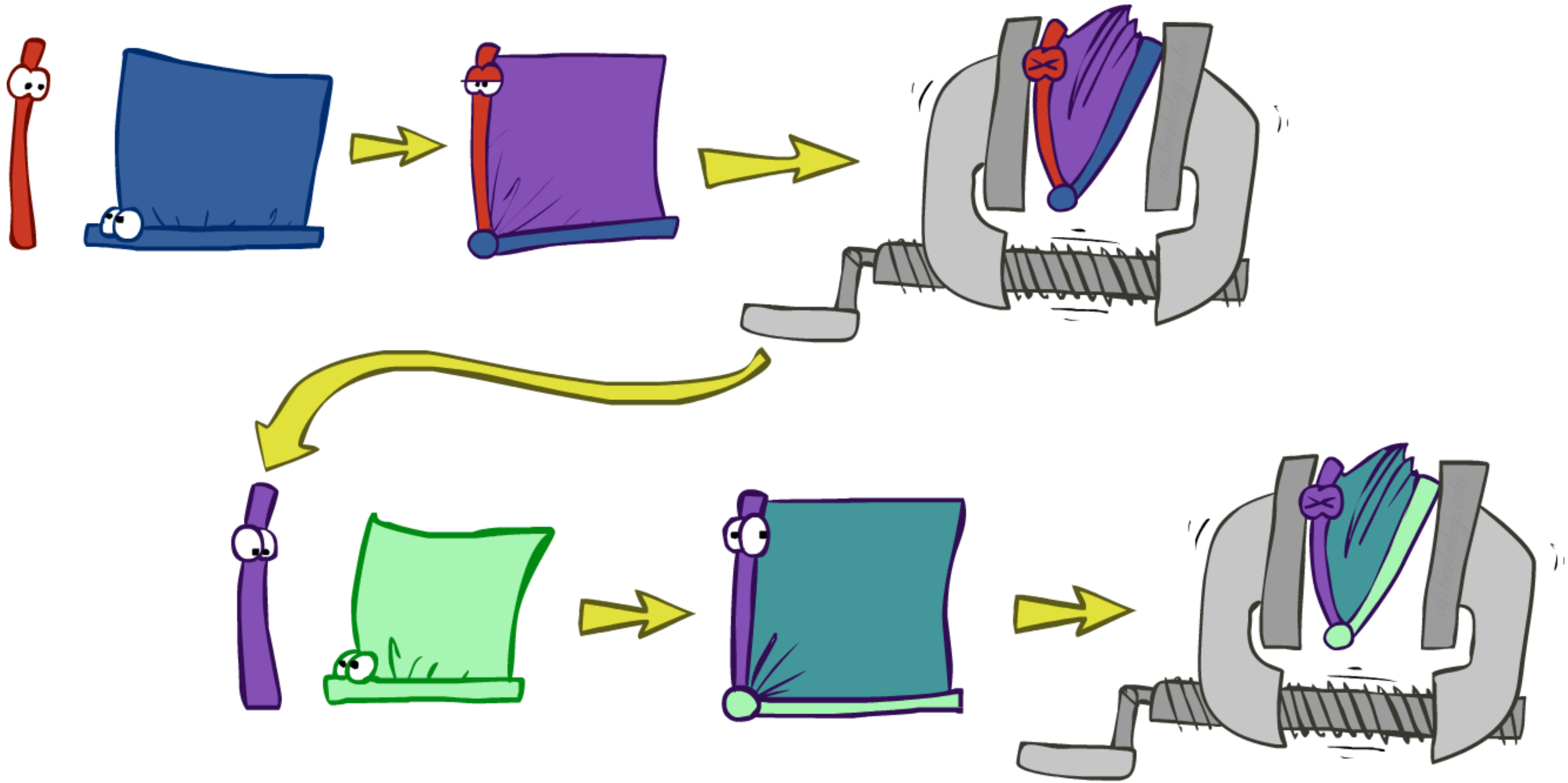
+l	0.134
-l	0.886



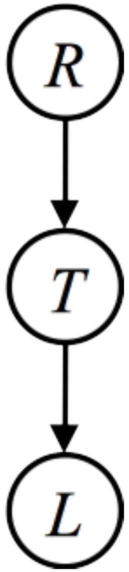
Thus far: Multiple join, multiple eliminate (= inference by enumeration)



Marginalizing early (= variable elimination)



Traffic domain



$$P(L) = \ ?$$

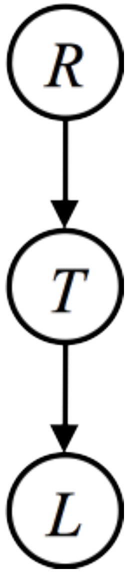
- Inference by Enumeration

$$= \underbrace{\sum_t \underbrace{\sum_r \underbrace{P(L|t)\underbrace{P(r)P(t|r)}_{\text{Join on r}}}_{\text{Join on t}}}_{\text{Eliminate r}}}_{\text{Eliminate t}}$$

- Variable Elimination

$$= \underbrace{\sum_t \underbrace{P(L|t)\underbrace{\sum_r \underbrace{P(r)P(t|r)}_{\text{Join on r}}}_{\text{Eliminate r}}}_{\text{Join on t}}}_{\text{Eliminate t}}$$

Variable elimination



$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

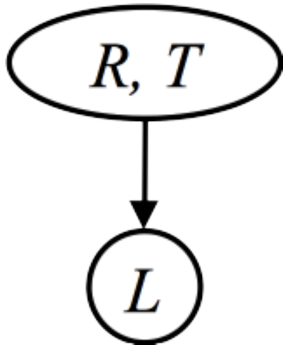
$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

Join R →

$P(R,T)$

+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81



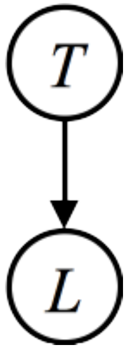
$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

Sum out R →

$P(T)$

+t	0.17
-t	0.83



$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

Join T →

$P(T,L)$

+t	+l	0.051
+t	-l	0.119
-t	+l	0.083
-t	-l	0.747

Sum out T →

$P(L)$

+l	0.134
-l	0.866

Evidence

- If evidence, start with factors that select that evidence
  - No evidence uses these initial factors:

$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

  - Computing $P(L|+r)$, the initial factors become:

$P(+r)$

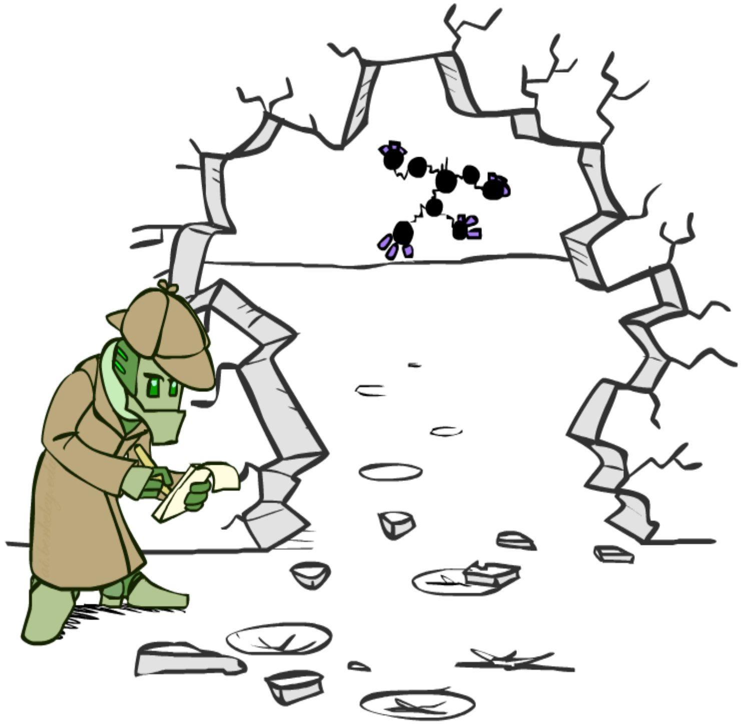
+r	0.1

$P(T|+r)$

+r	+t	0.8
+r	-t	0.2

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9



- We eliminate all variables other than query + evidence

Evidence

- Result will be a selected joint of query and evidence
  - E.g. for P(L | +r), we would end up with:

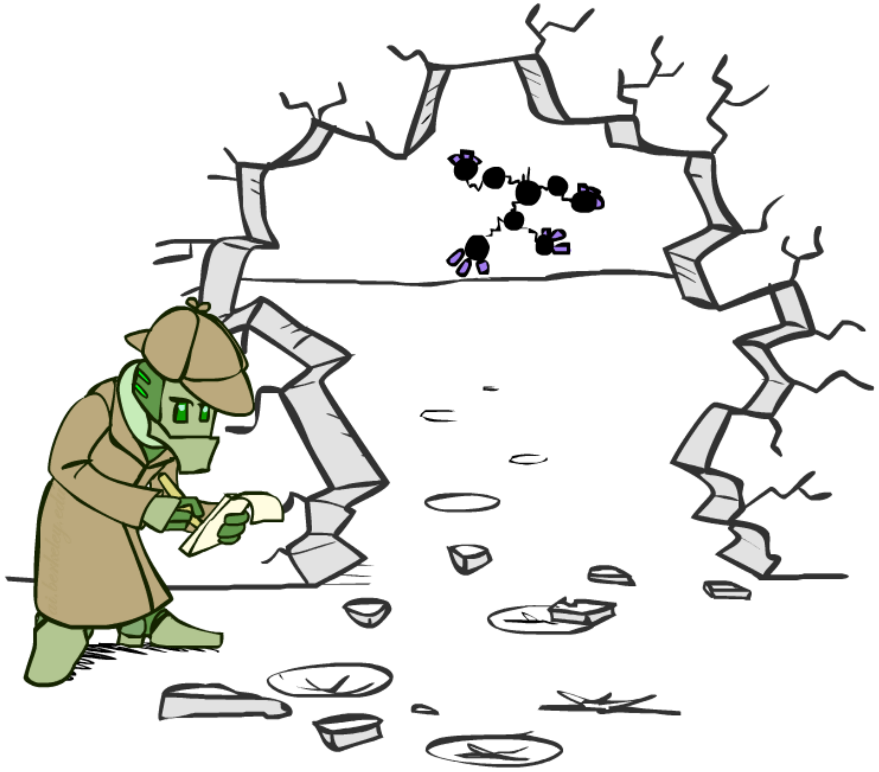
$P(+r, L)$

+r	+l	0.026
+r	-l	0.074

Normalize

$P(L| + r)$

+l	0.26
-l	0.74



- To get our answer, just normalize this!

- That ’s it!

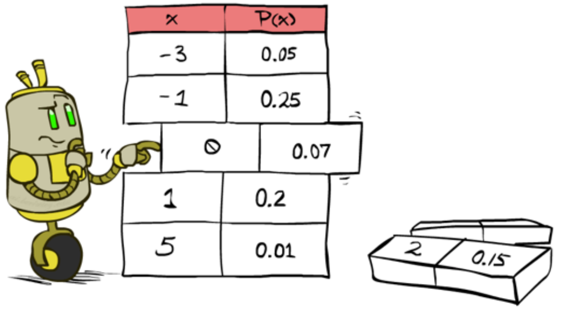
General variable elimination

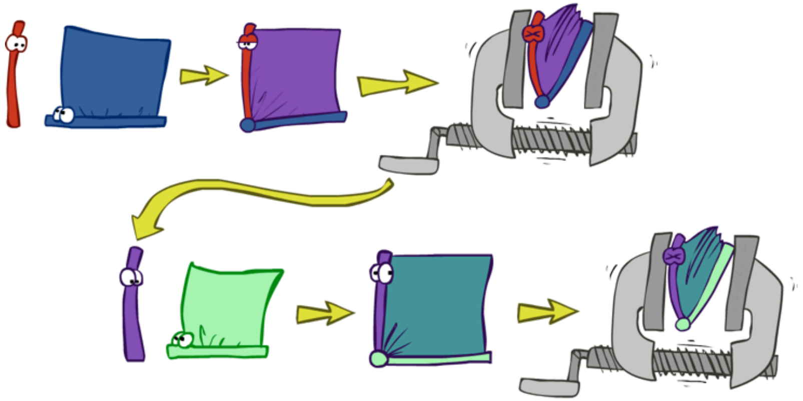
- Query: $P(Q|E_1 = e_1, \dots E_k = e_k)$

- Start with initial factors:
  - Local CPTs (but instantiated by evidence)

- While there are still hidden variables (not Q or evidence):
  - Pick a hidden variable H
  - Join all factors mentioning H
  - Eliminate (sum out) H

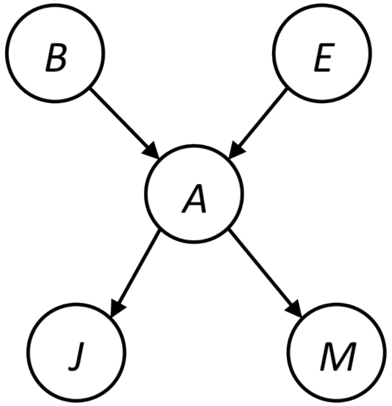
- Join all remaining factors and normalize





$\times \frac{1}{Z}$

Example



$$P(B|j,m) \propto P(B,j,m)$$

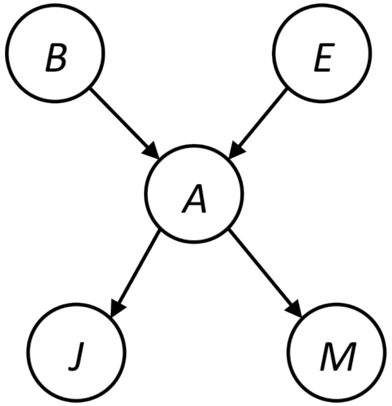
$$\boxed{P(B) \qquad P(E) \qquad P(A|B,E) \qquad P(j|A) \qquad P(m|A)}$$

Choose A

$$\begin{matrix} P(A|B,E) \\ P(j|A) \\ P(m|A) \end{matrix} \xrightarrow{\times} P(j,m,A|B,E) \xrightarrow{\sum} P(j,m|B,E)$$

$$\boxed{P(B) \qquad P(E) \qquad P(j,m|B,E)}$$

Example



$$\boxed{P(B) \qquad P(E) \qquad P(j,m|B,E)}$$

Choose E

$$\begin{matrix} P(E) \\ P(j,m|B,E) \end{matrix} \xrightarrow{\times} P(j,m,E|B) \xrightarrow{\sum} P(j,m|B)$$

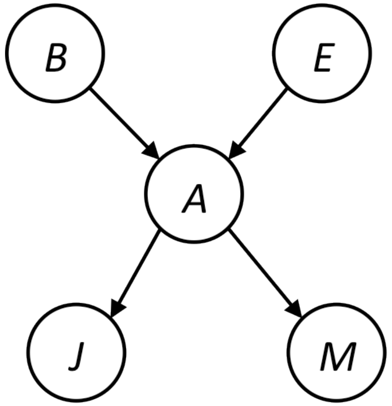
$$\boxed{P(B) \qquad P(j,m|B)}$$

Finish with B

$$\begin{matrix} P(B) \\ P(j,m|B) \end{matrix} \xrightarrow{\times} P(j,m,B) \xrightarrow{\text{Normalize}} P(B|j,m)$$

Same example in equations

$$P(B|j,m) \propto P(B,j,m)$$



$$\boxed{P(B) \qquad P(E) \qquad P(A|B,E) \qquad P(j|A) \qquad P(m|A)}$$

$$\begin{aligned}
P(B|j,m) &\propto P(B,j,m) \\
&= \sum_{e,a} P(B,j,m,e,a) \\
&= \sum_{e,a} P(B)P(e)P(a|B,e)P(j|a)P(m|a) \\
&= \sum_{e} P(B)P(e)\sum_{a} P(a|B,e)P(j|a)P(m|a) \\
&= \sum_{e} P(B)P(e)f_1(B,e,j,m) \\
&= P(B)\sum_{e} P(e)f_1(B,e,j,m) \\
&= P(B)f_2(B,j,m)
\end{aligned}$$

marginal can be obtained from joint by summing out

use Bayes' net joint distribution expression

use x*(y+z) = xy + xz

joining on a, and then summing out gives f_1

use x*(y+z) = xy + xz

joining on e, and then summing out gives f_2

All we are doing is exploiting uwy + uwz + uxy + uxz + vwy + vwz + vxy +vxz = (u+v)(w+x)(y+z) to improve computational efficiency!

Variable ordering

$$P(B|j,m) = \alpha P(B)\sum_{e} P(e)\sum_{a} P(a \mid B,e)P(j \mid a)P(m \mid a)$$

$$P(B|j,m) = \alpha P(B)\sum_{a} P(j \mid a)P(m \mid a)\sum_{e} P(e)P(a \mid B,e)$$

Complexity depends on factor size

Irrelevant variables

Consider the query $P(JohnCalls|Burglary = true)$

$$P(J \mid b) = \alpha P(b) \sum_{e} P(e) \sum_{a} P(a \mid b, e) P(J \mid a) \sum_{m} P(m \mid a)$$

Sum over m is identically 1; M is *irrelevant* to the query

Thm 1: Y is irrelevant unless $Y \in Ancestors(Q \cup E)$

Here, $Q = \{JohnCalls\}$, $E = \{Burglary\}$, and $Ancestors(Q \cup E) = \{Alarm, Earthquake\}$

so $MaryCalls$ is irrelevant

VE: Computation and space complexity

- The computational and space complexity of variable elimination is determined by the largest factor
- The elimination ordering can greatly affect the size of the largest factor
  - E.g., 2^n vs. 2
- Does there always exist an ordering that only results in small factors?
  - No!

Complexity of exact inference

Complexity in *polytrees* is linear in number of variables

A polytree has no *undirected* cycles

In general, inference in Bayesian networks is *NP hard*

In worst case, it is probably exponential

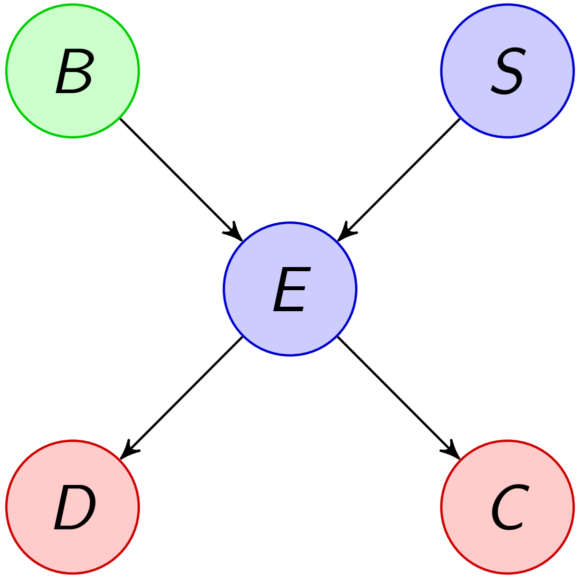
Variable elimination algorithm relies on a heuristic ordering of variables to eliminate in sequence

Often linear, sometimes exponential

Belief propagation propagates ``messages'' through the network: Linear for polytrees, not exact for other graphs

Approximate inference

Suppose we want to infer the distribution $P(B{=}true \mid D{=}true, C{=}true)$

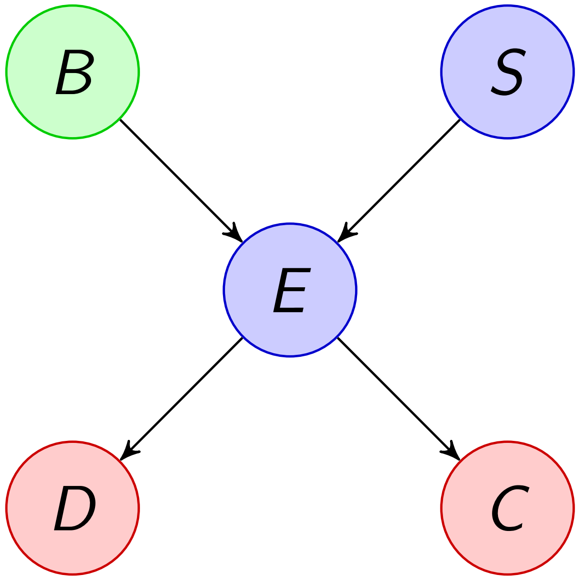


Query variables: B

Evidence variables: D, C

Hidden variables: S, E

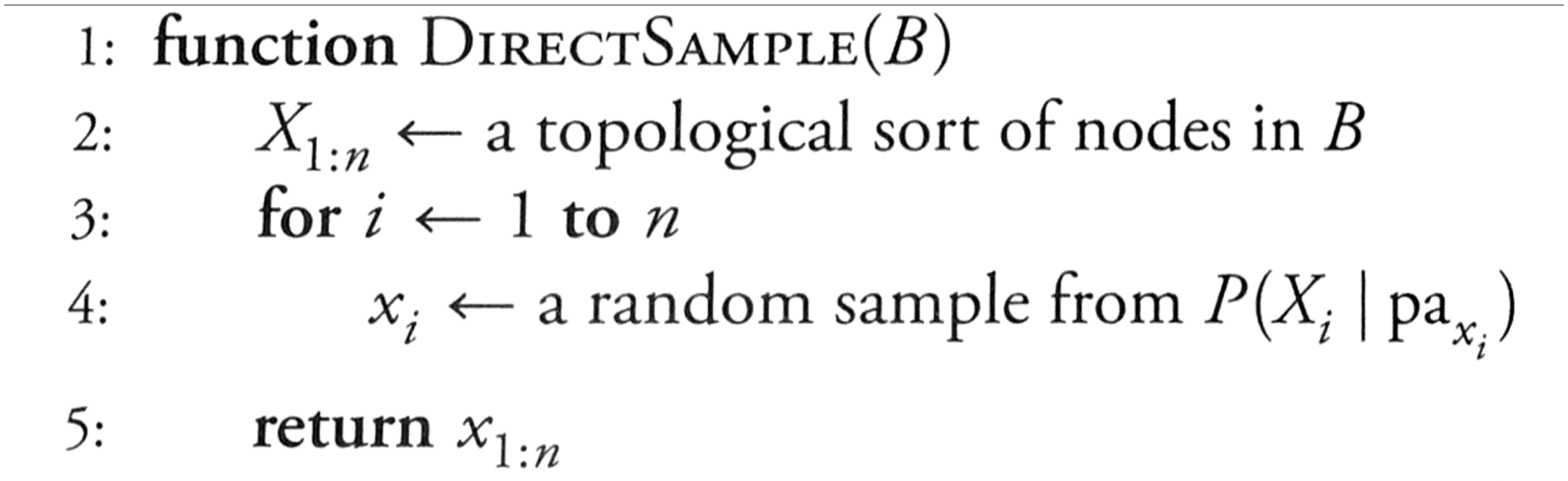
Topological sort



List the nodes in order

If there is an edge $\mathrm{A} \rightarrow \mathrm{B}$, then A comes before B in the list

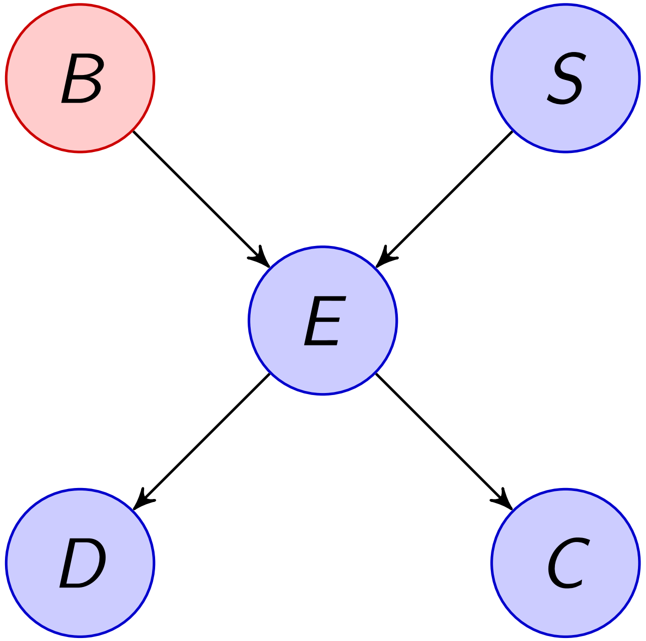
Approximate inference through sampling



```
function DirectSample($B$)
    $X_{1:n} \leftarrow$ a topological sort of nodes in $B$
    for $i \leftarrow 1$ to $n$
        $x_i \leftarrow$ a random sample from $P(X_i \mid \mathrm{pa}_{x_i})$
    return $x_{1:n}$
```

Direct sampling in a Bayes Net

Approximate inference through sampling (Direct sampling)

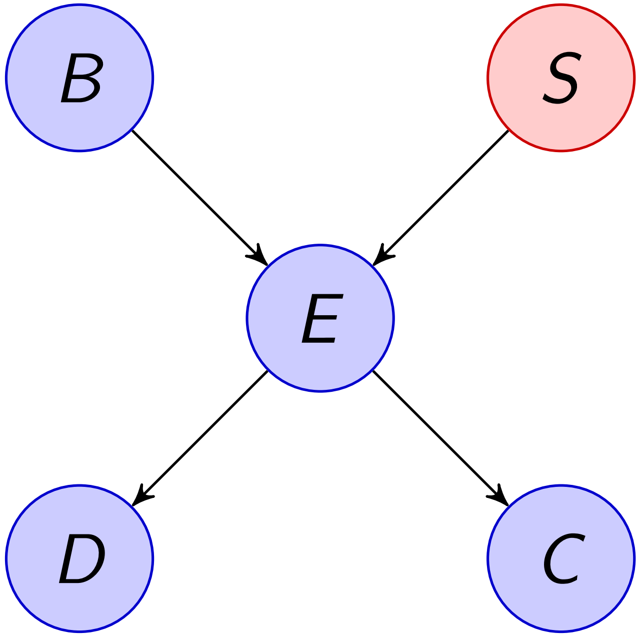


B	S	E	D	C
1				

In topological order

Sample from the condition probability distribution of X, given the sampled parent values

Approximate inference through sampling (Direct sampling)

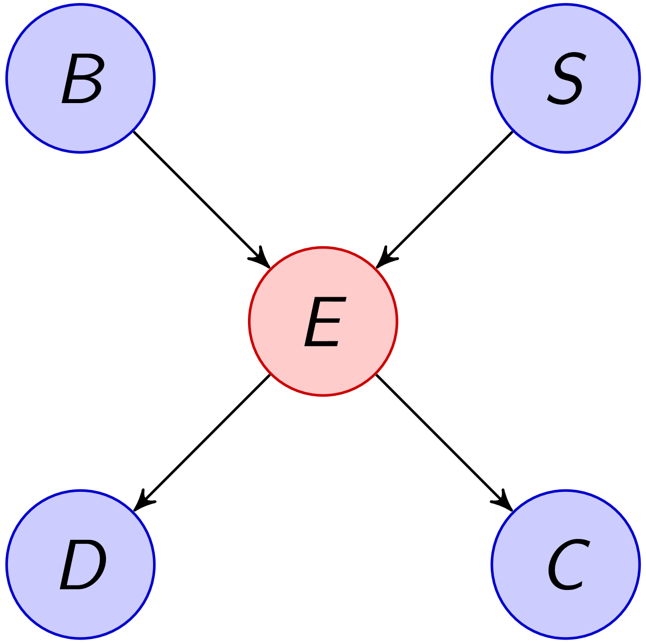


B	S	E	D	C
1	1			

In topological order

Sample from the condition probability distribution of X, given the sampled parent values

Approximate inference through sampling (Direct sampling)

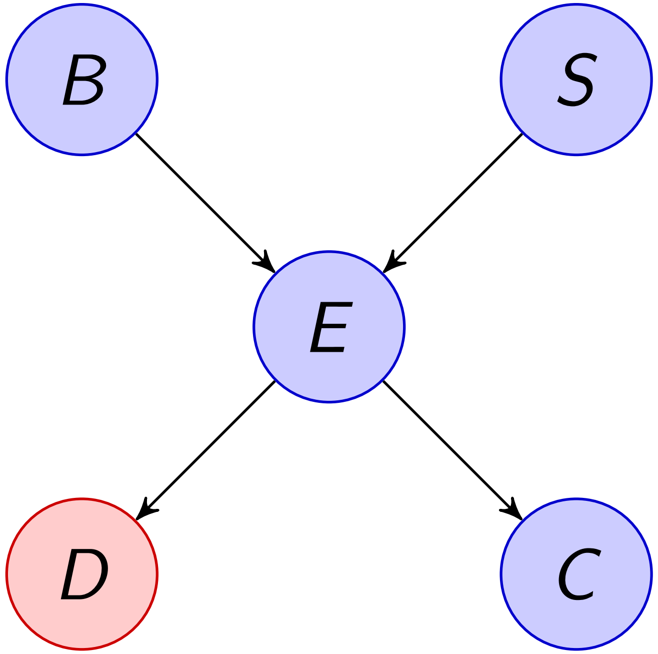


B	S	E	D	C
1	1	1		

In topological order

Sample from the condition probability distribution of X, given the sampled parent values

Approximate inference through sampling (Direct sampling)

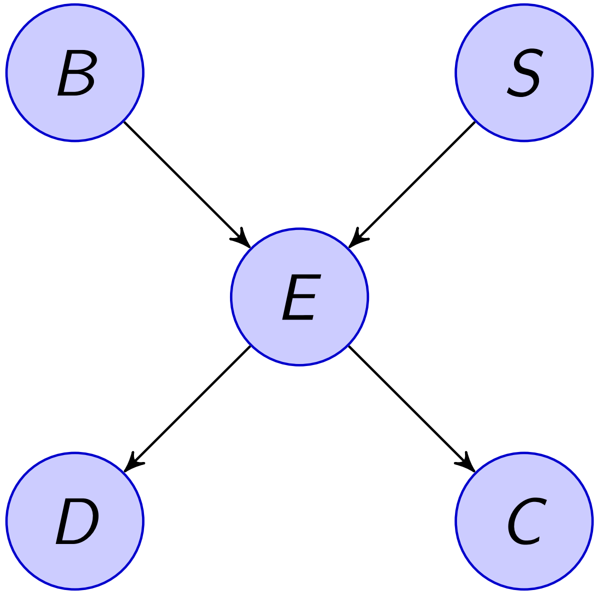


B	S	E	D	C
1	1	1	0	

In topological order

Sample from the condition probability distribution of X, given the sampled parent values

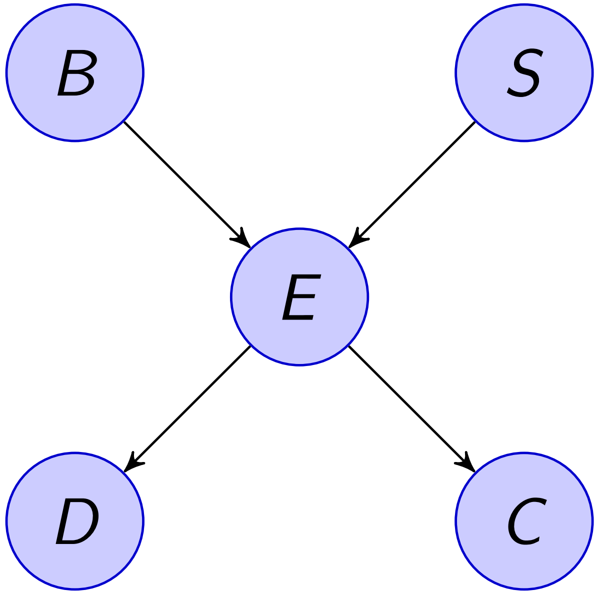
Approximate inference through sampling (Direct sampling)



B	S	E	D	C
1	1	1	0	0
0	1	0	1	0
0	0	0	1	1
1	0	1	1	1
1	0	0	0	0
0	0	0	0	1
1	0	0	0	1
0	0	1	0	0

What is the current approximation for $P(B{=}true \mid D{=}true, C{=}true)$?

Approximate inference through sampling (Direct sampling)



B	S	E	D	C
1	1	1	0	0
0	1	0	1	0
0	0	0	1	1
1	0	1	1	1
1	0	0	0	0
0	0	0	0	1
1	0	0	0	1
0	0	1	0	0

If likelihood of evidence is small, then many samples are required

Approximate sampling approaches

Likelihood weighting involves generating samples that are consistent with evidence and weighting them

Gibbs sampling involves making random local changes to samples (form of Markov Chain Monte Carlo)

Many other approaches