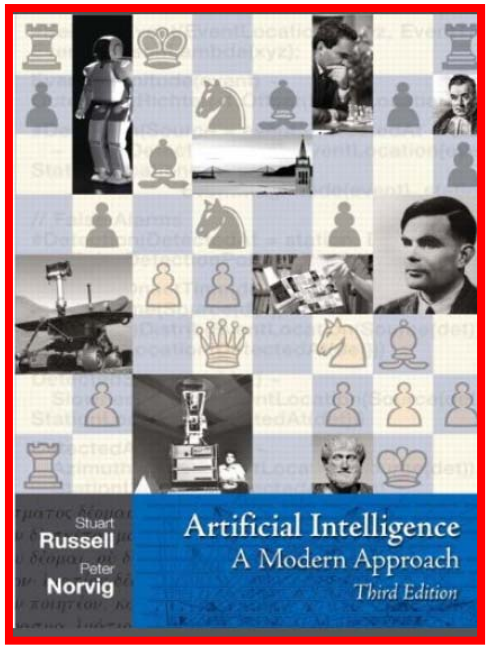


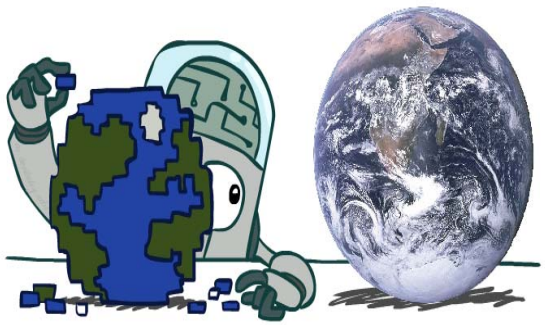
(Chapter-14)

BAYES NETS

Yanmei Zheng

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

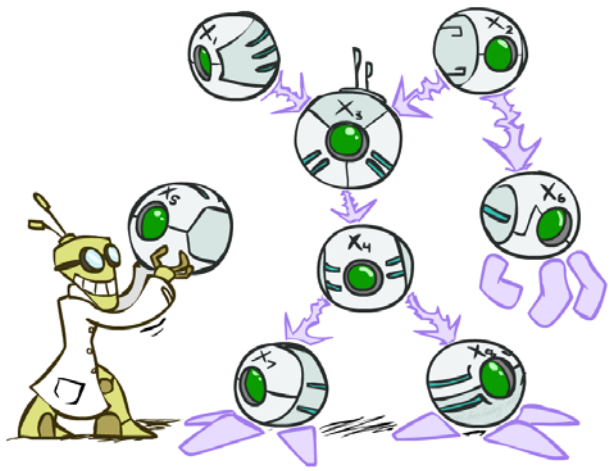


Bayes Nets: Big Picture

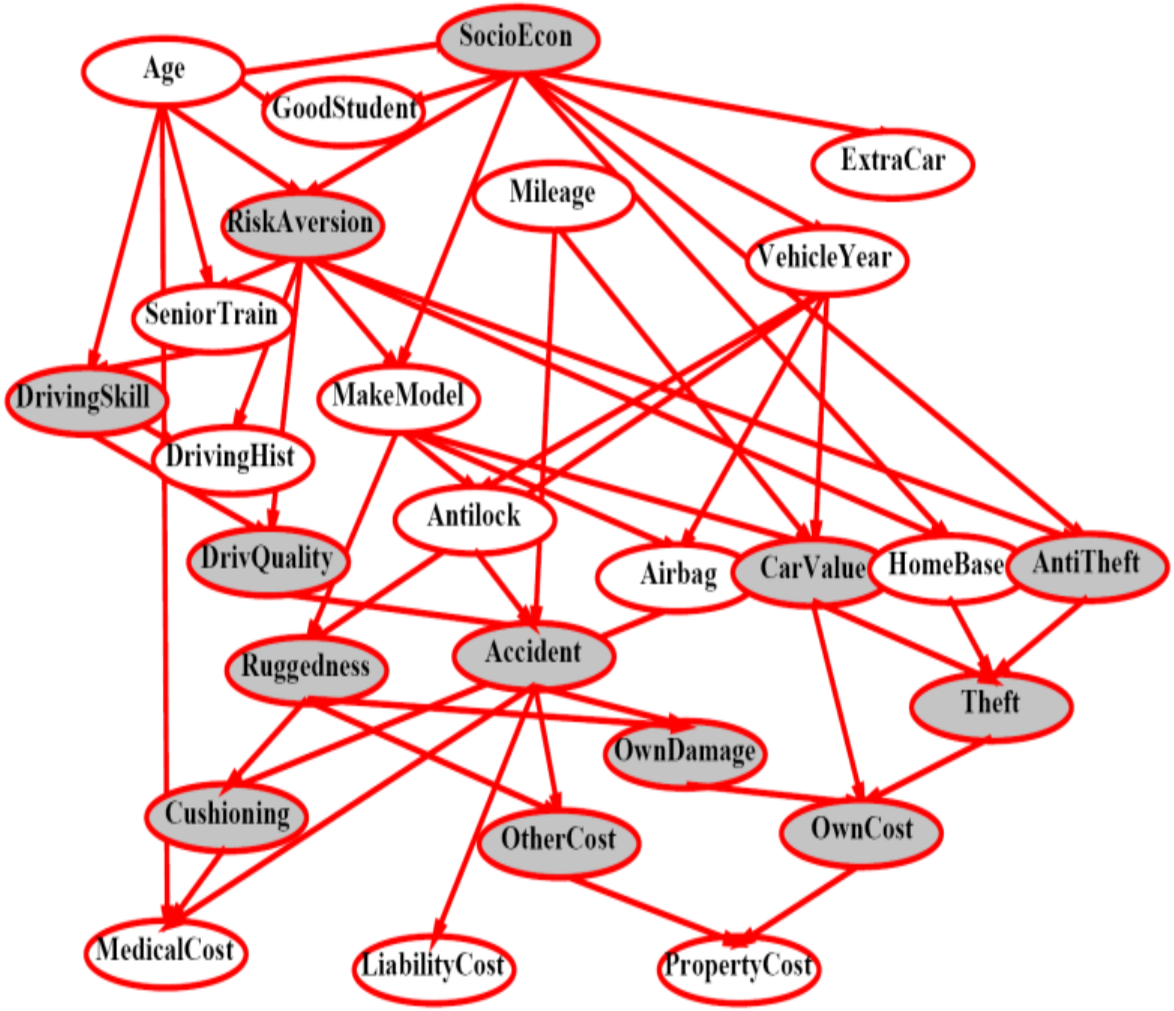
- Two problems with using full joint distribution tables as our probabilistic models:
  - ✓ Unless there are only a few variables, the joint is WAY too **big** to represent explicitly
  - ✓ **Hard** to learn (estimate) anything empirically about more than a few variables at a time

Bayes Nets: Big Picture

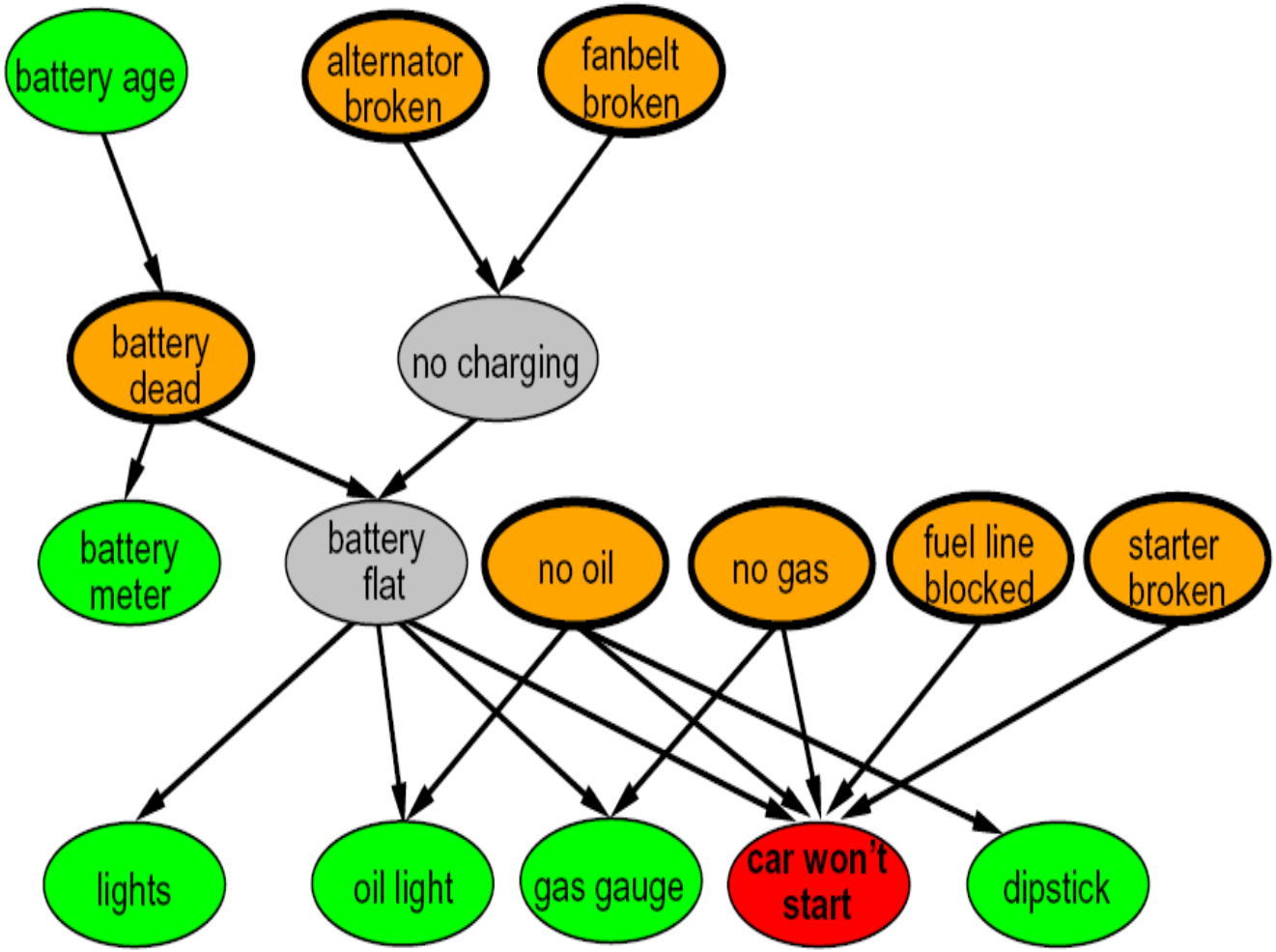
- **Bayes nets:** a technique for describing complex joint distributions (models) using simple, local distributions (conditional probabilities)
  - ✓ More properly called **graphical models**
  - ✓ We describe how variables locally interact
  - ✓ Local interactions chain together to give global, indirect interactions
  - ✓ For about 10 min, we will be vague about how these interactions are specified



Example Bayes Net: Insurance

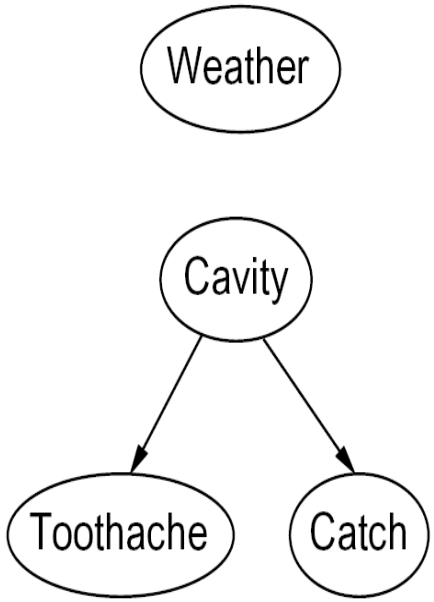


Example Bayes Net: Car



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)
- For now: imagine that arrows mean direct causation



Example: Coin Flips

- N independent coin flips

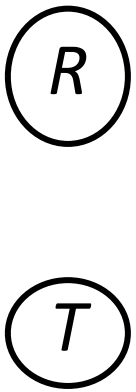
$$X_1 \qquad X_2 \qquad \cdots \qquad X_n$$

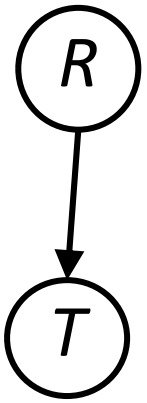
- No interactions between variables: **absolute independence**

Example: Traffic

- Variables:
  - ✓ R: It rains
  - ✓ T: There is traffic
- Model 1: independence

- Model 2: rain causes traffic





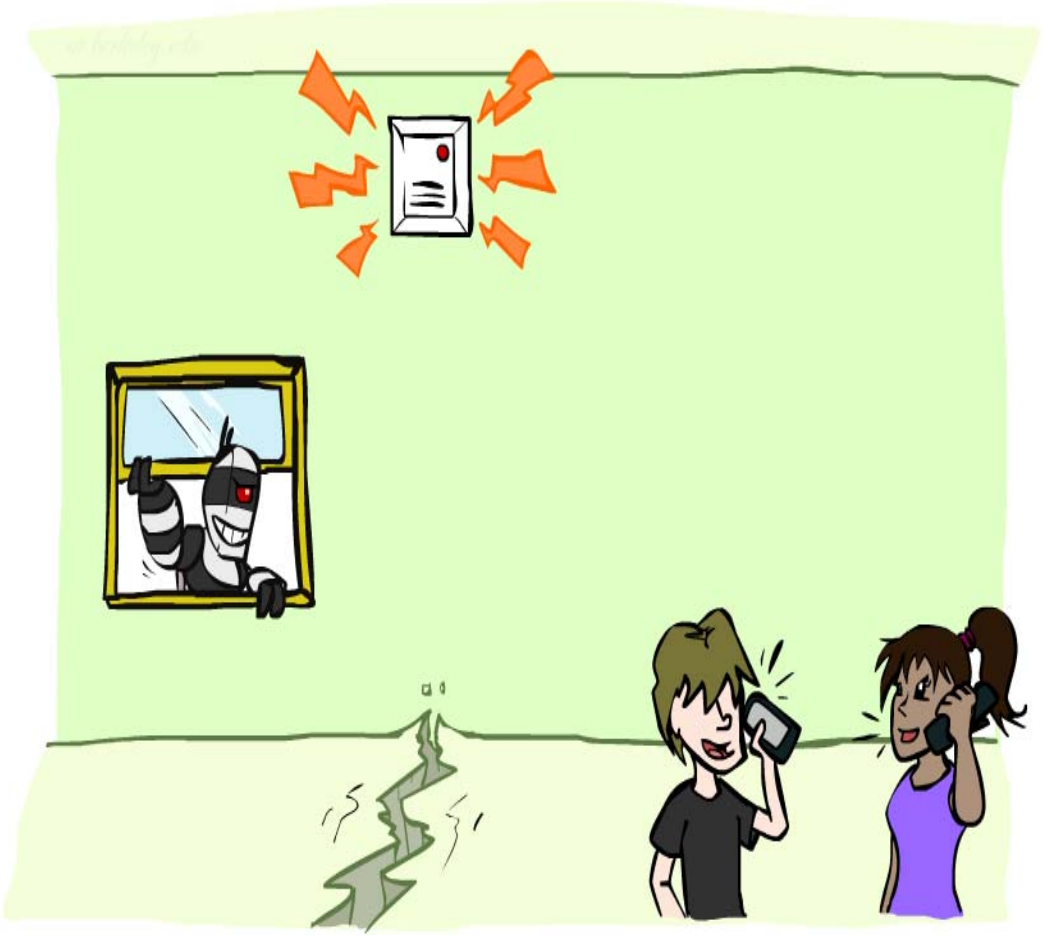
- Why is an agent using model 2 better?

Example: Traffic II

- Let's build a causal graphical model!
- Variables
  - ✓ T: Traffic
  - ✓ R: It rains
  - ✓ L: Low pressure
  - ✓ D: Roof drips
  - ✓ B: Ballgame
  - ✓ C: Cavity

Example: Alarm Network

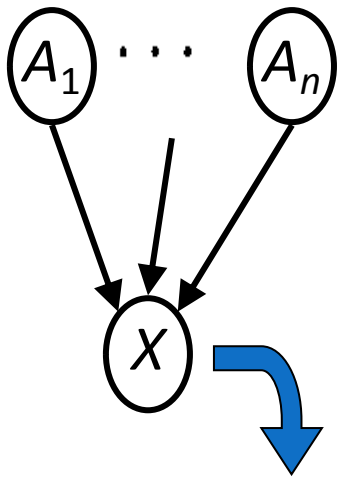
- Variables
  - ✓ B: Burglary
  - ✓ A: Alarm goes off
  - ✓ M: Mary calls
  - ✓ J: John calls
  - ✓ E: Earthquake!



Bayes Net Semantics

- 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
  - ✓ Description of a noisy "causal" process



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

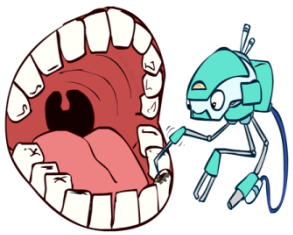
A Bayes net = Topology (graph) + Local Conditional Probabilities

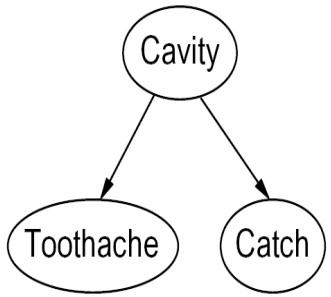
Probabilities in BNs

- **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:**

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

  - ✓ **Example:**





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

Probabilities 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):** $P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i | x_1 \ldots x_{i-1})$

- **<u>Assume</u> 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**

Example: Coin Flips

X_1 X_2 $\cdots$ X_n

$P(X_1)$

h	0.5
t	0.5

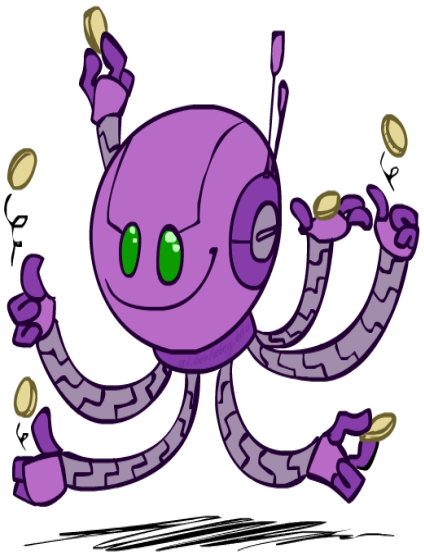
$P(X_2)$

h	0.5
t	0.5

$\cdots$

$P(X_n)$

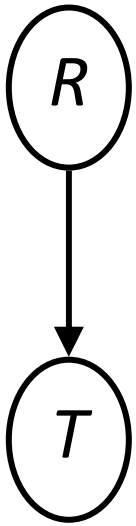
h	0.5
t	0.5



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

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

Example: Traffic



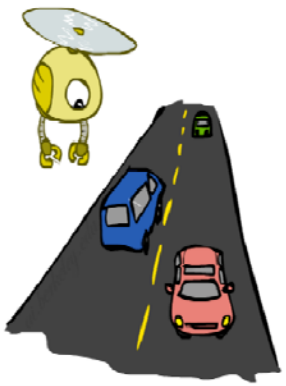
$P(R)$

+r	1/4
-r	3/4

$P(+r, -t) =$

$P(T|R)$

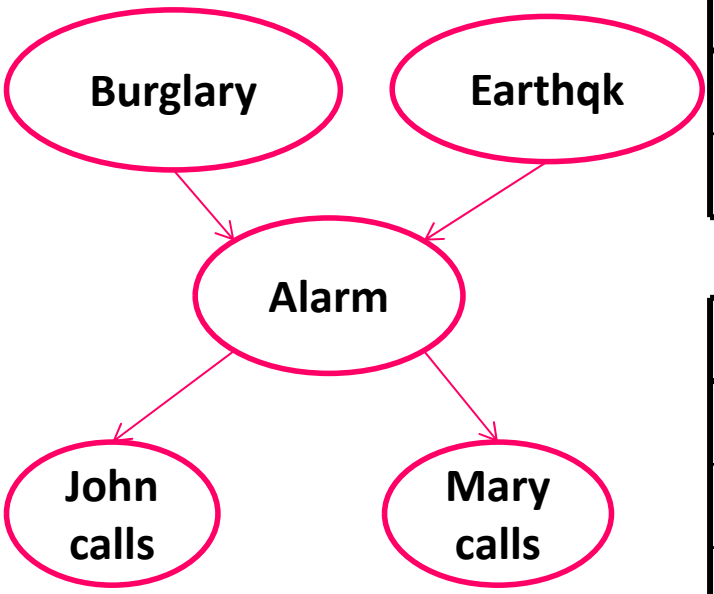
+r	+t	3/4
	-t	1/4
-r	+t	1/2
	-t	1/2



Dr.Yanmei Zheng

Example: Alarm Network

B	P(B)
+b	0.001
-b	0.999



E	P(E)
+e	0.002
-e	0.998

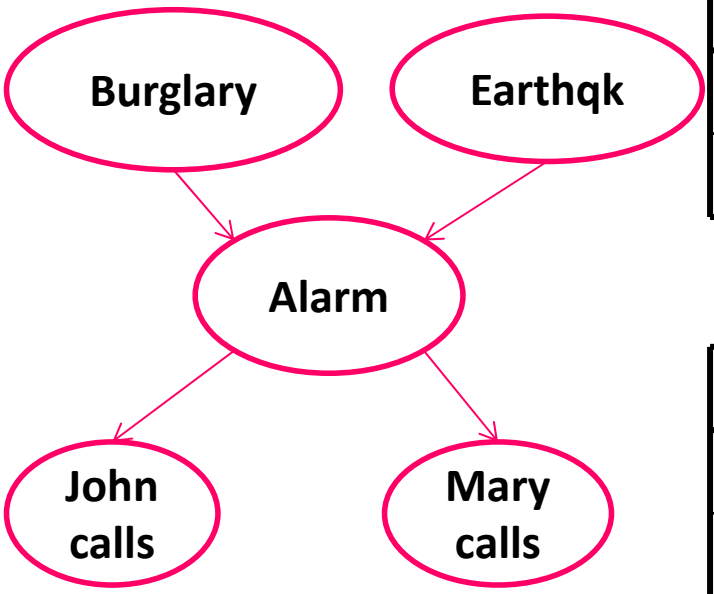
B	E	A	P(A|B,E)
+b	+e	+a	0.95
+b	+e	-a	0.05
+b	-e	+a	0.94
+b	-e	-a	0.06
-b	+e	+a	0.29
-b	+e	-a	0.71
-b	-e	+a	0.001
-b	-e	-a	0.999

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

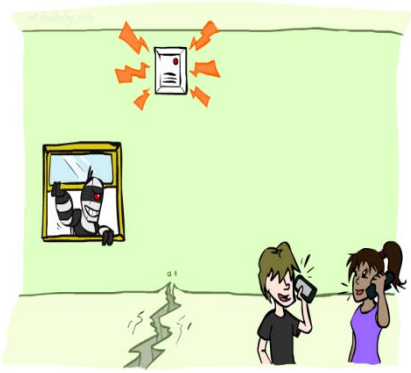
A	M	P(M|A)
+a	+m	0.7
+a	-m	0.3
-a	+m	0.01
-a	-m	0.99

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

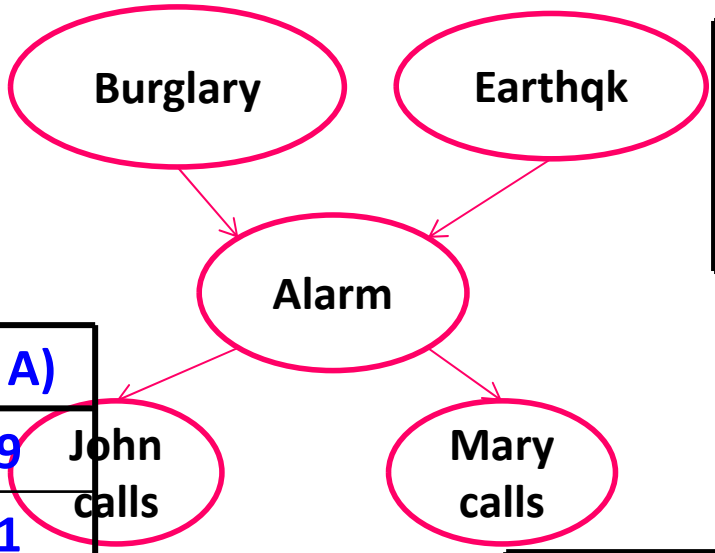
A	M	P(M|A)
+a	+m	0.7
+a	-m	0.3
-a	+m	0.01
-a	-m	0.99

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

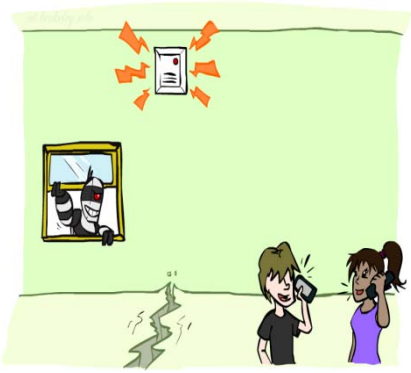
$$P(+b, -e, +a, -j, +m) =$$

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

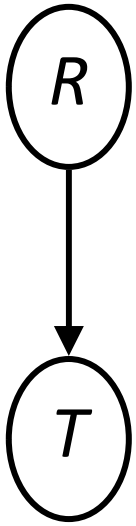
$$P(+b,-e,+a,-j,+m)$$

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

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

Example: Traffic

- Causal direction



$P(R)$

+r	1/4
-r	3/4

$P(T|R)$

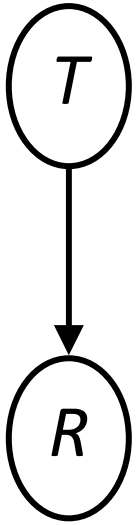
+r	+t	3/4
	-t	1/4
-r	+t	1/2
	-t	1/2

$P(T,R)$

+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

$P(R|T)$

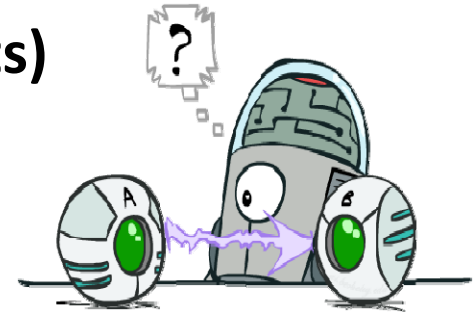
+t	+r	1/3
	-r	2/3
-t	+r	1/7
	-r	6/7

$P(T, R)$

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

Causality?

- When Bayesnets 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 variables *Traffic* and *Drips*
  - ✓ 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

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

Bayes Nets

- So far: how a Bayes net encodes a joint distribution
- Next: how to answer queries about that distribution
  - First assembled BNs using an intuitive notion of conditional independence as causality
  - Then saw that key property is conditional independence
  - Main goal: answer queries about conditional independence and influence
- After that: how to answer numerical queries (inference)

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

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

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

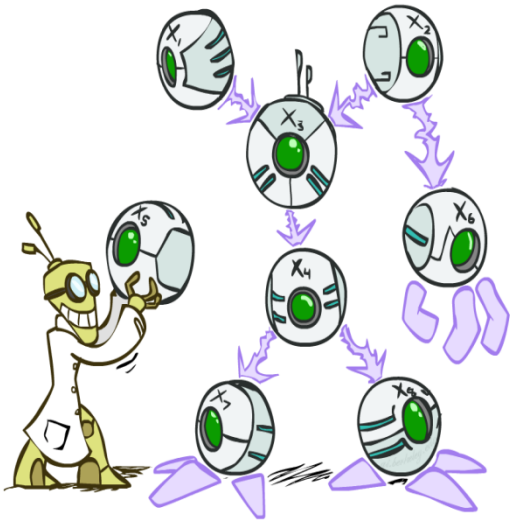
Ch14.4 Bayes Nets III Inference

Dr.Yanmei Zheng

Bayes Net Representation

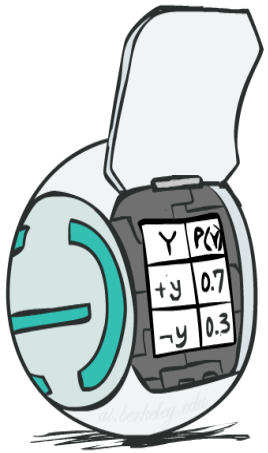
- A directed, acyclic graph, one node per random variable
- A conditional probability table (CPT) for each node
  - ✓ A collection of distributions over X, one for each combination of parents' values

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



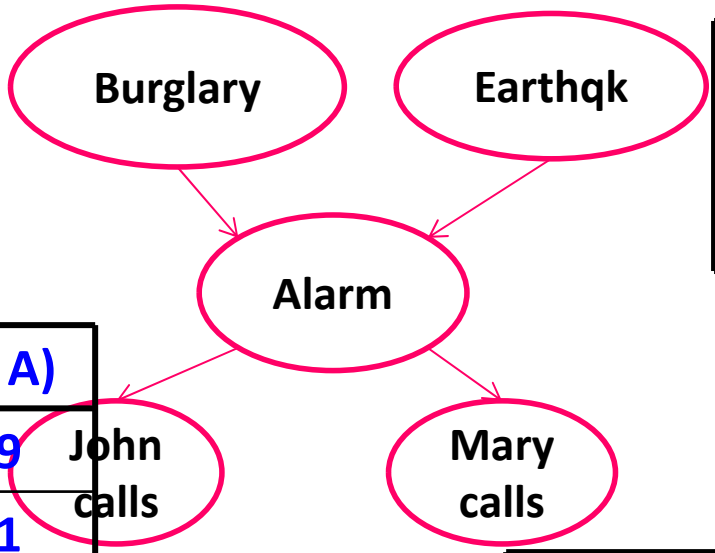
- Bayes' nets implicitly encode joint distributions
  - ✓ As a product of local conditional distributions
  - ✓ To see what probability a BN gives to a full assignment, multiply all the relevant conditionals together:

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

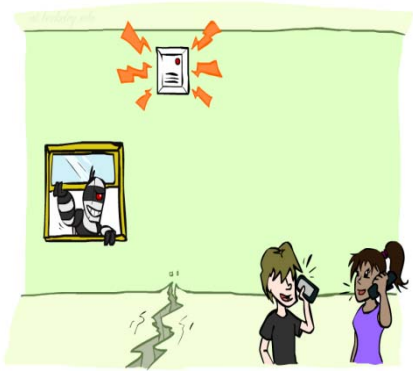


Example: Alarm Network

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

$$P(+b,-e,+a,-j,+m) =$$
$$P(+b)P(-e)P(+a|+b,-e)P(-j|+a)P(+m|+a) =$$
$$0.001 \times 0.998 \times 0.94 \times 0.1 \times 0.7$$

Bayes Nets

- Representation
- Conditional Independences
- Probabilistic Inference
  - ✓ Enumeration (exact, exponential complexity)
  - ✓ Variable elimination (exact, worst-case exponential complexity, often better)
  - ✓ Inference is NP-complete
  - ✓ Sampling (approximate)
- Learning Bayes Nets from Data

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

Inference by Enumeration

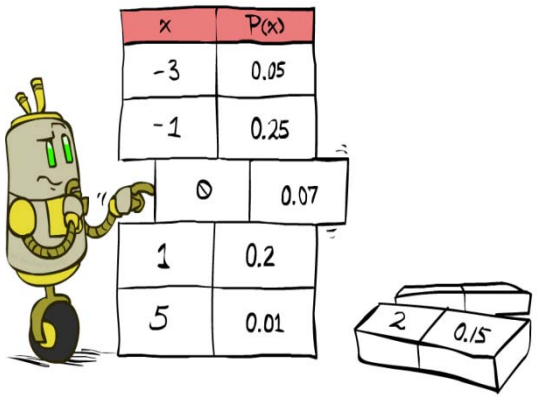
- **General case:**
  - ✓ **Evidence variables:** $E_1 \dots E_k = e_1 \dots e_k$
  - ✓ **Query* variable:** Q
  - ✓ **Hidden variables:** $H_1 \dots H_r$

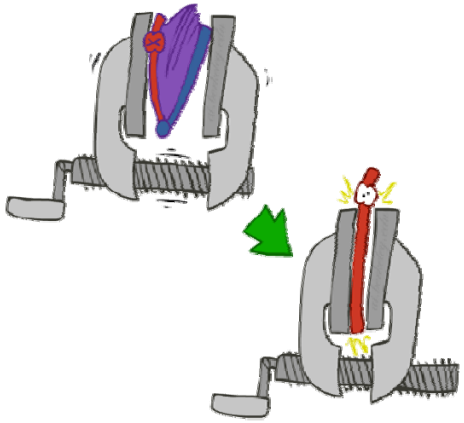
$\Big\}$ $X_1, X_2, \dots X_n$ *All variables*

- **We want:*** **Works fine with multiple query variables, too**

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

- **Step 1: Select the entries consistent with the evidence**
- **Step 2: Sum out H to get joint of Query and evidence**
- **Step 3: Normalize**





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

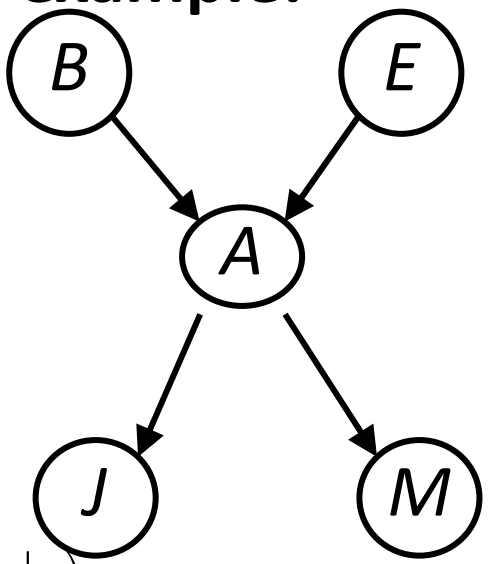
$$P(Q, e_1 \dots e_k) = \sum_{h_1 \dots h_r} \underbrace{P(Q, h_1 \dots h_r, e_1 \dots e_k)}_{X_1, X_2, \dots X_n}$$

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

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

Inference by Enumeration in Bayes' Net

- Given unlimited time, inference in BNs is easy
- Reminder of inference by enumeration by example:



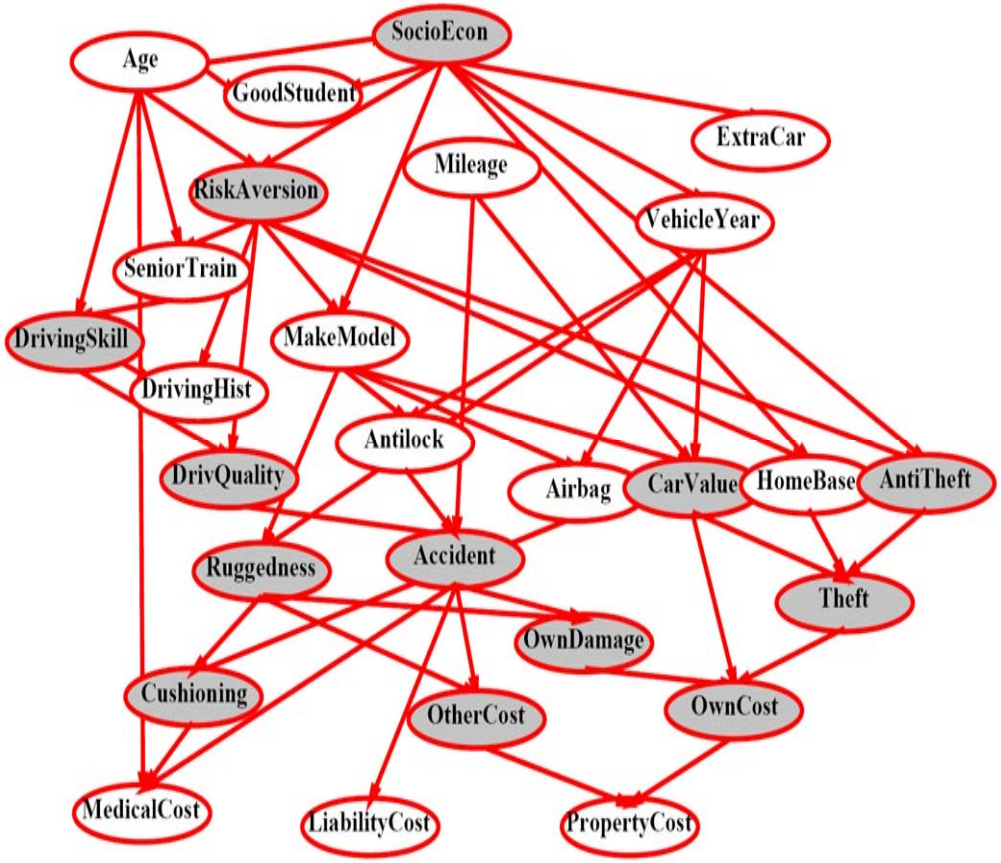
$$P(B \mid +j, +m) \propto_B P(B, +j, +m)$$

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

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

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

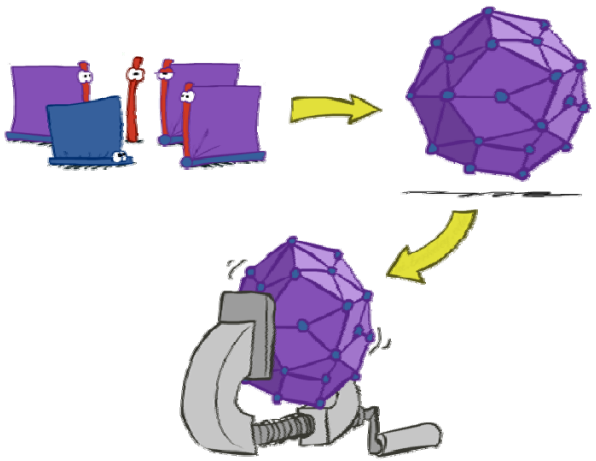
Inference by Enumeration?

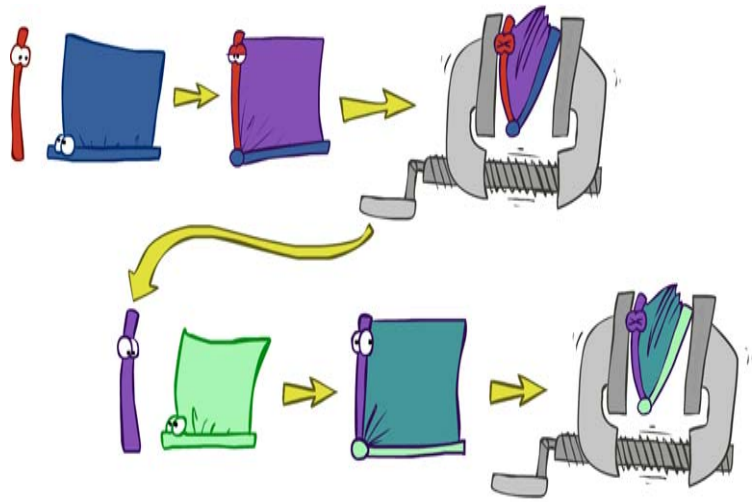


$$P(Antilock|observed\ variables) = ?$$

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"
  - ✓ Still NP-hard, but usually much faster than inference by enumeration
  - ✓ First we'll need some new notation: factors





Factor I

- Joint distribution: P(X,Y)
  - ✓ Entries P(x,y) for all x, y
  - ✓ Sums to 1
- Selected joint: P(x,Y)
  - ✓ A slice of the joint distribution
  - ✓ Entries P(x,y) for fixed x, all y
  - ✓ Sums to P(x)
- Number of capitals = dimensionality of the table

$P(T,W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

$P(cold,W)$

T	W	P
cold	sun	0.2
cold	rain	0.3

Factor II

- **Single conditional: P(Y | x)**
  - ✓ Entries P(y | x) for fixed x, all y
  - ✓ Sums to 1

$P(W|cold)$

T	W	P
cold	sun	0.4
cold	rain	0.6

- **Family of conditionals: P(X |Y)**
  - ✓ Multiple conditionals
  - ✓ Entries P(x | y) for all x, y
  - ✓ Sums to |Y|

$P(W|T)$

T	W	P
hot	sun	0.8
hot	rain	0.2
cold	sun	0.4
cold	rain	0.6

$P(W|hot)$

$P(W|cold)$

Factor III

- **Specified family: P(y | X)**
  - ✓ **Entries P(y | x) for fixed y, but for all x**
  - ✓ **Sums to … who knows!**

$$P(rain|T)$$

T	W	P
hot	rain	0.2
cold	rain	0.6

$P(rain|hot)$

$P(rain|cold)$

Factor Summary

- 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!**

$$P(L) = ?$$

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

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

$P(R)$

+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

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

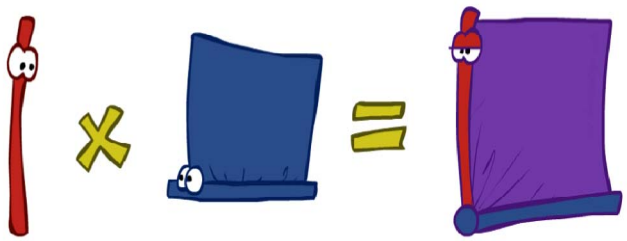
$P(+\ell|T)$

+t	+l	0.3
-t	+l	0.1

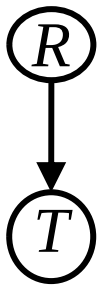
- Procedure: Join all factors, then eliminate all hidden variables

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
  - ✓ Computation for each entry: pointwise products



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

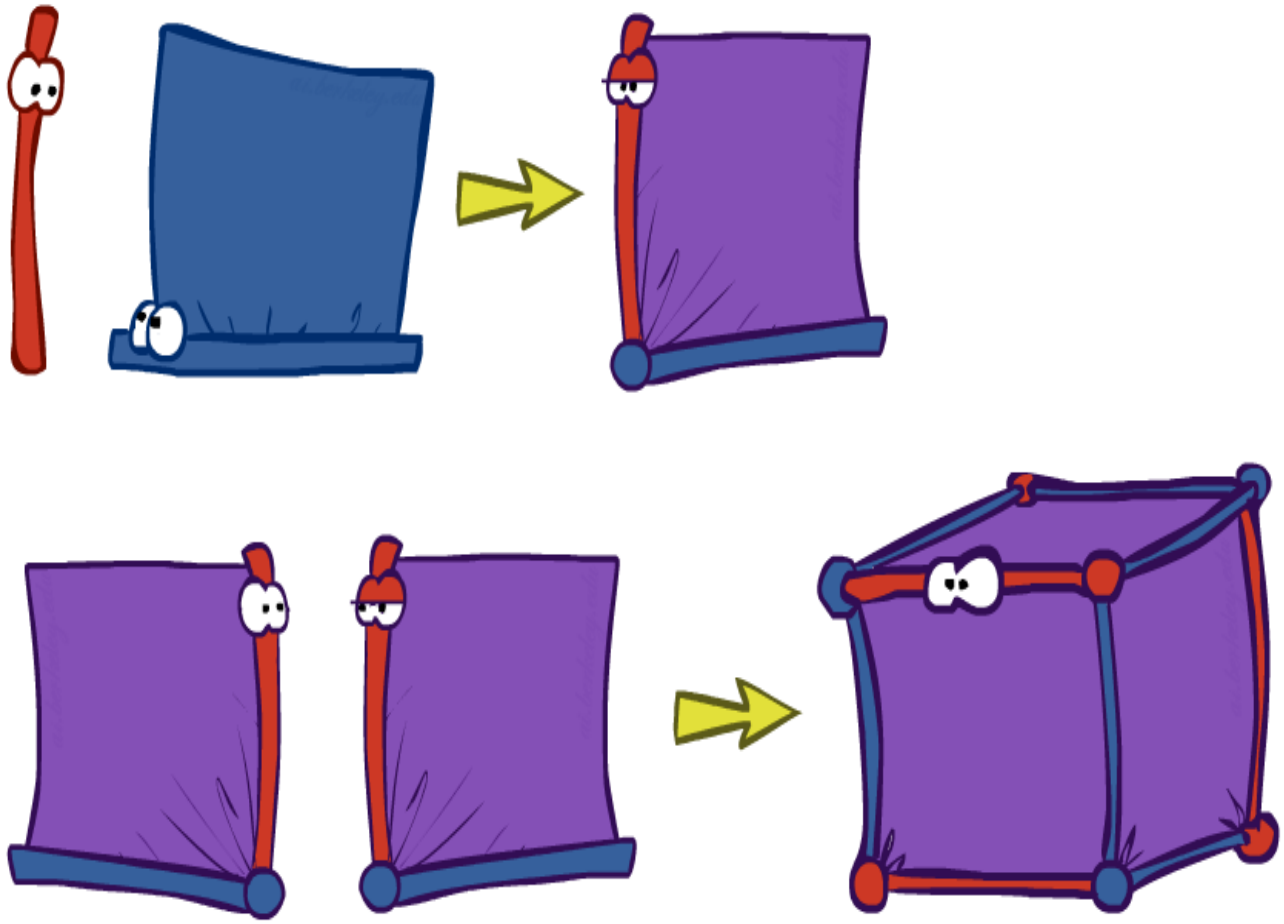
+r	0.1
-r	0.9

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

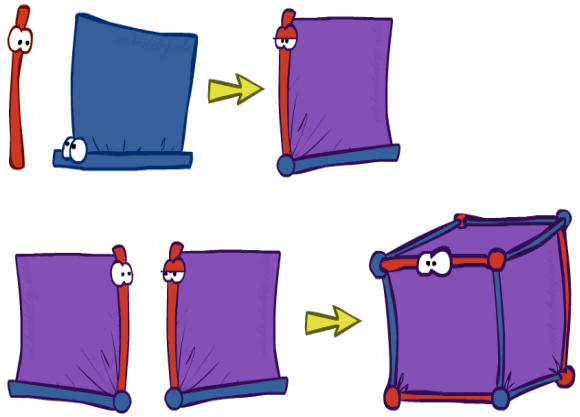
+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

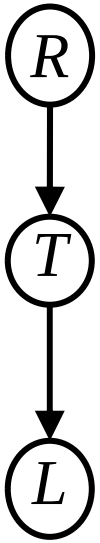
$$\forall r, t: \quad P(r,t) = P(r) \cdot P(t|r)$$

Example: Multiple Joins



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

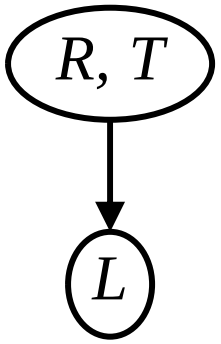
Join R

$P(R,T)$

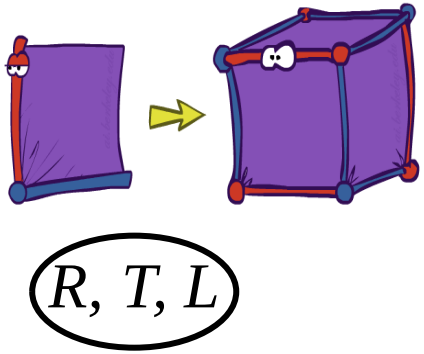
+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

$P(L|T)$

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



Join T

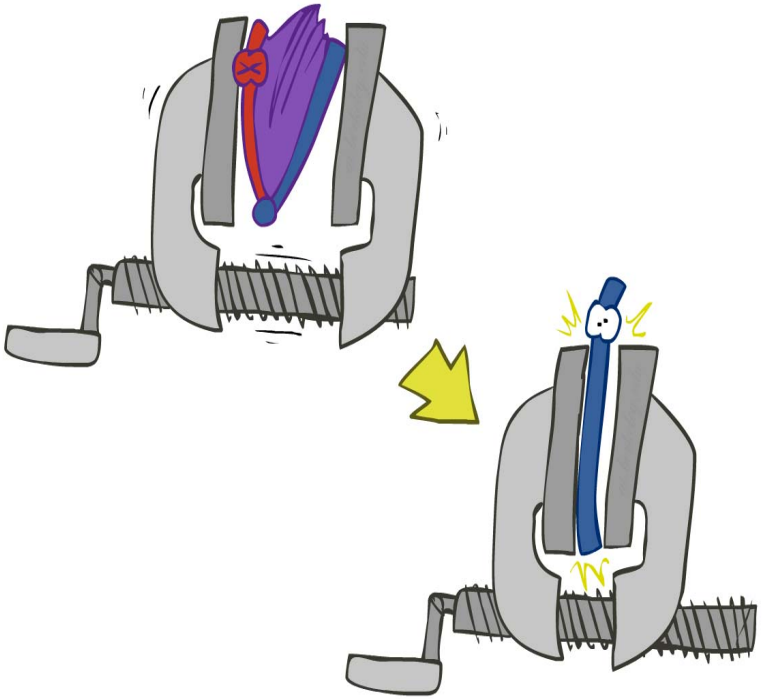


$P(R,T,L)$

+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

$P(T,L)$

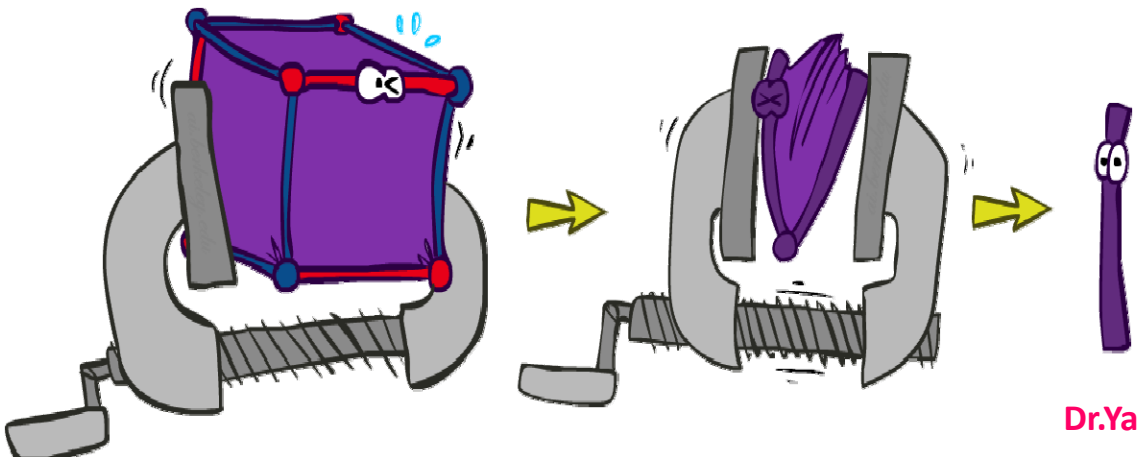
+t	+l	0.051
+t	-l	0.119
-t	+l	0.083
-t	-l	0.747

Sum out T

L

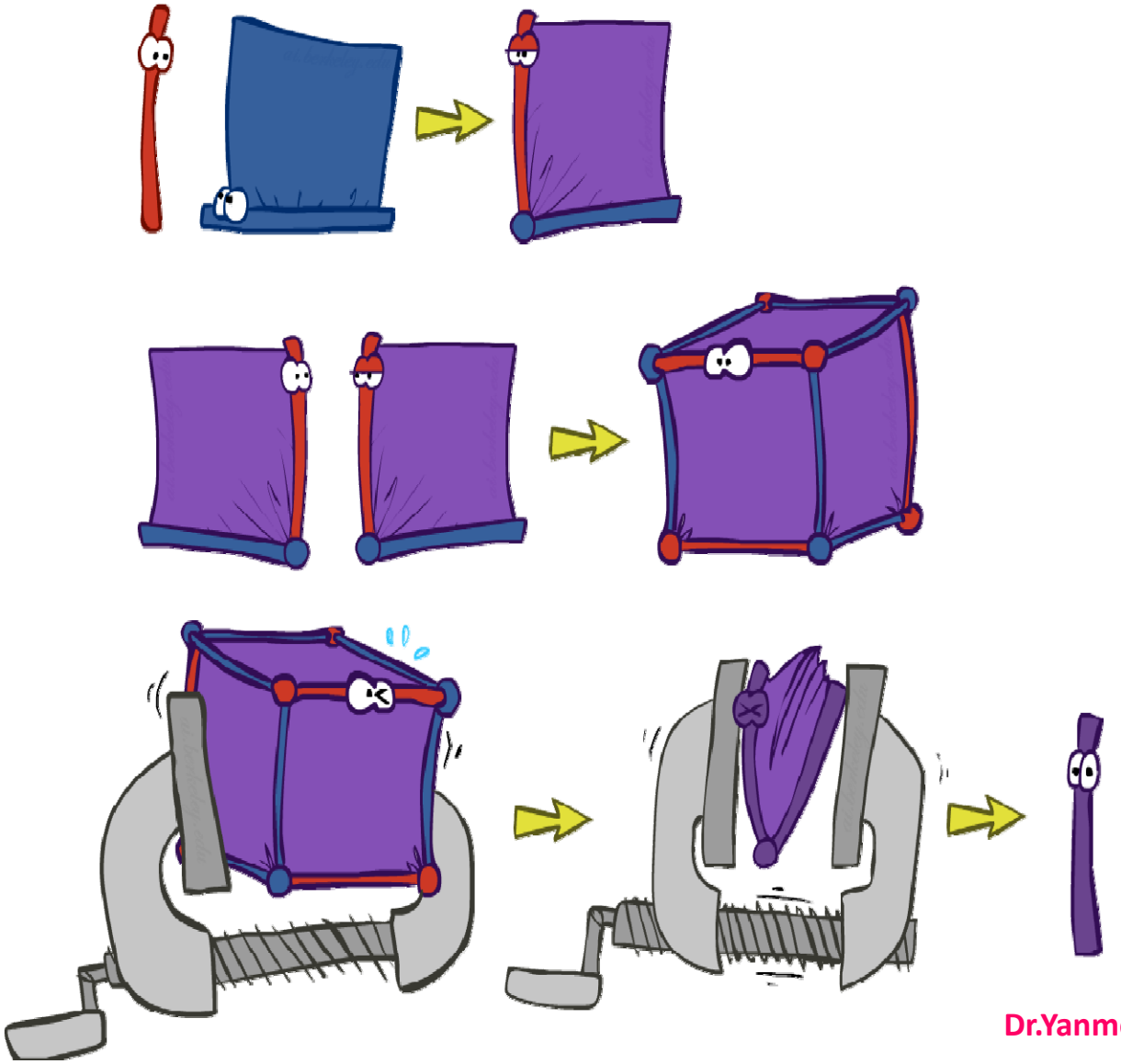
$P(L)$

+l	0.134
-l	0.886



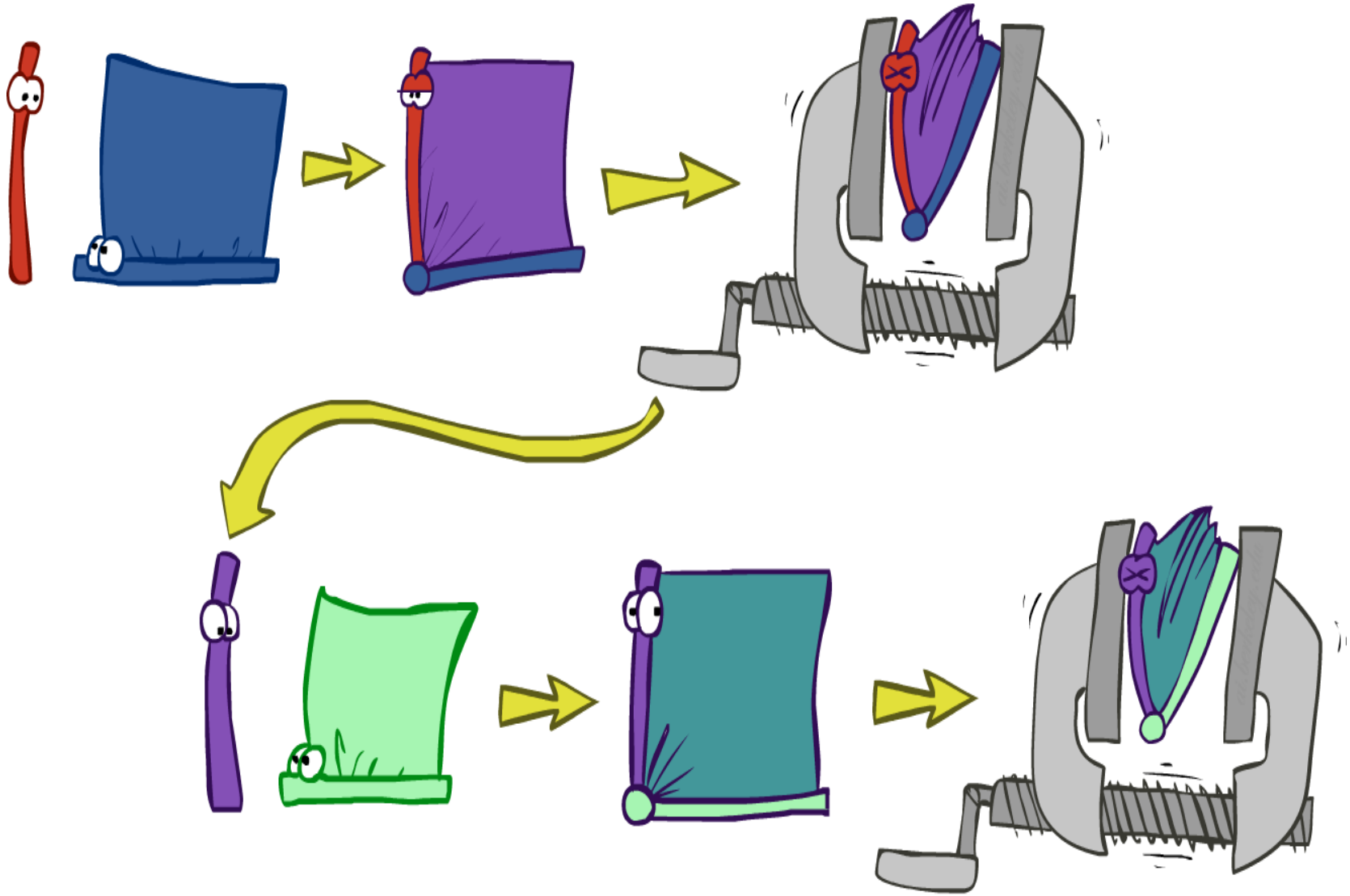
Dr.Yanmei Zheng

Thus Far: Multiple Join, Multiple Eliminate (= Inference by Enumeration)



Dr.Yanmei Zheng

Marginalizing Early (= Variable Elimination)



Dr.Yanmei Zheng

Traffic Domain

$$P(L) = ?$$

- **Inference by Enumeration**

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

Eliminate t

- Variable Elimination

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

Eliminate t

Marginalizing Early! (aka VE)

$P(R)$

+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

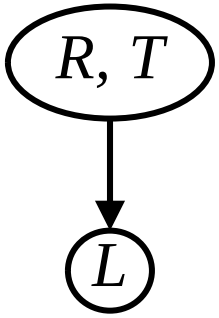
$P(L|T)$

+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



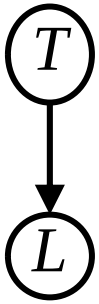
$P(L|T)$

+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 II

- 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

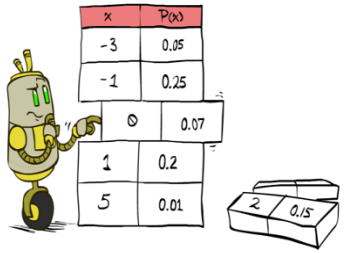
- To get our answer, just normalize this!

- That ’s it!

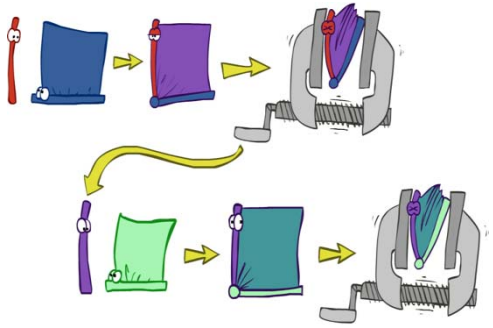
General Variable Elimination

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

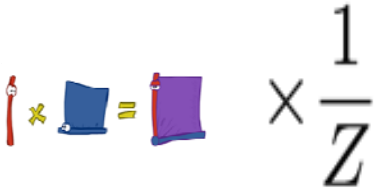
- **Start with initial factors:**
  - ✓ Local CPTs (but instantiated by evidence)



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

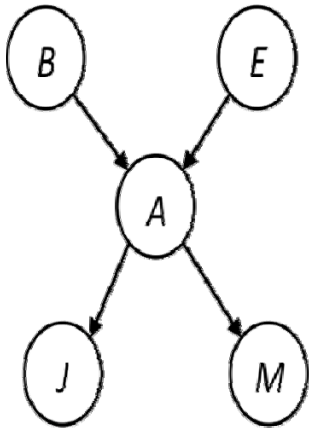


- **Join all remaining factors and normalize**



Example

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



$P(B)$	$P(E)$	$P(A\|B,E)$	$P(j\|A)$	$P(m\|A)$

Choose A

$P(A|B,E)$

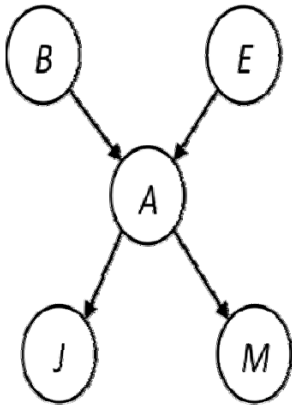
$P(j|A)$ $\xrightarrow{\times}$ $P(j,m,A|B,E)$ $\xrightarrow{\Sigma}$ $P(j,m|B,E)$

$P(m|A)$

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

Example

$P(B)$ $P(E)$ $P(j,m|B,E)$



Choose E

$P(E)$

$P(j,m|B,E)$

$\times$ $P(j,m,E|B)$ Σ $P(j,m|B)$

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

Finish with B

$P(B)$

$P(j,m|B)$

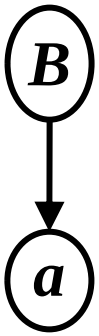
$\times$ $P(j,m,B)$ Normalize $P(B|j,m)$

Example 2: P(B|a)

Start / Select

$P(B)$

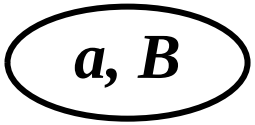
B	P
+b	0.1
¬b	0.9



$P(A|B) \rightarrow P(a|B)$

B	A	P
+b	+a	0.8
~~b~~	~~¬a~~	0.2
¬b	+a	0.1
~~¬b~~	~~¬a~~	0.9

Join on B



$P(a, B)$

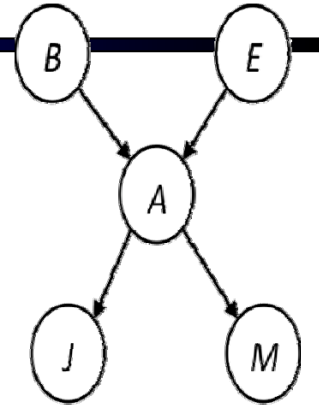
A	B	P
+a	+b	0.08
+a	¬b	0.09

Normalize

$P(B|a)$

A	B	P
+a	+b	8/17
+a	¬b	9/17

Same Example in Equations



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

$P(B)$	$P(E)$	$P(A\|B,E)$	$P(j\|A)$	$P(m\|A)$

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

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!

Another Variable Elimination Example

Query: $P(X_3|Y_1 = y_1, Y_2 = y_2, Y_3 = y_3)$

Start by inserting evidence, which gives the following initial factors:

$$p(Z)p(X_1|Z)p(X_2|Z)p(X_3|Z)p(y_1|X_1)p(y_2|X_2)p(y_3|X_3)$$

Eliminate X_1, this introduces the factor $f_1(Z, y_1) = \sum_{x_1} p(x_1|Z)p(y_1|x_1)$, and we are left with:

$$p(Z)f_1(Z, y_1)p(X_2|Z)p(X_3|Z)p(y_2|X_2)p(y_3|X_3)$$

Eliminate X_2, this introduces the factor $f_2(Z, y_2) = \sum_{x_2} p(x_2|Z)p(y_2|x_2)$, and we are left with:

$$p(Z)f_1(Z, y_1)f_2(Z, y_2)p(X_3|Z)p(y_3|X_3)$$

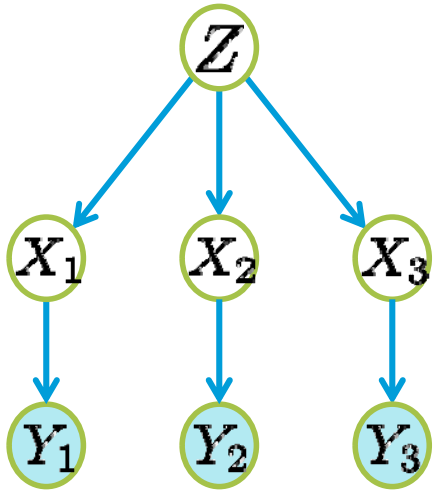
Eliminate Z, this introduces the factor $f_3(y_1, y_2, X_3) = \sum_z p(z)f_1(z, y_1)f_2(z, y_2)p(X_3|z)$, and we are left:

$$p(y_3|X_3), f_3(y_1, y_2, X_3)$$

No hidden variables left. Join the remaining factors to get:

$$f_4(y_1, y_2, y_3, X_3) = P(y_3|X_3)f_3(y_1, y_2, X_3).$$

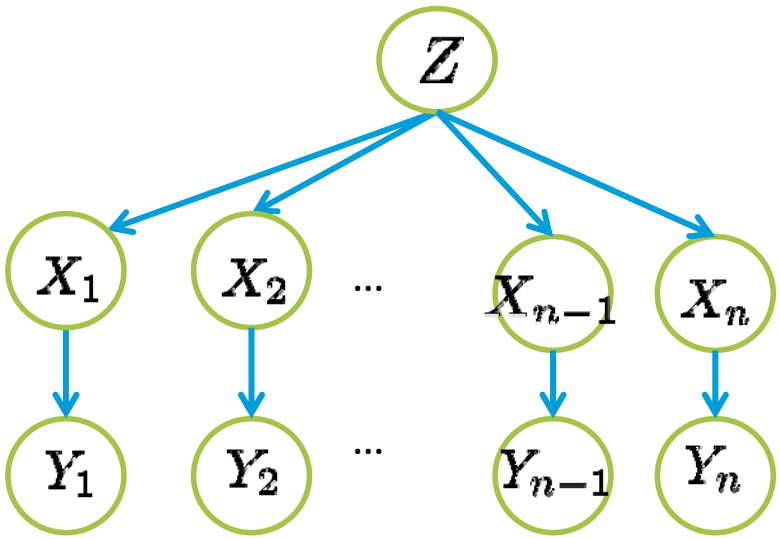
Normalizing over X_3 gives $P(X_3|y_1, y_2, y_3)$.



Computational complexity critically depends on the largest factor being generated in this process. Size of factor = number of entries in table. In example above (assuming binary) all factors generated are of size 2 --- as they all only have one variable (Z, Z, and X_3 respectively).

Variable Elimination Ordering

- For the query $P(X_n|y_1,\ldots,y_n)$ work through the following two different orderings as done in previous slide: $Z, X_1, \ldots, X_{n-1}$ and $X_1, \ldots, X_{n-1}, Z$. What is the size of the maximum factor generated for each of the orderings?



- Answer: 2^{n+1} versus 2^2 (assuming binary)
- In general: the ordering can greatly affect efficiency.

VE: Computational 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., previous slide's example 2^n vs. 2

- Does there always exist an ordering that only results in small factors?
  - ✓ No!

Bayes Nets

- Representation
- Conditional Independences
- Probabilistic Inference
  - ✓ Enumeration (exact, exponential complexity)
  - ✓ Variable elimination (exact, worst-case exponential complexity, often better)
  - ✓ Inference is NP-complete
  - ✓ Sampling (approximate)
- Learning Bayes Nets from Data

Ch14.5 Bayes Nets: Sampling

Dr.Yanmei Zheng

Bayes Net Representation

- 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

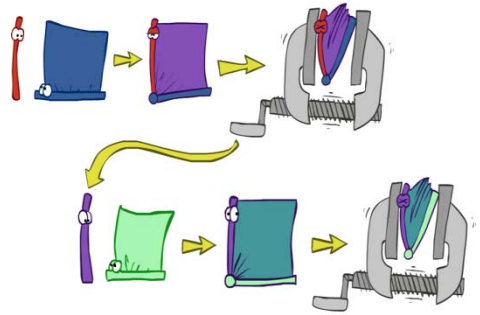
$$P(X|a_1 \ldots a_n)$$

- 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:

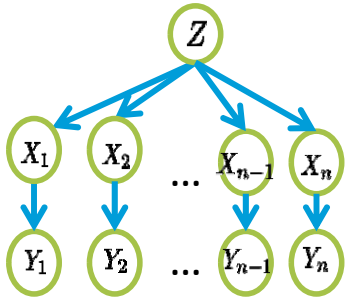
$$P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i|parents(X_i))$$

Variable Elimination

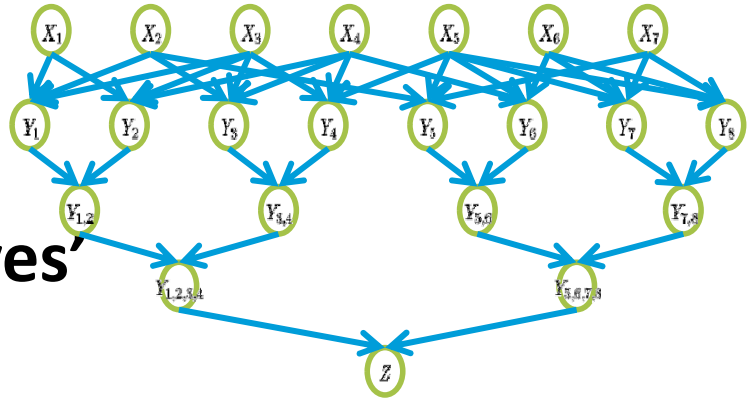
- Interleave joining and marginalizing



- d^k entries computed for a factor over k variables with domain sizes d



- Ordering of elimination of hidden variables can affect size of factors generated



- Worst case: running time exponential in the size of the Bayes' net

Sampling

- Sampling is a lot like repeated simulation
  - ✓ Predicting the weather, basketball games, …
- Basic idea
  - ✓ Draw N samples from a sampling distribution S
  - ✓ Compute an approximate posterior probability
  - ✓ Show this converges to the true probability P

- Why sample?
  - Learning: get samples from a distribution you don't know
  - Inference: getting a sample is faster than computing the right answer (e.g. with variable elimination)

Sampling

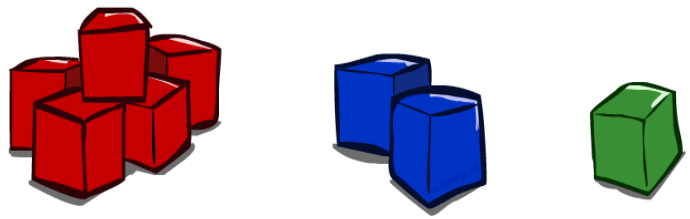
Sampling from given distribution

- ✓ Step 1: Get sample *u* from uniform distribution over [0, 1)
  - E.g. random() in python
- ✓ Step 2: Convert this sample *u* into an outcome for the given distribution by having each outcome associated with a sub-interval of [0,1) with sub-interval size equal to probability of the outcome

C	P(C)
red	0.6
green	0.1
blue	0.3

$$0 \leq u < 0.6, \rightarrow C = red$$
$$0.6 \leq u < 0.7, \rightarrow C = green$$
$$0.7 \leq u < 1, \rightarrow C = blue$$

- **Example**
  - If random() returns *u* = 0.83, then our sample is *C* = blue
  - E.g, after sampling 8 times:



Sampling in Bayes Nets

- Prior Sampling
- Rejection Sampling
- Likelihood Weighting
- Gibbs Sampling

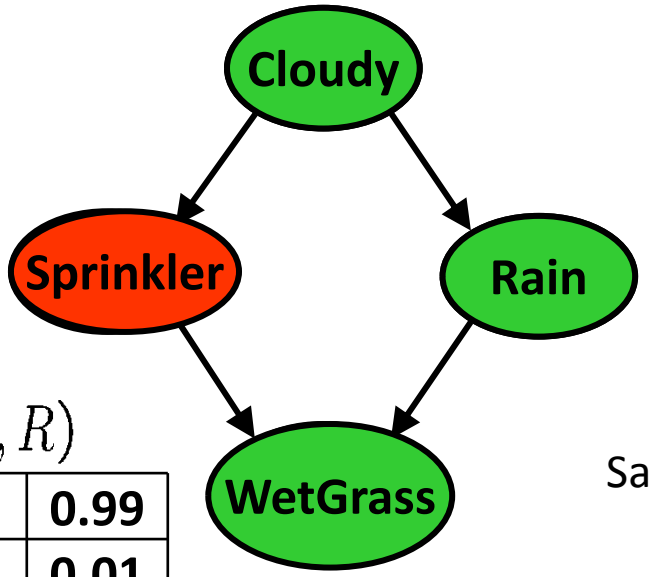
Prior Sampling

$P(C)$

+c	0.5
-c	0.5

$P(S|C)$

+c	+s	0.1
	-s	0.9
-c	+s	0.5
	-s	0.5



$P(R|C)$

+c	+r	0.8
	-r	0.2
-c	+r	0.2
	-r	0.8

$P(W|S,R)$

+s	+r	+w	0.99
		-w	0.01
	-r	+w	0.90
		-w	0.10
-s	+r	+w	0.90
		-w	0.10
	-r	+w	0.01
		-w	0.99

Samples:

+c, -s, +r, +w
-c, +s, -r, +w
…

Prior Sampling

- For i=1, 2, …, n
  - ✓ Sample x_i from $P(X_i \mid Parents(X_i))$
- Return $(x_1, x_2, \ldots, x_n)$

Prior Sampling

- This process generates samples with probability:

$$S_{PS}(x_1 \ldots x_n) = \prod_{i=1}^{n} P(x_i|\text{Parents}(X_i)) = P(x_1 \ldots x_n)$$

…i.e. the BN's joint probability

- Let the number of samples of an event be $N_{PS}(x_1 \ldots x_n)$

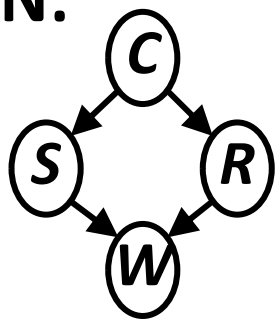
- Then

$$\begin{aligned} \lim_{N\to\infty} \hat{P}(x_1,\ldots,x_n) &= \lim_{N\to\infty} N_{PS}(x_1,\ldots,x_n)/N \\ &= S_{PS}(x_1,\ldots,x_n) \\ &= P(x_1 \ldots x_n) \end{aligned}$$

- I.e., the sampling procedure is consistent

Example

- We'll get a bunch of samples from the BN:



  +c, -s, +r, +w

  +c, +s, +r, +w

  -c, +s, +r, -w

  +c, -s, +r, +w

  -c, -s, -r, +w

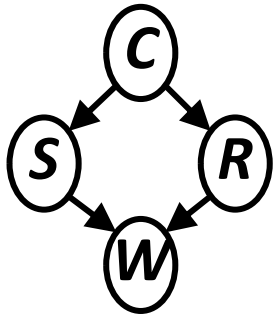
- If we want to know P(W)
  - ✓ We have counts <+w:4, -w:1>
  - ✓ Normalize to get P(W) = <+w:0.8, -w:0.2>
  - ✓ This will get closer to the true distribution with more samples
  - ✓ Can estimate anything else, too
  - ✓ What about P(C| +w)? P(C| +r, +w)? P(C| -r, -w)?
  - ✓ Fast: can use fewer samples if less time (what's the drawback?)

Sampling in Bayes' Nets

- Prior Sampling
- Rejection Sampling
- Likelihood Weighting
- Gibbs Sampling

Rejection Sampling

- Let's say we want P(C)
  - ✓ No point keeping all samples around
  - ✓ Just tally counts of C as we go



- Let's say we want P(C| +s)
  - ✓ Same thing: tally C outcomes, but ignore (reject) samples which don't have S=+s
  - ✓ This is called rejection sampling
  - ✓ It is also consistent for conditional probabilities (i.e., correct in the limit)

+c, -s, +r, +w

+c, +s, +r, +w

-c, +s, +r, -w

+c, -s, +r, +w

-c, -s, -r, +w

Rejection Sampling

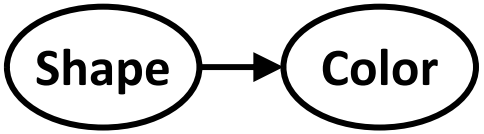
- IN: evidence instantiation
- For i=1, 2, …, n
  - ✓ Sample x_i from $P(X_i \mid Parents(X_i))$
  - ✓ If x_i not consistent with evidence
    - Reject: Return, and no sample is generated in this cycle
- Return $(x_1, x_2, \ldots, x_n)$

Sampling in Bayes' Nets

- Prior Sampling
- Rejection Sampling
- Likelihood Weighting
- Gibbs Sampling

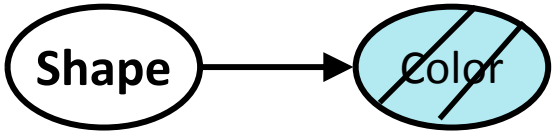
Likelihood Weighting

- Problem with rejection sampling:
  - ✓ If evidence is unlikely, rejects lots of samples
  - ✓ Evidence not exploited as you sample
  - ✓ Consider P(Shape|blue)



~~pyramid, green~~
~~pyramid, red~~
sphere, blue
~~cube, red~~
~~sphere, green~~

- Idea: fix evidence variables and sample the rest
  - Problem: sample distribution not consistent!
  - Solution: weight by probability of evidence given parents

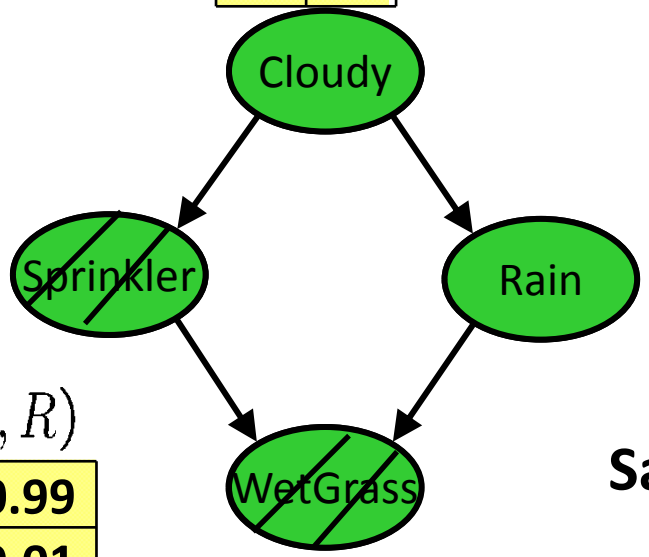


pyramid, blue
pyramid, blue
sphere, blue
cube, blue
sphere, blue

Likelihood Weighting

$P(C)$

+c	0.5
-c	0.5



$P(S|C)$

+c	+s	0.1
	-s	0.9
-c	+s	0.5
	-s	0.5

$P(R|C)$

+c	+r	0.8
	-r	0.2
-c	+r	0.2
	-r	0.8

$P(W|S,R)$

+s	+r	+w	0.99
		-w	0.01
	-r	+w	0.90
		-w	0.10
-s	+r	+w	0.90
		-w	0.10
	-r	+w	0.01
		-w	0.99

Samples:

+c, +s, +r, +w

…

$$w = 1.0 \times 0.1 \times 0.99$$

Likelihood Weighting

- IN: evidence instantiation
- w = 1.0
- for i=1, 2, …, n
  - ✓ if X_i is an evidence variable
    - X_i = observation x_i for X_i
    - Set w = w * $P(x_i \mid Parents(X_i))$
  - ✓ else
    - Sample x_i from $P(X_i \mid Parents(X_i))$
- return $(x_1, x_2, \ldots, x_n)$, w

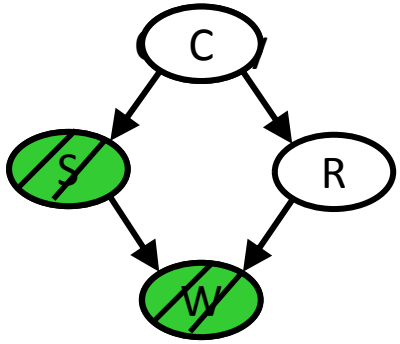
Likelihood Weighting

- Sampling distribution if z sampled and e fixed evidence

$$S_{WS}(\mathbf{z}, \mathbf{e}) = \prod_{i=1}^{l} P(z_i | \text{Parents}(Z_i))$$

- Now, samples have weights

$$w(\mathbf{z}, \mathbf{e}) = \prod_{i=1}^{m} P(e_i | \text{Parents}(E_i))$$



- Together, weighted sampling distribution is consistent

$$S_{\text{WS}}(z, e) \cdot w(z, e) = \prod_{i=1}^{l} P(z_i | \text{Parents}(z_i)) \prod_{i=1}^{m} P(e_i | \text{Parents}(e_i))$$

$$= P(\mathbf{z}, \mathbf{e})$$

Likelihood Weighting

- Likelihood weighting is good
  - ✓ We have taken evidence into account as we generate the sample
  - ✓ E.g. here, W' s value will get picked based on the evidence values of S, R
  - ✓ More of our samples will reflect the state of the world suggested by the evidence
- Likelihood weighting doesn't solve all our problems
  - ✓ Evidence influences the choice of downstream variables, but not upstream ones (C isn't more likely to get a value matching the evidence)
- We would like to consider evidence when we sample every variable

  → Gibbs sampling

Sampling in Bayes' Nets

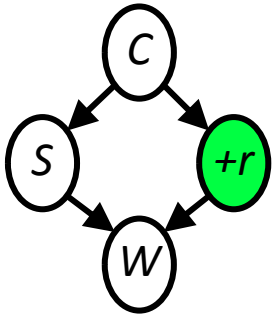
- Prior Sampling
- Rejection Sampling
- Likelihood Weighting
- Gibbs Sampling

Gibbs Sampling

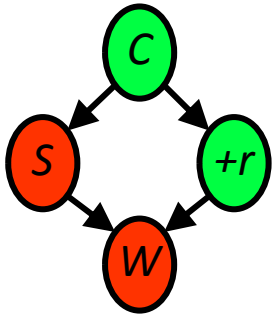
- ***Procedure:*** **keep track of a full instantiation $x_1, x_2, \ldots, x_n$. Start with an arbitrary instantiation consistent with the evidence. Sample one variable at a time, conditioned on all the rest, but keep evidence fixed. Keep repeating this for a long time.**
- ***Property:*** **in the limit of repeating this infinitely many times the resulting sample is coming from the correct distribution.**
- ***Rationale*****: both upstream and downstream variables condition on evidence.**
- **In contrast: likelihood weighting only conditions on upstream evidence, and hence weights obtained in likelihood weighting can sometimes be very small. Sum of weights over all samples is indicative of how many "effective" samples were obtained, so want high weight.**

Gibbs Sampling Example: P(S | +r)

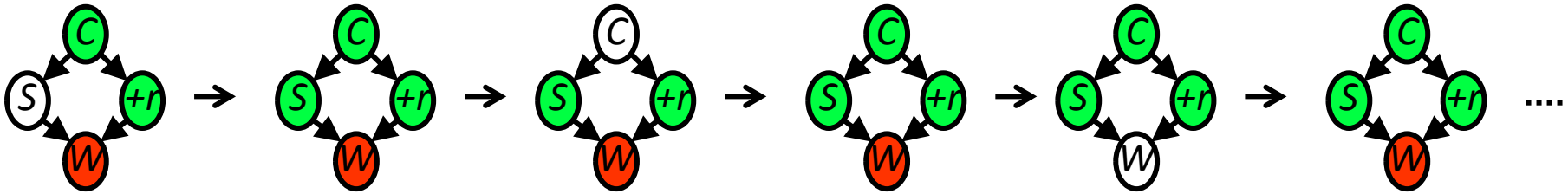
- **Step 1: Fix evidence**
  - ✓ R = +r



- **Step 2: Initialize other variables**
  - Randomly



- **Steps 3: Repeat**
  - ✓ Choose a non-evidence variable X
  - ✓ Resample X from P(X | all other variables)



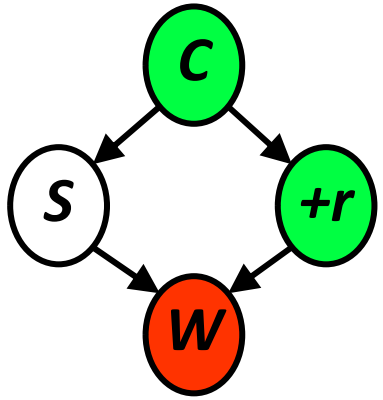
Sample from $P(S|+c,-w,+r)$ Sample from $P(C|+s,-w,+r)$ Sample from $P(W|+s,+c,+r)$

Gibbs Sampling

- How is this better than sampling from the full joint?
  - ✓ In a Bayes' Net, sampling a variable given all the other variables (e.g. P(R|S,C,W)) is usually much easier than sampling from the full joint distribution
    - Only requires a join on the variable to be sampled (in this case, a join on R)
    - The resulting factor only depends on the variable's parents, its children, and its children's parents (this is often referred to as its Markov blanket)

Efficient Resampling of One Variable

- Sample from P(S | +c, +r, -w)



$$
\begin{aligned}
P(S|+c,+r,-w) &= \frac{P(S,+c,+r,-w)}{P(+c,+r,-w)} \\
&= \frac{P(S,+c,+r,-w)}{\sum_s P(s,+c,+r,-w)} \\
&= \frac{P(+c)P(S|+c)P(+r|+c)P(-w|S,+r)}{\sum_s P(+c)P(s|+c)P(+r|+c)P(-w|s,+r)} \\
&= \frac{P(+c)P(S|+c)P(+r|+c)P(-w|S,+r)}{P(+c)P(+r|+c)\sum_s P(s|+c)P(-w|s,+r)} \\
&= \frac{P(S|+c)P(-w|S,+r)}{\sum_s P(s|+c)P(-w|s,+r)}
\end{aligned}
$$

- Many things cancel out – only CPTs with S remain!
- More generally: only CPTs that have resampled variable need to be considered, and joined together

Sampling in Bayes' Nets

- Prior Sampling
- Rejection Sampling
- Likelihood Weighting
- Gibbs Sampling

Thank you

End of Chapter 14