

# Bays Nets and Markov Chains

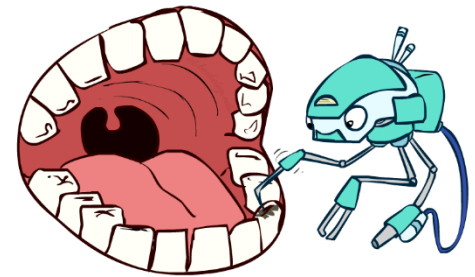
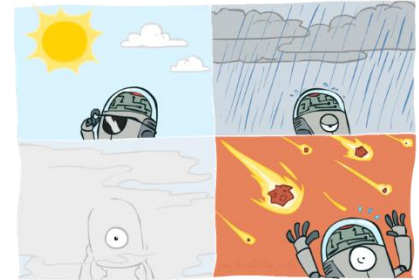
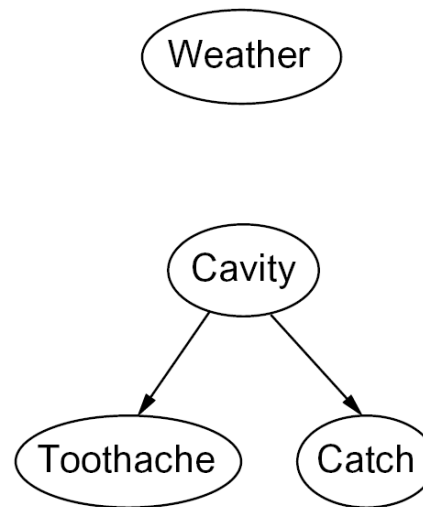
With slides from Dan Klein and Stuart Russell

# Today

- Inference in Bayes Nets
  - Enumeration
- Quiz
- Special case of Bayes Nets: Markov models
- Hidden Markov Models
- Project 4: Ghost busters (Q&A)

# 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
- For now: imagine that arrows mean direct causation (not true in general)

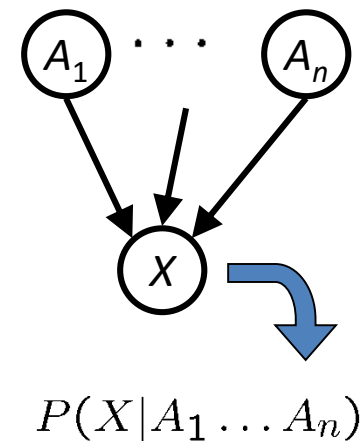


# Bayes' Net Semantics

- 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

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

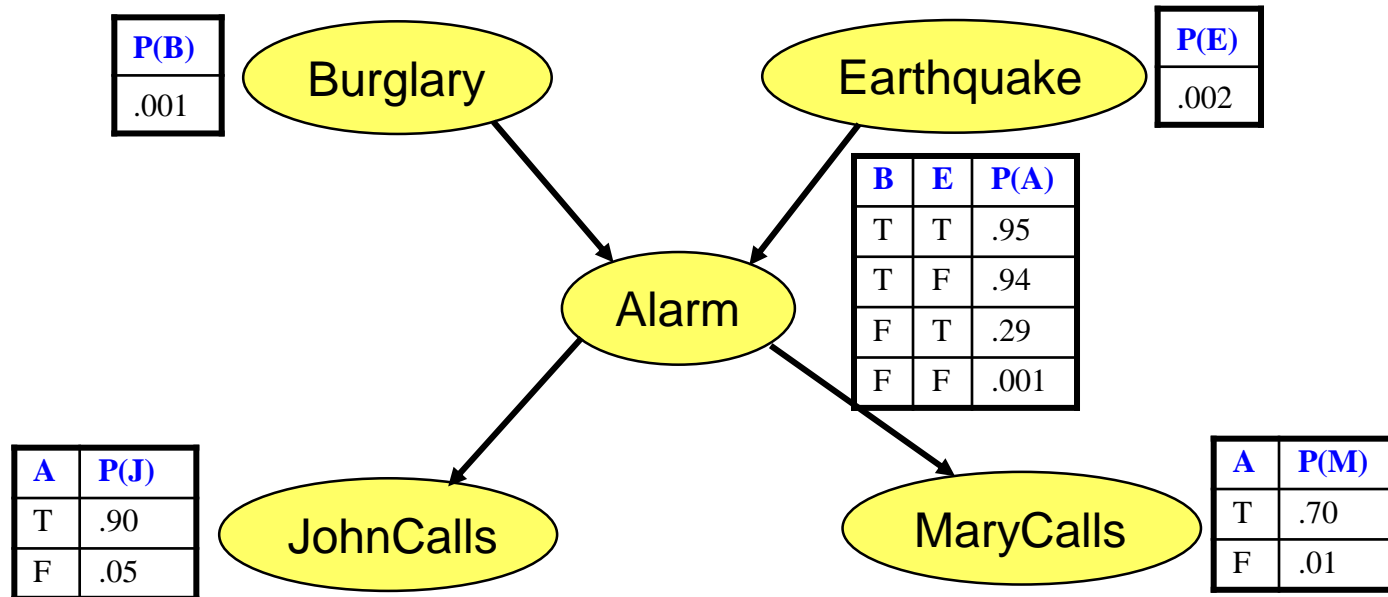
- CPT: conditional probability table
- Description of a noisy “causal” process



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

# Conditional Probability Tables

- Each node has a **conditional probability table (CPT)** that gives the probability of each of its values given every possible combination of values for its parents (conditioning case).
  - Roots (sources) of the DAG that have no parents are given prior probabilities.



# CPT Comments

- Probability of Node=false not given, can subtract from 1:

B	E	P(A=T)
T	T	.95



B	E	P(A=F)
T	T	.05

- CPT rows do not need to add up to one – they are NOT NORMALIZED. (convenient for inference)
- Example requires 10 parameters rather than  $2^5 - 1 = 31$  for specifying the full joint distribution.
- Number of parameters in the CPT for a node is exponential in the number of parents (fan-in).

# Joint Distributions for Bayes Nets

- A Bayesian Network implicitly defines a joint distribution.

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^n P(x_i \mid \text{Parents}(X_i))$$

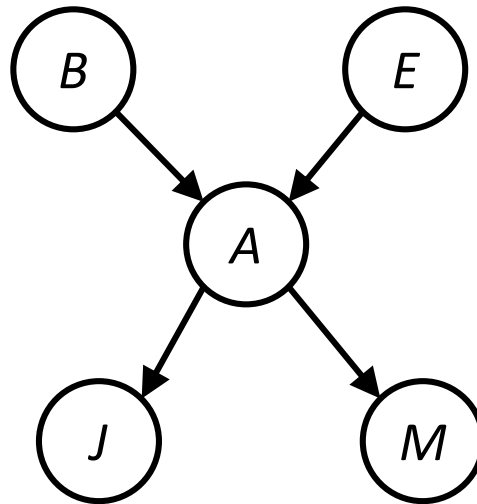
- Example

$$\begin{aligned} &P(J \wedge M \wedge A \wedge \neg B \wedge \neg E) \\ &= P(J \mid A)P(M \mid A)P(A \mid \neg B \wedge \neg E)P(\neg B)P(\neg E) \\ &= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998 = 0.00062 \end{aligned}$$

- An inefficient approach to inference is:
  - 1) Compute the joint distribution using this equation.
  - 2) Compute any desired conditional probability using the joint distribution.

# Example: Alarm Network

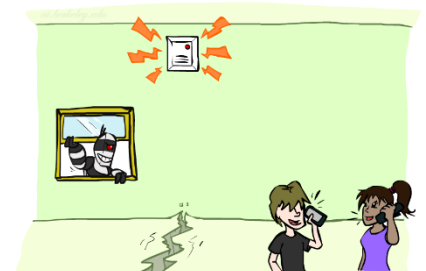
B	P(B)
+b	0.001
-b	0.999



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



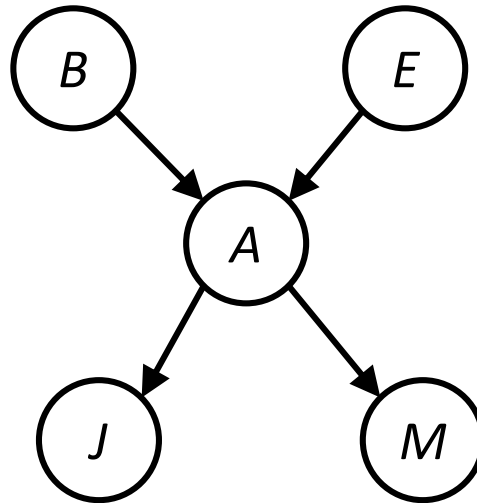
$$P(+b, -e, +a, -j, +m) =$$

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



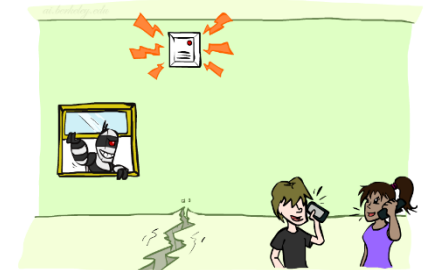
# Example: Alarm Network

B	P(B)
+b	0.001
-b	0.999



E	P(E)
+e	0.002
-e	0.998

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



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

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

$$\begin{aligned}
 P(+b, -e, +a, -j, +m) &= \\
 P(+b)P(-e)P(+a|+b, -e)P(-j|+a)P(+m|+a) &= \\
 0.001 \times 0.998 \times 0.94 \times 0.1 \times 0.7 &
 \end{aligned}$$

# Inference by Enumeration

- General case:

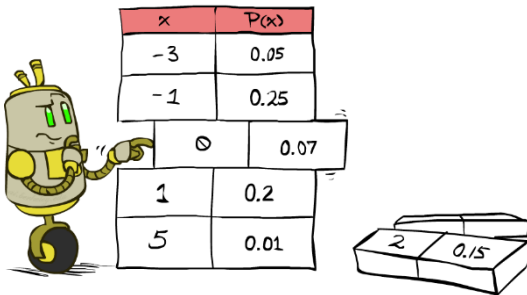
- Evidence variables:  $E_1 \dots E_k = e_1 \dots e_k$
  - Query\* variable:  $Q$
  - Hidden variables:  $H_1 \dots H_r$
- $$\left. \begin{array}{l} E_1 \dots E_k = e_1 \dots e_k \\ Q \\ H_1 \dots H_r \end{array} \right\} \begin{array}{l} X_1, X_2, \dots X_n \\ \text{All} \\ \text{variables} \end{array}$$

- We want:

*\* Works fine with multiple query variables, too*

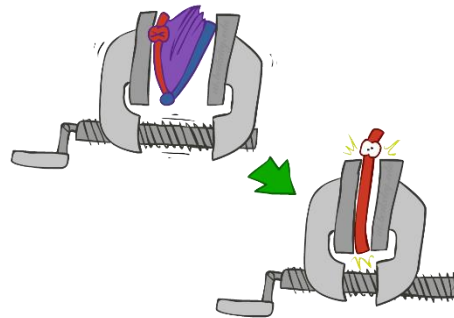
$$P(Q|e_1 \dots e_k)$$

- Step 1: Select the entries consistent with the evidence



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

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



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

- Step 3: Normalize

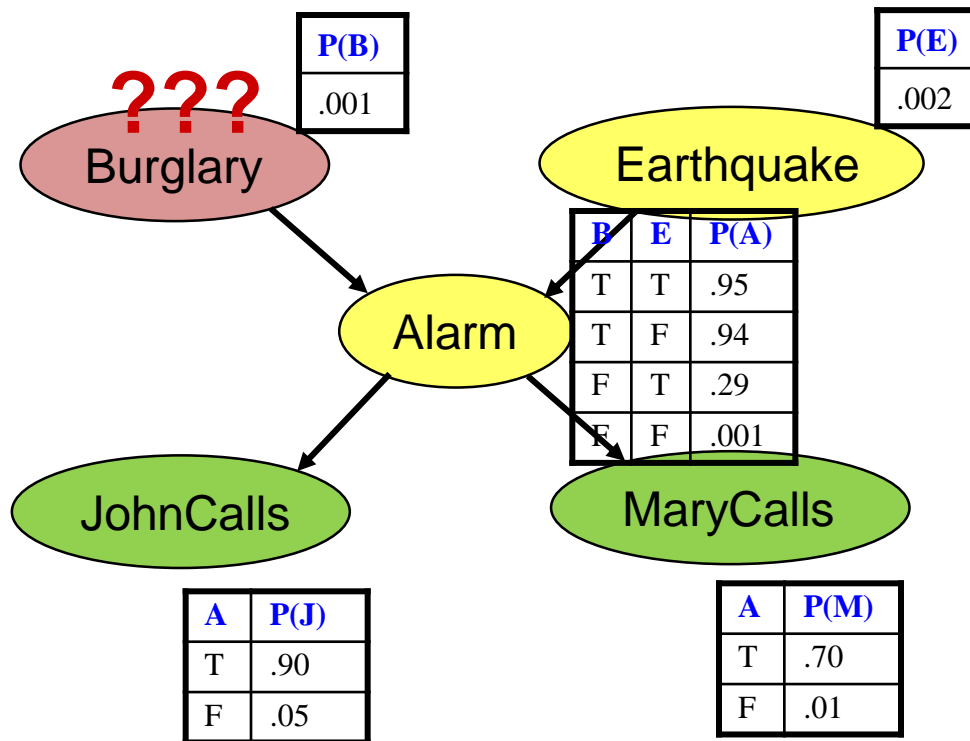
$$\times \frac{1}{Z}$$

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

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

# Bayes Net Inference: Example

- Example: Given that John and Mary call (+j, +m), what is the probability that there is a Burglary (+b)?



$$P(+b \mid +j, +m) = ?$$

# Inference by Enumeration in Bayes' Net

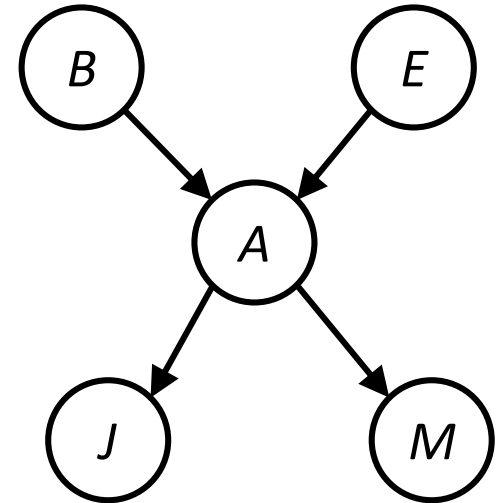
- Given unlimited time, inference in BNs is easy
- Reminder of inference by enumeration by example:

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

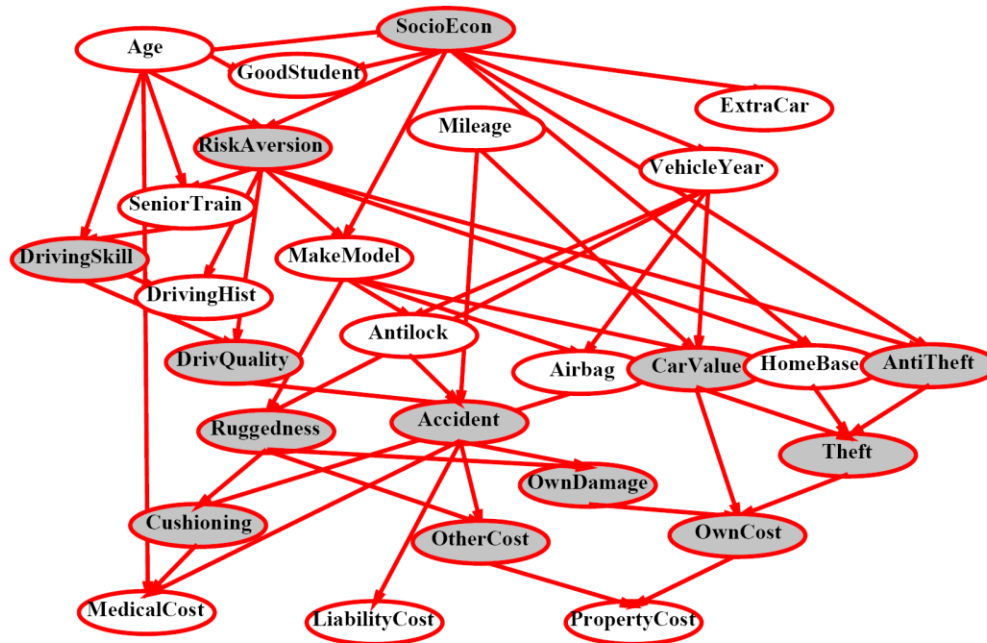
$$= \sum_{e,a} P(B, e, a, +j, +m)$$

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

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



# Inference by Enumeration?



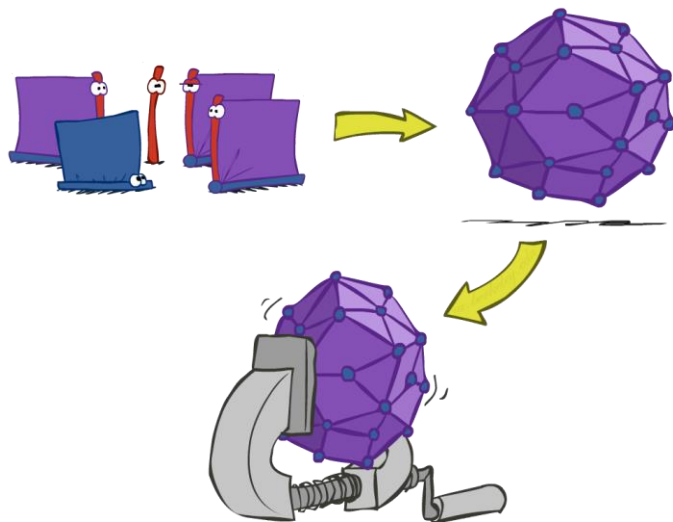
$$P(Antilock|observed\ variables) = ?$$

# Complexity of Bayes Net Inference

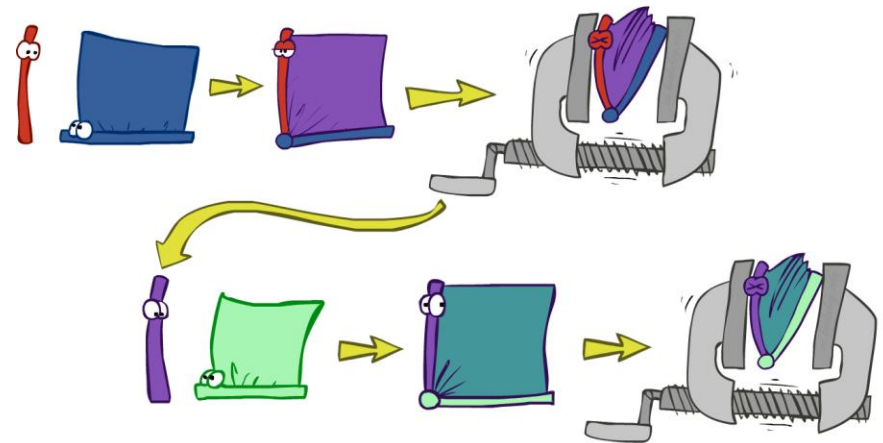
- In general, the problem of Bayes Net inference is NP-hard (exponential in the size of the graph).
- For **singly-connected networks** or **polytrees** in which there are no undirected loops, there are linear-time algorithms based on **belief propagation**.
  - Each node sends local evidence messages to their children and parents.
  - Each node updates belief in each of its possible values based on incoming messages from its neighbors and propagates evidence on to its neighbors.
- There are approximations to inference for general networks based on **loopy belief propagation** that iteratively refines probabilities that converge to accurate values in the limit.

# 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

# Inference by Enumeration: Procedural Outline

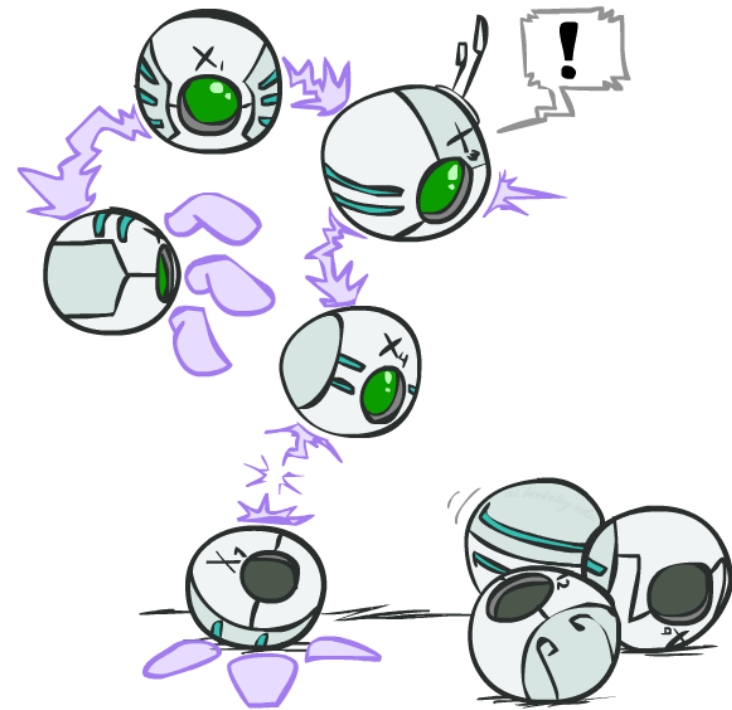
- Track objects called **factors**
- Initial factors are local CPTs (one per node)

$P(R)$		$P(T R)$			$P(L T)$		
+r	0.1	+r	+t	0.8	+t	+l	0.3
-r	0.9	+r	-t	0.2	+t	-l	0.7
		-r	+t	0.1	-t	+l	0.1
		-r	-t	0.9	-t	-l	0.9

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

$P(R)$		$P(T R)$			$P(+l T)$		
+r	0.1	+r	+t	0.8	+t	+l	0.3
-r	0.9	+r	-t	0.2	-t	+l	0.1
		-r	+t	0.1			
		-r	-t	0.9			

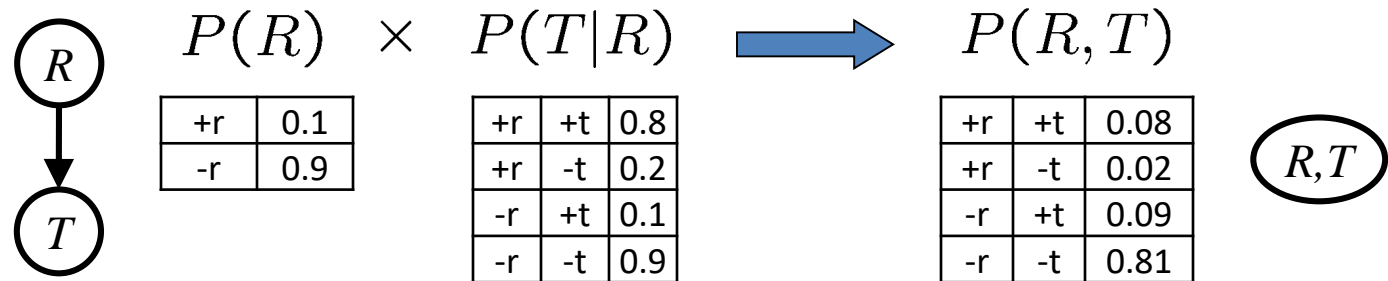
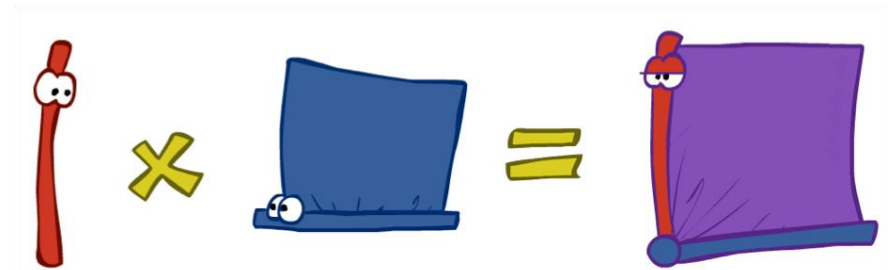
- Procedure: Join all factors, then eliminate all hidden variables





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

# 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:

$$P(R, T)$$

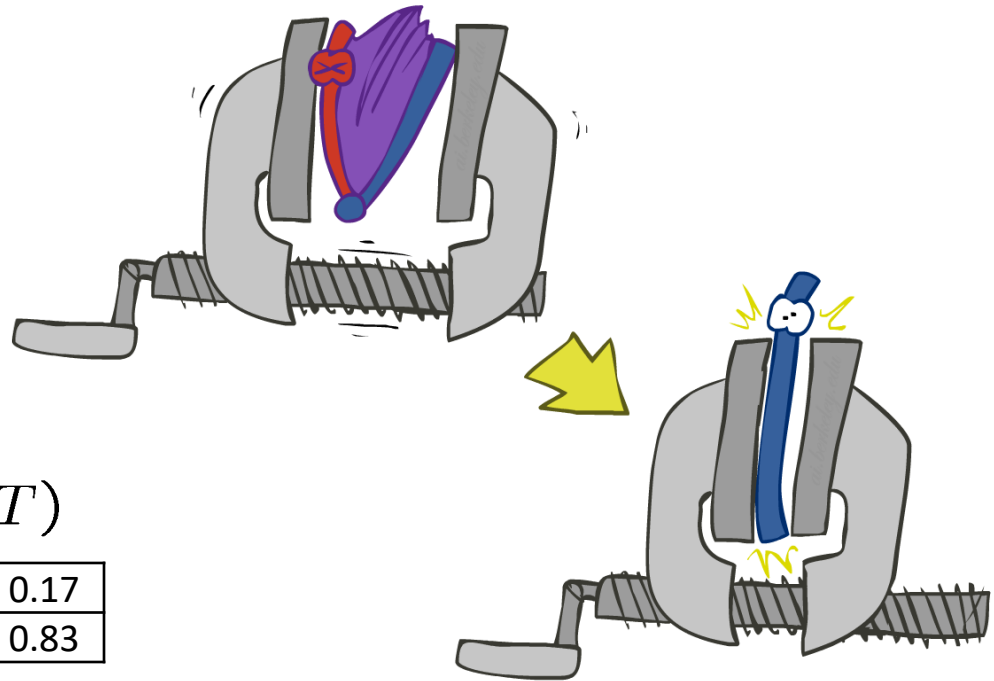
+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

sum  $R$

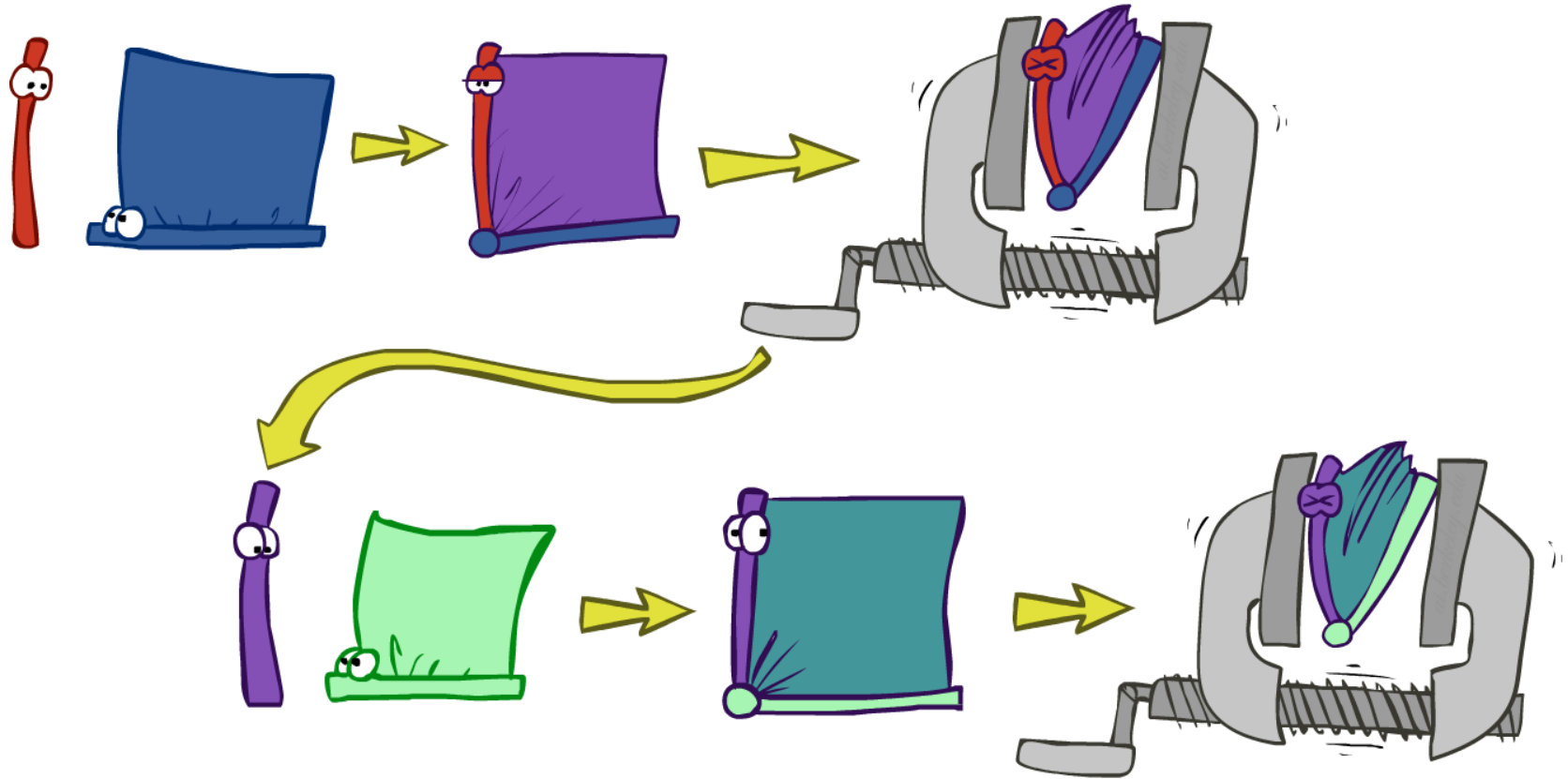


$$P(T)$$

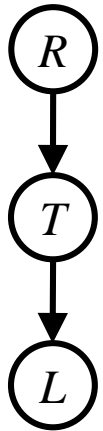
+t	0.17
-t	0.83



# Marginalizing Early (= Variable Elimination)



# Traffic Domain



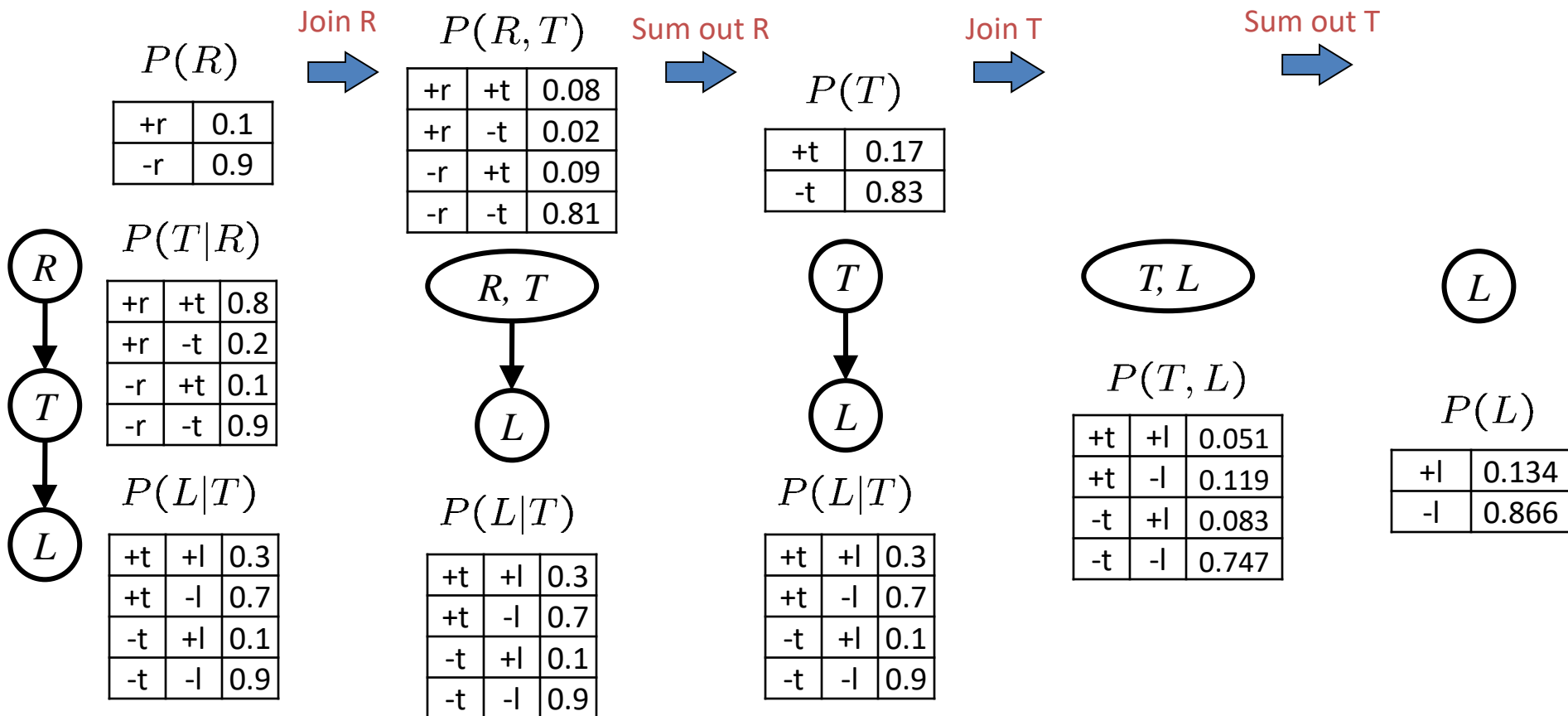
- $P(L) = ?$   
Inference by Enumeration

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

- Variable Elimination

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

# Marginalizing Early! (aka VE)



# 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

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

$$P(+r)$$

+r	0.1
----	-----

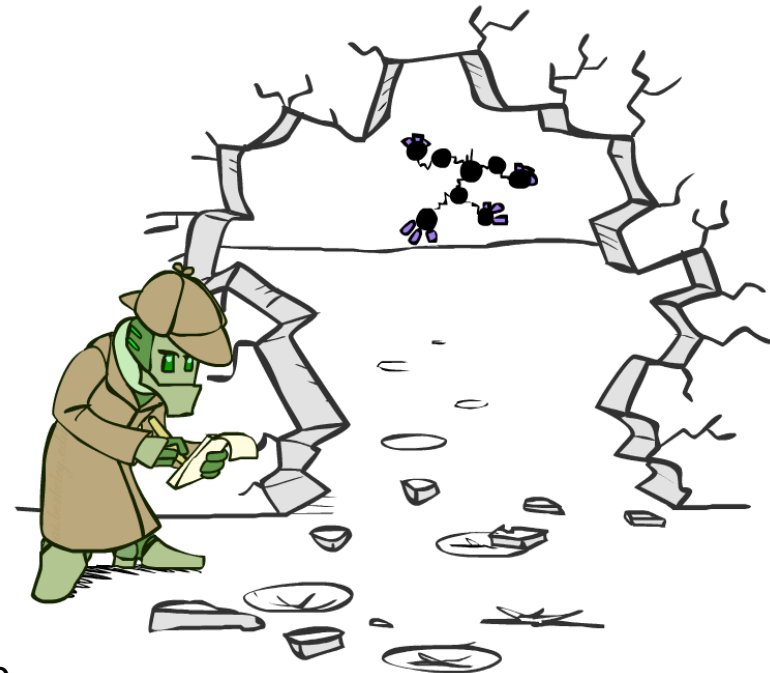
$$P(T|+r)$$

+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 vars other than query + evidence



# Evidence II

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

$$P(+r, L)$$

+r	+l	0.026
+r	-l	0.074

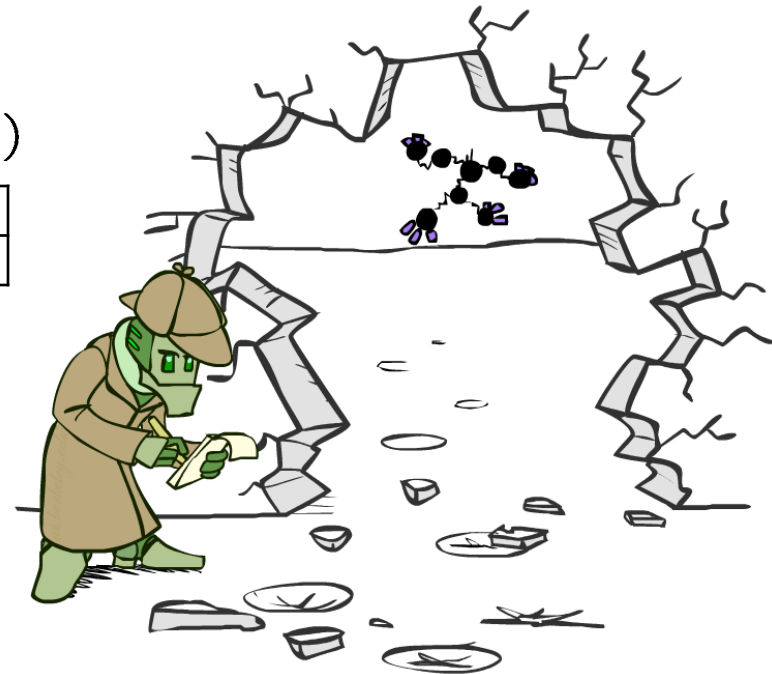
Normalize



$$P(L \mid +r)$$

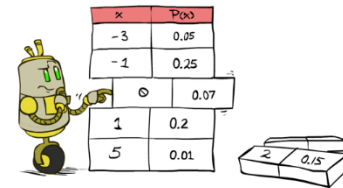
+l	0.26
-l	0.74

- To get our answer, just normalize this!
- That's it!

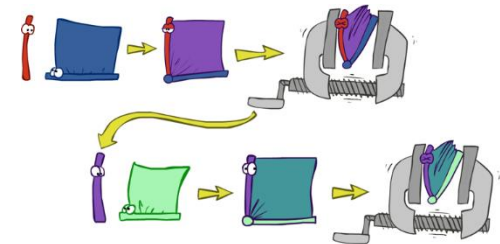


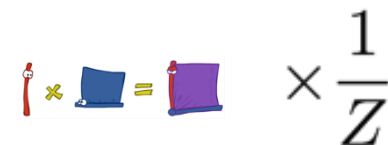
# General Variable Elimination

- 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



x	p(x)
-3	0.05
-1	0.25
0	0.07
1	0.2
5	0.01



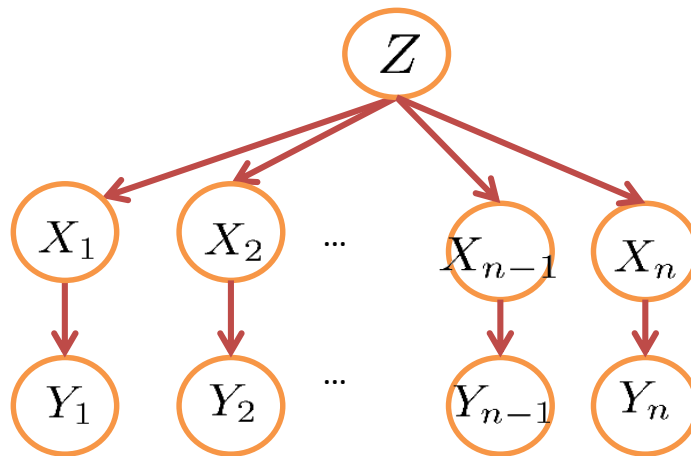


$$\text{red line} \times \text{blue square} = \text{purple square} \times \frac{1}{Z}$$



# Variable Elimination Ordering

- For the query  $P(X_n | y_1, \dots, y_n)$  work through the following two different orderings as done in previous slide:  $Z, X_1, \dots, X_{n-1}$  and  $X_1, \dots, X_{n-1}, Z$ . 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

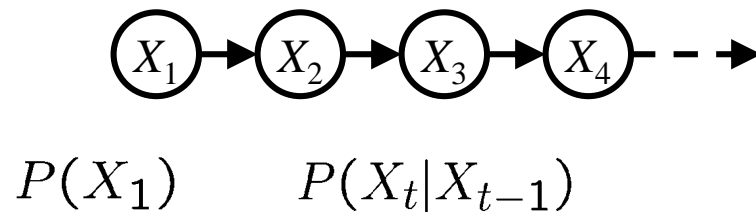
- The computational and space complexity of variable elimination is determined by the largest factor
- The elimination ordering can greatly affect the size of the largest factor.
  - E.g., previous slide's example  $2^n$  vs. 2
- Does there always exist an ordering that only results in small factors?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

# Quiz: BN Inference

---

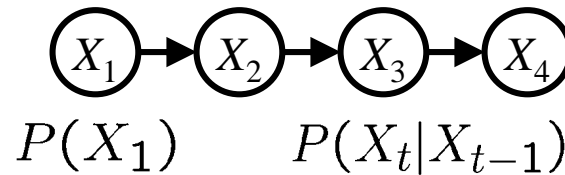
# Markov Models

- Value of  $X$  at a given time is called the **state**



- Parameters: called **transition probabilities** or dynamics, specify how the state evolves over time (also, initial state probabilities)
- Stationarity assumption: transition probabilities the same at all times
- Same as MDP transition model, but no choice of action

# Joint Distribution of a Markov Model



- Joint distribution:

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X_3|X_2)P(X_4|X_3)$$

- More generally:

$$\begin{aligned} P(X_1, X_2, \dots, X_T) &= P(X_1)P(X_2|X_1)P(X_3|X_2) \dots P(X_T|X_{T-1}) \\ &= P(X_1) \prod_{t=2}^T P(X_t|X_{t-1}) \end{aligned}$$

- Questions to be resolved:

- Does this indeed define a joint distribution?
- Can every joint distribution be factored this way, or are we making some assumptions about the joint distribution by using this factorization?

# Markov Models Recap

- Explicit assumption for all  $t$ :  $X_t \perp\!\!\!\perp X_1, \dots, X_{t-2} \mid X_{t-1}$
- Consequence, joint distribution can be written as:

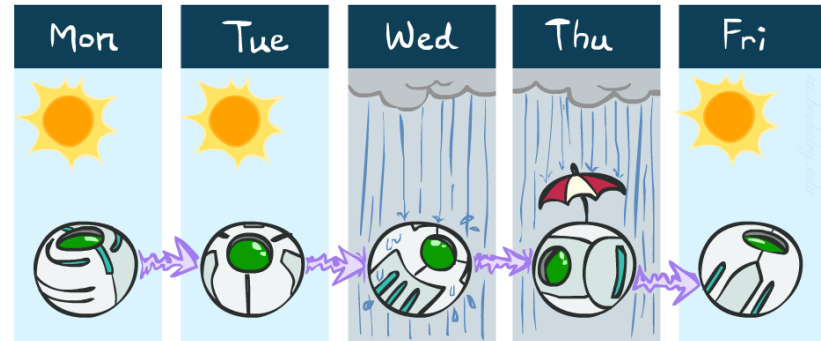
$$\begin{aligned} P(X_1, X_2, \dots, X_T) &= P(X_1)P(X_2|X_1)P(X_3|X_2) \dots P(X_T|X_{T-1}) \\ &= P(X_1) \prod_{t=2}^T P(X_t|X_{t-1}) \end{aligned}$$

- Implied conditional independencies: (try to prove this!)
    - Past variables independent of future variables given the present
- i.e., if  $t_1 < t_2 < t_3$  or  $t_1 > t_2 > t_3$  then:  $X_{t_1} \perp\!\!\!\perp X_{t_3} \mid X_{t_2}$
- Additional explicit assumption:  $P(X_t \mid X_{t-1})$  is the same for all  $t$

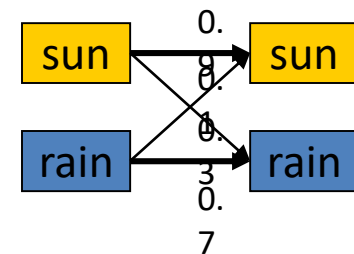
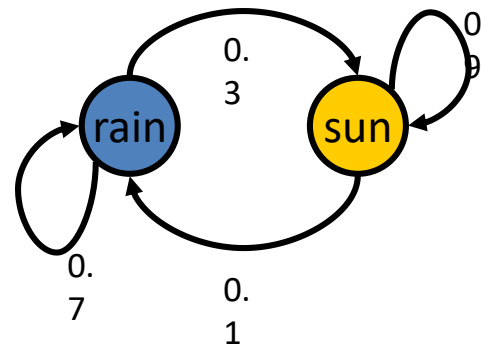
# Example Markov Chain: Weather

- States:  $X = \{\text{rain}, \text{sun}\}$
- Initial distribution: 1.0 sun
- CPT  $P(X_t | X_{t-1})$ :

$X_{t-1}$	$X_t$	$P(X_t   X_{t-1})$
sun	sun	0.9
sun	rain	0.1
rain	sun	0.3
rain	rain	0.7

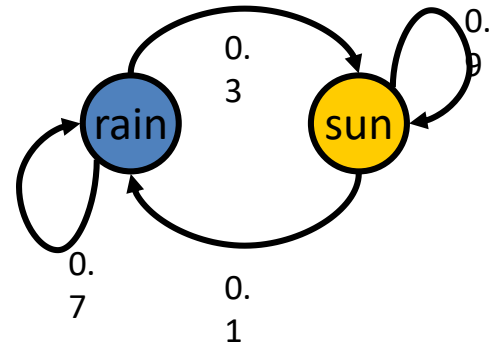


Two new ways of representing the same CPT



# Example Markov Chain: Weather

- Initial distribution: 1.0 sun



- What is the probability distribution after one step?

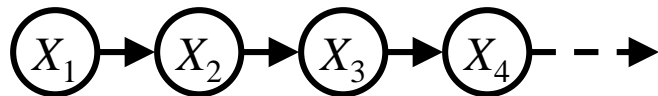
$$P(X_2 = \text{sun}) = P(X_2 = \text{sun} | X_1 = \text{sun})P(X_1 = \text{sun}) + P(X_2 = \text{sun} | X_1 = \text{rain})P(X_1 = \text{rain})$$

$$0.9 \cdot 1.0 + 0.3 \cdot 0.0 = 0.9$$



# Mini-Forward Algorithm

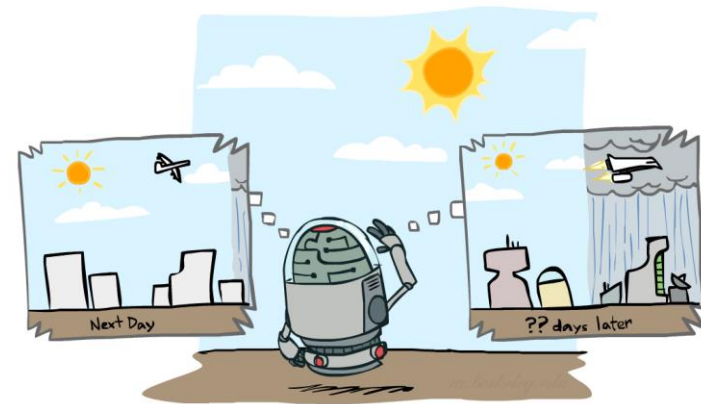
- Question: What's  $P(X)$  on some day  $t$ ?



$P(x_1)$  = known

$$\begin{aligned} P(x_t) &= \sum_{x_{t-1}} P(x_{t-1}, x_t) \\ &= \sum_{x_{t-1}} P(x_t \mid x_{t-1}) P(x_{t-1}) \end{aligned}$$

*Forward simulation* ←



## Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\begin{array}{ccccc}
 \left\langle \begin{array}{c} 1.0 \\ 0.0 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.9 \\ 0.1 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.84 \\ 0.16 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.804 \\ 0.196 \end{array} \right\rangle & \longrightarrow \left\langle \begin{array}{c} 0.75 \\ 0.25 \end{array} \right\rangle \\
 P(X_1) & P(X_2) & P(X_3) & P(X_4) & P(X_\infty)
 \end{array}$$

- From initial observation of rain

$$\begin{array}{ccccc}
 \left\langle \begin{array}{c} 0.0 \\ 1.0 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.3 \\ 0.7 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.48 \\ 0.52 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.588 \\ 0.412 \end{array} \right\rangle & \longrightarrow \left\langle \begin{array}{c} 0.75 \\ 0.25 \end{array} \right\rangle \\
 P(X_1) & P(X_2) & P(X_3) & P(X_4) & P(X_\infty)
 \end{array}$$

- From yet another initial distribution  $P(X_1)$ :

$$\begin{array}{ccc}
 \left\langle \begin{array}{c} p \\ 1 - p \end{array} \right\rangle & \dots & \longrightarrow \left\langle \begin{array}{c} 0.75 \\ 0.25 \end{array} \right\rangle \\
 P(X_1) & & P(X_\infty)
 \end{array}$$

# Stationary Distributions

- For most chains:

- Influence of the initial distribution gets less and less over time.
- The distribution we end up in is independent of the initial distribution

- **Stationary distribution:**

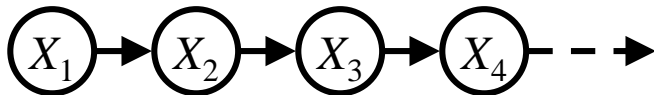
- The distribution we end up with is called the **stationary distribution** of the chain  $P_\infty$
- It satisfies

$$P_\infty(X) = P_{\infty+1}(X) = \sum_x P(X|x)P_\infty(x)$$



# Example: Stationary Distributions

- Question: What's  $P(X)$  at time  $t = \text{infinity}$ ?



$$P_{\infty}(\text{sun}) = P(\text{sun}|\text{sun})P_{\infty}(\text{sun}) + P(\text{sun}|\text{rain})P_{\infty}(\text{rain})$$

$$P_{\infty}(\text{rain}) = P(\text{rain}|\text{sun})P_{\infty}(\text{sun}) + P(\text{rain}|\text{rain})P_{\infty}(\text{rain})$$

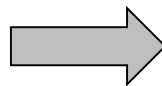
$$P_{\infty}(\text{sun}) = 0.9P_{\infty}(\text{sun}) + 0.3P_{\infty}(\text{rain})$$

$$P_{\infty}(\text{rain}) = 0.1P_{\infty}(\text{sun}) + 0.7P_{\infty}(\text{rain})$$

$$P_{\infty}(\text{sun}) = 3P_{\infty}(\text{rain})$$

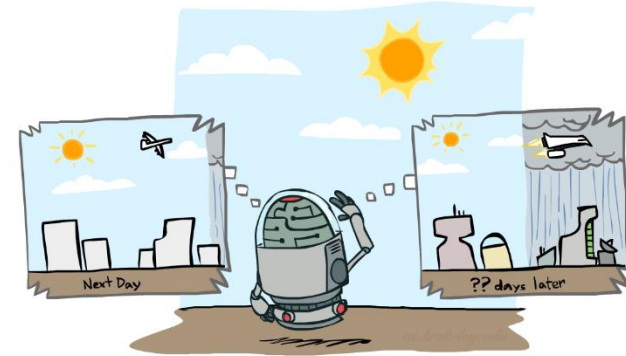
$$P_{\infty}(\text{rain}) = 1/3P_{\infty}(\text{sun})$$

Also:  $P_{\infty}(\text{sun}) + P_{\infty}(\text{rain}) = 1$



$$P_{\infty}(\text{sun}) = 3/4$$

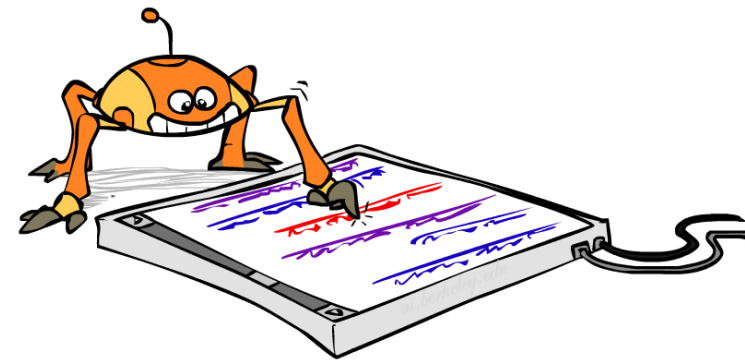
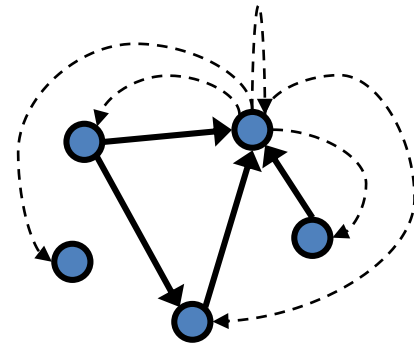
$$P_{\infty}(\text{rain}) = 1/4$$



$X_{t-1}$	$X_t$	$P(X_t X_{t-1})$
sun	sun	0.9
sun	rain	0.1
rain	sun	0.3
rain	rain	0.7

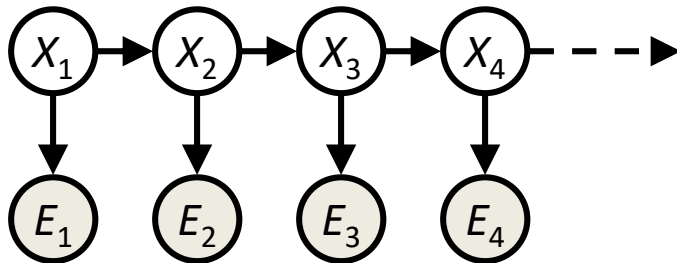
# Application of Stationary Distribution: Web Link Analysis

- PageRank over a web graph
  - Each web page is a state
  - Initial distribution: uniform over pages
  - Transitions:
    - With prob.  $c$ , uniform jump to a random page (dotted lines, not all shown)
    - With prob.  $1-c$ , follow a random outlink (solid lines)
- Stationary distribution
  - Will spend more time on highly reachable pages
  - E.g. many ways to get to the Acrobat Reader download page
  - Somewhat robust to link spam
  - Google 1.0 returned the set of pages containing all your keywords in decreasing rank, now all search engines use link analysis along with many other factors (rank actually getting less important over time)

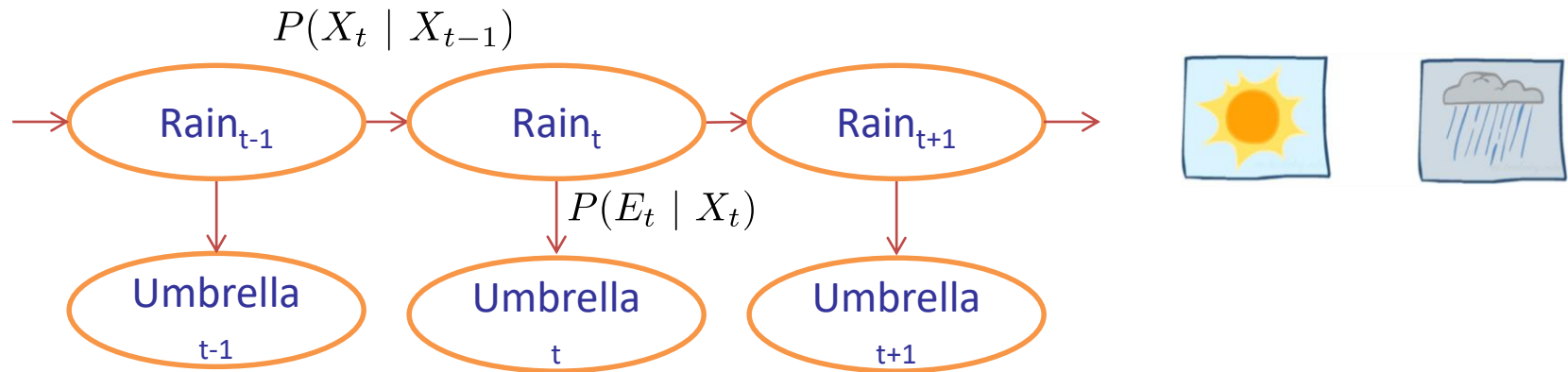


# Hidden Markov Models

- Markov chains not so useful for most agents
  - Need observations to update your beliefs
- Hidden Markov models (HMMs)
  - Underlying Markov chain over states  $X$
  - You observe outputs (effects) at each time step



# Example: Weather HMM



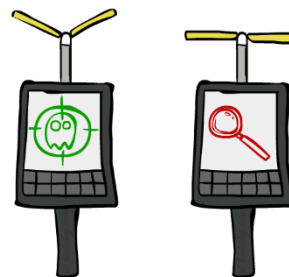
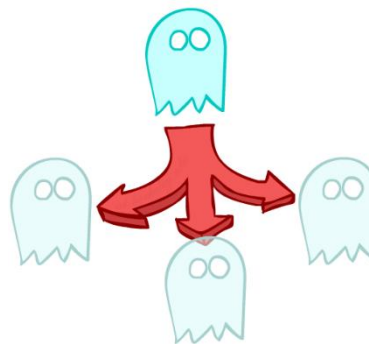
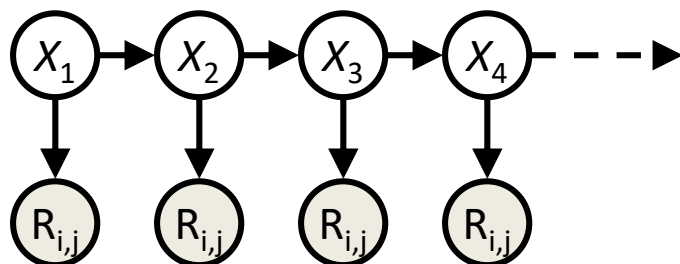
- An HMM is defined by:
  - Initial distribution:  $P(X_1)$
  - Transitions:  $P(X_t | X_{t-1})$
  - Emissions:  $P(E_t | X_t)$

$R_t$	$R_{t+1}$	$P(R_{t+1}   R_t)$
+r	+r	0.7
+r	-r	0.3
-r	+r	0.3
-r	-r	0.7

$R_t$	$U_t$	$P(U_t   R_t)$
+r	+u	0.9
+r	-u	0.1
-r	+u	0.2
-r	-u	0.8

# Example: Ghostbusters HMM

- $P(X_1) = \text{uniform}$
- $P(X|X')$  = usually move clockwise, but sometimes move in a random direction or stay in place
- $P(R_{ij}|X)$  = same sensor model as before: red means close, green means far away.



1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

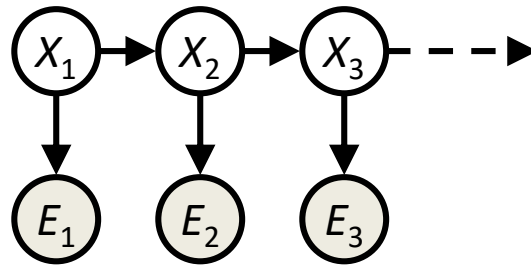
$P(X_1)$

1/6	1/6	1/2
0	1/6	0
0	0	0

$P(X|X' = \langle 1, 2 \rangle)$



# Joint Distribution of an HMM



– Joint distribution:

$$P(X_1, E_1, X_2, E_2, X_3, E_3) = P(X_1)P(E_1|X_1)P(X_2|X_1)P(E_2|X_2)P(X_3|X_2)P(E_3|X_3)$$

– More generally:

$$P(X_1, E_1, \dots, X_T, E_T) = P(X_1)P(E_1|X_1) \prod_{t=2}^T P(X_t|X_{t-1})P(E_t|X_t)$$

– Questions to be resolved:

- Does this indeed define a joint distribution?
- Can every joint distribution be factored this way, or are we making some assumptions about the joint distribution by using this factorization?

# Real HMM Examples

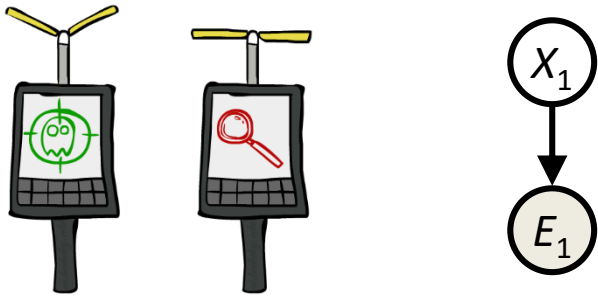
---

- Speech recognition HMMs:
  - Observations are acoustic signals (continuous valued)
  - States are specific positions in specific words (so, tens of thousands)
- Machine translation HMMs:
  - Observations are words (tens of thousands)
  - States are translation options
- Robot tracking:
  - Observations are range readings (continuous)
  - States are positions on a map (continuous)

# Filtering / Monitoring

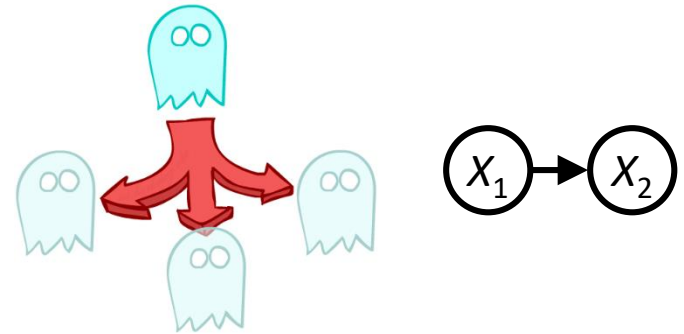
- Filtering, or monitoring, is the task of tracking the distribution  $B_t(X) = P_t(X_t \mid e_1, \dots, e_t)$  (the belief state) over time
- We start with  $B_1(X)$  in an initial setting, usually uniform
- As time passes, or we get observations, we update  $B(X)$
- The Kalman filter was invented in the 60's and first implemented as a method of trajectory estimation for the Apollo program

# Inference: Base Cases



$$P(X_1|e_1)$$

$$\begin{aligned} P(x_1|e_1) &= P(x_1, e_1)/P(e_1) \\ &\propto_{X_1} P(x_1, e_1) \\ &= P(x_1)P(e_1|x_1) \end{aligned}$$



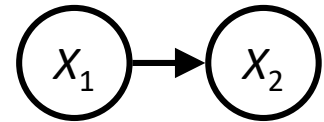
$$P(X_2)$$

$$\begin{aligned} P(x_2) &= \sum_{x_1} P(x_1, x_2) \\ &= \sum_{x_1} P(x_1)P(x_2|x_1) \end{aligned}$$

# Passage of Time

- Assume we have current belief  $P(X \mid \text{evidence to date})$

$$B(X_t) = P(X_t | e_{1:t})$$



- Then, after one time step passes:

$$\begin{aligned} P(X_{t+1} | e_{1:t}) &= \sum_{x_t} P(X_{t+1}, x_t | e_{1:t}) \\ &= \sum_{x_t} P(X_{t+1} | x_t, e_{1:t}) P(x_t | e_{1:t}) \\ &= \sum_{x_t} P(X_{t+1} | x_t) P(x_t | e_{1:t}) \end{aligned}$$

- Or compactly:

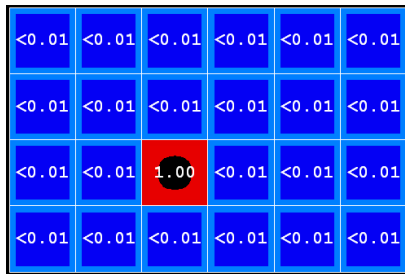
$$B'(X_{t+1}) = \sum_{x_t} P(X' | x_t) B(x_t)$$

- Basic idea: beliefs get “pushed” through the transitions
  - With the “B” notation, we have to be careful about what time step  $t$  the belief is about, and what evidence it includes

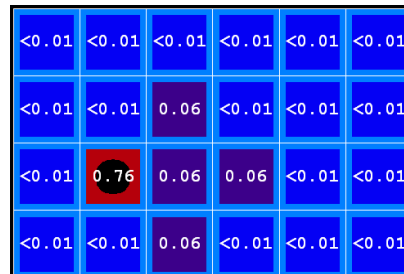
# Example: Passage of Time

- As time passes, uncertainty “accumulates”

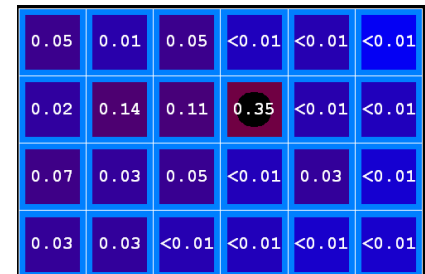
(Transition model: ghosts usually go clockwise)



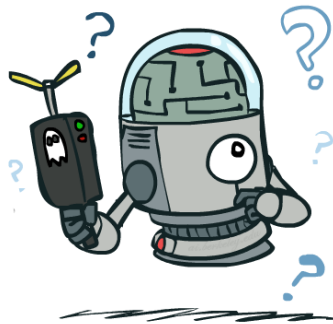
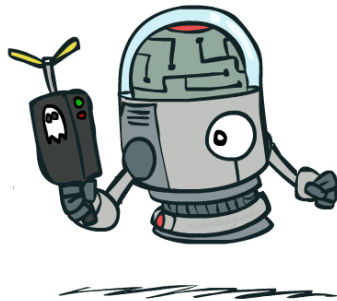
T = 1



T = 2



T = 5



# Observation

- Assume we have current belief  $P(X \mid \text{previous evidence})$ :

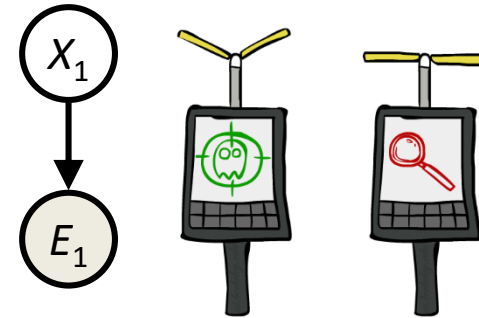
$$B'(X_{t+1}) = P(X_{t+1} | e_{1:t})$$

- Then, after evidence comes in:

$$\begin{aligned} P(X_{t+1} | e_{1:t+1}) &= P(X_{t+1}, e_{t+1} | e_{1:t}) / P(e_{t+1} | e_{1:t}) \\ &\propto_{X_{t+1}} P(X_{t+1}, e_{t+1} | e_{1:t}) \\ &= P(e_{t+1} | e_{1:t}, X_{t+1}) P(X_{t+1} | e_{1:t}) \\ &= P(e_{t+1} | X_{t+1}) P(X_{t+1} | e_{1:t}) \end{aligned}$$

- Or, compactly:

$$B(X_{t+1}) \propto_{X_{t+1}} P(e_{t+1} | X_{t+1}) B'(X_{t+1})$$



- Basic idea: beliefs “reweighted” by likelihood of evidence
- Unlike passage of time, we have to renormalize

# Example: Observation

- As we get observations, beliefs get reweighted, uncertainty “decreases”

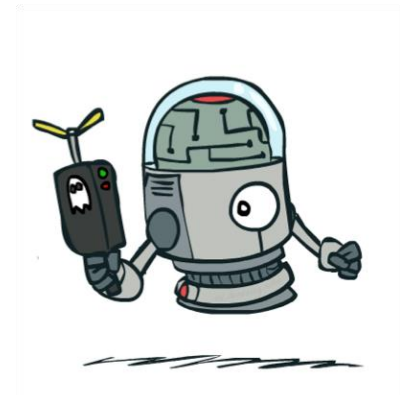
0.05	0.01	0.05	<0.01	<0.01	<0.01
0.02	0.14	0.11	0.35	<0.01	<0.01
0.07	0.03	0.05	<0.01	0.03	<0.01
0.03	0.03	<0.01	<0.01	<0.01	<0.01

Before observation

<0.01	<0.01	<0.01	<0.01	0.02	<0.01
<0.01	<0.01	<0.01	0.83	0.02	<0.01
<0.01	<0.01	0.11	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01

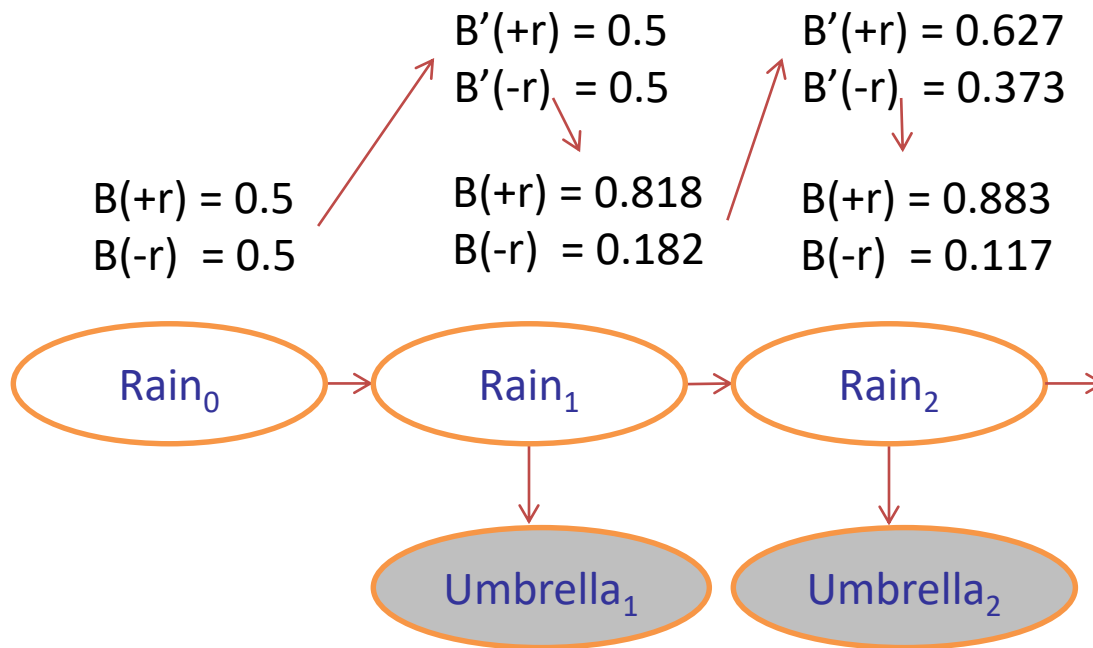
After observation

$$B(X) \propto P(e|X)B'(X)$$





# Example: Weather HMM



$R_t$	$R_{t+1}$	$P(R_{t+1}   R_t)$
$+r$	$+r$	0.7
$+r$	$-r$	0.3
$-r$	$+r$	0.3
$-r$	$-r$	0.7

$R_t$	$U_t$	$P(U_t   R_t)$
$+r$	$+u$	0.9
$+r$	$-u$	0.1
$-r$	$+u$	0.2
$-r$	$-u$	0.8

# The Forward Algorithm

- We are given evidence at each time and want to know

$$B_t(X) = P(X_t|e_{1:t})$$

- We can derive the following updates

$$\begin{aligned} P(x_t|e_{1:t}) &\propto_X P(x_t, e_{1:t}) \\ &= \sum_{x_{t-1}} P(x_{t-1}, x_t, e_{1:t}) \\ &= \sum_{x_{t-1}} P(x_{t-1}, e_{1:t-1}) P(x_t|x_{t-1}) P(e_t|x_t) \\ &= P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1}) \end{aligned}$$

We can normalize as we go if we want to have  $P(x|e)$  at each time step, or just once at the end...

# Online Belief Updates

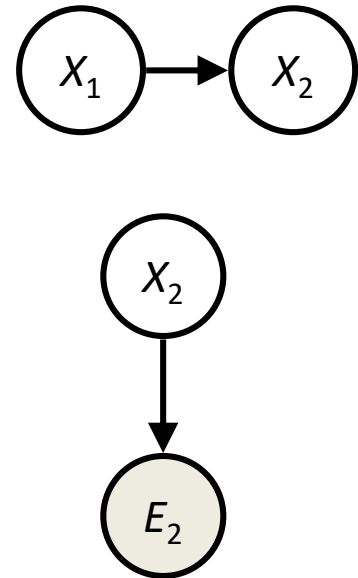
- Every time step, we start with current  $P(X \mid \text{evidence})$
- We update for time:

$$P(x_t | e_{1:t-1}) = \sum_{x_{t-1}} P(x_{t-1} | e_{1:t-1}) \cdot P(x_t | x_{t-1})$$

- We update for evidence:

$$P(x_t | e_{1:t}) \propto_X P(x_t | e_{1:t-1}) \cdot P(e_t | x_t)$$

- The forward algorithm does both at once (and doesn't normalize)



# Project 4: Ghost Busters

- Due **Wednesday, April 5**
- <http://www.mathcs.emory.edu/~eugene/cs325/p4/>

# Next Time: Particle Filtering and Applications of HMMs

---