Announcements

Assignments

- HW10
  - Due Wed 4/17
- HW11
  - Plan: Out tomorrow, due Wed 4/24
- P5
  - Plan: Out tonight, due 5/2

Sampling Wrap-up

Likelihood Weighting

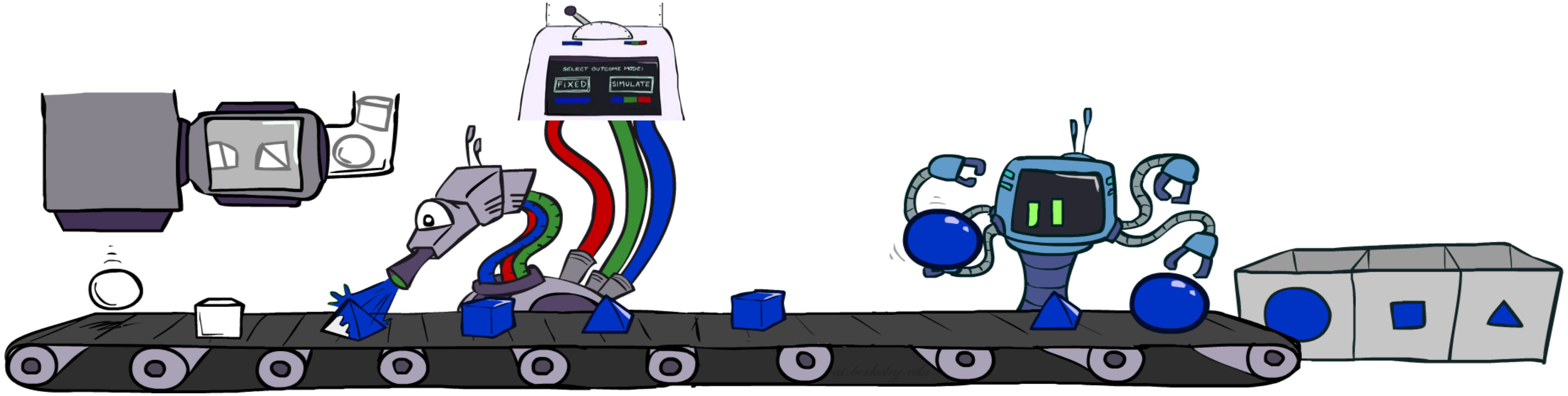
IN: evidence instantiation

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 | Parents(X_i))
- else
  - Sample x_i from P(X_i | Parents(X_i))

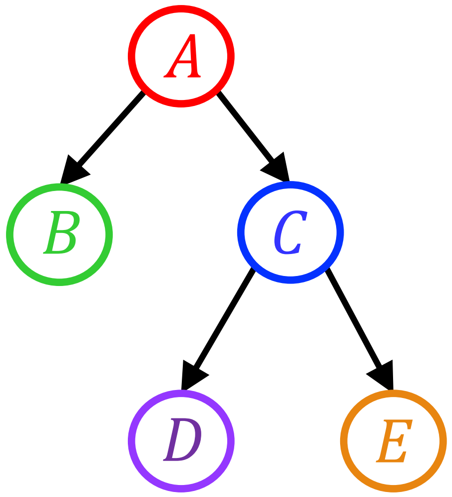
return (x_1, x_2, …, x_n), w



Likelihood Weighting

No evidence: Prior Sampling	Some evidence: Likelihood Weighted Sampling	All evidence: Likelihood Weighted
Input: no evidence	Input: evidence instantiation	Input: evidence instantiation
	w = 1.0	w = 1.0
for i=1, 2, …, n	for i=1, 2, …, n	for i=1, 2, …, n
	if X_i is an evidence variable	
	▪ X_i = observation x_i for X_i	
	▪ Set w = w * $P(x_i \mid Parents(X_i))$	▪ Set w = w * $P(x_i \mid Parents(X_i))$
	else	
▪ Sample x_i from $P(X_i \mid Parents(X_i))$	▪ Sample x_i from $P(X_i \mid Parents(X_i))$	
return $(x_1, x_2, \ldots, x_n)$	return $(x_1, x_2, \ldots, x_n)$, w	return w

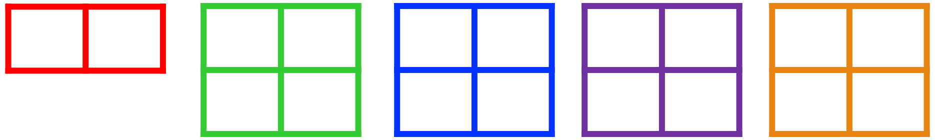
Likelihood Weighting Distribution



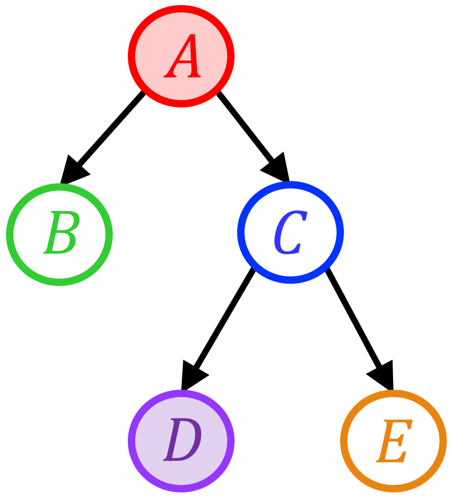
Consistency of likelihood weighted sampling distribution

Joint from Bayes nets

$$P(A,B,C,D,E) = P(A)\, P(B|A)\, P(C|A)\, P(D|C)\, P(E|C)$$



Likelihood Weighting Distribution

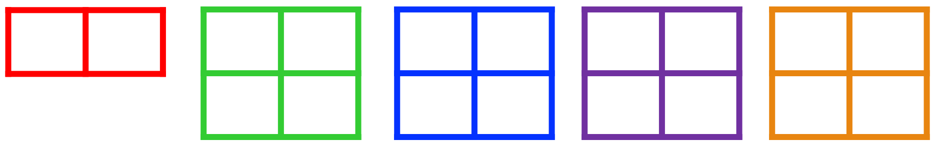


Consistency of likelihood weighted sampling distribution

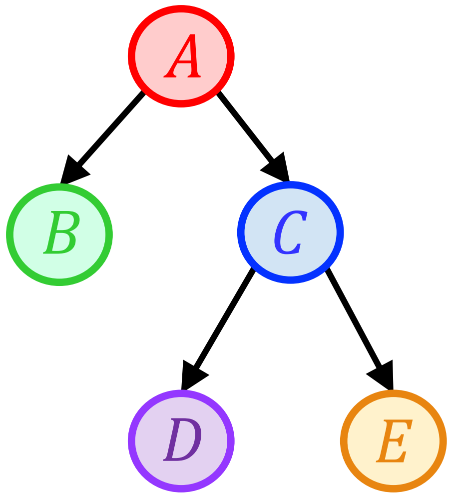
Evidence: $+a, -d$

Joint from Bayes nets

$$P(A,B,C,D,E) = P(+a)\,P(B|+a)\,P(C|+a)\,P(-d|C)\,P(E|C)$$



Likelihood Weighting Distribution

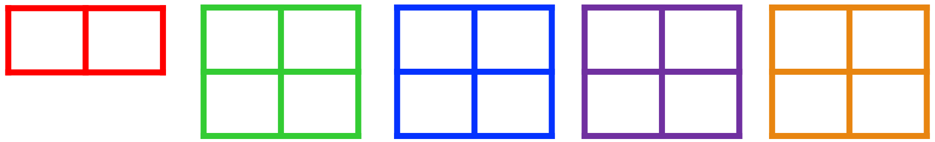


Consistency of likelihood weighted sampling distribution

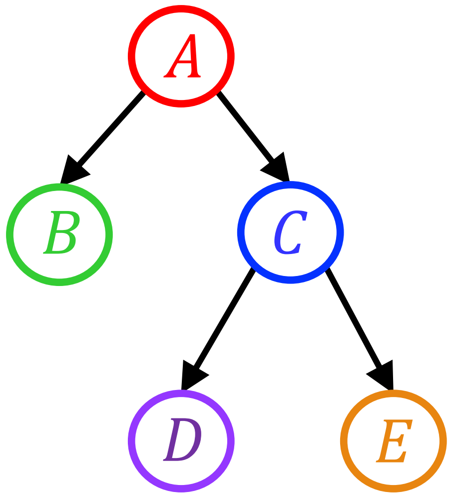
Evidence: $+a, +b, -c, -d, +e$

Joint from Bayes nets

$$P(A,B,C,D,E) = P(+a)\, P(+b|+a)\, P(-c|+a)\, P(-d|-c)\, P(+e|-c)$$



Likelihood Weighting Distribution

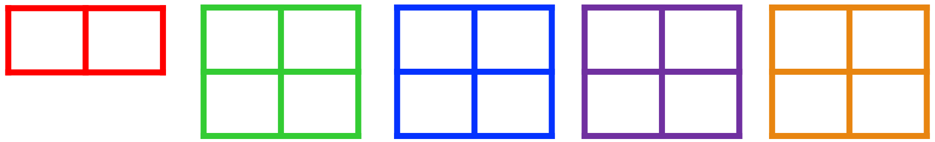


Consistency of likelihood weighted sampling distribution

Evidence: None

Joint from Bayes nets

$$P(A,B,C,D,E) = P(A)\,P(B|A)\,P(C|A)\,P(D|C)\,P(E|C)$$



Piazza Poll 1

Two identical samples from likelihood weighted sampling will have the same exact weights.

A. True
B. False
C. It depends
D. I don’t know

Piazza Poll 2

What does the following likelihood weighted value approximate?

$$\text{weight}_{(+a,-b,+c)} \cdot \frac{N(+a,-b,+c)}{N}$$

A. $P(+a, -b, +c)$

B. $P(+a, -b \mid +c)$

C. I’m not sure

Likelihood Weighting

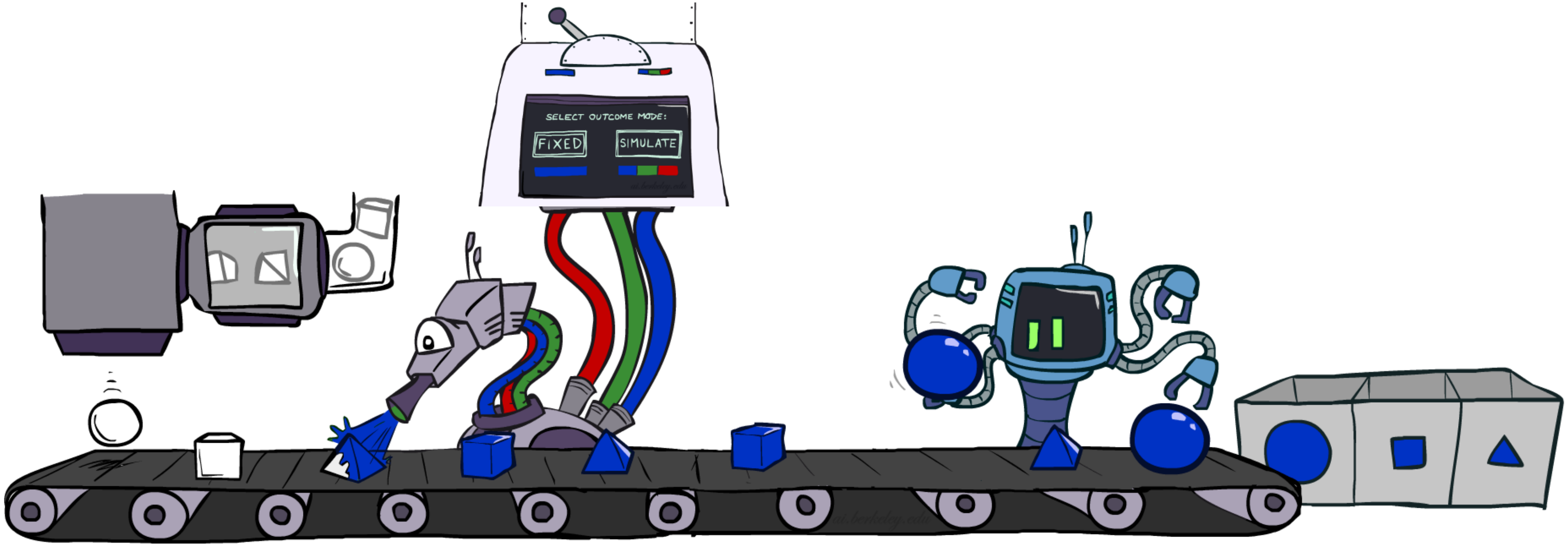
Likelihood weighting is good

- 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



Likelihood Weighting

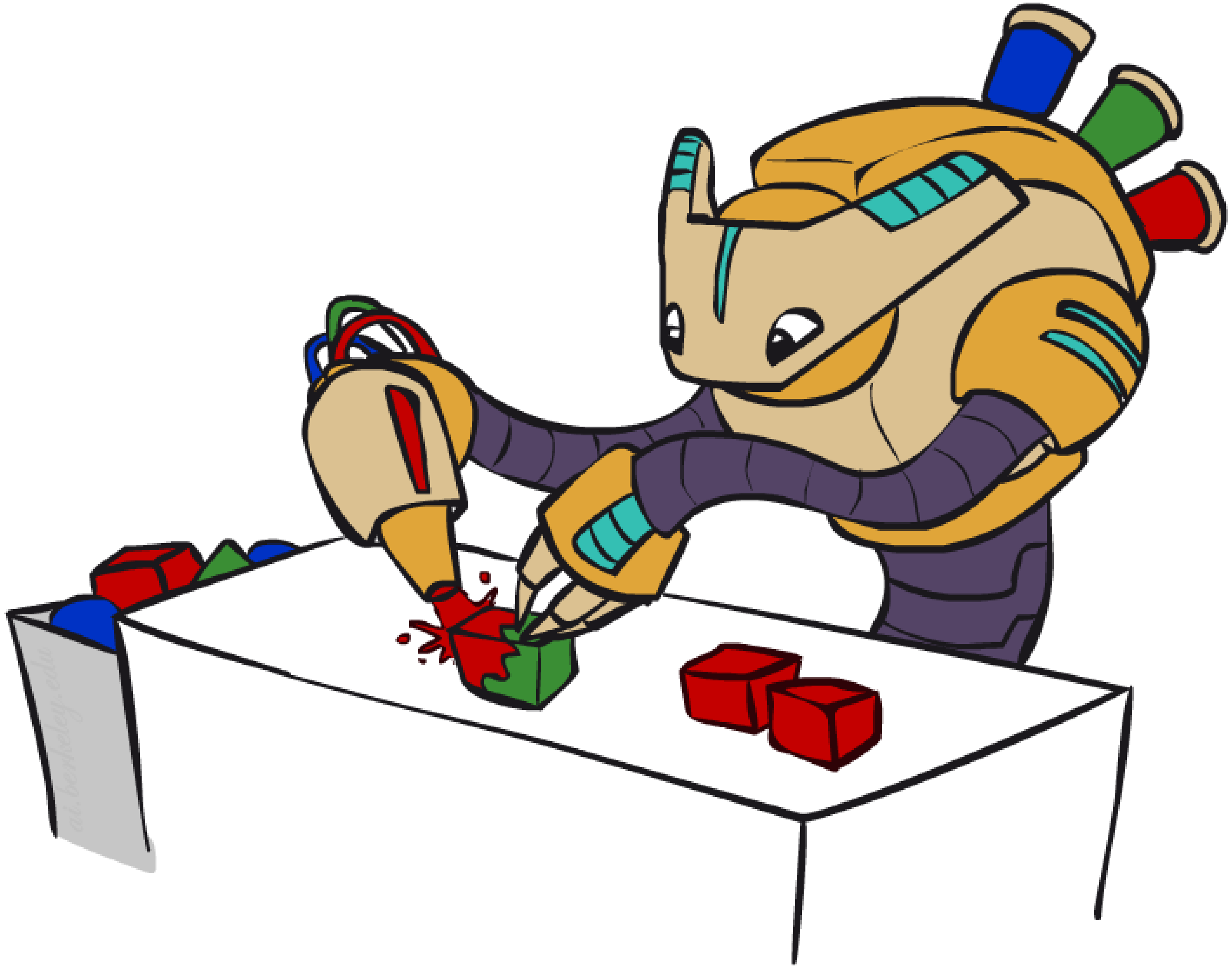
Likelihood weighting doesn't solve all our problems

- 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

Gibbs Sampling



Gibbs Sampling

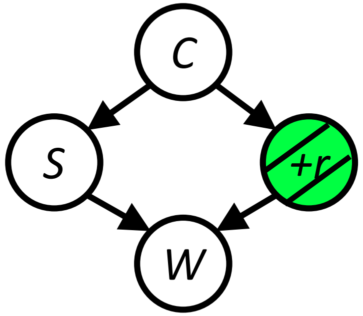
Procedure: keep track of a full instantiation $x_1, x_2, \ldots, x_n$.

1. Start with an arbitrary instantiation consistent with the evidence.
2. Sample one variable at a time, conditioned on all the rest, but keep evidence fixed.
3. Keep repeating this for a long time.

Gibbs Sampling Example: P(S | +r)

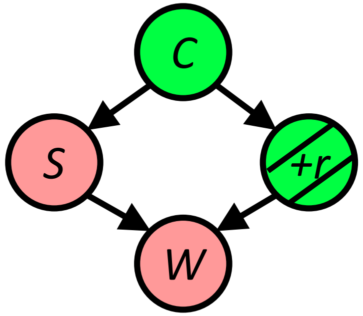
Step 1: Fix evidence

- R = +r



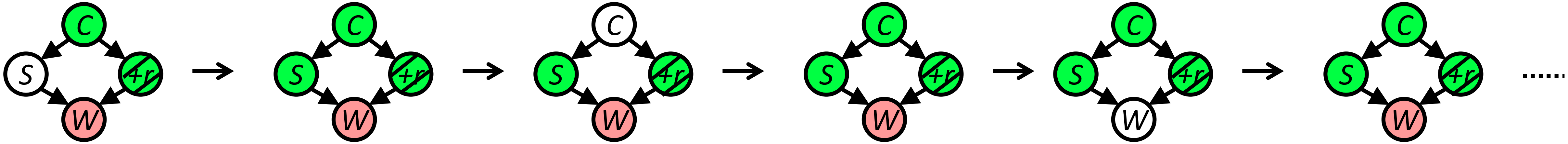
Step 2: Initialize other variables

- Randomly



Steps 3: Repeat

- Choose a non-evidence variable X
- Resample X from P(X | all other variables)

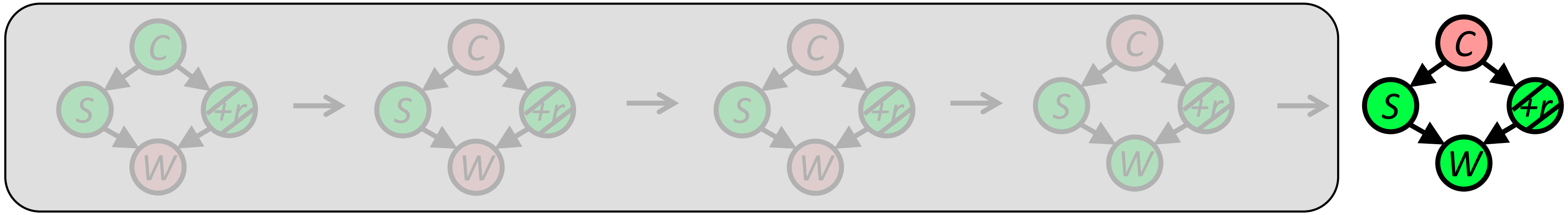


Sample from $P(S|+c,-w,+r)$ Sample from $P(C|+s,-w,+r)$ Sample from $P(W|+s,+c,+r)$

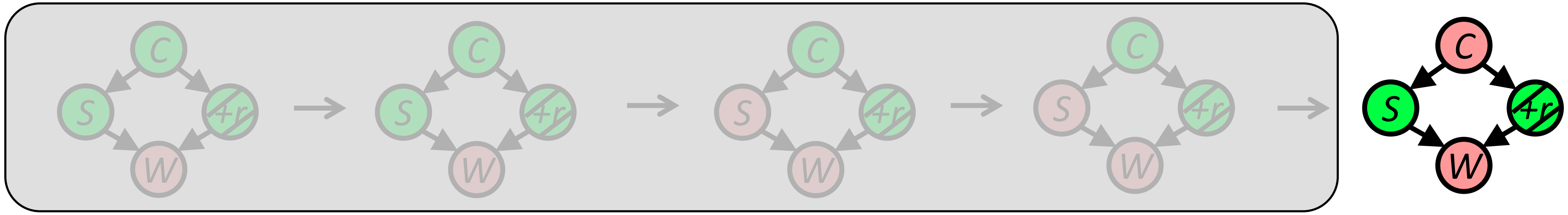
Gibbs Sampling Example: P(S | +r)

Keep only the last sample from each iteration:

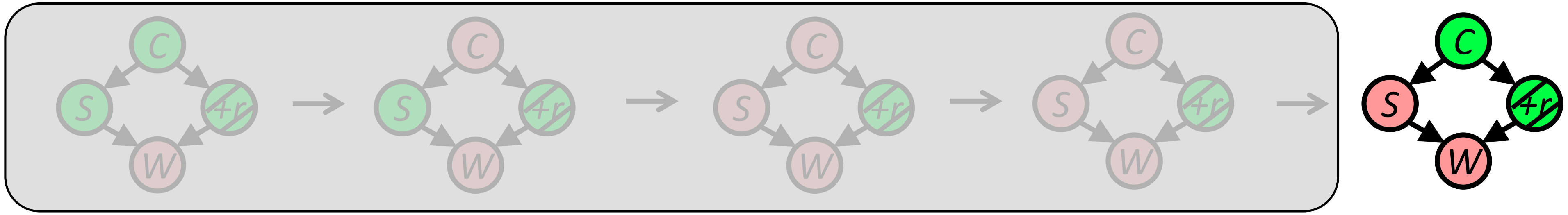
1.



2.



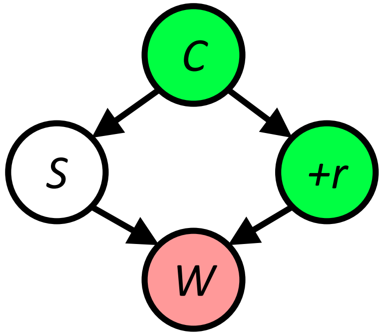
3.



Efficient Resampling of One Variable

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

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



Many things cancel out – only CPTs with S remain!

More generally: only CPTs that have resampled variable need to be considered, and joined together

Gibbs Sampling

Procedure: keep track of a full instantiation x_1, x_2, …, x_n.

1. Start with an arbitrary instantiation consistent with the evidence.
2. Sample one variable at a time, conditioned on all the rest, but keep evidence fixed.
3. 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

Gibbs sampling produces sample from the query distribution P(Q | e) in limit of re-sampling infinitely often

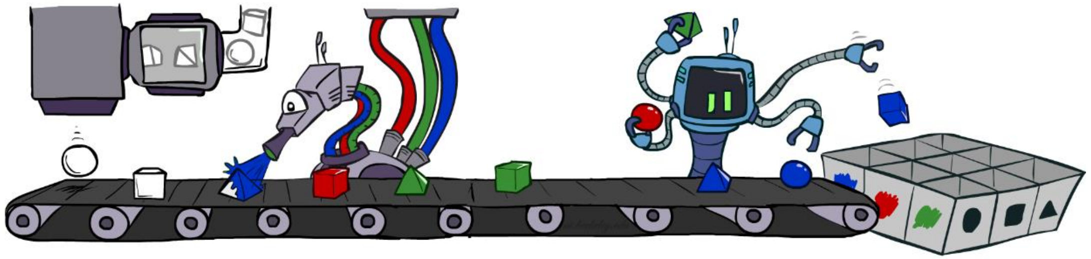
Gibbs sampling is a special case of more general methods called Markov chain Monte Carlo (MCMC) methods

- Metropolis-Hastings is one of the more famous MCMC methods (in fact, Gibbs sampling is a special case of Metropolis-Hastings)

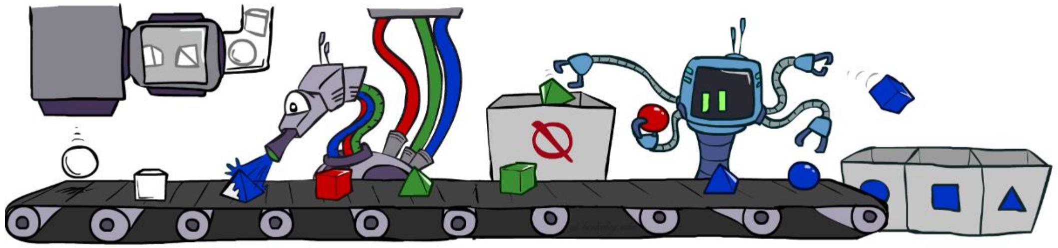
You may read about Monte Carlo methods – they're just sampling

Bayes' Net Sampling Summary

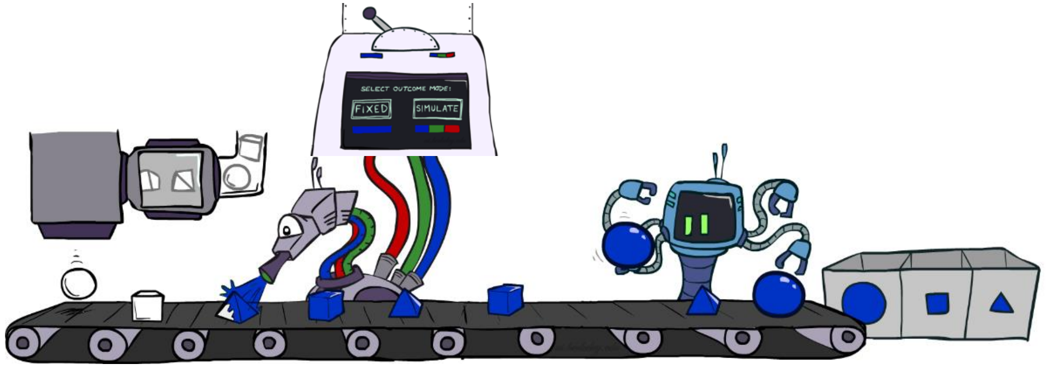
Prior Sampling P(Q, E)



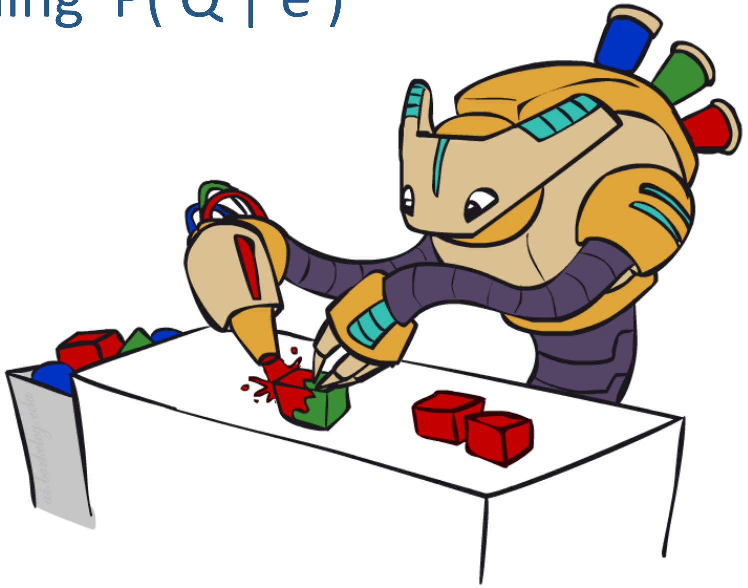
Rejection Sampling P(Q | e)



Likelihood Weighting P(Q , e)



Gibbs Sampling P(Q | e)



AI: Representation and Problem Solving

Hidden Markov Models

Instructors: Pat Virtue & Stephanie Rosenthal

Slide credits: CMU AI and http://ai.berkeley.edu

Reasoning over Time or Space

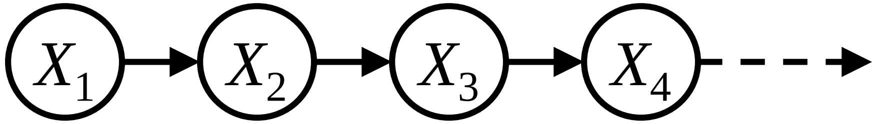
Often, we want to reason about a sequence of observations

- Speech recognition
- Robot localization
- User attention
- Medical monitoring

Need to introduce time (or space) into our models

Markov Models

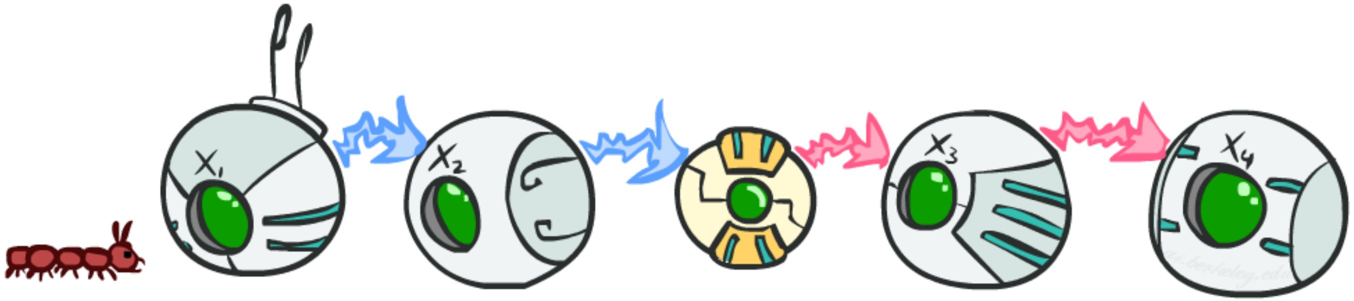
- Value of X at a given time is called the state



$P(X_1)$ $\qquad$ $P(X_t|X_{t-1})$

- 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

Conditional Independence



Basic conditional independence:

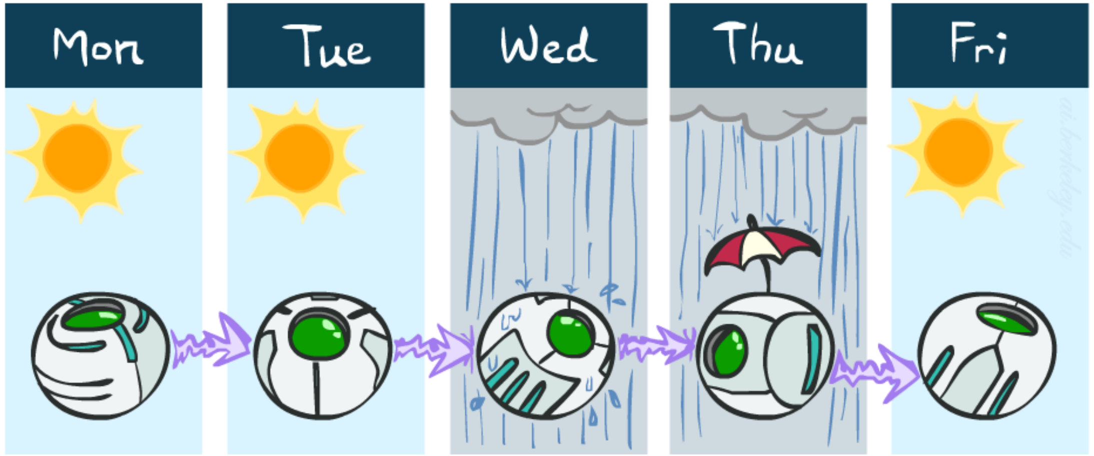
- Past and future independent given the present
- Each time step only depends on the previous
- This is called the (first order) Markov property

Note that the chain is just a (growable) BN

- We can always use generic BN reasoning on it if we truncate the chain at a fixed length

Example Markov Chain: Weather

States: X = {rain, sun}

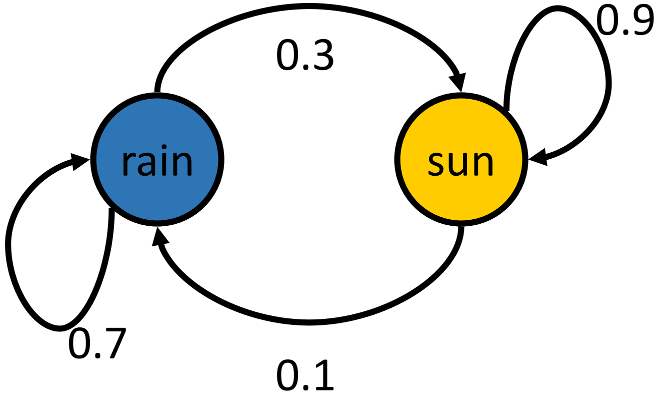


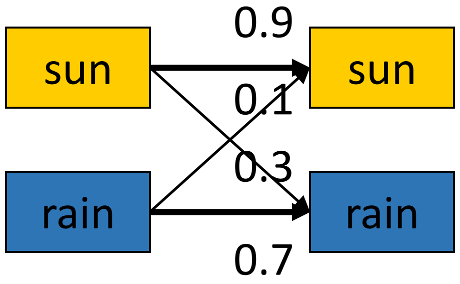
- Initial distribution: 1.0 sun

- CPT $P(X_t \mid X_{t-1})$:

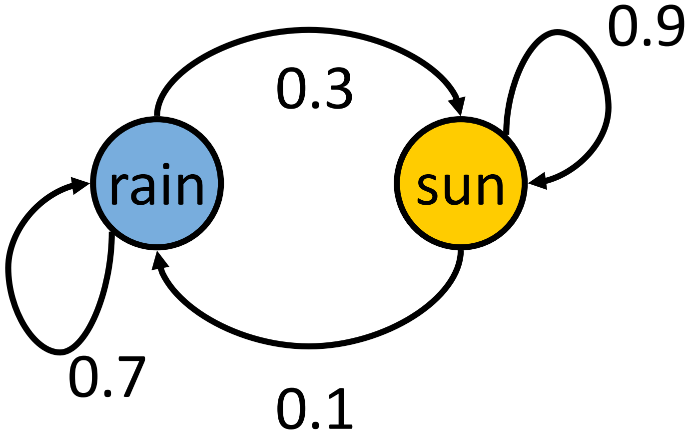
X_{t-1}	X_t	$P(X_t \mid X_{t-1})$
sun	sun	0.9
sun	rain	0.1
rain	sun	0.3
rain	rain	0.7

Two new ways of representing the same CPT





Piazza Poll 3



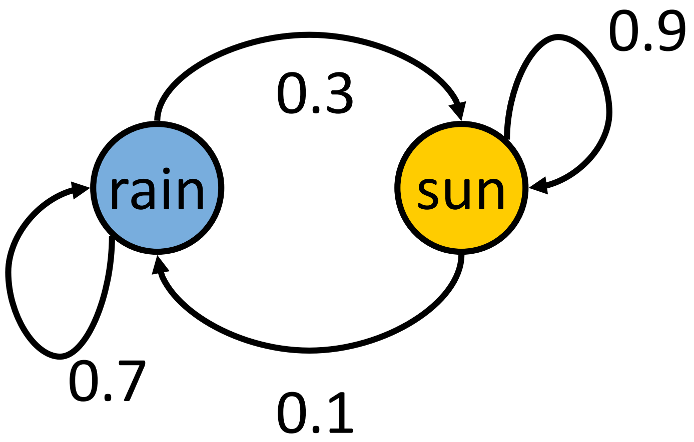
Initial distribution: $P(X_1 = sun) = 1.0$

What is the probability distribution after one step?

$P(X_2 = sun) = ?$

A) 0
B) 0.3
C) 0.9
D) 1.0
E) 1.2

Piazza Poll 3



Initial distribution: $P(X_1 = sun) = 1.0$

What is the probability distribution after one step?

$P(X_2 = sun) = ?$

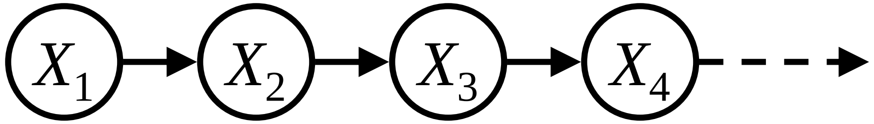
A) 0
B) 0.3
C) 0.9
D) 1.0
E) 1.2

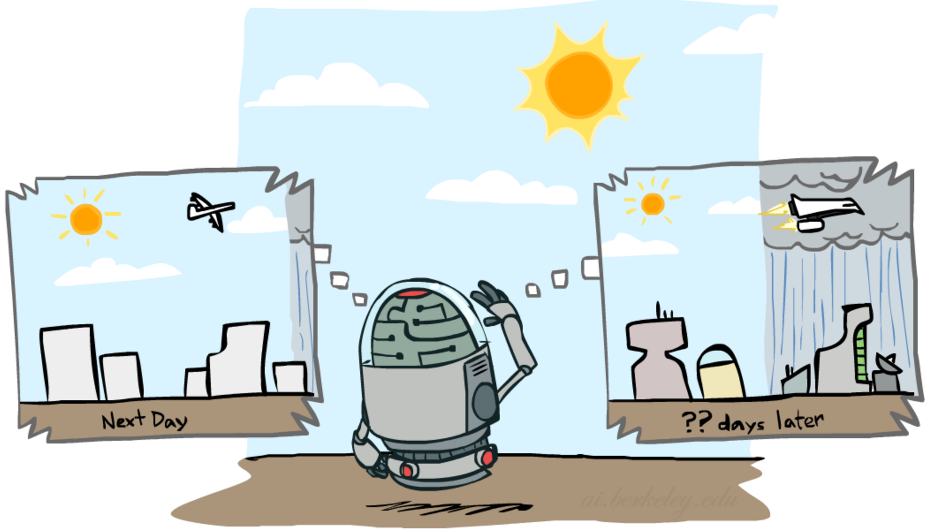
$$P(X_2 = \text{sun}) = P(X_2 = \text{sun}|X_1 = \text{sun})P(X_1 = \text{sun}) + P(X_2 = \text{sun}|X_1 = \text{rain})P(X_1 = \text{rain})$$

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

Mini-Forward Algorithm

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





$$P(x_1) = \text{known}$$

$$P(x_t) = \sum_{x_{t-1}} P(x_{t-1}, x_t)$$

$$= \sum_{x_{t-1}} P(x_t \mid x_{t-1}) P(x_{t-1})$$

Forward simulation

Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\begin{pmatrix} 1.0 \\ 0.0 \end{pmatrix} \begin{pmatrix} 0.9 \\ 0.1 \end{pmatrix} \begin{pmatrix} 0.84 \\ 0.16 \end{pmatrix} \begin{pmatrix} 0.804 \\ 0.196 \end{pmatrix} \Rightarrow \begin{pmatrix} 0.75 \\ 0.25 \end{pmatrix}$$

$$P(X_1) \quad P(X_2) \quad P(X_3) \quad P(X_4) \quad P(X_\infty)$$

- From initial observation of rain

$$\begin{pmatrix} 0.0 \\ 1.0 \end{pmatrix} \begin{pmatrix} 0.3 \\ 0.7 \end{pmatrix} \begin{pmatrix} 0.48 \\ 0.52 \end{pmatrix} \begin{pmatrix} 0.588 \\ 0.412 \end{pmatrix} \Rightarrow \begin{pmatrix} 0.75 \\ 0.25 \end{pmatrix}$$

$$P(X_1) \quad P(X_2) \quad P(X_3) \quad P(X_4) \quad P(X_\infty)$$

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

$$\begin{pmatrix} p \\ 1-p \end{pmatrix} \quad \cdots \quad \Rightarrow \begin{pmatrix} 0.75 \\ 0.25 \end{pmatrix}$$

$$P(X_1) \qquad\qquad P(X_\infty)$$

[Demo: L13D1,2,3]

Demo Ghostbusters Basic Dynamics

Demo Ghostbusters Circular Dynamics

Demo Ghostbusters Whirlpool Dynamics

Stationary Distributions

For most chains:

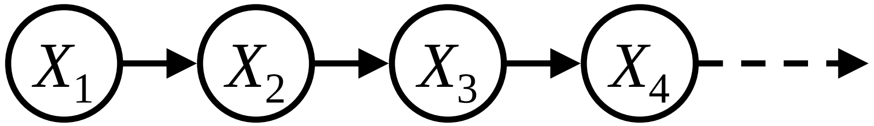
- Influence of the initial distribution gets less and less over time.
- The distribution we end up in is independent of the initial distribution

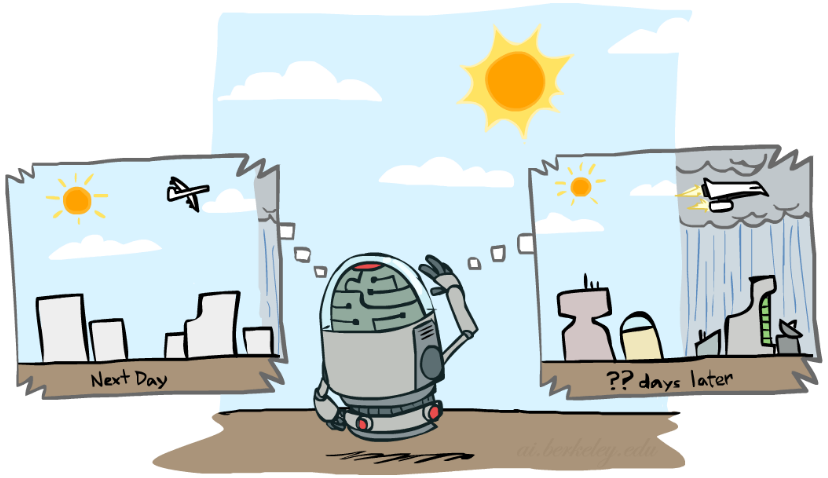
- Stationary distribution:
  - The distribution we end up with is called the stationary distribution P_∞ of the chain
  - It satisfies

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

Example: Stationary Distributions

Question: What's P(X) at time t = infinity?





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

$$P_\infty(sun) = 0.9P_\infty(sun) + 0.3P_\infty(rain)$$
$$P_\infty(rain) = 0.1P_\infty(sun) + 0.7P_\infty(rain)$$

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

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

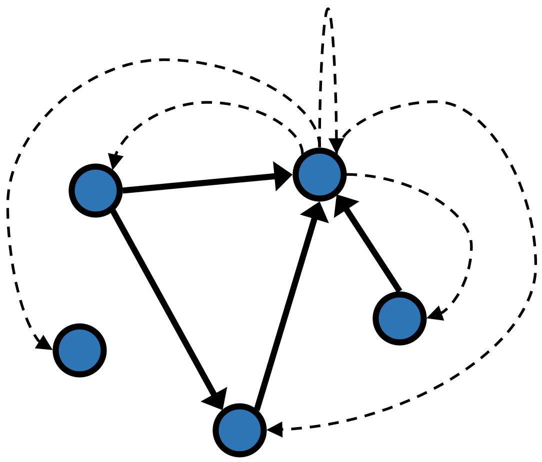
Also: $P_\infty(sun) + P_\infty(rain) = 1$

$$P_\infty(sun) = 3/4$$
$$P_\infty(rain) = 1/4$$

Application of Stationary Distribution: Web Link Analysis

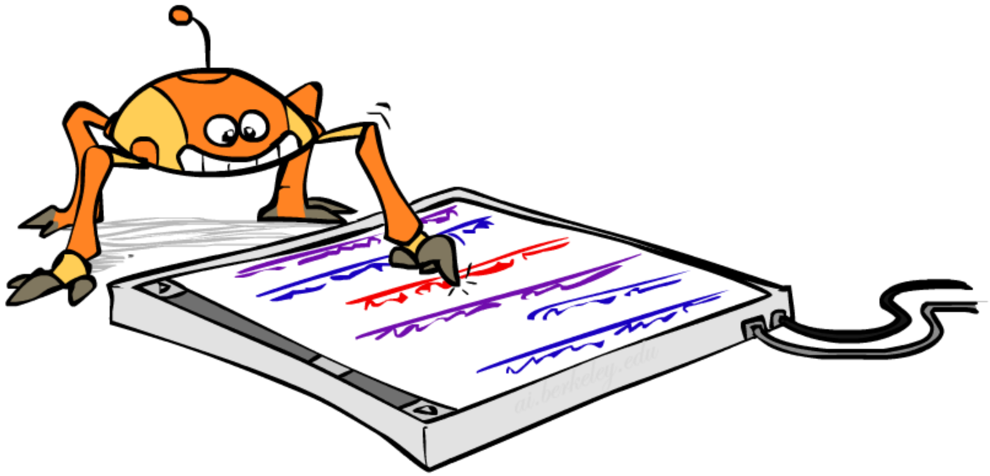
PageRank over a web graph

- 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)



Application of Stationary Distributions: Gibbs Sampling*

Each joint instantiation over all hidden and query variables is a state: $\{X_1, \ldots, X_n\} = H \cup Q$

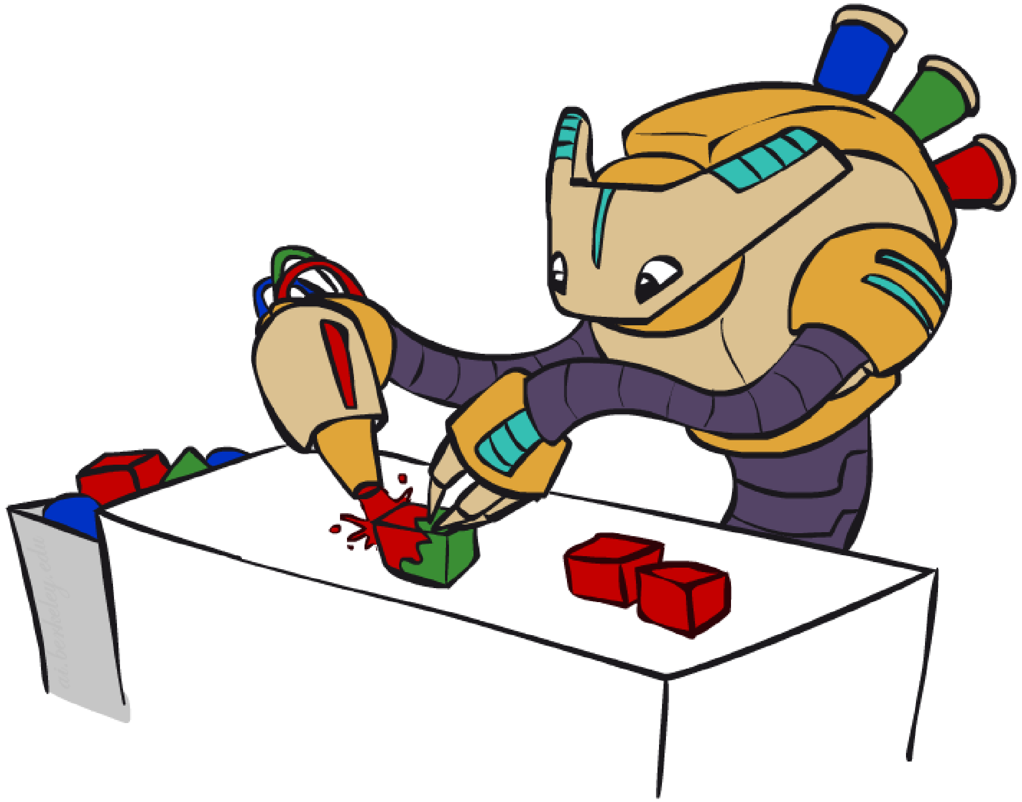
Transitions:

- With probability 1/n resample variable X_j according to

$$P(X_j \mid x_1, x_2, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n, e_1, \ldots, e_m)$$

Stationary distribution:

- Conditional distribution $P(X_1, X_2, \ldots, X_n \mid e_1, \ldots, e_m)$
- Means that when running Gibbs sampling long enough we get a sample from the desired distribution
- Requires some proof to show this is true!

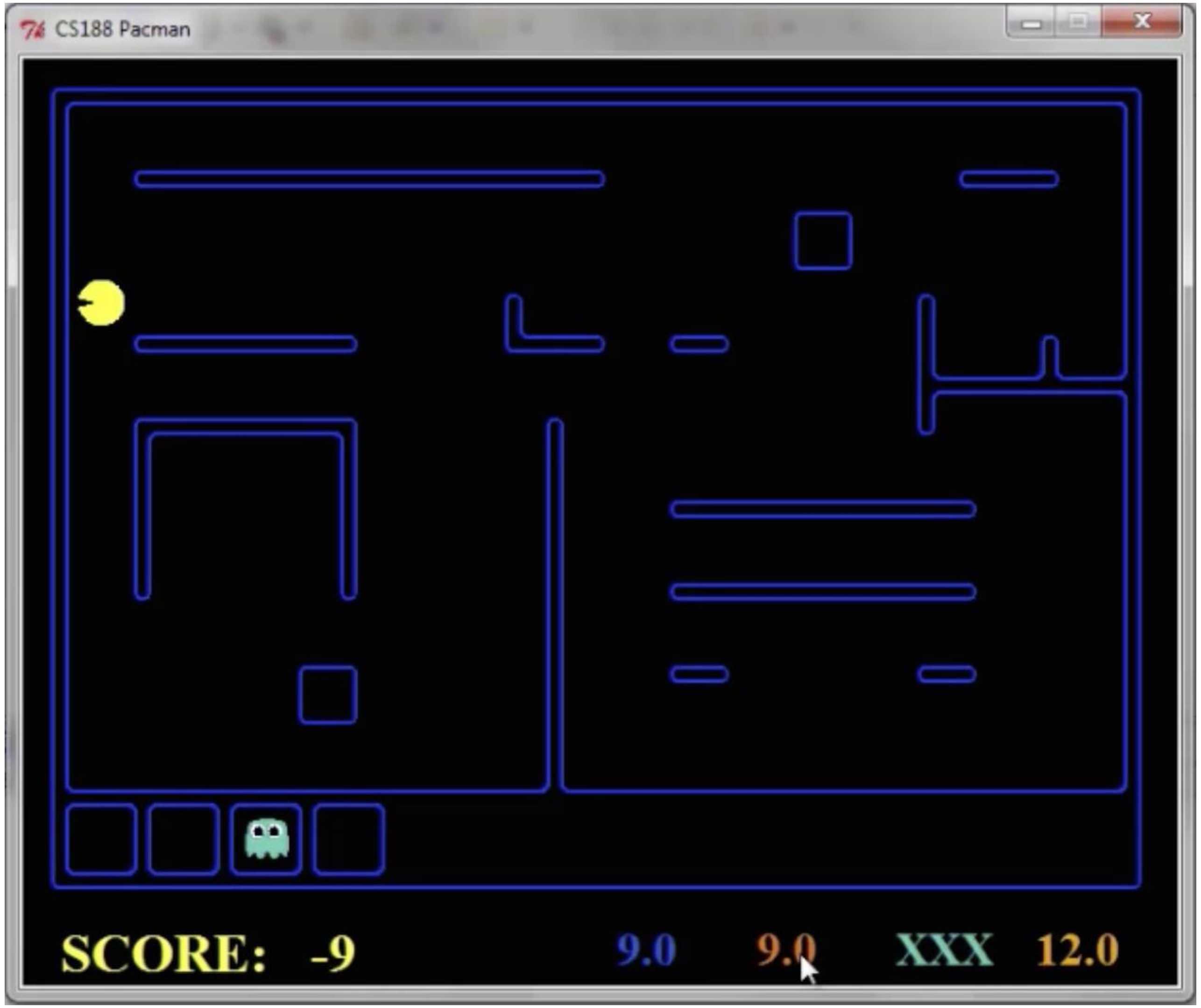


Quick Break

[How the real Ghostbusters make decisions]

Hidden Markov Models

Pacman – Sonar (P5)



[Demo: Pacman – Sonar – No Beliefs(L14D1)]

Demo Pacman – Sonar (no beliefs)

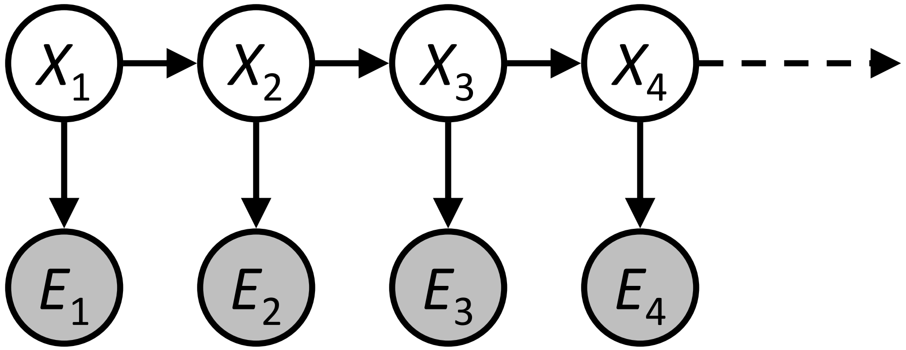
Hidden Markov Models

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

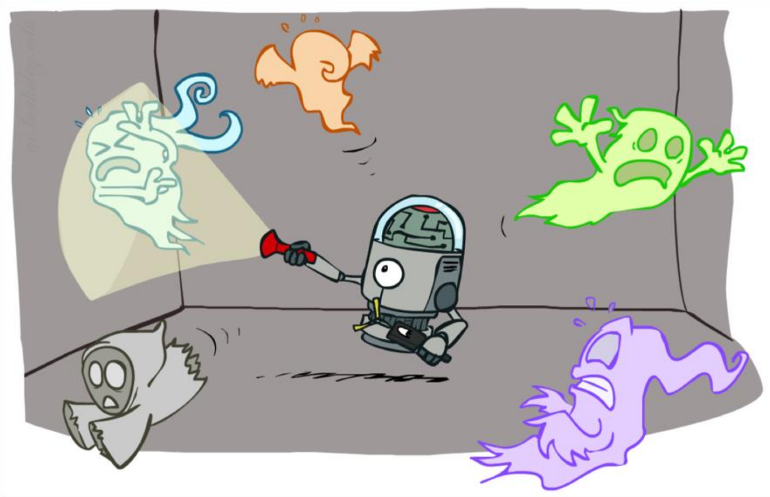


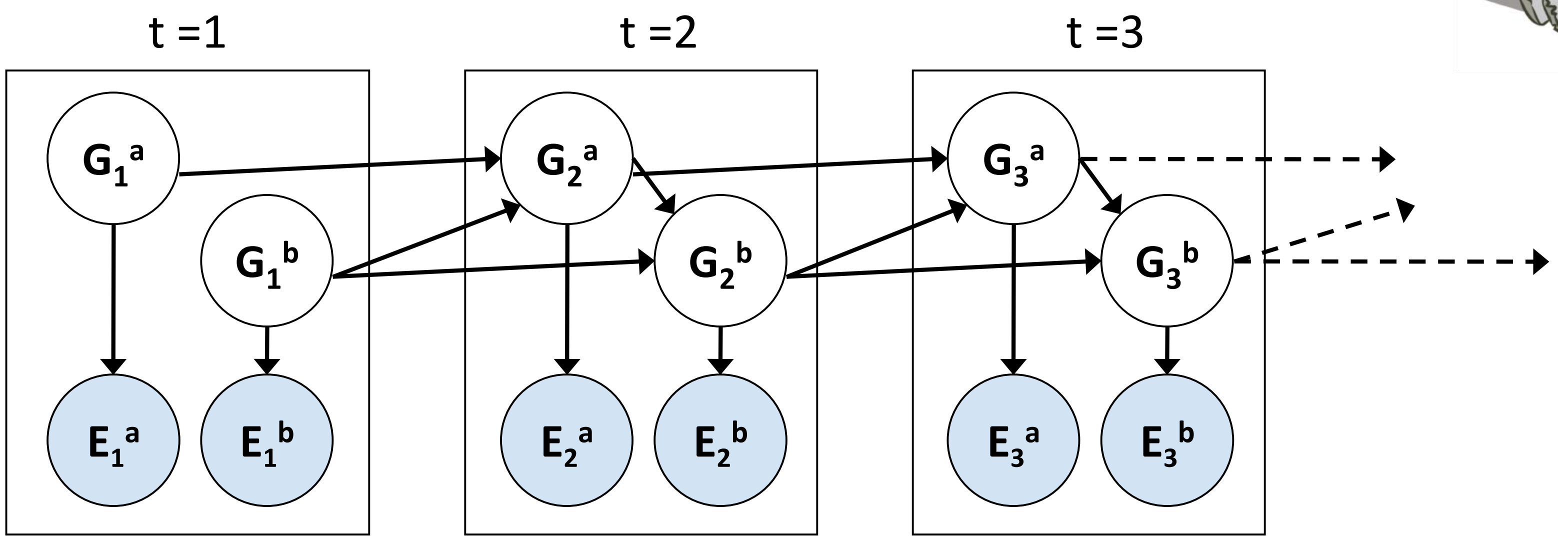
Dynamic Bayes Nets (DBNs)

We want to track multiple variables over time, using multiple sources of evidence

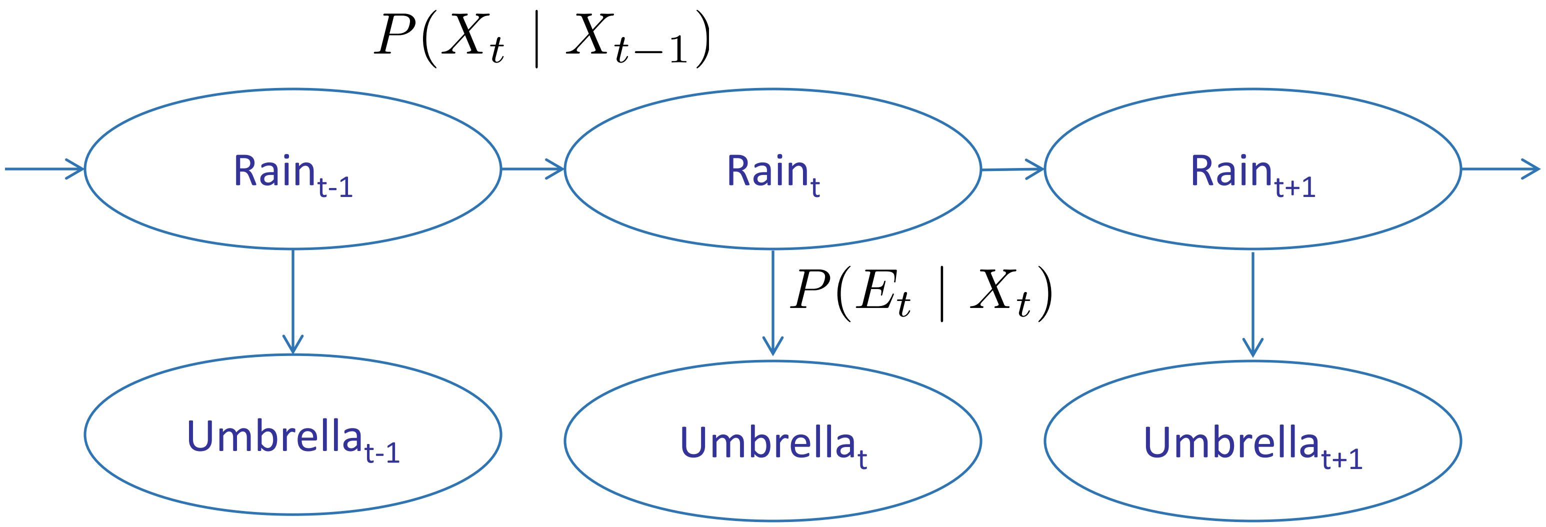
Idea: Repeat a fixed Bayes net structure at each time

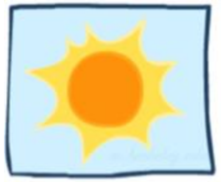
Variables from time *t* can condition on those from *t-1*





Example: Weather HMM





An HMM is defined by:

- Initial distribution: $P(X_1)$
- Transitions: $P(X_t \mid X_{t-1})$
- Emissions: $P(E_t \mid X_t)$

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

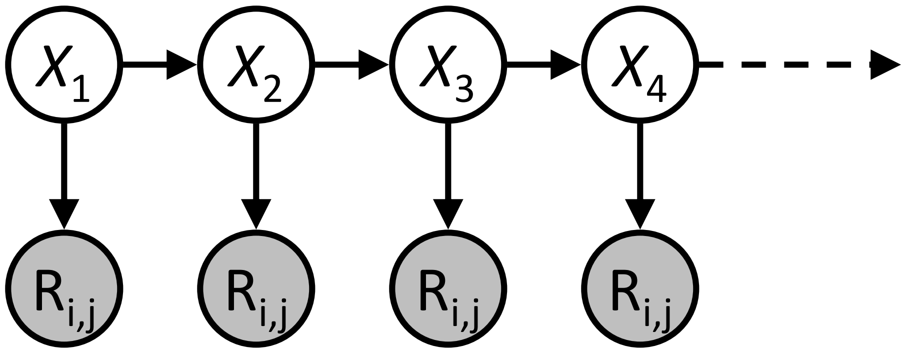
R_t	U_t	$P(U_t \mid R_t)$
+r	+u	0.9
+r	-u	0.1
-r	+u	0.2
-r	-u	0.8

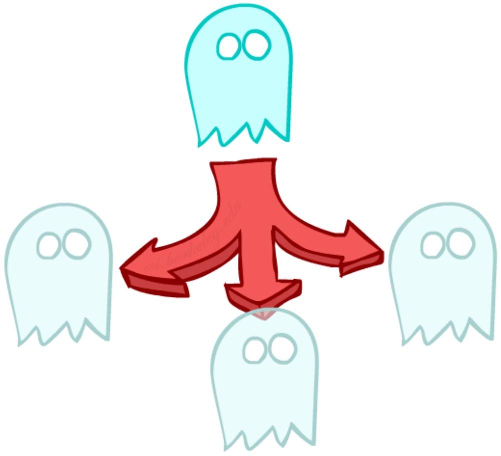
Example: Ghostbusters HMM

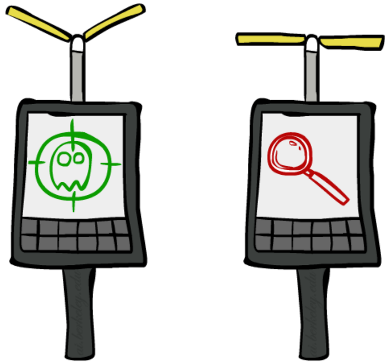
$P(X_1)$ = uniform

$P(X|X')$ = 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

$P(X|X'=<1,2>)$

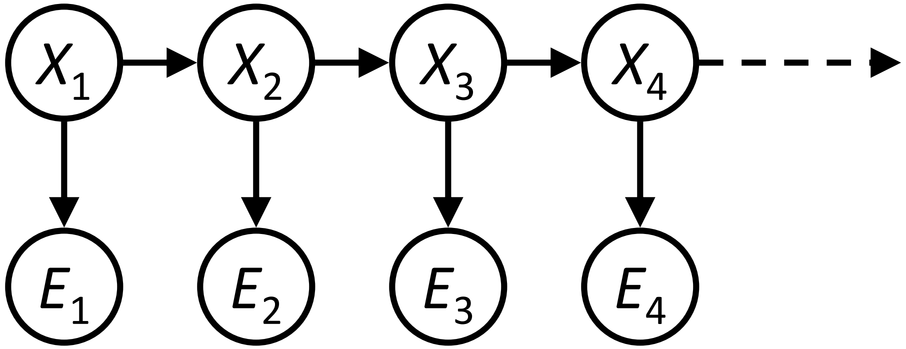
[Demo: Ghostbusters – Circular Dynamics – HMM (L14D2)]

Demo Ghostbusters – Circular Dynamics -- HMM

Conditional Independence

HMMs have two important independence properties:

- Markov hidden process: future depends on past via the present
- Current observation independent of all else given current state



Does this mean that evidence variables are guaranteed to be independent?

- [No, they tend to correlated by the hidden state]

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 $B_t(X) = P_t(X_t \mid e_1, \ldots, e_t)$ (the belief state) over time

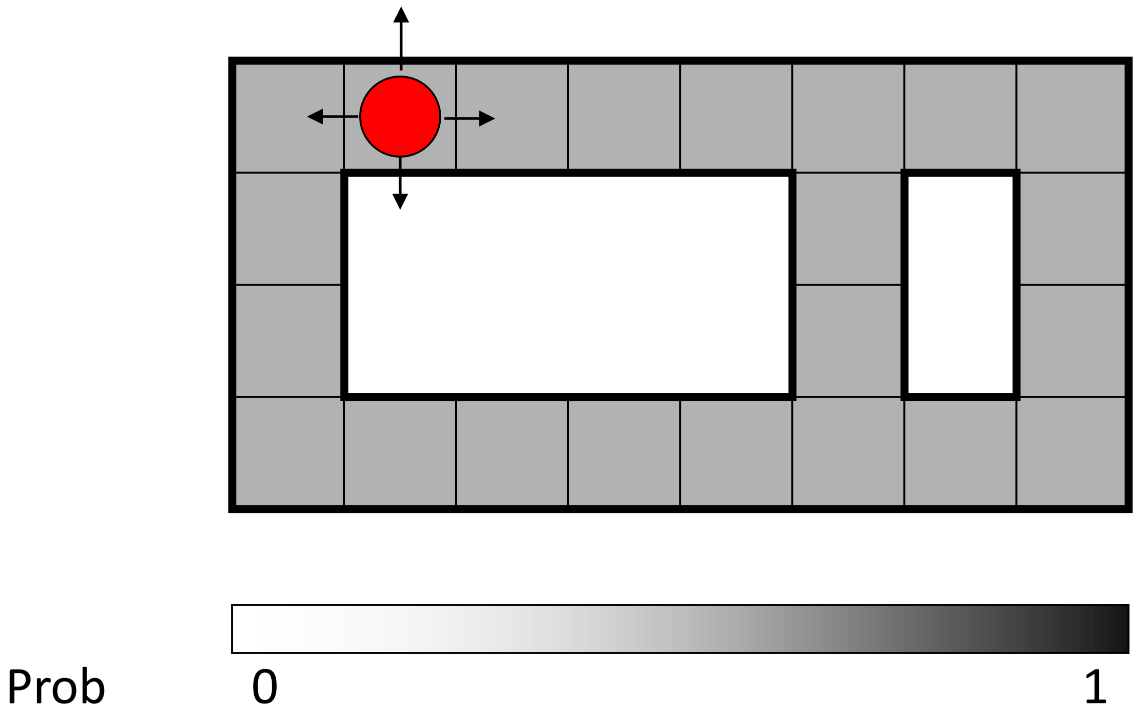
We start with $B_1(X)$ in an initial setting, usually uniform

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

Example: Robot Localization

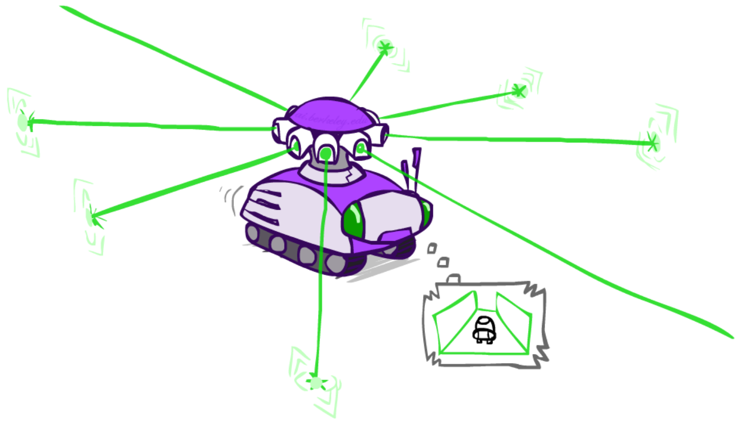
Example from
Michael Pfeiffer



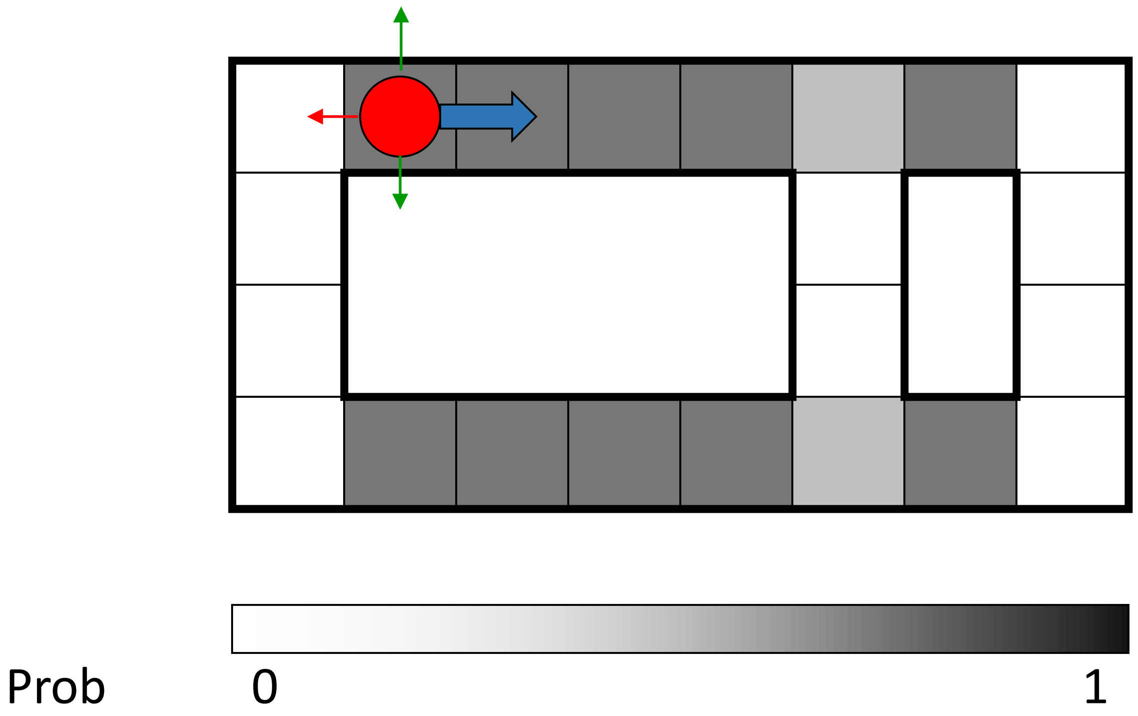
t=0

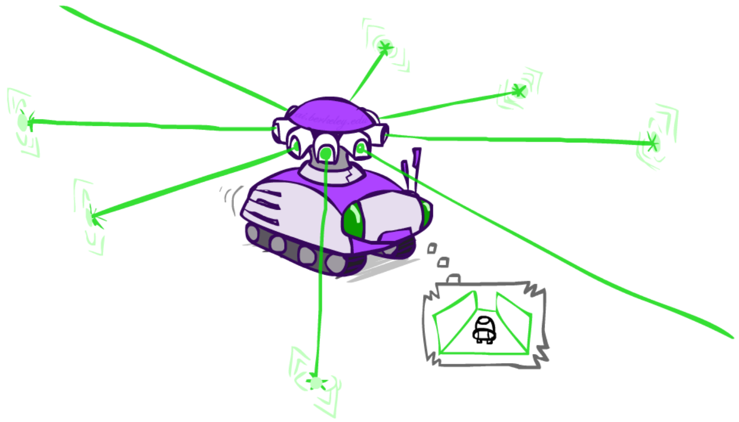
Sensor model: can read in which directions there is a wall, never more than 1 mistake

Motion model: may not execute action with small prob.



Example: Robot Localization

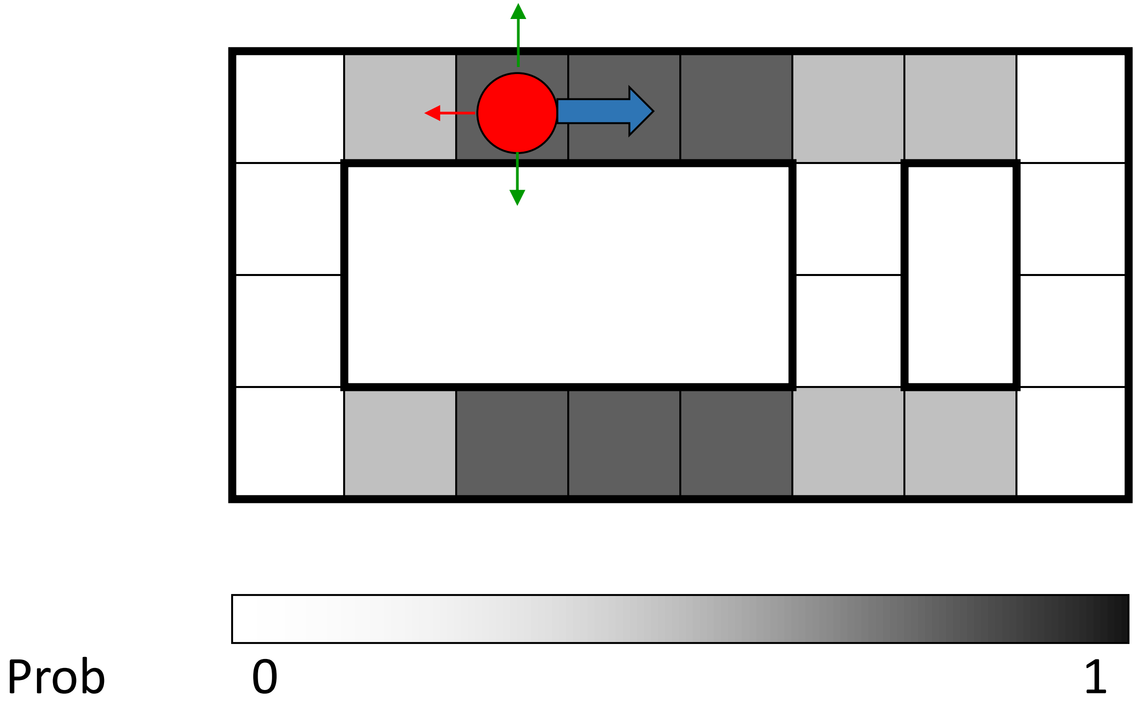




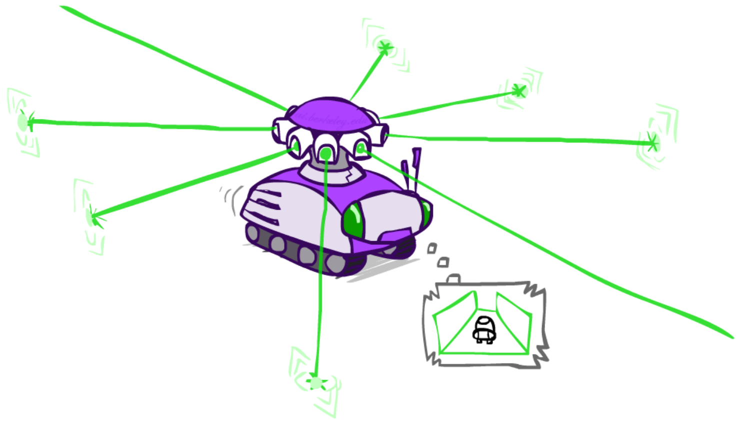
t=1

Lighter grey: was possible to get the reading, but less likely b/c required 1 mistake

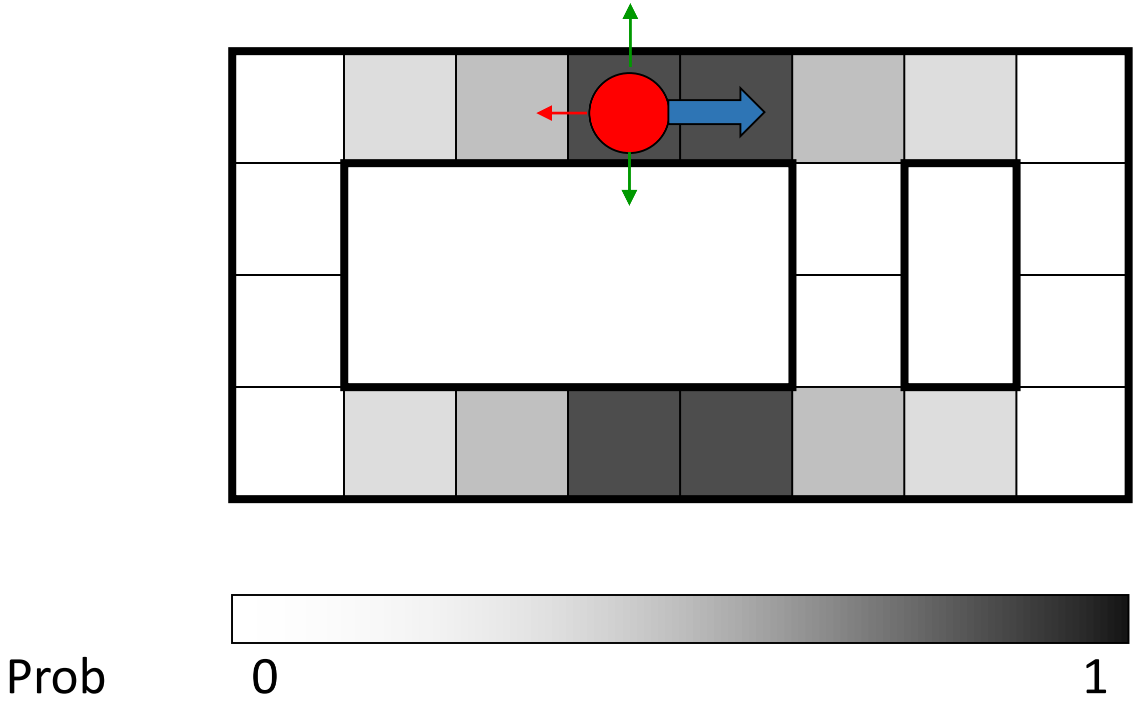
Example: Robot Localization



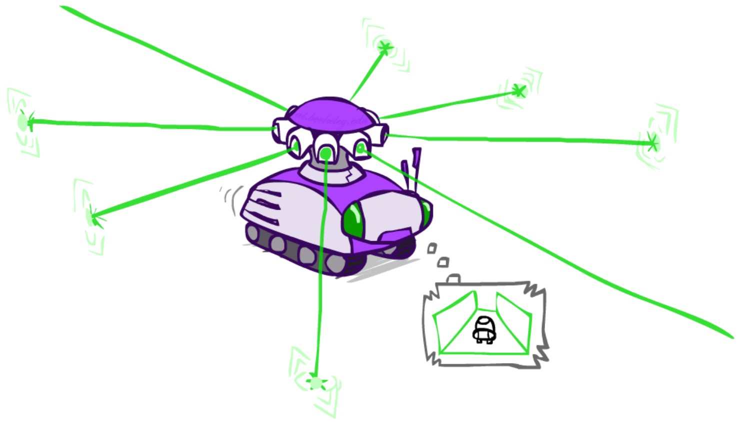
t=2



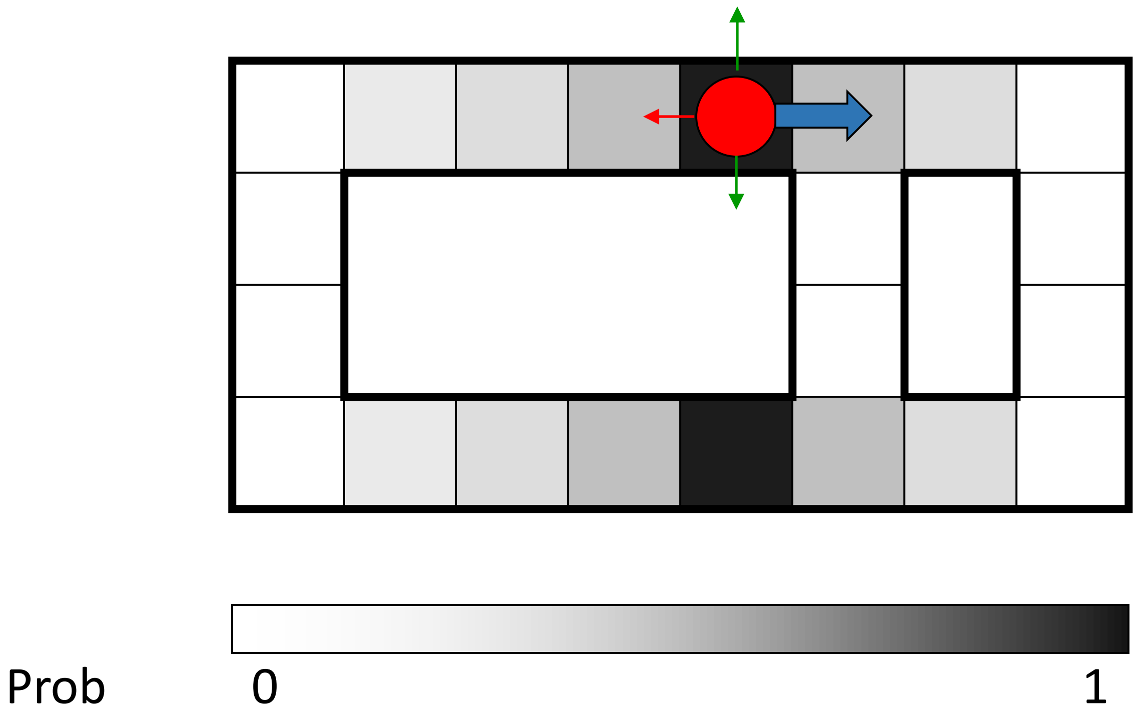
Example: Robot Localization



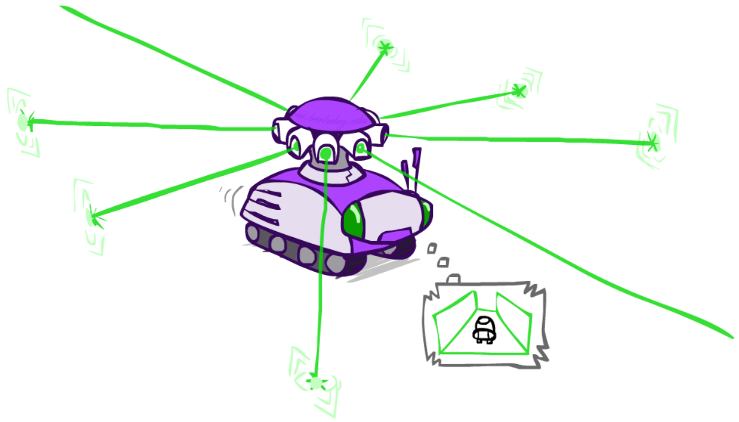
t=3



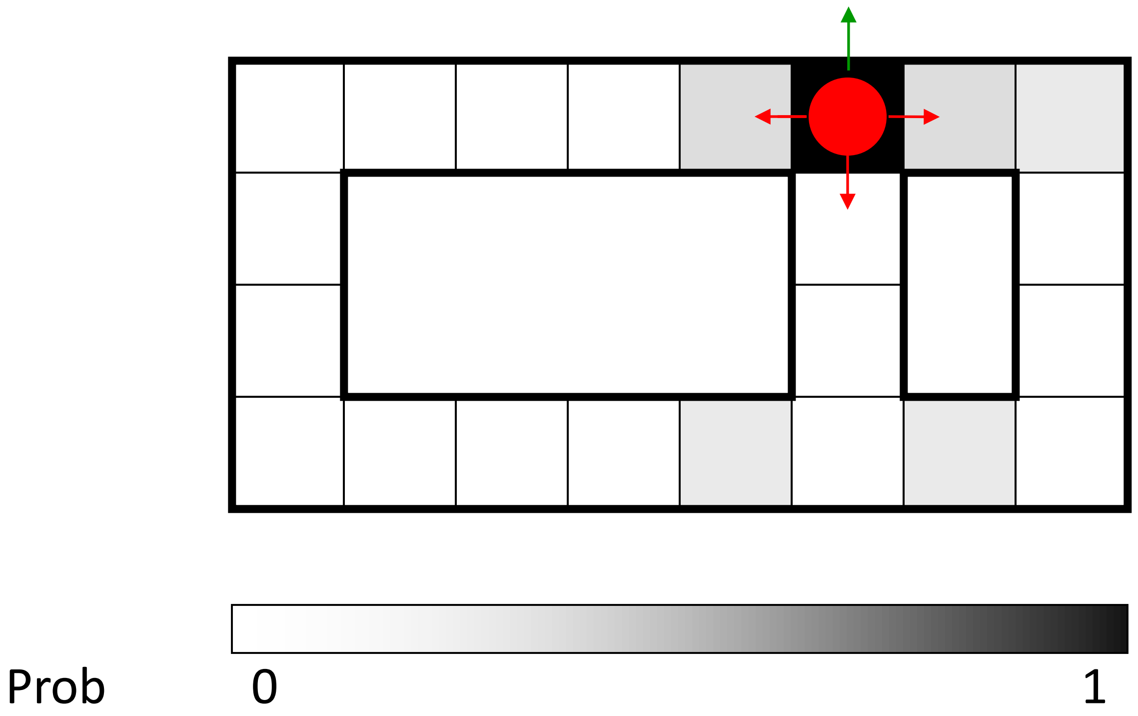
Example: Robot Localization

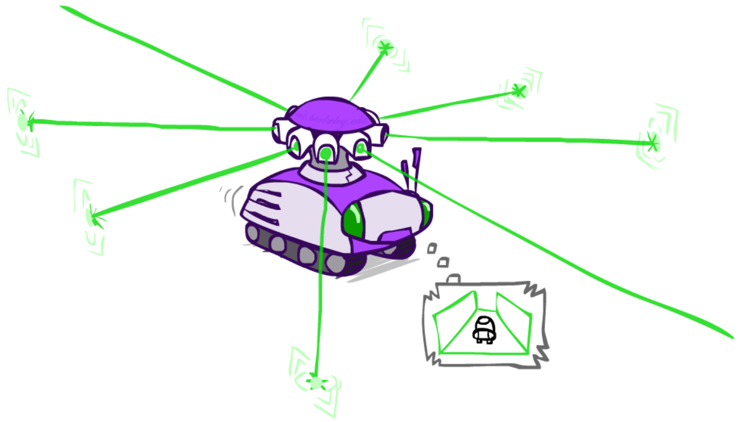


t=4



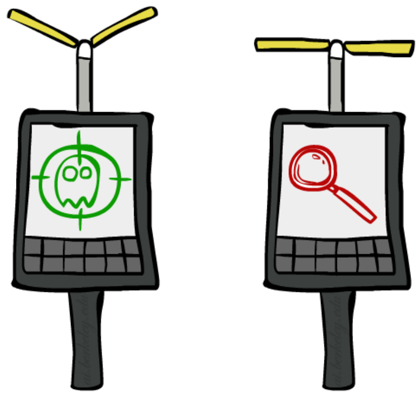
Example: Robot Localization

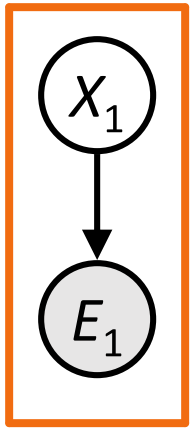


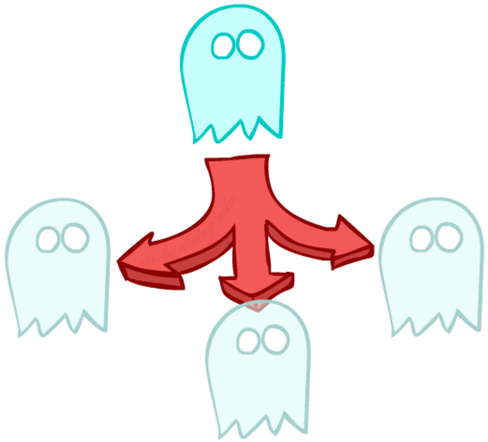


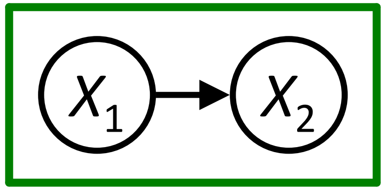
t=5

Inference: Base Cases









$$P(X_1|e_1)$$

$$P(x_1|e_1) = P(x_1, e_1)/P(e_1)$$
$$\propto_{X_1} P(x_1, e_1)$$
$$= P(x_1)P(e_1|x_1)$$

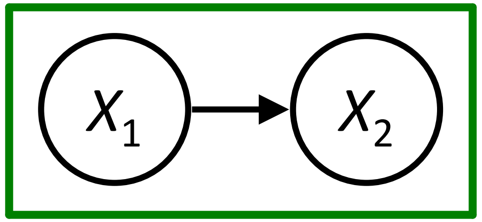
$$P(X_2)$$

$$P(x_2) = \sum_{x_1} P(x_1, x_2)$$
$$= \sum_{x_1} P(x_1)P(x_2|x_1)$$

Passage of Time

Assume we have current belief P(X | evidence to date)

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



Then, after one time step passes:

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

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

$$= \sum_{x_t} P(X_{t+1}|x_t)P(x_t|e_{1:t})$$

- 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)

<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	1.00	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01

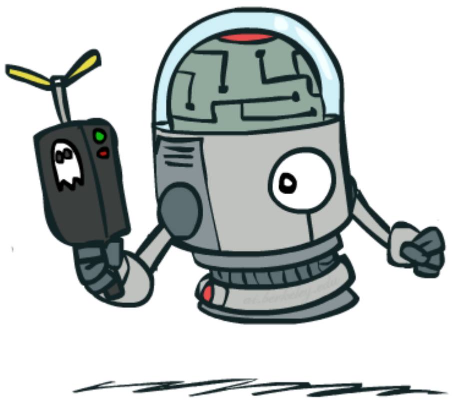
T = 1

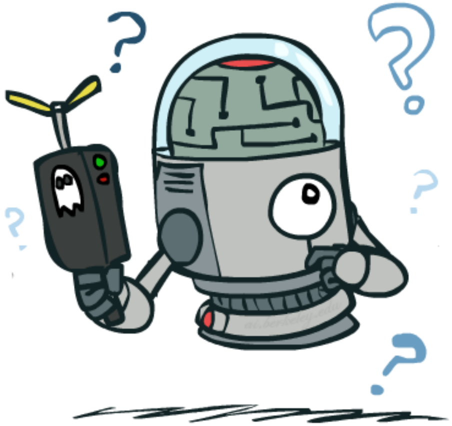
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	0.06	<0.01	<0.01	<0.01
<0.01	0.76	0.06	0.06	<0.01	<0.01
<0.01	<0.01	0.06	<0.01	<0.01	<0.01

T = 2

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

T = 5





Observation

Assume we have current belief P(X | previous evidence):

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

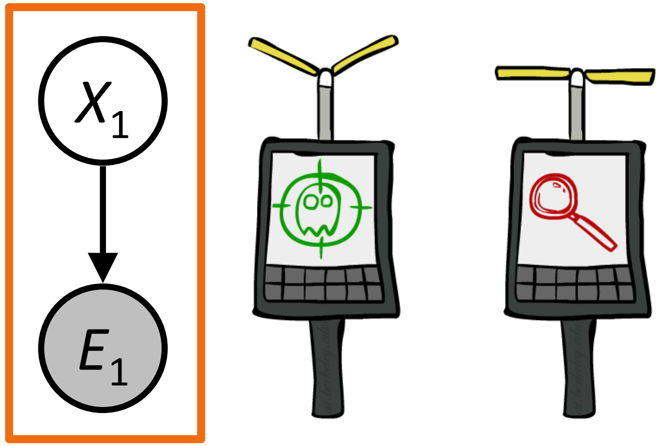
Then, after evidence comes in:

$$P(X_{t+1}|e_{1:t+1}) = P(X_{t+1}, e_{t+1}|e_{1:t})/P(e_{t+1}|e_{1:t})$$

$$\propto_{X_{t+1}} P(X_{t+1}, e_{t+1}|e_{1:t})$$

$$= P(e_{t+1}|e_{1:t}, X_{t+1})P(X_{t+1}|e_{1:t})$$

$$= P(e_{t+1}|X_{t+1})P(X_{t+1}|e_{1:t})$$



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"

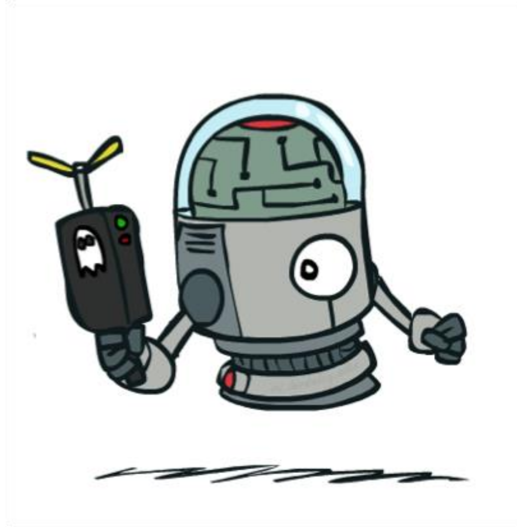
0.05	0.01	0.05	<0.01	<0.01	<0.01
0.02	0.14	0.11	0.35	<0.01	<0.01
0.07	0.03	0.05	<0.01	0.03	<0.01
0.03	0.03	<0.01	<0.01	<0.01	<0.01

Before observation

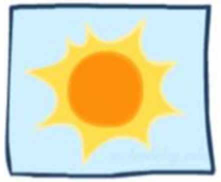
<0.01	<0.01	<0.01	<0.01	0.02	<0.01
<0.01	<0.01	<0.01	0.83	0.02	<0.01
<0.01	<0.01	0.11	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01

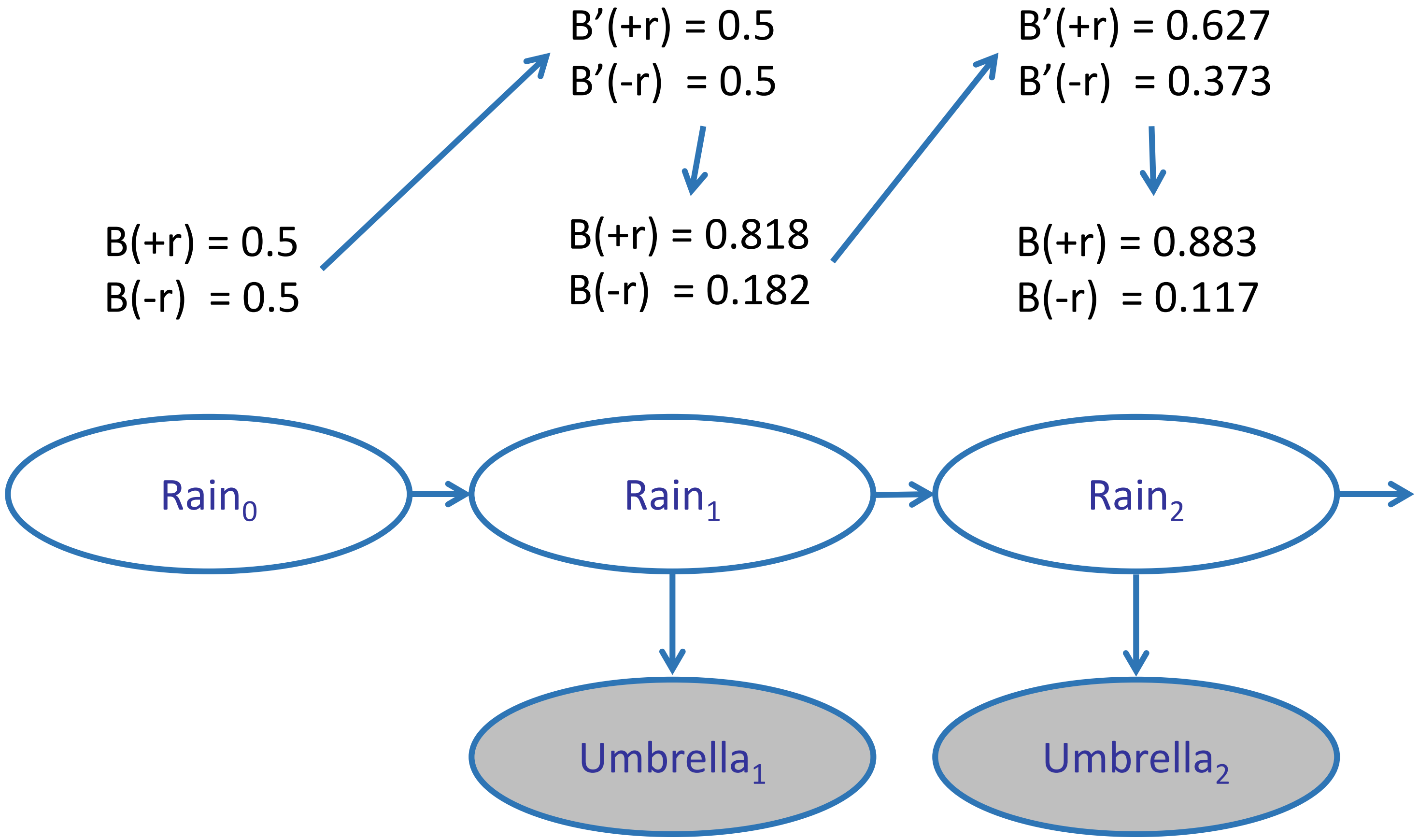
After observation

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



Example: Weather HMM



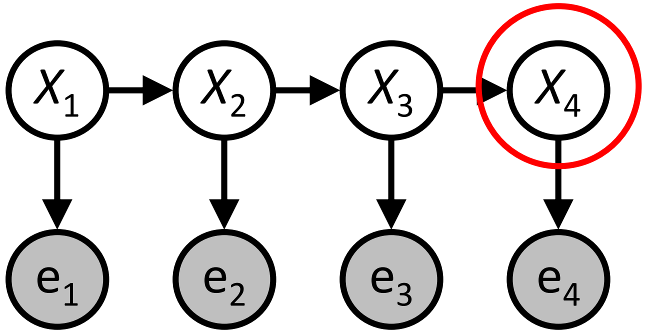


R_t	R_{t+1}	$P(R_{t+1} \mid R_t)$
+r	+r	0.7
+r	-r	0.3
-r	+r	0.3
-r	-r	0.7

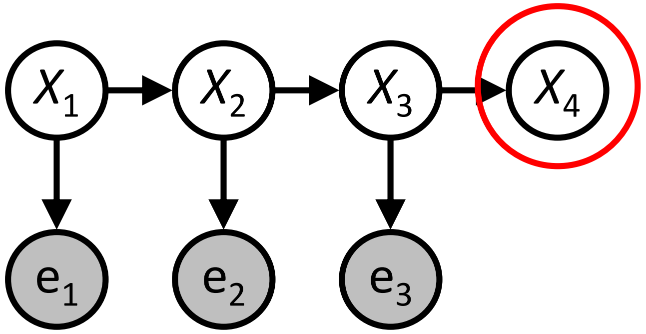
R_t	U_t	$P(U_t \mid R_t)$
+r	+u	0.9
+r	-u	0.1
-r	+u	0.2
-r	-u	0.8

Other HMM Queries

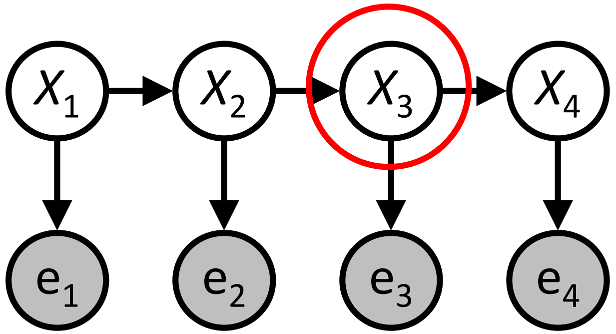
Filtering: $P(X_t | e_{1:t})$



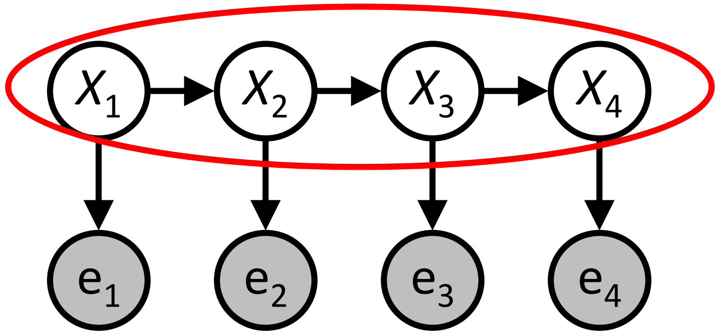
Prediction: $P(X_t | e_{1:t-1})$



Smoothing: $P(X_t | e_{1:N})$, $t<N$



Explanation: $P(X_{1:N} | e_{1:N})$

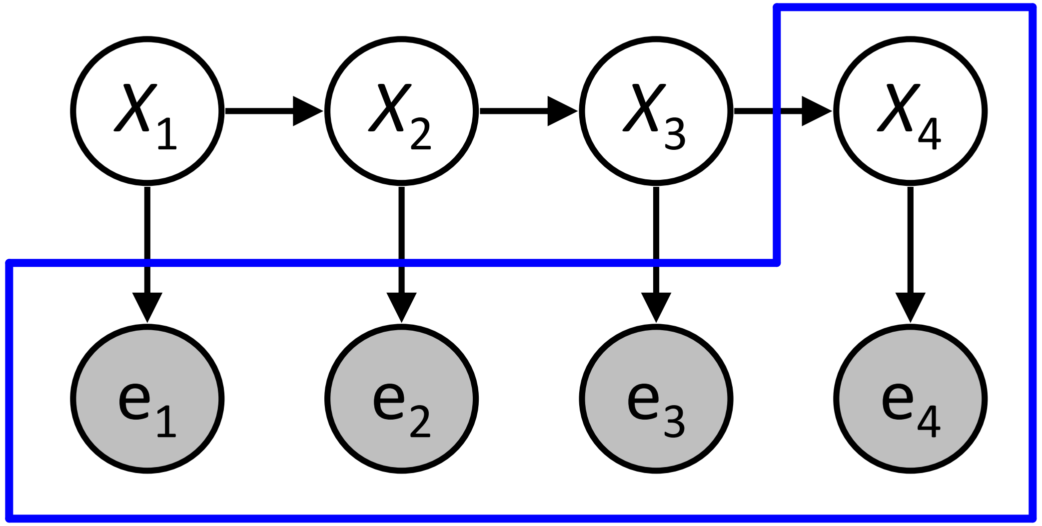


The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

$$P(x_t|e_{1:t}) \propto_{X_t} P(x_t, e_{1:t})$$



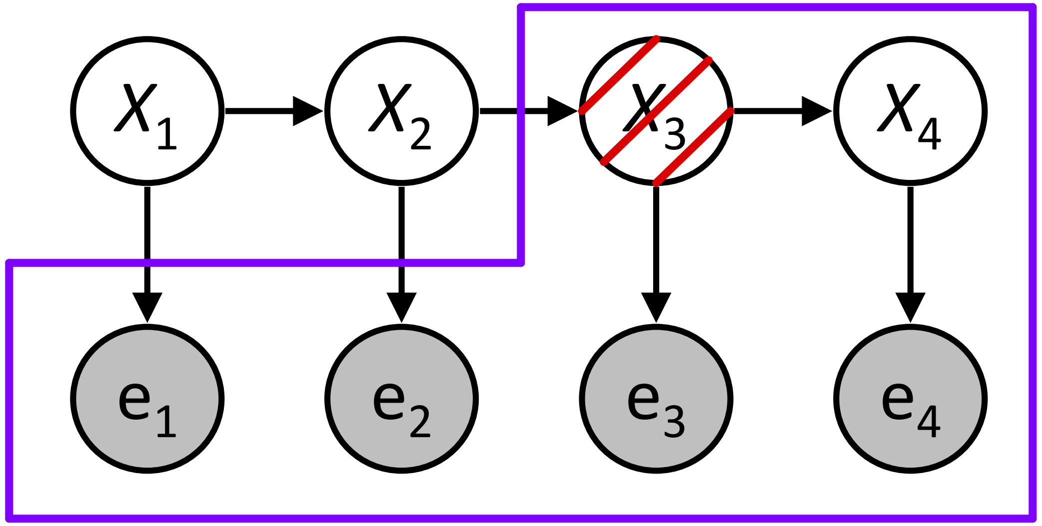
The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

$$P(x_t|e_{1:t}) \propto_{X_t} P(x_t, e_{1:t})$$

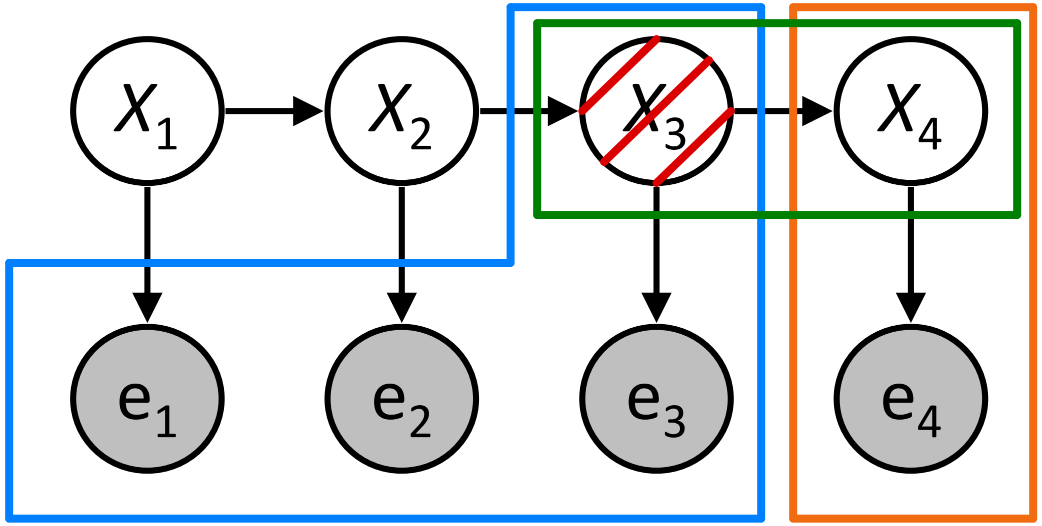
$$= \sum_{x_{t-1}} P(x_{t-1}, x_t, e_{1:t})$$



The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net



$$P(x_t|e_{1:t}) \propto_{X_t} P(x_t, e_{1:t})$$

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

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

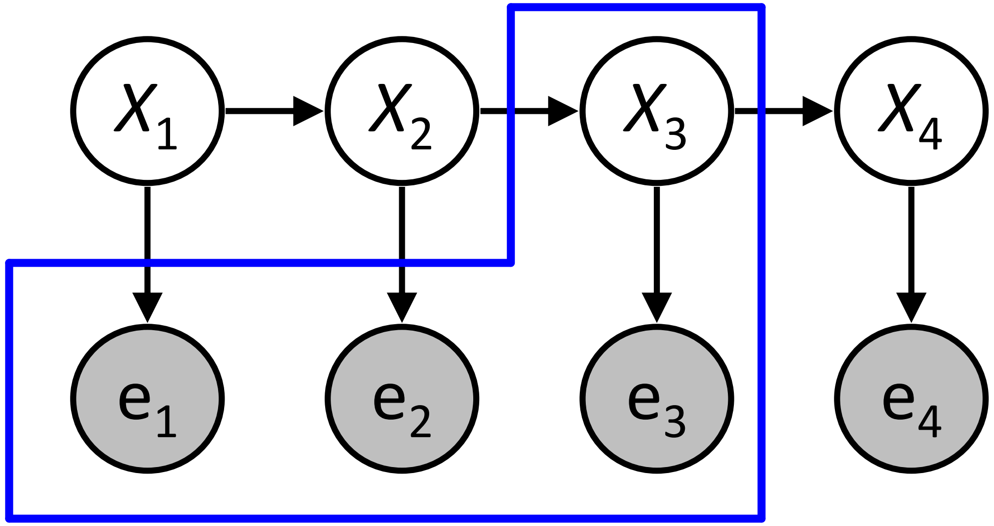
$$= P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1})\, P(x_{t-1}|e_{1:t-1})$$

The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

Recursive



$$P(x_t|e_{1:t}) \propto_{X_t} P(x_t, e_{1:t})$$

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

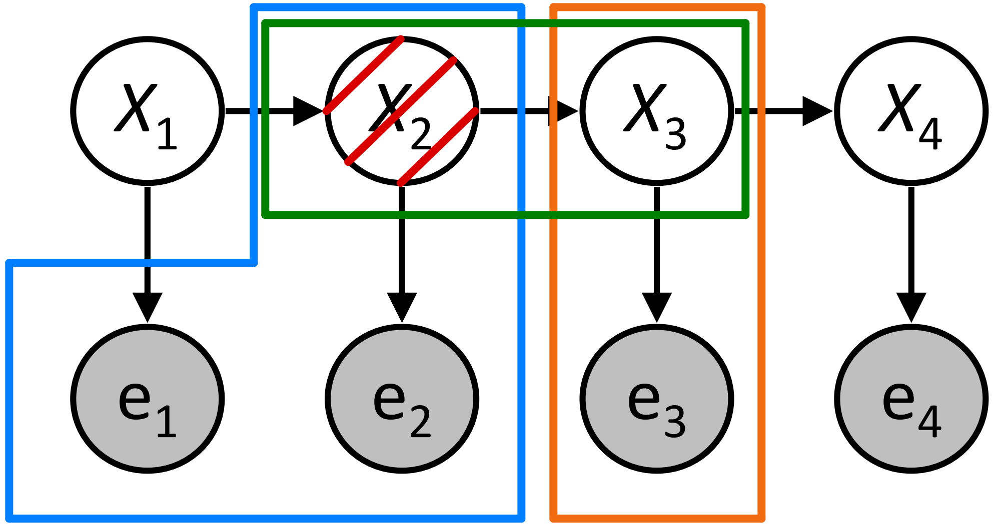
$$= \sum_{x_{t-1}} P(x_{t-1}|e_{1:t-1})\, P(x_t|x_{t-1})\, P(e_t|x_t)$$

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

The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net



$$P(x_t|e_{1:t}) \propto_{X_t} P(x_t, e_{1:t})$$

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

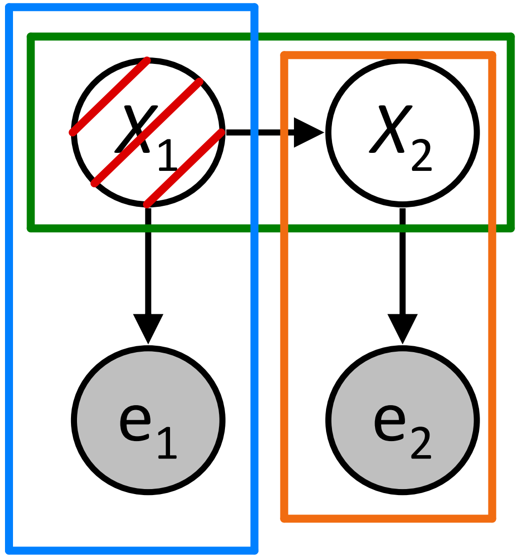
$$= \sum_{x_{t-1}} P(x_{t-1}|e_{1:t-1})\, P(x_t|x_{t-1})\, P(e_t|x_t)$$

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

The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

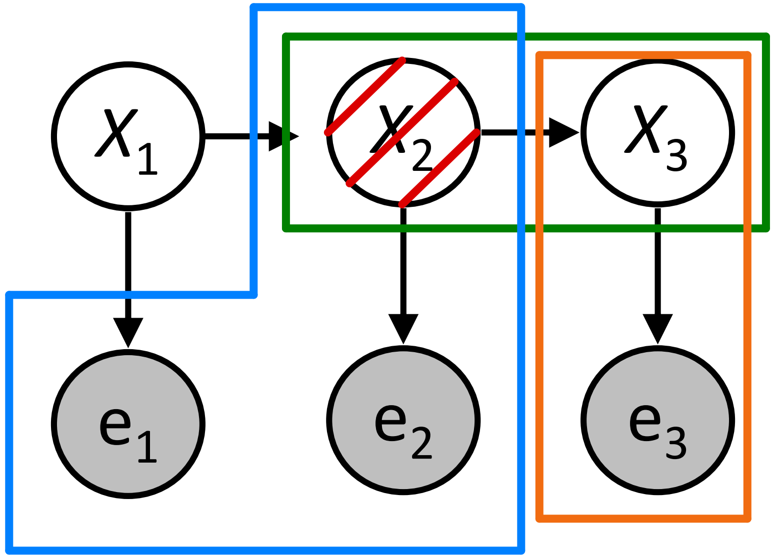
Marching **forward** through the HMM network



The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

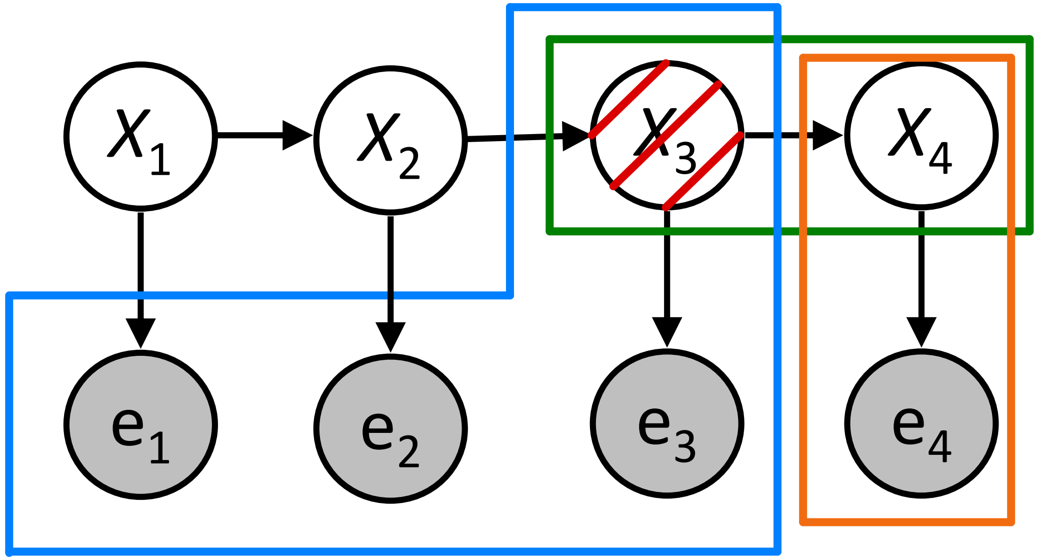
Marching **forward** through the HMM network



The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

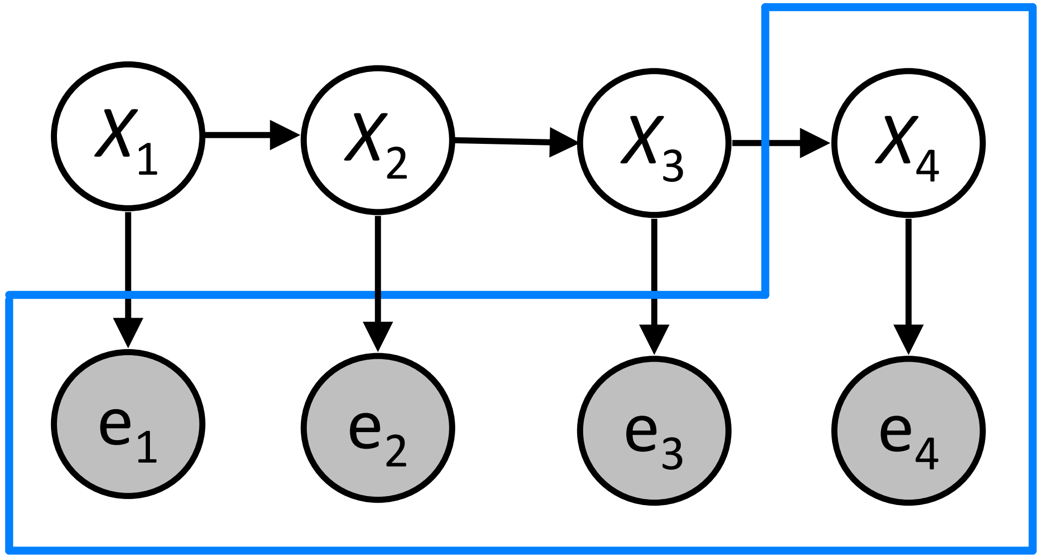
Marching **forward** through the HMM network



The Forward Algorithm

Query: What is the current state, given all of the current and past evidence?

Marching **forward** through the HMM network

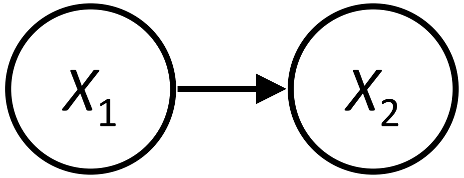


Online Belief Updates

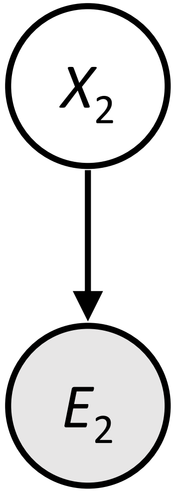
Every time step, we start with current P(X | evidence)

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)

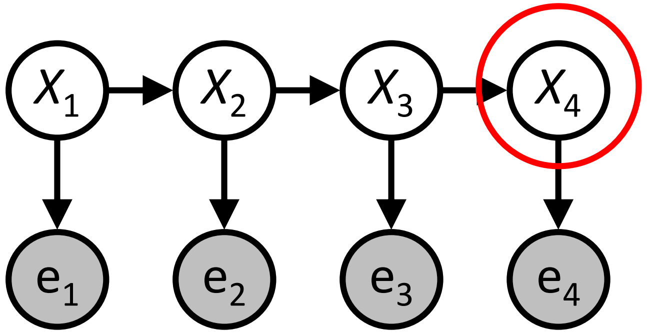
Pacman – Sonar (P5)

[Demo: Pacman – Sonar – No Beliefs(L14D1)]

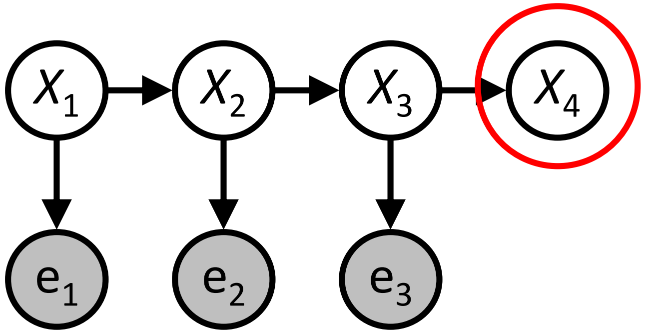
Video of Demo Pacman – Sonar (with beliefs)

Other HMM Queries

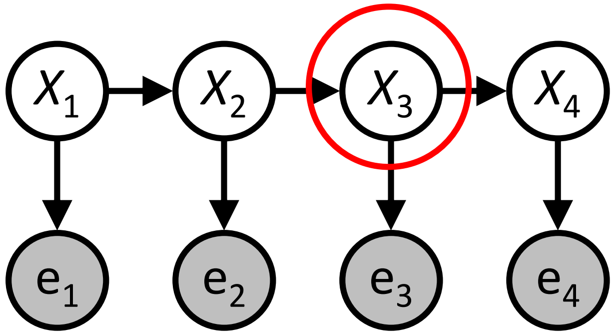
Filtering: $P(X_t | e_{1:t})$



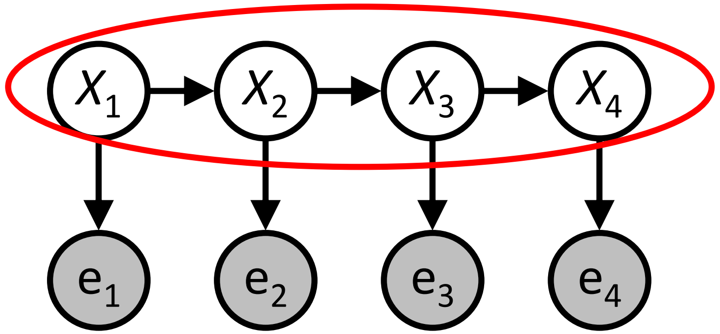
Prediction: $P(X_t | e_{1:t-1})$



Smoothing: $P(X_t | e_{1:N})$, $t<N$



Explanation: $P(X_{1:N} | e_{1:N})$



Next Time: Particle Filtering and Applications of HMMs