Markov Chains

Toolbox

- Search: uninformed/heuristic
- Adversarial search
- Probability
- Bayes nets
  - Naive Bayes classifiers

Reasoning over time

- In a Bayes net, each random variable (node) takes on one specific value.
  - Good for modeling static situations.

- What if we need to model a situation that is changing over time?

Example: Comcast

- In 2004 and 2007, Comcast had the worst customer satisfaction rating of any company or gov't agency, including the IRS.
- I have cable internet service from Comcast, and sometimes my router goes down. If the router is online, it will be online the next day with prob=0.8. If it's offline, it will be offline the next day with prob=0.4.
- How do we model the probability that my router will be online/offline tomorrow? In 2 days?

Example: Waiting in line

- You go to the Apple Store to buy the latest iPhone. Every minute, the first person in line is served with prob=0.5.
- Every minute, a new person joins the line with probability

  1 if the line length=0

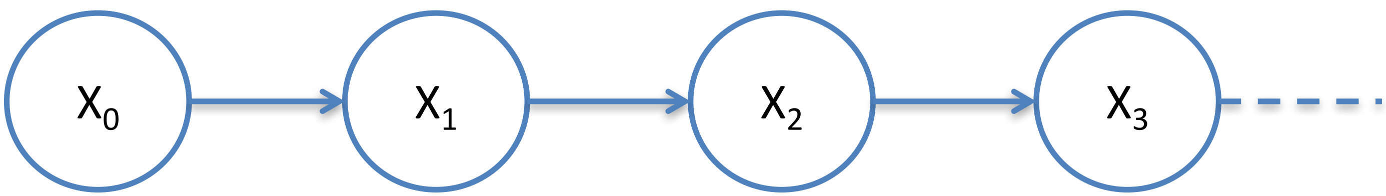
  2/3 if the line length=1

  1/3 if the line length=2

  0 if the line length=3
- How do we model what the line will look like in 1 minute? In 5 minutes?

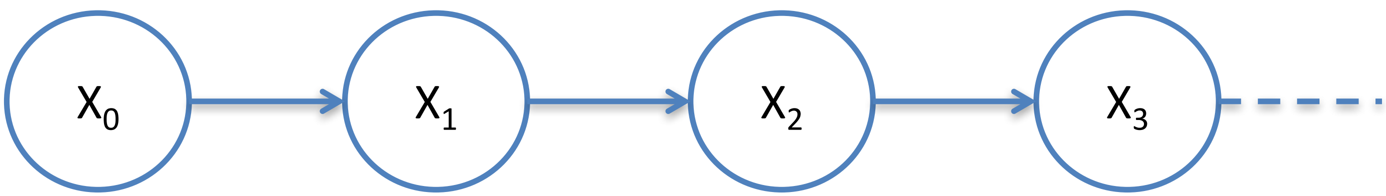
Markov Chains

- A Markov chain is a type of Bayes net with a potentially infinite number of variables (nodes).
- Each variable describes the state of the system at a given point in time (t).



Markov Chains

- Markov property:
  $P(X_t \mid X_{t-1}, X_{t-2}, X_{t-3}, \ldots) = P(X_t \mid X_{t-1})$
- Probabilities for each variable are identical:
  $P(X_t \mid X_{t-1}) = P(X_1 \mid X_0)$



Markov Chains

- Since these are just Bayes nets, we can use standard Bayes net ideas.
  - Shortcut notation: $X_{i:j}$ will refer to all variables X_i through X_j, inclusive.
- Common questions:
  - What is the probability of a specific event happening in the future?
  - What is the probability of a specific sequence of events happening in the future?

An alternate formulation

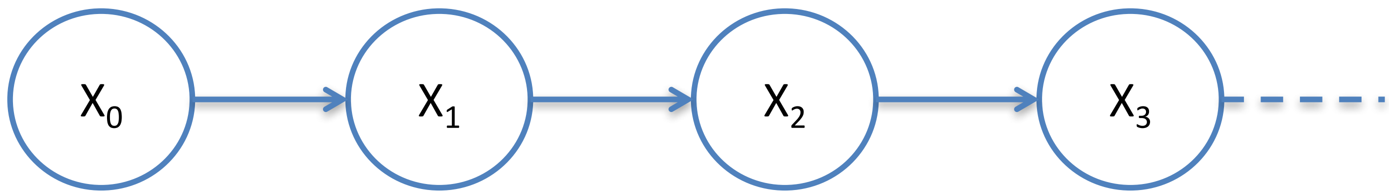
- We have a set of states, S.
- The Markov chain is always in *exactly one* state at any given time t.
- The chain transitions to a new state at each time t+1 based only on the current state at time t.

$$p_{ij} = P(X_{t+1} = j \mid X_t = i)$$

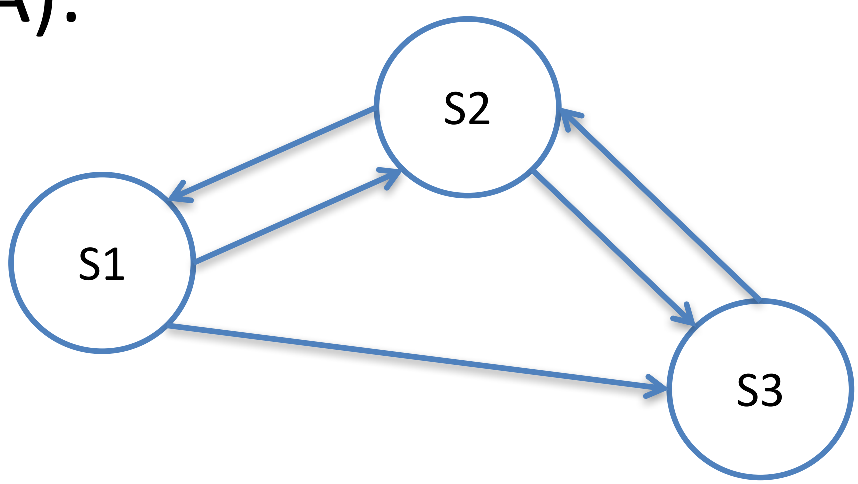
- Chain must specify p_{ij} for all i and j, and starting probabilities for $P(X_0 = j)$ for all j.

Two different representations

- As a Bayes net:



- As a state transition diagram (similar to a DFA/NFA):



Formulate Comcast in both ways

- I have cable internet service from Comcast, and sometimes my router goes down. If the router is online, it will be online the next day with prob=0.8. If it's offline, it will be offline the next day with prob=0.4.
- Let’s draw this situation in both ways.
- Assume on day 0, probability of router being down is 0.5.

Comcast

- What is the probability my router is offline for 3 days in a row (days 0, 1, and 2)?
  - $P(X_0=\text{off}, X_1=\text{off}, X_2=\text{off})$?
  - $P(X_0=\text{off}) * P(X_1=\text{off} | X_0=\text{off}) * P(X_2=\text{off} | X_1=\text{off})$
  - $P(X_0=\text{off}) * p_{\text{off,off}} * p_{\text{off,off}}$

$$P(x_{0:t}) = P(x_0) \prod_{i=1}^{t} P(x_i \mid x_{i-1})$$

More Comcast

- Suppose I don't know if my router is online right now (day 0). What is the prob it is offline tomorrow?
  - $P(X_1\text{=off})$
  - $P(X_1\text{=off}) = P(X_1\text{=off}, X_0\text{=on}) + P(X_1\text{=off}, X_0\text{=off})$
  - $P(X_1\text{=off}) = P(X_1\text{=off}|X_0\text{=on}) * P(X_0\text{=on})$
    $+ P(X_1\text{=off}|X_0\text{=off}) * P(X_0\text{=off})$

$$P(X_{t+1}) = \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t)$$

More Comcast

- Suppose I don't know if my router is online right now (day 0). What is the prob it is offline **the day after tomorrow**?
  - $P(X_2\text{=off})$
  - $P(X_2\text{=off}) = P(X_2\text{=off}, X_1\text{=on}) + P(X_2\text{=off}, X_1\text{=off})$
  - $P(X_2\text{=off}) = P(X_2\text{=off}|X_1\text{=on}) * P(X_1\text{=on})$
    $+ P(X_2\text{=off}|X_1\text{=off}) * P(X_1\text{=off})$

$$P(X_{t+1}) = \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t)$$

Markov chains with matrices

- Define a transition matrix for the chain:

$$T = \begin{bmatrix} 0.8 & 0.2 \\ 0.6 & 0.4 \end{bmatrix}$$

- Each row of the matrix represents the transition probabilities **leaving** a state.
- Let v_t = a row vector representing the probability that the chain is in each state at time t.
- $v_t = v_{t-1} * T$

Mini-forward algorithm

- Suppose we are given the values of X_0, X_1, ... X_t, and we want to know X_{t+1}.
- $P(X_{t+1} \mid X_0, X_1, ..., X_t)$
- Row vector $v_0 = P(X_0)$
- $v_1 = v_0 * T$
- $v_2 = v_1 * T = v_0 * T * T = v_0 * T^2$
- $v_3 = v_0 * T^3$
- $v_t = v_0 * T^t$

Back to the Apple Store...

- You go to the Apple Store to buy the latest iPhone. Every minute, the first person in line is served with prob=0.5.
- Every minute, a new person joins the line with probability

  1 if the line length=0

  2/3 if the line length=1

  1/3 if the line length=2

  0 if the line length=3
- Model this as a Markov chain, assuming the line starts empty. Draw the state transition diagram.
- What is T? What is v_0?

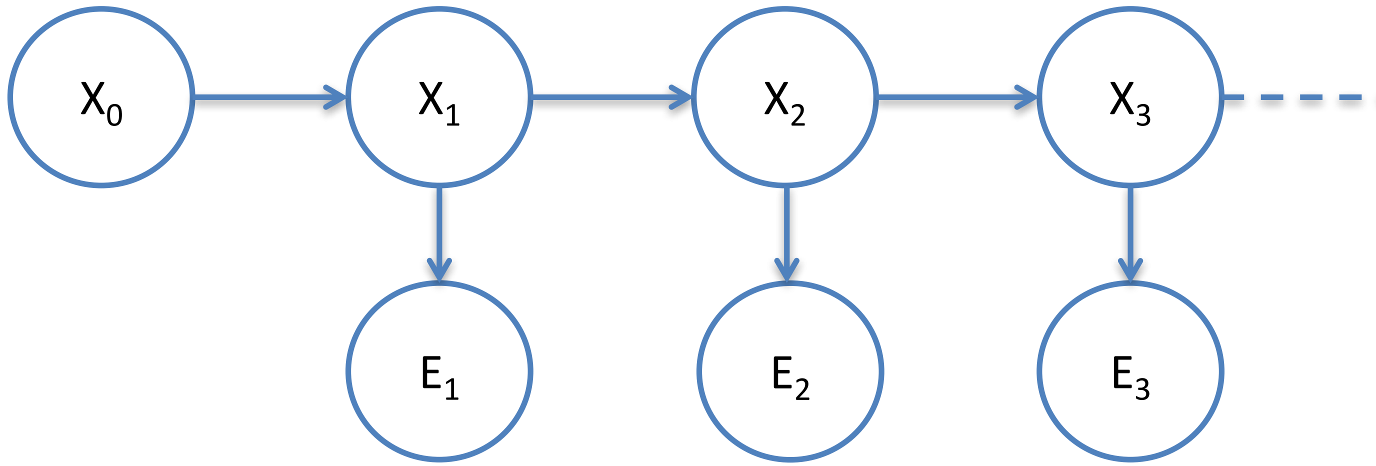
- Markov chains are pretty easy!
- But sometimes they aren't realistic…

- What if we can't directly know the states of the model, but we can see some indirect evidence resulting from the states?

Weather

- Regular Markov chain
  - Each day the weather is rainy or sunny.
  - $P(X_t = \text{rain} \mid X_{t-1} = \text{rain}) = 0.7$
  - $P(X_t = \text{sunny} \mid X_{t-1} = \text{sunny}) = 0.9$
- Twist:
  - Suppose you work in an office with no windows. All you can observe is weather your colleague brings their umbrella to work.

Hidden Markov Models

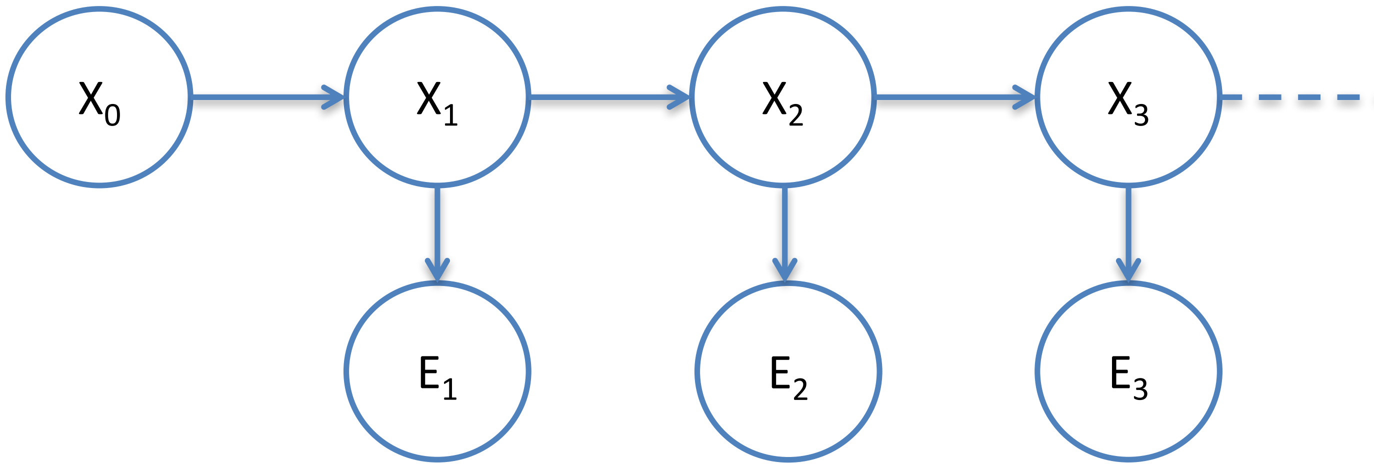


- The X's are the state variables (never directly observed).
- The E's are evidence variables.

Common real-world uses

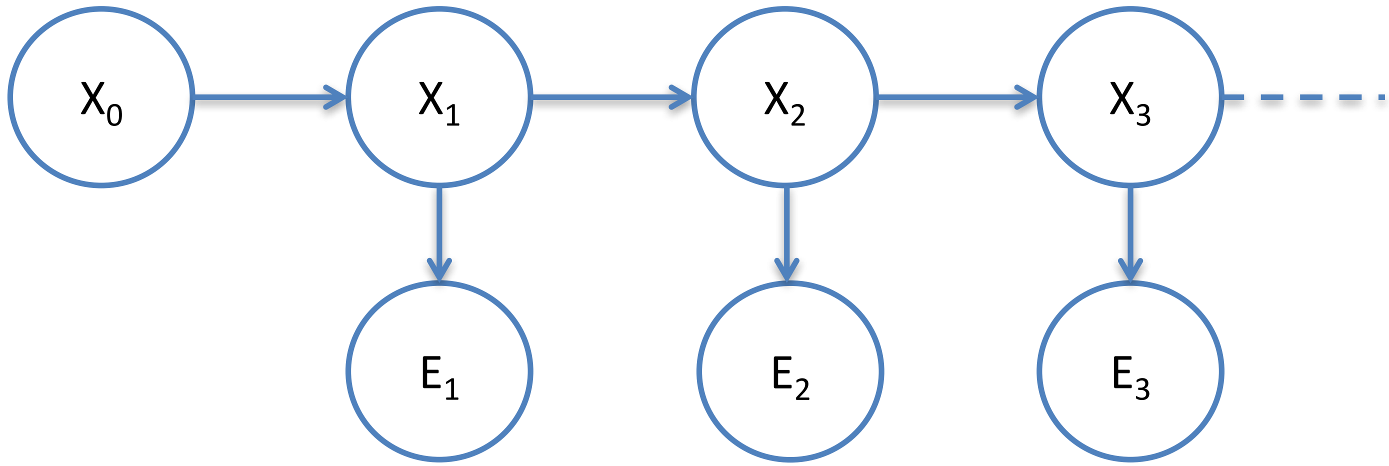
- Speech processing:
  - Observations are sounds, states are words.
- Localization:
  - Observations are inputs from video cameras or microphones, state is the actual location.
- Video processing (example):
  - Extracting a human walking from each video frame. Observations are the frames, states are the positions of the legs.

Hidden Markov Models



- $P(X_t \mid X_{t-1}, X_{t-2}, X_{t-3}, \ldots) = P(X_t \mid X_{t-1})$
- $P(X_t \mid X_{t-1}) = P(X_1 \mid X_0)$
- $P(E_t \mid X_{0:t}, E_{0:t-1}) = P(E_t \mid X_t)$
- $P(E_t \mid X_t) = P(E_1 \mid X_1)$

Hidden Markov Models



- What is $P(X_{0:t}, E_{1:t})$?

$$P(X_0)\prod_{i=1}^{t} P(X_i \mid X_{i-1}) P(E_i \mid X_i)$$

Common questions

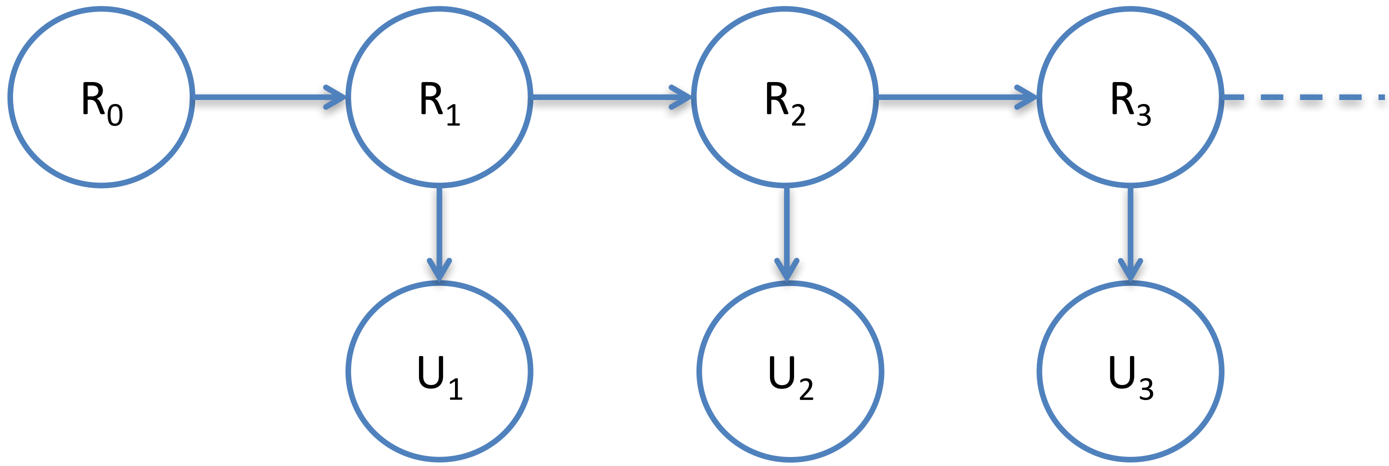
- **Filtering**: Given a sequence of observations, what is the most probable *current* state?
  - Compute $P(X_t \mid e_{1:t})$
- **Prediction**: Given a sequence of observations, what is the most probable *future* state?
  - Compute $P(X_{t+k} \mid e_{1:t})$ for some $k > 0$
- **Smoothing**: Given a sequence of observations, what is the most probable *past* state?
  - Compute $P(X_k \mid e_{1:t})$ for some $k < t$

Common questions

- **Most likely explanation:** Given a sequence of observations, what is the most probable sequence of states?
  - Compute $\underset{x_{1:t}}{\operatorname{argmax}} P(x_{1:t} \mid e_{1:t})$

- **Learning**: How can we estimate the transition and sensor models from real-world data? (Future machine learning class?)

Hidden Markov Models



- $P(R_t = yes \mid R_{t-1} = yes) = 0.7$
  $P(R_t = yes \mid R_{t-1} = no) = 0.1$
- $P(U_t = yes \mid R_t = yes) = 0.9$
  $P(U_t = yes \mid R_t = no) = 0.2$

Filtering

- Filtering is concerned with finding the most probable "current" state from a sequence of evidence.
- Let's compute this.

Forward algorithm

- Recursive computation of the probability distribution over current states.
- Say we have $P(X_t \mid e_{1:t})$

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

$$\alpha P(e_{t+1} \mid X_{t+1}) \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t \mid e_{1:t})$$

Forward algorithm

- Markov chain version:

$$P(X_{t+1}) = \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t)$$

- Hidden Markov model version:

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

$$\alpha P(e_{t+1} \mid X_{t+1}) \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t \mid e_{1:t})$$

Forward algorithm

- Today is Day 2, and I've been pulling all-nighters for two days!
- My colleague brought their umbrella on days 1 and 2.
- What is the probability it is raining today?

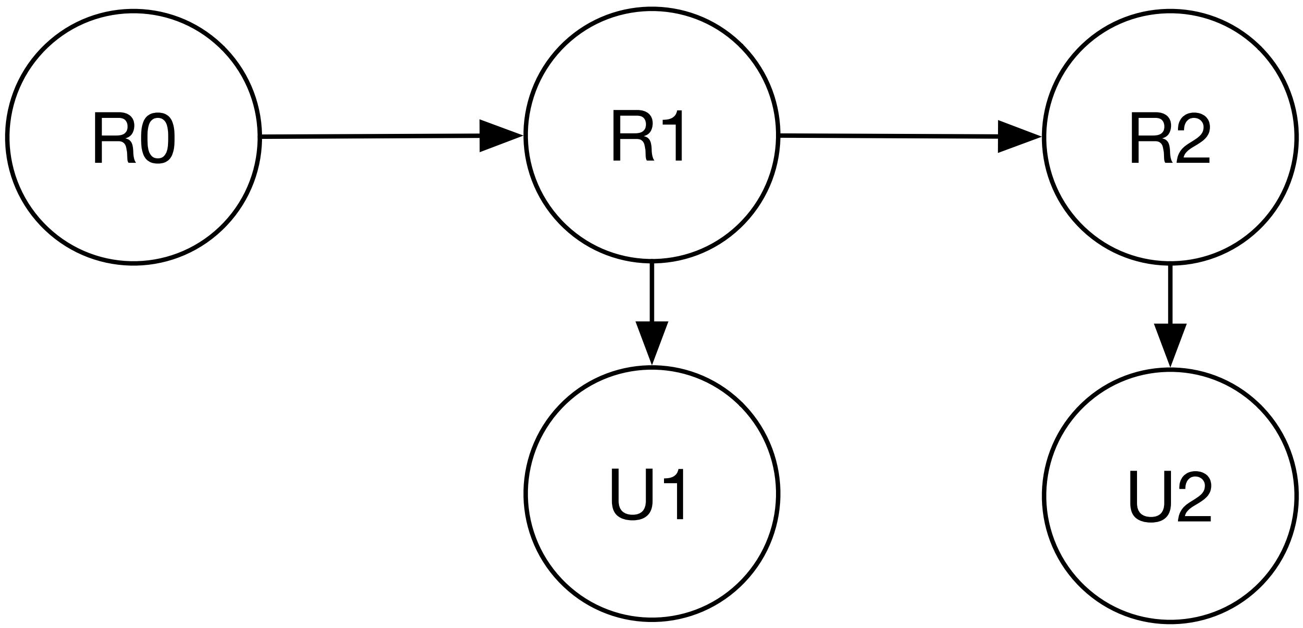
Matrices to the rescue!

- Define a transition matrix T as normal.
- Define a sequence of observation matrices O_1 through O_t.
- Each O matrix is a diagonal matrix with the entries corresponding to that particular observation given each state.

$$f_{1:t+1} = \alpha f_{1:t} \cdot T \cdot O_{t+1}$$

where each f is a row vector containing the probability distribution at state t.

f1:0=[0.5, 0.5]   f1:1=[0.75, 0.25]   f1:2=[0.846, 0.154]



$$T = \begin{bmatrix} 0.7, & 0.3 \\ 0.1, & 0.9 \end{bmatrix}$$

$$O1 = \begin{bmatrix} 0.9, & 0.0 \\ 0.0, & 0.2 \end{bmatrix}$$

$$O2 = \begin{bmatrix} 0.9, & 0.0 \\ 0.0, & 0.2 \end{bmatrix}$$

f1:0 = P(R0) = [0.5, 0.5]

f1:1 = P(R1 | u1) = $\boldsymbol{\alpha}$ * f1:0 * T * O1 = $\boldsymbol{\alpha}$[0.36, 0.12] = [0.75, 0.25]

f1:2 = P(R2 | u1, u2) = $\boldsymbol{\alpha}$ * f1:1 * T * O2 = $\boldsymbol{\alpha}$[0.495, 0.09] = [.846, .154]

Forward algorithm

- Note that the forward algorithm only gives you the probability of X_t taking into account evidence at times 1 through t.
- In other words, say you calculate $P(X_1 \mid e_1)$ using the forward algorithm, then you calculate $P(X_2 \mid e_1, e_2)$.
  - Knowing e2 changes your calculation of X1.
  - That is, $P(X_1 \mid e_1) != P(X_1 \mid e_1, e_2)$

Backward algorithm

- Updates previous probabilities to take into account new evidence.
- Calculates $P(X_k \mid e_{1:t})$ for $k < t$
  - aka smoothing.

Backward matrices

- Main equations:

$$b_{k:t} = T \cdot O_k \cdot b_{k+1:t}$$

$$b_{t+1:t} = [1; \cdots ; 1]$$ (column vec of 1s)

$$P(X_k \mid e_{1:t}) = \alpha f_{1:k} \times b_{k+1:t}$$

f1:0=[0.5, 0.5]

b1:2=[0.4509, 0.1107]

mult=[0.803, 0.197]

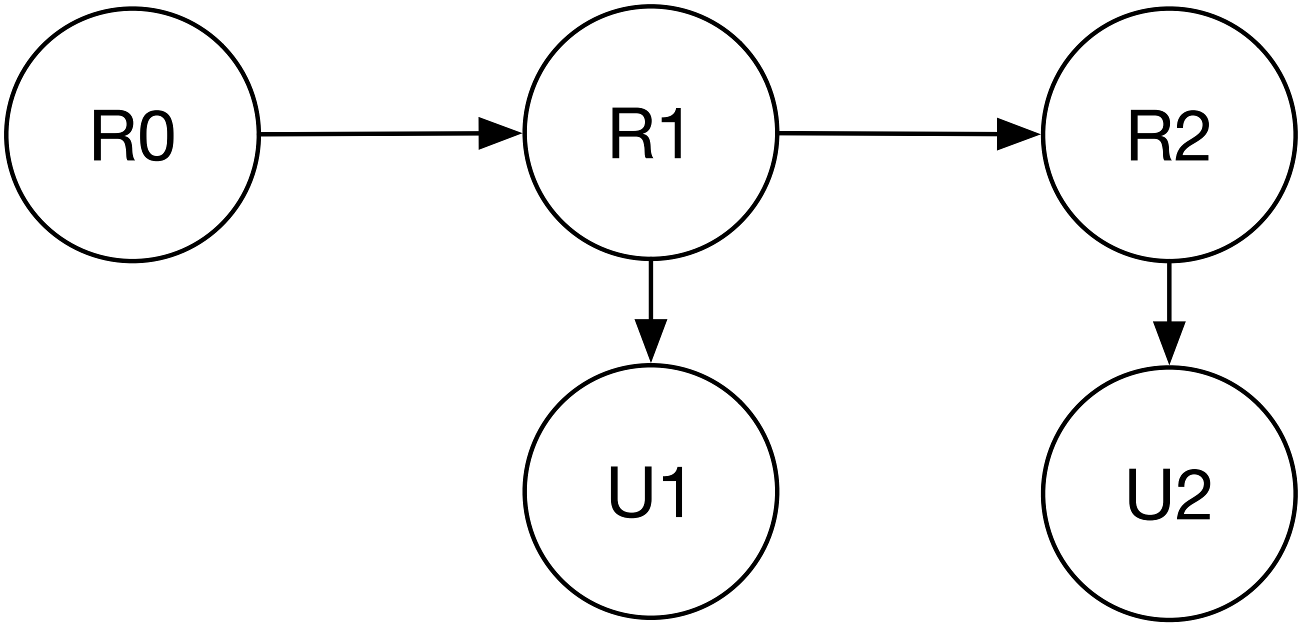
f1:1=[0.75, 0.25]

b2:2=[0.69, 0.27]

mult=[0.885, 0.115]

f1:2=[0.846, 0.154]

b3:2=[1; 1]



$$T = \begin{bmatrix} 0.7, & 0.3 \\ 0.1, & 0.9 \end{bmatrix}$$

$$O1 = \begin{bmatrix} 0.9, & 0.0 \\ 0.0, & 0.2 \end{bmatrix}$$

$$O2 = \begin{bmatrix} 0.9, & 0.0 \\ 0.0, & 0.2 \end{bmatrix}$$

b3:2 = [1; 1]

b2:2 = T * O2 * b3:2 = [0.69, 0.27]

P(R1 | u1, u2) = α f1:1 x b2:2 = α[0.5175, 0.0675] = [0.885, 0.115]

b1:2 = T * O1 * b2:2 = [0.4509, 0.1107]

P(R0 | u1, u2) = α f1:0 x b1:2 = α[0.5175, 0.0675] = [0.803, 0.197]

Forward-backward algorithm

$$f_{1:0} = P(X_0)$$

$$f_{1:t+1} = \alpha f_{1:t} \cdot T \cdot O_{t+1}$$

Compute these forward from X_0 to wherever you want to stop (X_t)

$$b_{t+1:t} = [1; \cdots ; 1]$$

$$b_{k:t} = T \cdot O_k \cdot b_{k+1:t}$$

$$P(X_k \mid e_{1:t}) = \alpha f_{1:k} \times b_{k+1:t}$$

Compute these backwards from X_{t+1} to X_0.