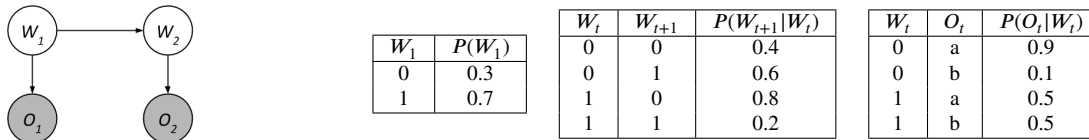


1 Particle Filtering

Let's use Particle Filtering to estimate the distribution of $P(W_2|O_1 = a, O_2 = b)$. Here's the HMM again. O_1 and O_2 are supposed to be shaded.



We start with two particles representing our distribution for W_1 .

$P_1 : W_1 = 0$

$P_2 : W_1 = 1$

Use the following random numbers to run particle filtering:

[0.22, 0.05, 0.33, 0.20, 0.84, 0.54, 0.79, 0.66, 0.14, 0.96]

1. **Observe:** Compute the weight of the two particles after evidence $O_1 = a$.

2. **Resample:** Using the random numbers, resample P_1 and P_2 based on the weights.

3. **Predict:** Sample P_1 and P_2 from applying the time update.

4. **Update:** Compute the weight of the two particles after evidence $O_2 = b$.

5. **Resample:** Using the random numbers, resample P_1 and P_2 based on the weights.

6. What is our estimated distribution for $P(W_2|O_1 = a, O_2 = b)$?

Q2. Independence in Hidden Markov Models

Below is a full derivation of the forward algorithm updates for Hidden Markov Models. As seen in lecture, we used $e_{1:t}$ to denote all the evidence variables e_1, e_2, \dots, e_t . Similarly, $e_{1:t-1}$ denotes e_1, e_2, \dots, e_{t-1} . For reference, the Bayes net corresponding to the usual Hidden Markov Model is shown on the right side of the derivation below.

$$P(x_t | e_{1:t}) \propto P(x_t, e_{1:t}) \quad (1)$$

$$= \sum_{x_{t-1}} P(x_{t-1}, x_t, e_{1:t}) \quad (2)$$

$$= \sum_{x_{t-1}} P(x_{t-1}, x_t, e_{1:t-1}, e_t) \quad (3)$$

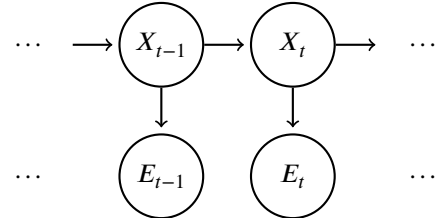
$$= \sum_{x_{t-1}} P(e_t | x_{t-1}, x_t, e_{1:t-1}) P(x_{t-1}, x_t, e_{1:t-1}) \quad (4)$$

$$= \sum_{x_{t-1}} P(e_t | x_t) P(x_{t-1}, x_t, e_{1:t-1}) \quad (5)$$

$$= \sum_{x_{t-1}} P(e_t | x_t) P(x_t | x_{t-1}, e_{1:t-1}) P(x_{t-1}, e_{1:t-1}) \quad (6)$$

$$= \sum_{x_{t-1}} P(e_t | x_t) P(x_t | x_{t-1}) P(x_{t-1}, e_{1:t-1}) \quad (7)$$

$$= P(e_t | x_t) \sum_{x_{t-1}} P(x_t | x_{t-1}) P(x_{t-1}, e_{1:t-1}) \quad (8)$$



(a) The following assumption(s) are needed to justify going from step (4) to step (5):

(select all that apply)

$E_t \perp\!\!\!\perp X_{t-1} | X_t$

$E_t \perp\!\!\!\perp E_k | X_{t-1}$ for all $1 \leq k \leq t-1$

$E_t \perp\!\!\!\perp E_k | X_t$ for all $1 \leq k \leq t-1$

$X_t \perp\!\!\!\perp E_{t+1} | X_{t+1}$

$E_t \perp\!\!\!\perp E_k$ for all $1 \leq k \leq t-1$

$X_t \perp\!\!\!\perp E_k | X_{t-1}$ for all $1 \leq k \leq t-1$

$E_t \perp\!\!\!\perp E_{t+1} | X_t$

 none

(b) The following assumption(s) are needed to justify going from step (5) to step (6):

(select all that apply)

$E_t \perp\!\!\!\perp X_{t-1} | X_t$

$E_t \perp\!\!\!\perp E_k | X_{t-1}$ for all $1 \leq k \leq t-1$

$E_t \perp\!\!\!\perp E_k | X_t$ for all $1 \leq k \leq t-1$

$X_t \perp\!\!\!\perp E_{t+1} | X_{t+1}$

$E_t \perp\!\!\!\perp E_k$ for all $1 \leq k \leq t-1$

$X_t \perp\!\!\!\perp E_k | X_{t-1}$ for all $1 \leq k \leq t-1$

$E_t \perp\!\!\!\perp E_{t+1} | X_t$

 none

(c) The following assumption(s) are needed to justify going from step (6) to step (7):

(select all that apply)

$E_t \perp\!\!\!\perp X_{t-1} | X_t$

$E_t \perp\!\!\!\perp E_k | X_{t-1}$ for all $1 \leq k \leq t-1$

$E_t \perp\!\!\!\perp E_k | X_t$ for all $1 \leq k \leq t-1$

$X_t \perp\!\!\!\perp E_{t+1} | X_{t+1}$

$E_t \perp\!\!\!\perp E_k$ for all $1 \leq k \leq t-1$

$X_t \perp\!\!\!\perp E_k | X_{t-1}$ for all $1 \leq k \leq t-1$

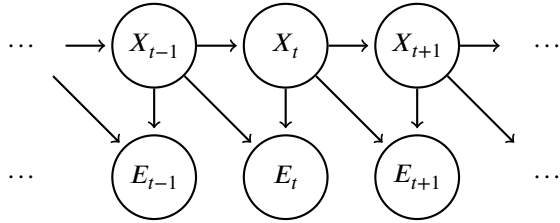
$E_t \perp\!\!\!\perp E_{t+1} | X_t$

 none

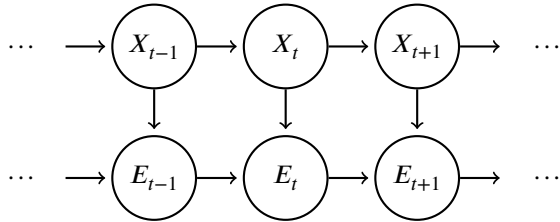
Hidden Markov Models can be extended in a number of ways to incorporate additional relations. Since the independence assumptions are different in these extended Hidden Markov Models, the forward algorithm updates will also be different.

Complete the forward algorithm updates for the extended Hidden Markov Models specified by the following Bayes nets:

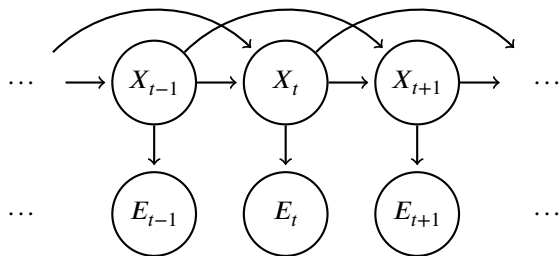
(d) $P(x_t | e_{1:t}) \propto \sum_{x_{t-1}} P(x_{t-1}, e_{1:t-1}) \cdot$ _____



(e) $P(x_t | e_{1:t}) \propto \sum_{x_{t-1}} P(x_{t-1}, e_{1:t-1}) \cdot$ _____



(f) $P(x_t, x_{t+1} | e_{1:t}) \propto \sum_{x_{t-1}} P(x_{t-1}, x_t, e_{1:t-1}) \cdot$ _____



Q3. HMMs

- (a) Consider a Markov model like the one above. For the first three parts of this problem, assume the domain of our variables is $\{a, b\}$. Fill in the table below with the probability of being in each state after a large number of transitions, when $P(V_n) = P(V_{n+1})$. If the values never reach a point when $P(V_n) = P(V_{n+1})$, write 'None'.

In the left part of the table, assume that we start with a uniform distribution ($P(V_0 = a) = P(V_0 = b) = 0.5$). In the right part of the table, assume that we start with the distribution that has $P(V_0 = a) = 1.0$.

Transition Probabilities		$P(V_n)$ given that $P(V_0)$ is uniform		$P(V_n)$ given that $P(V_0 = a) = 1.0$												
		$V_n = a$	$V_n = b$	$V_n = a$	$V_n = b$											
<table border="1"> <thead> <tr> <th rowspan="2">V_{i-1}</th> <th colspan="2">$P(V_i V_{i-1})$</th> </tr> <tr> <th>$V_i = a$</th> <th>$V_i = b$</th> </tr> </thead> <tbody> <tr> <td>a</td> <td>0.5</td> <td>0.5</td> </tr> <tr> <td>b</td> <td>0.5</td> <td>0.5</td> </tr> </tbody> </table>	V_{i-1}	$P(V_i V_{i-1})$		$V_i = a$	$V_i = b$	a	0.5	0.5	b	0.5	0.5					
		V_{i-1}	$P(V_i V_{i-1})$													
	$V_i = a$		$V_i = b$													
a	0.5	0.5														
b	0.5	0.5														
<table border="1"> <thead> <tr> <th rowspan="2">V_{i-1}</th> <th colspan="2">$P(V_i V_{i-1})$</th> </tr> <tr> <th>$V_i = a$</th> <th>$V_i = b$</th> </tr> </thead> <tbody> <tr> <td>a</td> <td>0.9</td> <td>0.1</td> </tr> <tr> <td>b</td> <td>0.3</td> <td>0.7</td> </tr> </tbody> </table>	V_{i-1}	$P(V_i V_{i-1})$		$V_i = a$	$V_i = b$	a	0.9	0.1	b	0.3	0.7					
		V_{i-1}	$P(V_i V_{i-1})$													
	$V_i = a$		$V_i = b$													
a	0.9	0.1														
b	0.3	0.7														
<table border="1"> <thead> <tr> <th rowspan="2">V_{i-1}</th> <th colspan="2">$P(V_i V_{i-1})$</th> </tr> <tr> <th>$V_i = a$</th> <th>$V_i = b$</th> </tr> </thead> <tbody> <tr> <td>a</td> <td>0.0</td> <td>1.0</td> </tr> <tr> <td>b</td> <td>1.0</td> <td>0.0</td> </tr> </tbody> </table>	V_{i-1}	$P(V_i V_{i-1})$		$V_i = a$	$V_i = b$	a	0.0	1.0	b	1.0	0.0					
		V_{i-1}	$P(V_i V_{i-1})$													
	$V_i = a$		$V_i = b$													
a	0.0	1.0														
b	1.0	0.0														

For this part our variables have the domain $\{a, b, c\}$. Fill in the table at the bottom with the probability of being in each state after a large number of transitions, when $P(V_n) = P(V_{n+1})$. In the left part of the table, assume that we start with a uniform distribution ($P(V_0 = a) = P(V_0 = b) = P(V_0 = c) = \frac{1}{3}$). In the right part of the table, assume that we start with the distribution that has $P(V_0 = a) = 1.0$.

V_{i-1}	$P(V_i V_{i-1})$		
	$V_i = a$	$V_i = b$	$V_i = c$
a	0.5	0.5	0.0
b	0.5	0.5	0.0
c	0.0	0.0	1.0

$P(V_n)$ given that $P(V_0)$ is uniform			$P(V_n)$ given that $P(V_0 = a) = 1.0$		
a	b	c	a	b	c

Now we will consider a Hidden Markov Model, and look at properties of the Viterbi Algorithm. The Viterbi algorithm finds the most probable sequence of hidden states $X_{1:S}$ given a sequence of observations $y_{1:S}$. Recall that for the canonical HMM structure, the Viterbi algorithm performs the following update at each time step:

$$m_t[x_t] = P(y_t|x_t) \max_{x_{t-1}} [P(x_t|x_{t-1})m_{t-1}[x_{t-1}]]$$

Assume we have an HMM where:

- The hidden variable X can take on H values
- The (observed) emission variable Y can take on E values
- Our sequence has S steps

(b) (i) What is the run time of the Viterbi algorithm?

- | | | |
|---------------------------------------|----------------------------------|---------------------------------|
| <input type="radio"/> $O(SEH)$ | <input type="radio"/> $O(SEH^2)$ | <input type="radio"/> $O(SH^2)$ |
| <input type="radio"/> $O(SH)$ | <input type="radio"/> $O(EH)$ | <input type="radio"/> $O(EH^2)$ |
| <input type="radio"/> $O(SH^2 + SEH)$ | | |

Ignoring the storage of the emission probabilities, $P(Y_t|X_t)$, and the transition probabilities, $P(X_t|X_{t-1})$, what are the storage requirements of the Viterbi algorithm?

- | | | |
|----------------------------------|----------------------------------|----------------------------------|
| <input type="radio"/> $O(S)$ | <input type="radio"/> $O(E)$ | <input type="radio"/> $O(H)$ |
| <input type="radio"/> $O(SH)$ | <input type="radio"/> $O(SE)$ | <input type="radio"/> $O(EH)$ |
| <input type="radio"/> $O(S + H)$ | <input type="radio"/> $O(S + E)$ | <input type="radio"/> $O(E + H)$ |
| <input type="radio"/> $O(SEH)$ | | |

Now, assume that most of the transitions in our HMM have probability zero. In particular, suppose that for any given hidden state value, there are only K possible next state values for which the transition probability is non-zero. To exploit this sparsity, we change the Viterbi Algorithm to only consider the non-zero transition edges during each max computation inside each update. You can think of this as the Viterbi algorithm ignoring edges that correspond to zero probability transitions in the transition lattice diagram.

(ii) What is the run time of this modified algorithm?

- | | | |
|---------------------------------------|--------------------------------------|---------------------------------|
| <input type="radio"/> $O(SEH)$ | <input type="radio"/> $O(SEH^2)$ | <input type="radio"/> $O(SH^2)$ |
| <input type="radio"/> $O(SH)$ | <input type="radio"/> $O(EH)$ | <input type="radio"/> $O(EH^2)$ |
| <input type="radio"/> $O(SH^2 + SEH)$ | | |
| <input type="radio"/> $O(SEK)$ | <input type="radio"/> $O(SEHK)$ | <input type="radio"/> $O(SHK)$ |
| <input type="radio"/> $O(SK)$ | <input type="radio"/> $O(EK)$ | <input type="radio"/> $O(EHK)$ |
| <input type="radio"/> $O(SK + SEK)$ | <input type="radio"/> $O(SHK + SEK)$ | |

Ignoring the storage of the emission probabilities, $P(Y_t|X_t)$, and the transition probabilities, $P(X_t|X_{t-1})$, what are the storage requirements of this modified Viterbi algorithm?

- | | | |
|----------------------------------|----------------------------------|----------------------------------|
| <input type="radio"/> $O(S)$ | <input type="radio"/> $O(E)$ | <input type="radio"/> $O(H)$ |
| <input type="radio"/> $O(SH)$ | <input type="radio"/> $O(SE)$ | <input type="radio"/> $O(EH)$ |
| <input type="radio"/> $O(S + H)$ | <input type="radio"/> $O(S + E)$ | <input type="radio"/> $O(E + H)$ |
| <input type="radio"/> $O(SEH)$ | | |
| <input type="radio"/> $O(K)$ | <input type="radio"/> $O(SK)$ | <input type="radio"/> $O(EK)$ |
| <input type="radio"/> $O(HK)$ | <input type="radio"/> $O(S + K)$ | <input type="radio"/> $O(E + K)$ |
| <input type="radio"/> $O(H + K)$ | <input type="radio"/> $O(SEK)$ | |