CSE 150. Assignment 6 *Fall 2018*

Out: *Tue Nov 13*
Due: *Tue Nov 20*

Supplementary reading:

- Russell & Norvig, Chapter 15.
- L. R. Rabiner (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77(2):257286.
  (Available at: https://tinyurl.com/y879md2u)

6.1 Viterbi algorithm

In this problem, you will decode an English phrase from a long sequence of non-text observations. To do so, you will implement the same algorithm used in modern engines for automatic speech recognition. In a speech recognizer, these observations would be derived from real-valued measurements of acoustic waveforms. Here, for simplicity, the observations only take on binary values, but the high-level concepts are the same.

Consider a discrete HMM with $n = 27$ hidden states $S_t \in \{1, 2, \ldots, 27\}$ and binary observations $O_t \in \{0, 1\}$. Download the four ASCII data files from the course website for this assignment:

- initialStateDistribution.txt, which contains parameter values for the initial state distribution $\pi_i = P(S_1 = i)$,
- transitionMatrix.txt, which contains the transition matrix $a_{ij} = P(S_{t+1} = j | S_t = i)$,
- emissionsMatrix.txt, which contains the emission matrix $b_{ik} = P(O_t = k | S_t = i)$, and
- observations.txt, which contains a long bit sequence of $T = 308000$ observations.

Use the Viterbi algorithm to compute the most probable sequence of hidden states conditioned on this particular sequence of observations. As always, you may program in the language of your choice. Make sure your submission to Gradescope includes the following:

(a) a PDF of your source code

(b) a plot of the most likely sequence of hidden states versus time.

To check your answer: suppose that the hidden states $\{1, 2, \ldots, 26\}$ represent the letters $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$ of the English alphabet, and suppose that hidden state 27 encodes a space between words. If you have implemented the Viterbi algorithm correctly, the most probable sequence of hidden states *(ignoring repeated elements)* will reveal a highly recognizable message, as well as an interesting commentary on our times.

6.2 Conditional independence

Consider the hidden Markov model (HMM) shown below, with hidden states S_t and observations O_t for times $t \in \{1, 2, \ldots, T\}$. Indicate whether the following statements are true or false.

__________ $P(S_t|S_{t-1}) = P(S_t|S_{t-1}, O_t)$

__________ $P(S_t|S_{t-1}) = P(S_t|S_{t-1}, O_{t-1})$

__________ $P(S_t|S_{t-1}) = P(S_t|S_{t-1}, S_{t+1})$

__________ $P(S_t|O_{t-1}) = P(S_t|O_1, O_2, \ldots, O_{t-1})$

__________ $P(O_t|S_{t-1}) = P(O_t|S_{t-1}, O_{t-1})$

__________ $P(O_t|O_{t-1}) = P(O_t|O_1, O_2, \ldots, O_{t-1})$

__________ $P(O_1, O_2, \ldots, O_T) = \prod_{t=1}^{T} P(O_t|O_1, \ldots, O_{t-1})$

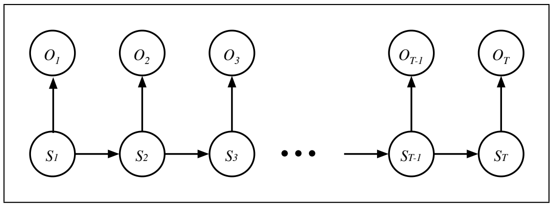
__________ $P(S_2, S_3, \ldots, S_T|S_1) = \prod_{t=2}^{T} P(S_t|S_{t-1})$

__________ $P(S_1, S_2, \ldots, S_{T-1}|S_T) = \prod_{t=1}^{T-1} P(S_t|S_{t+1})$

__________ $P(S_1, S_2, \ldots, S_T|O_1, O_2, \ldots, O_T) = \prod_{t=1}^{T} P(S_t|O_t)$

__________ $P(S_1, S_2, \ldots, S_T, O_1, O_2, \ldots, O_T) = \prod_{t=1}^{T} P(S_t, O_t)$

__________ $P(O_1, O_2, \ldots, O_T|S_1, S_2, \ldots, S_T) = \prod_{t=1}^{T} P(O_t|S_t)$



6.3 More conditional independence

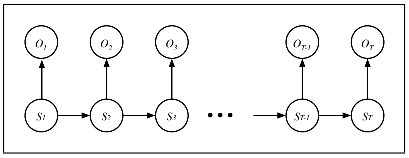
Indicate the **smallest** subset of evidence nodes that must be considered to compute each conditional probability shown below. The first two problems are done as examples. (You may assume everywhere that $2 < t < T - 1$, i.e., do not worry about special boundary cases.)

(a) (Optional)

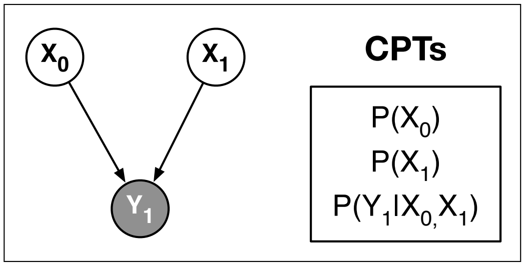
$$
\begin{aligned}
P(S_t|S_1, S_2, \ldots, S_{t-1}) &= P(S_t|S_{t-1}) \\
P(O_t|S_1, S_2, \ldots, S_T) &= P(O_t|S_t) \\
P(S_t|S_{t+1}, S_{t+2}, \ldots, S_T) &= \underline{\hspace{6cm}} \\
P(S_t|O_t, O_{t-1}, O_{t+1}) &= \underline{\hspace{6cm}} \\
P(S_t|O_t, O_{t+1}, \ldots, O_T) &= \underline{\hspace{6cm}} \\
P(O_t|O_1, O_2, \ldots, O_{t-1}) &= \underline{\hspace{6cm}} \\
P(O_t|S_{t-2}, S_{t-1}, S_{t+1}, S_{t+2}) &= \underline{\hspace{6cm}} \\
P(O_t|O_{t-1}, O_{t+1}, S_1, S_T) &= \underline{\hspace{6cm}}
\end{aligned}
$$

(b) **(NOT optional – will be graded)**

$$
\begin{aligned}
P(S_t|O_t, O_{t-1}, O_{t+1}, S_{t-1}, S_{t+1}) &= \underline{\hspace{6cm}} \\
P(S_t|S_1, S_T, O_1, O_t, O_T) &= \underline{\hspace{6cm}} \\
P(O_t|O_1, O_2, \ldots, O_{t-1}, S_{t-1}) &= \underline{\hspace{6cm}} \\
P(O_t|O_1, O_2, \ldots, O_{t-1}, S_{t-2}) &= \underline{\hspace{6cm}}
\end{aligned}
$$



6.4 Belief updating



Consider the simple belief network on the right with nodes X_0, X_1, and Y_1. To compute the posterior probability $P(X_1|Y_1)$, we can use Bayes rule:

$$P(X_1|Y_1) \;=\; \frac{P(Y_1|X_1)\,P(X_1)}{P(Y_1)}$$

(a) Show how to compute the term $P(Y_1|X_1)$ in the numerator of Bayes rule.

(b) Show how to compute the term $P(Y_1)$ in the denominator of Bayes rule.

Now consider the belief network shown at the bottom of the page. It does not have the same structure as an HMM, but using similar ideas we can derive efficient algorithms for inference. In particular, consider how to compute the posterior probability $P(X_t|Y_1, Y_2, \ldots, Y_t)$ that accounts for evidence up to and including time t. We can derive an efficient recursion from Bayes rule:

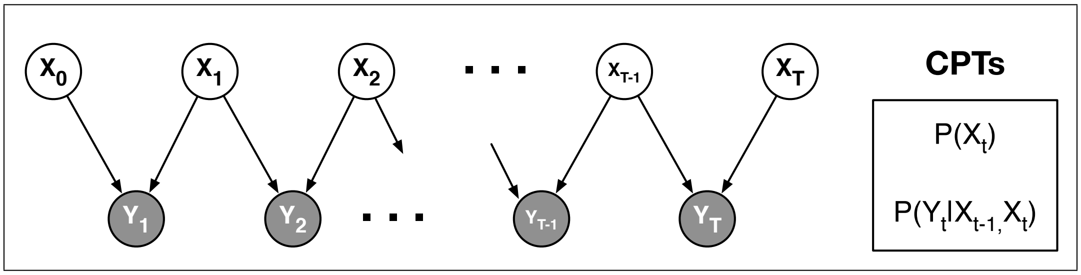
$$P(X_t|Y_1, Y_2, \ldots, Y_t) \;=\; \frac{P(Y_t|X_t, Y_1, Y_2, \ldots, Y_{t-1})\,P(X_t|Y_1, Y_2, \ldots, Y_{t-1})}{P(Y_t|Y_1, \ldots, Y_{t-1})}$$

where the nodes $Y_1, Y_2, \ldots, Y_{t-1}$ are treated as background evidence. In parts (c-e) of this problem you will compute the individual terms that appear in this version of Bayes rule. You should express your answers in terms of the CPTs of the belief network and the probabilities $P(X_{t-1}\!=\!x|Y_1, Y_2, \ldots, Y_{t-1})$, *which you may assume have been computed at a previous step of the recursion.* Your answers to parts (a) and (b) may be instructive for parts (d) and (e).

(c) Show how to simplify the term $P(X_t|Y_1, Y_2, \ldots, Y_{t-1})$ in the numerator of Bayes rule.

(d) Show how to compute the term $P(Y_t|X_t, Y_1, Y_2, \ldots, Y_{t-1})$ in the numerator of Bayes rule.

(e) Show how to compute the term $P(Y_t|Y_1, Y_2, \ldots, Y_{t-1})$ in the denominator of Bayes rule.



6.5 Most likely hidden states (Optional)

The Viterbi algorithm in HMMs computes the most likely *sequence* of hidden states for a particular sequence of observations:

$$\{s_1^*, s_2^*, \ldots, s_T^*\} \;=\; \operatorname*{argmax}_{\{s_1,s_2,\ldots,s_T\}} \left[P(s_1, s_2, \ldots, s_T|o_1, o_2, \ldots, o_T)\right]$$

Consider how these *collectively* optimal hidden states s_t^* differ (if at all) from the *individually* optimal hidden states $\hat{s}_t$:

$$\hat{s}_t \;=\; \operatorname*{argmax}_{i} \left[P(S_t\!=\!i|o_1, o_2, \ldots, o_T)\right].$$

Answer the following `[yes/no]` questions:

(a) Is it possible that $P(\hat{s}_1, \ldots, \hat{s}_T|o_1, \ldots, o_T) > P(s_1^*, \ldots, s_T^*|o_1, \ldots, o_T)$?

(b) Is it possible that $\hat{s}_t = s_t^*$ for all t?

(c) Is it necessarily true that $\hat{s}_t = s_t^*$ for all t?

(d) Is it necessarily true that $P(\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_T) > 0$?