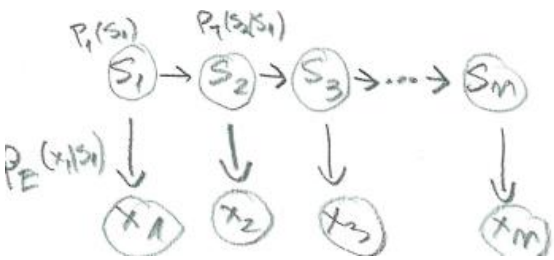
Alin Tomescu, http://people.csail.mit.edu/~alinush
6.867 Machine learning | Prof. Tommi Jaakkola | Week 12, Tuesday, November 22nd, 2013| Lecture 22

Lecture 22: More Hidden Markov Models

We can view HMMs as Bayesian network.



We showed how this graphical structure implies independency and how they are easier to learn.

$$s_1 \perp s_3 \mid s_2 \; (true)$$

$$x_1 \perp x_3 \mid s_2 \; (true)$$

$$x_1 \perp x_3 \mid x_2 \; (false)$$

What does $x_1 \perp x_3 \mid x_2 \; (false)$ say about the observable variables? That there are many possible ways to couple them together. The sequence of the observable variables is NOT a Markov model. They are *more* dependent on each other than a Markov model.

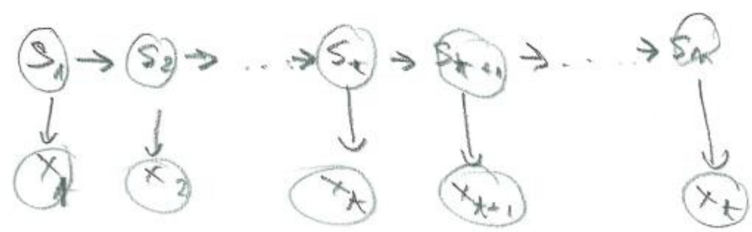


Figure 1: I have no clue where this figure was supposed to be inserted

Last time we looked over the tree problems we need to solve for an HMM.

(1) $P(x_1, \dots, x_n) = \sum_{s_1,\dots,s_n} P(x_1, \dots, x_n, s_1, \dots, s_n)$ and the Markov structure makes it easier to evaluate that joint
    a. Solved last time using forward and backward probabilities
(2) Learn $P_1(s_1), P_E(x|s), P_T(s'|s), x \in \mathcal{X} = \{1, \dots, k_E\}$
(3) Find the most likely underlying explanation for the observables in terms of states: $(\hat{s}_1, \dots, \hat{s}_n) = \underset{s_1,\dots,s_n}{\operatorname{argmax}} P(x_1, \dots, x_n, s_1, \dots, s_n)$.

Forward probabilities: $\alpha_t(s_t) = P(x_1, \dots, x_t, s_t)$

$$\alpha_1(s_1) = P_1(s_1)P_E(x_1|s_1), s_1 = 1, \dots, k$$

$$\alpha_t(s_t) = \sum_{s_{t-1}} \alpha_{t-1}(s_{t-1})P_T(s_t|s_{t-1})P_E(x_t|s_t), s_t = 1, \dots, k$$

Backward probabilities: $\beta_t(s_t) = P(x_{t+1}, \dots, x_n | s_t)$

$$\beta_n(s_n) = 1$$

$$\beta_t(s_t) = \sum_{s_{t+1}} P_T(s_{t+1}|s_t) P_E(x_{t+1}|s_{t+1}) \beta_{t+1}(s_{t+1}), s_t = 1, \dots, k$$

$$\sum_{s_n=1}^{k} \alpha_n(s_n) = P(x_1, \dots, x_n)$$

$$\sum_{s_t=1}^{k} \alpha_t(s_t)\beta_t(s_t) = P(x_1, \dots, x_n), \forall t = 1, \dots, n$$

Learning HMMs from data

Estimate $P_1(s_1), s_1 = 1, \dots, k$, $P_T(s'|s), s, s' = 1, \dots, k$, get a k^2 probability table, and $P_E(x|s), s = 1, \dots, k, x = 1, \dots, k_E$

Complete log likelihood (single input $x_1, \dots, x_n$ sequence):

$$\log P(x_1, \dots, x_n, s_1, \dots, s_n) = \log P_1(s_1) + \sum_{t=1}^{n} \log P_E(x_t|s_t) + \sum_{t=1}^{n} \log P_T(s_{t+1}|s_t)$$
$$= \sum_{s=1}^{k} n_1(s) \log P_1(S_1 = s) + \sum_{s',s} n_T(s, s') \log P_T\left(S_{next} = s' | S_{prev} = s\right)$$
$$+ \sum_{s,x} n_E(s, x) \log P_E(X = x | S = s)$$

$$n_1(s) = [[s = s_1]]$$

$$n_T(s, s') = \sum_{t=1}^{n-1} [[s = s_t]][[s' = s_{t+1}]]$$

$$n_E(s, x) = \sum_{t=1}^{n} [[s = s_t]][[x = x_t]]$$

ML estimates of the parameters given these counts:

$$\hat{P}_1(s) = \frac{n_1(s)}{\sum_{s'} n_1(s')}$$

$$\hat{P}_E(x|s) = \frac{n_E(s, x)}{\sum_{x'} n_E(s, x')}$$

$$\hat{P}_T(s'|s) = \frac{n_T(s, s')}{\sum_{s''} n_T(s, s'')}$$

What if we don't have complete data? We randomly initialize the model and compute:

$$n_1(s) \to \gamma_1(s) = P(s_1 = s|x_1, \dots, x_n) = \frac{\alpha_1(s)\beta_1(s)}{\sum_{s'} \alpha_1(s')\beta_1(s')}$$

$$n_E(s,x) \to \gamma_t(s) = P(s_t = s|x_1, \dots, x_n) = \frac{\alpha_t(s)\beta_t(s)}{\sum_{s'} \alpha_t(s')\beta_t(s')}$$

$$n_T(s,s') \to \xi_t(s,s') = P(s_t = s, s_{t+1} = s'|x_1, \dots, x_n) = \frac{\alpha_t(s)P_T(s'|s)P_E(x_{t+1}|s')\beta_{t+1}(s')}{\sum_{\tilde{s}} \alpha_t(\tilde{s})\beta_t(\tilde{s})}$$

EM algorithm (Forward-backward algorithm for estimating HMM)

Initialization: Initialize $P_1(s_1), P_T(s'|s), P_E(x|s)$ with a guess

E-step: Evaluate $\gamma_t(s), \xi(s,s'), \forall t = 1, \dots, n, \forall s, s' = 1, \dots, k$, for a single sequence

$$\tilde{n}_1(s) = \gamma_1(s) = P(s_1 = s|x_1, \dots, x_n)$$

$$\tilde{n}_T(s,s') = \sum_{t=1}^{n-1} \xi_t(s,s') = \sum_{t=1}^{n-1} P(s_t = s, s_{t+1} = s'|x_1, \dots, x_n)$$

$$\tilde{n}_E(s,x) = \sum_{t=1}^{n} \gamma_t(s)[[x = x_t]] = \sum_{t=1}^{n} P(s_t = s|x_1, \dots, x_n)[[x = x_t]]$$

M-step: Exactly as before, except we use the $\tilde{n}$ counts:

$$\hat{P}_1(s) = \frac{\tilde{n}_1(s)}{\sum_{s'} \tilde{n}_1(s')}$$

$$\hat{P}_E(x|s) = \frac{\tilde{n}_E(s,x)}{\sum_{x'} \tilde{n}_E(s,x')}$$

$$\hat{P}_T(s'|s) = \frac{\tilde{n}_T(s,s')}{\sum_{s''} \tilde{n}_T(s,s'')}$$

Example:

$$P_1(s_1) = \begin{cases} 1, s_1 = 1 \\ 0, s_1 = 2 \end{cases}$$

$$P_T(s'|s) = \begin{bmatrix} & s' = 1 & s' = 2 \\ s = 1 & .9 & .1 \\ s = 2 & 0 & 1 \end{bmatrix}$$

$$P_E(x|s) = \begin{bmatrix} & x = A & x = B \\ s = 1 & .5 & .5 \\ s = 2 & .1 & .9 \end{bmatrix}$$

Find the states:

$$(\hat{s}_1, \hat{s}_2) = \underset{s_1, s_2}{\operatorname{argmax}} P(x_1 = B, x_2 = B, s_1, s_2)$$

Consider the following probabilities:

$$\begin{aligned}(s_1 = 1, s_2 = 1) &\to P(x_1 = B, x_2 = B, s_1 = 1, s_2 = 1)\\ &= P(s_1 = 1)P_E(x_1 = B|s_1 = 1)P_T(s_2 = 1|s_1 = 1)P(x_2 = B|s_2 = 1) = 1 \cdot 0.5 \cdot 0.9 \cdot 0.5\end{aligned}$$

Similarly, we get:

$$(s_1 = 1, s_2 = 2) \to 1 \cdot 0.5 \cdot 0.1 \cdot 0.9$$

$$(s_1 = 2, s_2 = 1) \to 0\ prob$$

$$(s_1 = 2, s_2 = 2) \to 0\ prob$$

So, the higher likelihood answer is $(s_1 = 1, s_2 = 1)$

If I observed $B, B, B, B, \ldots, B$ the estimated ML sequence would have been 1,2,2,2,2, … ,2.

Viterbi algorithm

We can estimate the HMM model with the EM algorithm, but how can we find the most likely state sequence given some data? (Remember each state is in $\{1, \ldots, k\}$, so we have an exponential space of states to explore)

$$(\hat{s}_1, \ldots, \hat{s}_n) = \underset{s_1, \ldots, s_n \in \{1, \ldots, k\}}{\operatorname{argmax}} P(x_1, \ldots, x_n, s_1, \ldots, s_n)$$

We can use something very similar to the **forward probabilities**, except that instead of summing over all possible previous states we take the *maximum* instead. Let,

$$\delta_n(s_n) = \max_{s_1, \ldots, s_{n-1} \in \{1, \ldots, k\}} P(x_1, \ldots, x_n, s_1, \ldots, s_n)$$

If I have $\delta_n(s_n)$ how would I determine the ML for s_n?

$$\hat{s}_n = \underset{s_n = 1, \ldots, k}{\operatorname{argmax}} \delta_n(s_n)$$

…because $\max_{s_n} \delta_n(s_n) = \max_{s_1, \ldots, s_n} P(x_1, \ldots, x_n, s_1, \ldots, s_n)$

$$\delta_1(s_1) = P_1(s_1)P_E(x_1|s_1) = P(x_1, s_1)$$

$$\delta_2(s_2) = \max_{s_1 = 1, \ldots, k} P(x_1, x_2, s_1, s_2) = \max_{s_1 = 1, \ldots, k} P_1(s_1)P_E(x_1|s_1)P_T(s_2|s_1)P_E(x_2|s_2) = \max_{s_1 = 1, \ldots, k} \delta_1(s_1)P_T(s_2|s_1)P_E(x_2|s_2)$$

$$\begin{aligned}\delta_3(s_3) &= \max_{s_1, s_2 \in \{1, \ldots, k\}} P(x_1, x_2, x_3, s_1, s_2, s_3) = \max_{s_1, s_2 \in \{1, \ldots, k\}} P(x_1, x_2, s_1, s_2)\, P_T(s_3|s_2)P_E(x_3|s_3)\\ &= \max_{s_2 = 1, \ldots, k} \left(\max_{s_1 = 1, \ldots, k} P(x_1, x_2, s_1, s_2) \right) P_T(s_3|s_2)P_E(x_3|s_3) = \max_{s_2 = 1, \ldots, k} \delta_2(s_2)P_T(s_3|s_2)P_E(x_3|s_3)\end{aligned}$$

In general, we can prove that:

$$\delta_t(s_t) = \max_{s_{t-1}=1,\ldots,k} \delta_{t-1}(s_{t-1})\, P_T(s_t|s_{t-1}) P_E(x_t|s_t), \forall s_t = 1, \ldots, k$$

Backtracking iteration:

We can compute the $n \times k$ $\delta_i(j)$ table for all $i \in \{1, \ldots, n\}$ and for all $j \in \{1, \ldots, k\}$ in the following order:

$$\delta_1(1), \delta_1(2), \ldots, \delta_1(k); \delta_2(1), \ldots, \delta_2(n); \ldots; \delta_n(1), \ldots, \delta_n(k)$$

Then we can find the maximum sequence of states $(\hat{s}_1, \ldots, \hat{s}_n)$ by doing:

$$\hat{s}_n = \operatorname*{argmax}_{s_n} \delta_n(s_n)$$

$$\begin{aligned}\hat{s}_{n-1} = \left(\text{the } s_{n-1} \text{ that maximized } \delta_n(\hat{s}_n)\right) &= \operatorname*{argmax}_{s_{n-1}} \delta_{n-1}(s_{n-1})\, P_T(\hat{s}_n|s_{n-1}) P_E(x_n|\hat{s}_n) \\ &= \operatorname*{argmax}_{s_{n-1}} \delta_{n-1}(s_{n-1})\, P_T(\hat{s}_n|s_{n-1})\end{aligned}$$

$$\hat{s}_{n-2} = \operatorname*{argmax}_{s_{n-2}} \delta_{n-2}(s_{n-2})\, P_T(\hat{s}_{n-1}|s_{n-2})$$

$$\vdots$$

$$\hat{s}_1 = \operatorname*{argmax}_{s_1} \delta_1(s_1)\, P_T(\hat{s}_2|s_1)$$