Lecture 21: Hidden Markov Models

Final exam: Evening of December 10th, location and time to be announced.
- Hidden Markov models are sure to be on the final exam, because it is so easy to use them as a test of how well you understand generative modelling

Bayesian networks are graphical models that characterize how variables are independent of each other.

\[
P(s, x, y) = P(s)P(x|s)P(y|s)
\]

- \( s \) is a parent of \( x \)
- \( x \) is a child of \( s \)

**Hidden Markov models**
A particular type of Bayesian network. The graph gives us “parsimony of description” (a compact way of describing it). It also gives us efficiency of computation.

Notation change: The latent variables we don’t know about are denoted with the letter \( s \), which stands for “state.”

States are coupled with observations. I know something about each state.

By contrast, a simple mixture model looks like this:

6.867 Machine learning | Prof. Tommi Jaakkola | Week 12, Tuesday, November 19th, 2013 | Lecture 21

Example: $x_i$ can be a word and all the observations would constitute a sentence, such as:

“This course is {terrible, great} = $x_1, x_2, x_3, x_4$

You would like to give a part of speech tag for each of these words, as follows:

$s_1 = \text{det}, s_2 = \text{noun}, s_3 = \text{verb}, s_4 = \text{adjective}$

How can we write down the distribution for this graphical model, for this Bayesian network?

$P(x_1, ..., x_n, s_1, ..., s_n) = ?$

What independence properties are satisfied?

1. $x_1, ..., x_n$ are conditionally independent given $s_1, ..., s_n$

$$P(x_1, ..., x_n, s_1, ..., s_n) = P(x_1, ..., x_n | s_1, ..., s_n)P(s_1, ..., s_n) = \prod_{i=1}^{n} P(x_i | s_i)P(s_1, ..., s_n)$$

2. $s_1, s_2, ..., s_{i-2}$ and $s_i$ are conditionally independent given $s_{i-1}$

$$s_i \perp s_{i-2}, ..., s_1 | s_{i-1} \iff P(s_i, s_{i-2}, ..., s_1 | s_{i-1}) = P(s_i, s_{i-2}, ..., s_{i-1} | s_{i-1})P(s_i | s_{i-1})$$

$$P(x_1, ..., x_n, s_1, ..., s_n) = \prod_{i=1}^{n} P(x_i | s_i)P(s_1, ..., s_n)$$

$= \prod_{i=1}^{n} P(x_i | s_i)P(s_n | s_{n-1}, s_{n-2}, ..., s_1)P(s_{n-1}, s_{n-2}, ..., s_1) = ...$

$= \prod_{i=1}^{n} P(x_i | s_i)P(s_1 | s_2)P(s_2 | s_1)P(s_3 | s_2, s_1)P(s_n | s_{n-1}, ..., s_1)$

$= \prod_{i=1}^{n} P(x_i | s_i)P(s_1 | s_2)P(s_2 | s_1)P(s_n | s_{n-1})$

3. $x_i \perp$ all the other $x_i'$s and all the other $s_i'$s | $s_i$

$$P(x_1, ..., x_n, s_1, ..., s_n) = \prod_{i=1}^{n} P_{x_i}(x_i | s_i) \prod_{i=2}^{n} P_i(s_i | s_{i-1})$$

4. We will make an additional assumption here not shown in the graph: HMM is homogenous (the probabilities $P(z_i = z | z_{i-1} = z')$ do not depend on the position $i$ along the sequence)

$$P(x_1, ..., x_n, s_1, ..., s_n) = \prod_{i=1}^{n} P_{x_i}(x_i | s_i) \prod_{i=2}^{n} P_i(s_i | s_{i-1})$$

What do we need to specify an HMM?
What are the states? $s \in \{1, \ldots, k\}$

What are the outputs? $x \in \mathcal{X} = \{\mathbb{R}^d\}$

We need to specify the initial state distribution $P_1(S_1)$.

We need to specify emission output probabilities: $P_E(x|s)$, which is a table of probabilities, or it could be a Gaussian distribution with a mean that depends on the state $N(x; \mu_s, \sigma^2 I)$.

We need to model the transition probabilities: $P_T(s'|s)$

Example:

\[
P_1(s_1): \begin{bmatrix} 1 \\ 0 \end{bmatrix} s_1 = 1 \\
\begin{bmatrix} 0 \\ 1 \end{bmatrix} s_2 = 2
\]

\[
P_T(s_t|s_{t-1})
\begin{bmatrix} s_t = 1 \\ s_t = 2 \end{bmatrix}
\begin{bmatrix} s_{t-1} = 1 \\ s_{t-1} = 2 \end{bmatrix}
\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}
\]

$P_E(x|s) = N(x; \mu_s, \sigma^2 I), \mu_1 > \mu_2$

What does this model generate? What is a likely sequence of states?

$s_1, s_2, s_3, \ldots = 1, 2, 2, 2, \ldots$

In terms of observations, at time 1 I am always in state 1 and at time 2 or greater I am always going to be and remain in state 2.
How to use these HMM models?

We need to be able to solve a few problems: How likely is an observation sequence in this model, after specifying it. We need to evaluate:

\[
P(x_1, \ldots, x_n) = \sum_{\text{all } k^n \text{ possible } s_1, \ldots, s_n} P(x_1, \ldots, x_n, s_1, \ldots, s_n)
\]

We need to be able to estimate \( P_1(s_1), P_E(x|s), P_T(s'|s) \) from data \( \{x_1^{(1)}, \ldots, x_n^{(1)}, \ldots, x_1^{(T)}, \ldots, x_n^{(T)}\} \)

We need to estimate the prediction \( (\hat{s}_1, \ldots, \hat{s}_n) = \arg\max_{s_1, \ldots, s_n} P(x_1, \ldots, x_n, s_1, \ldots, s_n) \) for a particular data row of \( x_i \)'s in the above data matrix.

But how can we sum over \( k^n \) possible terms? We can perform the summation in time linear to the length of the sequence due to the independence relations.

The forward-backward algorithm

Gives us \( P(x_1, \ldots, x_n) \) in linear time.

Forward probabilities: Predictive probabilities. For a particular sequence \( x_1, \ldots, x_n \), with \( s_t \in \{1, \ldots, k\} \), we want to predict \( \alpha_t(i) = P(x_1, \ldots, x_t, s_t = i) \). Then we can predict \( P(s_t = i|x_1, \ldots, x_t) = \frac{\alpha_t(i)}{\sum_j \alpha_t(j)} \)

\[
\alpha_1(s_1) = P_1(s_1)P_E(x_1|s_1) = P(x_1, s_1)
\]

\[
\sum_{s_1} \alpha_1(s_1) = P(x_1)
\]
\[ \alpha_2(s_2) = \sum_{s_1} P(x_1, x_2, s_1, s_2) = \sum_{s_1} \left( P_1(s_1) P_E(x_1|s_1) P_T(s_2|s_1) P_E(x_2|s_2) \right) = \sum_{s_1} \alpha_2(s_1) P_T(s_2|s_1) P_E(x_2|s_2) \]

\[ \alpha_3(s_3) = \sum_{s_1, s_2} P(x_1, x_2, x_3, s_1, s_2, s_3) = \sum_{s_2} \left( \sum_{s_1} P(x_1, x_2, s_1, s_2) \right) P_T(s_3|s_2) P_E(x_3|s_3) = \sum_{s_2} \alpha_2(s_2) P_T(s_3|s_2) P_E(x_3|s_3) \]

In general, we get:

\[ \alpha_t(s_t) = P(x_1, x_2, ..., x_t, s_t) = \sum_{s_1, s_2, ..., s_{t-1}} P(x_1, x_2, ..., x_t, s_1, s_2, ..., s_t) = \sum_{s_{t-1}} \alpha_{t-1}(s_{t-1}) P_T(s_t|s_{t-1}) P_E(x_t|s_t), \]

\[ \forall s_t = 1, ..., k \]

\[ \sum_{s_t} \alpha_t(s_t) = P(x_1, x_2, ..., x_t) \]

For \( \alpha_1(s_1) \), we have \( k \) possible values, corresponding to each \( s_1 \in \{1, ..., k\} \).

What is the computational cost of evaluating \( P(x_1, x_2, ..., x_n) \)? \( O(nk^2) \), because I have \( k \) numbers to fill in for \( \alpha_t \) and each one involves summing over the \( k \) previous \( \alpha_{t-1} \) values. Note that \( t \in \{1, ..., n\} \) hence the \( O(nk^2) \).

**Note:** Increasing the number of values \( k \) for the hidden states in an HMM has much greater effect on the computational cost of \( O(nk^2) \) forward-backward algorithm than increasing the length \( n \) of the observation sequence.

**Backward probabilities:** The complement of forward probabilities. *Diagnostic* probabilities.
\[ \beta_t(s_t) = P(x_{t+1}, ..., x_n | s_t) \]

If I start from that state, then what is the probabilities of generating all the future observations?

\[ \beta_n(s_n) = 1 \]

\[ B_{n-1}(s_{n-1}) = P(x_n | s_{n-1}) = \sum_{s_n} P_T(s_n | s_{n-1}) P_E(x_n | s_n) \]

\[ B_{n-2}(s_{n-2}) = P(x_{n-1}, x_n | s_{n-2}) = \sum_{s_{n-1}} P_T(s_{n-1} | s_{n-2}) P_E(x_{n-1} | s_{n-1}) P_T(s_n | s_{n-1}) P_E(x_n | s_n) \]

\[ = \sum_{s_{n-1}} \left( \sum_{s_n} P_T(s_n | s_{n-1}) P_E(x_n | s_n) \right) P_T(s_{n-1} | s_{n-2}) P_E(x_{n-1} | s_{n-1}) \]

\[ = \sum_{s_{n-1}} B_{n-1}(s_{n-1}) P_T(s_{n-1} | s_{n-2}) P_E(x_{n-1} | s_{n-1}) \]

\[ \beta_t(s_t) = \sum_{s_{t+1}} P_T(s_{t+1} | s_t) P_E(x_{t+1} | s_{t+1}) \beta_{t+1}(s_{t+1}) \]

How to evaluate the **posterior probability of a particular state**:

\[ P(s_t = s | x_1, ..., x_n) = \frac{P(x_1, ..., x_n, s_t = s)}{P(x_1, ..., x_n)} = \frac{P(x_1, ..., x_t, s_t = s) P(x_{t+1}, ..., x_n | s_t = s)}{P(x_1, ..., x_n)} = \frac{\alpha_t(s) \beta_t(s)}{\sum_{s} \alpha_t(s) \beta_t(s)} \]

How to evaluate the **probability of the data set**:

\[ P(x_1, x_2, ..., x_n) = \sum_{s_n} \alpha_n(s_n) \]

\[ P(x_1, x_2, ..., x_n) = \sum_{s_1} P(s_1) P(x_1 | s_1) \beta_1(s_1) \]

\[ P(x_1, x_2, ..., x_n) = \sum_{s_t} \alpha_t(s_t) \beta_t(s_t) \]

How to evaluate the posterior probability that the HMM went \( s \to s' \) at time \( t \).

\[ P(s_t = s, s_{t+1} = s' | x_1, ..., x_n) = \frac{\alpha_t(s) P_T(s' | s) P_E(x_{t+1} | s') \beta_{t+1}(s')}{\sum_{s} \alpha_t(s) \beta_t(s)} \]