HMMs

Introduction to Machine Learning (CSCI 1950-F), Summer 2011

In these exercises, we use the Viterbi algorithm on a hidden Markov model (HMM), applied to synthetic data. These instructions (this file) can be downloaded from the course website along with code and data for use in the exercises below.

We will use a synthetic data set inspired by a problem in computational biology. (I'm not an expert in this area, but this is my best attempt to explain the rough idea.) In this problem, a sequence of measurements is made, one for each block of B nucleotides in a strand of DNA or RNA. These measurements are expected to be roughly normally distributed with a very slowly changing mean. However, it can happen that a short strand of nucleotides from a different part of the genome can be inserted into the original strand, and since it has a different mean, the sequence of measurements would be expected to shift to the new mean for a short period of time and then return to the old mean. The task is to find where these insertions have occurred. We will grossly oversimplify the problem by supposing that the measurements are one-dimensional with two possible means, and modeling this as a hidden Markov model with known transition matrix and emission probabilities.

The data set consists of a sequence of real-valued observations $x_1, \ldots, x_n$ generated by the following HMM. The hidden variables $z_1, \ldots, z_n$ can each take one of two possible states, 1 or 2. The transition matrix is

$$T = \begin{pmatrix} 1 - p_{switch} & p_{switch} \\ p_{switch} & 1 - p_{switch} \end{pmatrix},$$

where p_{switch} is the probability of switching from one state to the other. We assume that p_{switch} is known. (Note: When p_{switch} is small, the chain tends to stay in the same state.) The emission probability distributions are Gaussian:

$$p(x_t|z_t) = \mathcal{N}(x_t|\mu_{z_t}, \sigma^2_{z_t}),$$

where the parameters $\mu_1, \mu_2 \in \mathbb{R}$ and $\sigma_1^2, \sigma_2^2 > 0$ are known. The initial distribution is uniform:

$$\pi(Z_1 = 1) = \pi(Z_1 = 2) = 1/2.$$

(1) Generate the data set

The script `main.m` defines the distributions to be used in the HMM, using the following parameters: $p_{switch} = 0.01$, $\mu_1 = -2$, $\mu_2 = 2$, $\sigma_1 = 6$, $\sigma_2 = 6$. Next, the script is intended to randomly sample a sequence $z_1, \ldots, z_n$ of states and corresponding observations $x_1, \ldots, x_n$, however, this section of the code is missing. Your task is to fill in this missing section. Use $n = 1000$. Print out and submit the code you wrote.

(2) Visualize your sample

Run the script to generate the data set (using the code you wrote in (1)) and plot the observations $x_1, \ldots, x_n$. Print out and submit this plot.

Without plotting the sequence of hidden states $z_1, \ldots, z_n$, using your knowledge of how the data was generated, try to guess the sequence of hidden states by visually inspecting the plot. You don't need to submit this, but just for fun, write down your best estimate of the sequence of hidden states.

(3) Run the Viterbi algorithm

Now, enable the section with the Viterbi algorithm by setting `enabled = 1`. Run the script again, to run the Viterbi algorithm on your data set, and plot the Viterbi path (the sequence $z_1^*, \ldots, z_n^*$ maximizing $p(z^*|x)$) on top of the true hidden states $z_1, \ldots, z_n$. Print out and submit the resulting plot, along with your answers to the following questions.

Roughly speaking, how well did Viterbi do compared to your visual estimate?

Now, set `reset_random = 0` at the beginning of the script, and run the script several more times to visualize the results for different randomly generated data sets. Usually, in this problem the Viterbi algorithm will get several short sequences incorrect in any given data set. How would you characterize these sequences of points that Viterbi gets wrong?

(4) Complexity of the Viterbi algorithm

What is the computational complexity of the Viterbi algorithm, in terms of the length of the sequence n and the number of hidden states K? (Pretty cool, huh?)