Hidden Markov Models

Guy Lebanon

November 9, 2006

The Hidden Markov Model (HMM) is a discrete-time random process which means that it assigns probabilities to sequences $Y_1, Y_2 \ldots, Y_n$ of RVs of arbitrary lengths. In the case of HMM, $Y_i = (X_i, Z_i)$ where $X_i \in \{1, \ldots, b\}$ is observed and $Z_i\{1, \ldots, a\}$ is unobserved. The precise definition of the HMM model is

$$P(X_1, Z_1, \ldots, X_m, Z_m) = P(X_1) \prod_{i=2}^{m} P(Z_i|Z_{i-1}) \prod_{i=1}^{m} P(X_i|Z_i). \tag{1}$$

In other words, the $Z_1, \ldots, Z_m$ form a Markov chain and the X_i is conditionally independent of all other variables given Z_i. The parameters of the process are the emission probabilities $P(X_i = k|Z_i = l) = \kappa_{lk}$ and the transition probabilities $P(Z_i = k|Z_{i-1} = l) = \theta_{lk}$ and the initial probabilities $\nu_i = P(Z_1 = i)$. By defining the matrices $\Theta = [\theta_{ij}] \in \mathbb{R}^{b \times b}$, $K = [\kappa_{ij}] \in \mathbb{R}^{b \times a}$ and $\nu = [\nu_i] \in \mathbb{R}^b$ the model (1) is defined for sequences of any length m.

HMM is widely used in statistical genetics, speech recognition, digital communication etc. to model observed sequences $X_1, \ldots, X_m$ that are affected by an unobserved or latent markov chain $Z_1, \ldots, Z_m$. For example, in speech recognition, we observe the phonetic sound segments or phoneme sequences $X_1, \ldots, X_m$ which are "noisy" observations of the underlying spoken syllables $Z_1, \ldots, Z_m$. The unobserved spoken syllables are assumed to follow a Markov model and the phoneme sequence elements are the observed (or heard) emissions of the spoken syllables. Useful references for additional information concerning HMMs and their applications are [2, 1].

Since the sequence $Z_1, \ldots, Z_n$ is unobserved we often need to marginalize (1) over all possible missing sequences. Such summation grows exponentially with n and is not likely to produce efficient computations. The popularity of HMM stems from the fact that using dynamic programming much of the computations that would otherwise be unfeasible become simple and fast.

There are three main tasks associated with HMMs (i) Computing the probability of an observed sequence (ii) Predicting the most likely hidden sequence and (iii) estimating the model parameters by maximum likelihood for the observed data. The third problem is solved by EM as described in an earlier note. We outline below efficient solutions to problems (i) and (ii).

Computing the probability of an observed sequence

A naive computation of the probability of the observed sequence

$$P(x_1, \ldots, x_m) = \sum_{z_1, \ldots, z_m} P(z_1) \prod_{i=2}^{m} P(z_i|z_{i-1}) \prod_{i=1}^{m} P(x_i|x_i)$$

would be exponential in m. We show below an efficient dynamic programming polynomial time method to compute it. In a future note we will introduce the sum product algorithm which is a more general framework for such computations.

The dynamic programming method is based on the observation that

$$P(x_1, \ldots, x_m) = \sum_{i=1}^{b} \alpha_i(m+1) \quad \text{where} \quad \alpha_i(t) \stackrel{\text{def}}{=} P(X_1 = x_1, \ldots, X_{t-1} = x_{t-1}, Z_t = i)$$

and that the computation of $\alpha_i(t)$ depends only on $\{\alpha_j(t-1) : j = 1, \ldots, b\}$ using the recursive formula $\alpha_j(t+1) = \sum_{i=1}^{b} \alpha_i(t)\, \theta_{ij}\, \kappa_{i,x_t}$. The above recursion relation leads to a $O(b^2 t)$ time computation of $\alpha_i(t)$ for all i, t as well as $P(x_1, \ldots, x_m)$ using the above observation.

The quantities $\alpha_i(t)$ are called forward probabilities and their analogs $\beta_i(t) = P(x_t, \ldots, x_m | z_t = i)$ are called backward probabilities. The backward probabilities, computable by the recursive relation $\beta_i(t) = \sum_{j=1}^{b} \theta_{ij} \kappa_{i,x_t} \beta_j(t+1)$ also provide the required probability via $P(x_1, \ldots, x_m) = \sum_{i=1}^{b} \pi_i \beta_i(1)$.

The forward and backward probabilities may be combined to efficiently compute the following probability

$$\begin{aligned}
&P(X_1 = x_1, \ldots, X_m = x_m, Z_t = i) \\
=&P(X_1 = x_1, \ldots, X_{t-1} = x_{t-1}, Z_t = i)\, P(X_t = x_t, \ldots, X_m = x_m | X_1 = x_1, \ldots, X_{t-1} = x_{t-1}, Z_t = i) \\
=&P(X_1 = x_1, \ldots, X_{t-1} = x_{t-1}, Z_t = i)\, P(X_t = x_i, \ldots, X_m = x_m | Z_t = i) = \alpha_i(t)\beta_i(t).
\end{aligned}$$

Predicting the most likely hidden sequence

Given an observation sequence we are often interesting in predicting the most likely sequence of missing values that corresponds to it

$$(z_1^*, \ldots, z_m^*) = \arg\max_{z_1, \ldots, z_m} P(z_1, \ldots, z_m | x_1, \ldots, x_m) = \arg\max_{z_1, \ldots, z_m} P(x_1, z_1, \ldots, x_m, z_m).$$

For example, in speech recognition, this "optimal decoding" problem produces the most likely written syllable sequence that correspond to the sequence of sounds segments. As in the previous case, a naive decoding would need to go over an exponential number of possible sequences before choosing the optimal one.

Andrew Viterbi invented a dynamic programming solution to the above problem which popularized the use of HMMs in digital communications. The algorithm is based on the quantity

$$\delta_j(t) = \max_{z_1, \ldots, z_{t-1}} P(Z_1 = z_1, \ldots, Z_{t-1} = z_{t-1}, X_1 = x_1, \ldots, X_{t-1} = x_{t-1}, Z_t = j)$$

and is described as follows.

We start by initializing $\delta_j(1) = \pi_j$ and proceed to compute the following quantities by iterating $t = 2, 3, \ldots, m$

$$\begin{aligned}
\delta_j(t+1) &= \max_{i=1,\ldots,b} \delta_i(t)\, \theta_{ij}\, \kappa_{i,x_t} \\
\psi_j(t+1) &= \arg\max_{i=1,\ldots,b} \delta_i(t)\, \theta_{ij}\, \kappa_{i,x_t}.
\end{aligned}$$

The optimal sequences $\hat{z}_1, \ldots, \hat{z}_m$ may be obtained by following the back pointers stored in $\psi_j(t)$: $\hat{z}_m = \arg\max_{i=1,\ldots,b} \delta_i(m)$, $\hat{z}_t = \psi_{\hat{z}_{t+1}}(t+1)$ for $t = m-1, \ldots, 1$. The probability itself is $P(\hat{z}_1, \ldots, \hat{z}_m) = \max_{i=1,\ldots,b} \delta_i(m)$.

References

[1] Christopher D. Manning and Hinrich Schutze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

[2] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.