CS6220-03 Data Mining Techniques – Fall 2016

Hidden Markov Models, Exponential Families, and the Forward-backward Algorithm

Jan-Willem van de Meent, 19 November 2016

1 Hidden Markov Models

A hidden Markov model (HMM) defines a joint probability distribution of a series of observations x_t and hidden states z_t for $t = 1, \ldots, T$. We use $x_{1:T}$ and $z_{1:T}$ to refer to the full sequence of observations and states respectively. In a HMM the prior on the state sequence is assumed to satisfy the Markov property, which is to say that the probability of each state z_t depends only on the previous state z_{t-1}. In the broadest definition of a HMM, the states z_t can be either discrete or continuous valued. Here we will discuss the more commonly considered case where z_t takes on a discrete values $z_t \in [K]$ where $[K] := \{1, \ldots, K\}$. We will use a matrix $A \in \mathbb{R}^{K \times K}$ and a vector $\pi \in \mathbb{R}^K$ to define the transition probabilities and the probability for the first state z_1 respectively

$$p(z_t = l \,|\, z_{t-1} = k, A) := A_{kl}, \tag{1}$$

$$p(z_1 = k \,|\, \pi) := \pi_k. \tag{2}$$

The rows A_k of the transition matrix and entries of π sum to one

$$\sum_l A_{kl} = 1 \qquad \forall k \in [K], \qquad\qquad \sum_k \pi_k = 1. \tag{3}$$

The prior probability for the state sequence $z_{1:T}$ can be expressed as a product over conditional probabilities

$$p(z_{1:T}|\pi, A) = p(z_1|\pi) \prod_{t=2}^{T} p(z_t|z_{t-1}, A). \tag{4}$$

The observations in an HMM are assumed to be independent conditioned on the sequence of states, which is to say

$$p(x_{1:T}|z_{1:T}, \eta) = \prod_{t=1}^{T} p(x_t|z_t, \eta). \tag{5}$$

Here $\eta = \eta_{1:K}$ is a set of for the density $f(x_t; \eta_k)$ associated with each state $k \in [K]$,

$$p(x_t \,|\, z_t = k, \eta) := f(x_t; \eta_k). \tag{6}$$

The observations x_t can be either discrete or continuous valued. We will here assume that $f(x_t; \eta_k)$ is an exponential family distribution, which we will describe in more detail below.

1.1 Expectation Maximization

We will use $\theta = \{\eta, A, \pi\}$ to refer to the parameters of the HMM. We would like to find the set of parameters that maximizes the complete data likelihood,

$$\theta^* = \operatorname*{argmax}_\theta p(x_{1:T} \,|\, \theta) = \operatorname*{argmax}_\theta \sum_{z_{1:T} \in [K]^T} p(x_{1:T}, z_{1:T} \,|\, \theta) \tag{7}$$

Expectation maximization (EM) methods define a lower bound on the log likelihood a variational distribution $q(z_{1:T})$

$$\begin{aligned} \mathcal{L}(q(z_{1:T}), \theta) &:= \mathbb{E}_{q(z_{1:T})} \left[\log \frac{p(x_{1:T}, z_{1:T} \,|\, \theta)}{q(z_{1:T})} \right], & (8) \\ &= \sum_{z_{1:T} \in [K]^T} q(z_{1:T}) \log \frac{p(x_{1:T}, z_{1:T} \,|\, \theta)}{q(z_{1:T})} \leq \log p(x_{1:T} \,|\, \theta). & (9) \end{aligned}$$

When the variational distribution on the state sequence is equal to the posterior, that is

$$q^*(z_{1:T}) := p(z_{1:T} \,|\, x_{1:T}, \theta), \tag{10}$$

the the lower bound is tight (i.e. the lower bound is equal to the log likelihood)

$$\begin{aligned} \mathcal{L}(q^*(z_{1:T}), \theta) &= \sum_{z_{1:T} \in [K]^T} p(z_{1:T} \,|\, x_{1:T}, \theta) \log \frac{p(x_{1:T}, z_{1:T} \,|\, \theta)}{p(z_{1:T} \,|\, x_{1:T}, \theta)}, & (11) \\ &= \sum_{z_{1:T} \in [K]^T} p(z_{1:T} \,|\, x_{1:T}, \theta) \log \frac{p(z_{1:T} \,|\, x_{1:T}, \theta) p(x_{1:T} \,|\, \theta)}{p(z_{1:T} \,|\, x_{1:T}, \theta)}, & (12) \\ &= \log p(x_{1:T} \,|\, \theta) \sum_{z_{1:T} \in [K]^T} p(z_{1:T} \,|\, x_{1:T}, \theta) = \log p(x_{1:T} \,|\, \theta). & (13) \end{aligned}$$

The EM algorithm iterates between two steps:

1. The expectation step: Optimize the lower bound with respect to $q(z_{1:T})$

$$q^i(z_{1:T}) = \operatorname*{argmax}_q \mathcal{L}(q(z_{1:T}), \theta^{i-1}) = p(z_{1:T} | x_{1:T}, \theta^{i-1}). \tag{14}$$

2. The maximization step: Optimize the lower bound with respect to θ

$$\theta^i = \operatorname*{argmax}_\theta \mathcal{L}(q^i(z_{1:T}), \theta). \tag{15}$$

At a first glance it is not obvious whether EM for hidden Markov models is computationally tractable. Calculation of the lower bound via direct summation over $z_{1:T}$ requires $O(K^T)$ computation. Fortunately it turns out that we can exploit the Markov property of the state sequence to perform this summation in $O(K^2 T)$ time.

To see how we can reduce the complexity of the estimation problem we will introduce some new notation. We will define to $z_{t,k} := I[z_t = k]$ to be the one-hot vector representation of the state at time t, which is sometimes also know as an indicator vector. Given this representation we can re-express the probabilities for states and observations as

$$p(x_t \mid z_t, \eta) = \prod_k f(x_t; \eta_k)^{z_{t,k}}, \tag{16}$$

$$p(z_t|z_{t-1}, A) = \prod_{k,l} A_{kl}^{z_{t-1,k} z_{t,l}}, \tag{17}$$

$$p(z_1|\pi) = \prod_k \pi_k^{z_{1,k}}. \tag{18}$$

Using this representation, we can now express the lower bound as

$$\mathcal{L}(q(z_{1:T}), \theta) = \mathbb{E}_{q(z_{1:T})} \left[\log p(x_{1:T}|z_{1:T}, \theta) + \log p(z_{1:T}|\theta)\right] - \mathbb{E}_{q(z_{1:T})} \left[\log q(z_{1:T})\right] \tag{19}$$

$$= \mathbb{E}_{q(z_{1:T})} \Big[\sum_{t=1}^{T} \sum_{k=1}^{K} z_{t,k} \log f(x_t; \eta_k) + \sum_k z_{1,k} \log \pi_k \tag{20}$$

$$+ \sum_{t=2}^{T} \sum_{k=1,l=1}^{K} z_{t-1,k} z_{t,l} \log A_{kl} \Big] \tag{21}$$

$$- \mathbb{E}_{q(z_{1:T})} \Big[\log q(z_{1:T}) \Big] \tag{22}$$

In this expression we see a number of terms that are a product of a term that depends on z and a term that does not. If we pull all terms that are independent of z out of the expectation, then we obtain

$$\mathcal{L}(q(z_{1:T}), \theta) = \sum_{t=1}^{T} \sum_{k=1}^{K} \mathbb{E}_{q(z_{1:T})}[z_{t,k}] \log f(x_t; \eta_k) + \sum_k \mathbb{E}_{q(z_{1:T})}[z_{1,k}] \log \pi_k \tag{23}$$

$$+ \sum_{t=2}^{T} \sum_{k=1,l=1}^{K} \mathbb{E}_{q(z_{1:T})}[z_{t-1,k} z_{t,l}] \log A_{kl} - \mathbb{E}_{q(z_{1:T})} \Big[\log q(z_{1:T}) \Big] \tag{24}$$

We can now derive the maximization step of the EM algorithm by differentiating the lower bound and solving for zero. For example, for the parameters η_k we need to solve the identity

$$\nabla_{\eta_k} \mathcal{L}(q(z_{1:T}), \theta) = \sum_{t=1}^{T} \mathbb{E}_{q(z_{1:T})}[z_{t,k}] \frac{\nabla_{\eta_k} f(x_t; \eta_k)}{f(x_t; \eta_k)} = 0 \tag{25}$$

The first thing to observe here is that the only dependence on z in this identity is through the terms $\mathbb{E}_{q(z_{1:T})}[z_{t,k}]$. Similarly, the gradients of $\mathcal{L}$ with respect to π and A will depend on $\mathbb{E}_{q(z_{1:T})}[z_{1,k}]$ and $\mathbb{E}_{q(z_{1:T})}[z_{t-1,k} z_{t,l}]$ respectively. These terms do not depend directly on the parameters θ, so we can treat them as constants during the maximization step. We will from now on use the following substitutions

$$\gamma_{tk} := \mathbb{E}_{q(z_{1:T})}[z_{t,k}], \tag{26}$$

$$\xi_{tkl} := \mathbb{E}_{q(z_{1:T})}[z_{t,k} z_{t+1,l}]. \tag{27}$$

If we can come up with a way of calculating γ_{tk} and ξ_{tkl} in a manner that avoids direct summation over all sequences $z_{1:T}$, then we can perform EM on hidden Markov models. It turns out that such a procedure does in fact exists and we will describe it in the section on the forward-backward algorithm below.

2 Maximization of Exponential Family distributions

Before we turn to the computation of γ and ξ, let us look at how to perform the maximization step once γ and ξ are known. We will start with maximization with respect to the parameters η_k. In general, the objective in equation 25 could be minimized using any number of gradient-based minimization methods (e.g. LBFGS). However, we can do better. We can obtain a closed-form solution when the observation density $f(x_t; \eta_k)$ belongs to a so called exponential family. Exponential family distributions can be either discrete or continuous valued. In both cases, the mass or density function must be expressible in the following form

$$f(x;\eta) = h(x)\exp[\eta^\top T(x) - A(\eta)]. \tag{28}$$

In this expression, $h(x)$ is referred to as the base measure, η is a vector of parameters, which are often referred to as the natural parameters, $T(x)$ is a vector of sufficient statistics and $A(\eta)$ is known as a log normalizer. It turns out that many distributions that we use most commonly can be cast into this form, including the univariate and multivariate normal, the discrete and multinomial, the Poisson, the gamma, and the Dirichlet.

As an example, the univariate normal distribution Normal$(x; \mu, \sigma)$ can be cast in exponential family form by defining

$$h(x) = \frac{1}{\sqrt{2\pi}}, \tag{29}$$

$$\eta = \left(\frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2}\right), \tag{30}$$

$$T(x) = (x, x^2), \tag{31}$$

$$A(\eta) = -\frac{\eta_1^2}{4\eta_2} + \frac{1}{2}\log(-2\eta_2) = \mu^2/2\sigma^2 + \log\sigma. \tag{32}$$

Similarly, for a discrete distribution Discrete$(x; \mu)$ with support $d \in \{1, \ldots, D\}$ we can define

$$h(x) = 1, \tag{33}$$

$$\eta_d = \log\mu_d, \tag{34}$$

$$T_d(x) = I[x = d], \tag{35}$$

$$A(\eta) = \log\sum_{d=1}^{D}\exp\eta_d = \log\sum_{d=1}^{D}\mu_d. \tag{36}$$

Exponential family distributions have a number of properties that turn out to be very beneficial when performing expectation maximization. Because of its exponential form, the gradient gradient of an exponential family density takes on a convenient form

$$\nabla_\eta f(x;\eta) = [T(x) - \nabla_\eta A(\eta)]f(x;\eta). \tag{37}$$

If we subsitute this form back into equation 25 then we obtain

$$\nabla_{\eta_k} A(\eta_k) = \frac{\sum_{t=1}^{T} \gamma_{tk} T(x_t)}{\sum_{t=1}^{T} \gamma_{tk}}. \tag{38}$$

The term on the right has a clear interpretation: it is the empirical expectation of the sufficient statistics $T(x)$ over the data points associated with state k. The term on the left has a clear interpretation as well. To see this, we need to derive an identity that holds for all exponential family distributions. We start by observing that, since $f(x;\eta)$ is normalized, the following must hold for all η

$$1 = \int dx f(x;\eta). \tag{39}$$

If we now take the gradient with respect to η on both sides of the equation we obtain

$$\begin{aligned} 0 &= \int dx\ \nabla_\eta f(x;\eta), && (40)\\ &= \int dx\ f(x;\eta)[T(x) - \nabla_\eta A(\eta)], && (41)\\ &= E_{f(x;\eta)}[T(x)] - \nabla_\eta A(\eta). && (42) \end{aligned}$$

In other words the gradient $\nabla_\eta A(\eta)$ is equal to the expected value of the sufficient statistics $E_{f(x;\eta)}[T(x)]$. If we substitute this relationship back into equation 38 then we obtain the condition

$$E_{f(x;\eta_k)}[T(x)] = \frac{\sum_{t=1}^{T} \gamma_{tk} T(x_t)}{\sum_{t=1}^{T} \gamma_{tk}}. \tag{43}$$

The interpretation of this condition is that we can perform the maximization step in the EM algorithm by finding the value η_k for which the expected value of the sufficient statistics $T(x)$ is equal to the empirical expectation for the observations associated with state k. This type of relationship is an example of a so called moment-matching condition.

As an example, if the observation distribution $f(x;\eta_k)$ is a univariate normal, then the sufficient statistics are $T(x) = (x, x^2)$. The maximization step updates then become

$$\begin{aligned} E_{f(x;\eta_k)}[x] &= \mu_k = \sum_t \gamma_{tk}\, x_t / \sum_t \gamma_{tk}, && (44)\\ E_{f(x;\eta_k)}[x^2] &= \sigma_k^2 + \mu_k^2 = \sum_t \gamma_{tk}\, x_t^2 / \sum_t \gamma_{tk}. && (45) \end{aligned}$$

Suppose the observation distribution is discrete, with D possible values. If we use $x_d = I[x = d]$ to refer to the entries of the one-hot representation, then $T_d(x) = I[x = d] = x_d$ and the maximization-step updates become

$$E_{f(x;\eta_k)}[x_d] = \pi_{k,d} = \sum_t \gamma_{tk} x_{t,d} / \sum_t \gamma_{tk}. \tag{46}$$

Of course, now that we know know how to do the maximization step for any exponential family, we can also make use of these relationships when in deriving the updates for π and A_k. For these variables we now obtain the particularly simple relations

$$\pi_k = \gamma_{1k}, \tag{47}$$

$$A_{kl} = \sum_{t=1}^{T-1} \xi_{tkl} / \sum_{t=1}^{T-1} \sum_{l=1}^{K} \xi_{tkl}. \tag{48}$$

3 The Forward-backward Algorithm

During the expectation step, we update $q(z_{1:T})$ to the posterior

$$q(z_{1:T}) = p(z_{1:T} | x_{1:T}, \theta). \tag{49}$$

For small values of T we could in principle calculate this posterior via direct summation

$$q(z_{1:T}) = \frac{p(x_{1:T}, z_{1:T} | \theta)}{\sum_{z_{1:T} \in [K]^T} p(x_{1:T}, z_{1:T} | \theta)}. \tag{50}$$

However, this quickly becomes infeasible since there are K^T distinct sequences for an HMM with K states. Luckily it turns out that the maximization step can be performed by calculating two sets of expected values

$$\gamma_{tk} := \mathbb{E}_{p(z_{1:T} | x_{1:T}, \theta)}[z_{t,k}], \tag{51}$$

$$\xi_{tkl} := \mathbb{E}_{p(z_{1:T} | x_{1:T}, \theta)}[z_{t,k} z_{t+1,l}]. \tag{52}$$

As we will see below these two quantities can be calculated in $O(K^2 T)$ time using a dynamic programming method known as the forward-backward algorithm.

3.1 Forward and Backward Recursion

We will begin by expressing γ_{tk} as a product of two terms. The first is the joint probability $a_{t,k} := p(x_{1:t}, z_t = k \,|\, \theta)$ of all observations up to time t and the current state z_t. The second is the probability of all future observations $\beta_{t,k} := p(x_{t+1:T} \,|\, z_t = k, \theta)$ conditioned on $z_t = k$. We can express γ in terms of α and β as

$$\gamma_{t,k} = \frac{p(x_{1:T}, z_t = k \,|\, \theta)}{p(x_{1:T} \,|\, \theta)} = \frac{p(x_{t+1:T} | z_t, \theta) p(x_{1:t}, z_t = k \,|\, \theta)}{p(x_{1:T} \,|\, \theta)} = \frac{\alpha_{t,k} \beta_{t,k}}{p(x_{1:T})}. \tag{53}$$

We can similarly express $\xi_{t,kl}$ in terms of α and β as

$$\xi_{t,kl} = p(z_t = k, z_{t+1} = l | x_{1:T}, \theta) = \frac{p(x_{1:T}, z_{t+1} = l, z_t = k | \theta)}{p(x_{1:T} | \theta)} \tag{54}$$

$$= \frac{p(x_{t+2:T} | z_t + 1 = l, \theta) p(x_{t+1} | z_{t+1} = l, \eta) p(z_{t+1} = l | z_{t+1} = k, A) p(x_{1:t}, z_t = k | \theta)}{p(x_{1:T} | \theta)} \tag{55}$$

$$= \frac{\beta_{t+1,l} f(x_t; \eta_l) A_{kl} \alpha_{tk}}{p(x_{1:T} \,|\, \theta)} \tag{56}$$

We will now derive a recursion relation for both α and β. For the first time point, we can calculate $\alpha_{1,k}$ directly

$$\alpha_{1,k} = p(x_1, z_1 = k) = f(x_1; \eta_k)\pi_k \tag{57}$$

For all $t > 1$ we can define a recursion relation that expresses $\alpha_{t,l}$ in terms of $\alpha_{t-1,k}$

$$\begin{aligned}
\alpha_{t,l} &:= p(x_{1:t}, z_t = l|\theta), && (58)\\
&= \sum_{k=1}^{K} p(x_t|z_t = l, \eta) p(z_t = l \mid z_{t-1} = k, A) p(x_{1:t-1}, z_{t-1} = k|\theta). && (59)\\
&= \sum_{k=1}^{K} f(x_t; \eta_l) A_{kl} \alpha_{k,t-1}. && (60)
\end{aligned}$$

For $\beta_{t,k}$ we begin by defining the value at the final time point $t = T$. At this point there are no future observations, so we can set the probability of future observations to 1,

$$\beta_{T,k} = p(\emptyset|z_T = k, \theta) = 1. \tag{61}$$

For all preceding $t < T$ we can express $\beta_{t,k}$ in terms of $\beta_{t+1,l}$

$$\begin{aligned}
\beta_{t,k} &:= p(x_{t+1:T}|z_t = k, \theta), && (62)\\
&= \sum_{l=1}^{K} p(x_{t+2:T}|z_{t+1}{=}l, \theta) p(x_{t+1}|z_{t+1}{=}l, \eta) p(z_{t+1}{=}l \mid z_t{=}k, A), && (63)\\
&= \sum_{l=1}^{K} \beta_{t+1,l}\, f(x_{t+1}; \eta_l)\, A_{kl}. && (64)
\end{aligned}$$

In short, we can calculate γ by combining a forward recursion for α with a backward recursion for β. Note that each step in both recursion requires $O(K^2)$, since we the calculation of each of the K entries requires a sum over K terms. This means the total computation required by the forward-backward algorithm is $O(K^2T)$.

3.2 Normalized computation

In practice we use a slightly different recursion relation when implementing the forward-backward algorithm. If we were to calculate α and β through the naive recursion relations defined above, then we could calculate γ by simple normalization

$$\gamma_{t,k} = \frac{\alpha_{t,k}\beta_{t,k}}{p(x_{1:T} \mid \theta)} = \frac{\alpha_{t,k}\beta_{t,k}}{\sum_{k'} \alpha_{t,k}\beta_{t,k}}. \tag{65}$$

At each step, the normalization term should then in principle be equal to the marginal likelihood of the data

$$p(x_{1:T} \mid \theta) = \sum_{k'} \alpha_{t,k}\beta_{t,k}. \tag{66}$$

In practice $p(x_{1:T} \,|\, \theta)$ will be either a small (or sometimes a very large) number for large values of T, which means naive computation of α and β can lead to numerical underflow (or overflow). A simple solution to this problem is to normalize α_{tk} during each step, by defining

$$\alpha_{t,l} = \sum_{k=1}^{K} f(x_t; \eta_l) A_{kl} \alpha'_{k,t-1}. \tag{67}$$

$$c_t = \sum_{k} \alpha_{tk}, \tag{68}$$

$$\alpha'_{tl} = \alpha_{tl} / c_t. \tag{69}$$

The normalized values of $\alpha'_{t,k}$ now represent the partial posterior $p(z_t = k | x_{1:t}, \theta)$ instead of the partial joint $p(x_{1:t}, z_t = k | \theta)$, which means that the normalized values $\alpha_{t,k}$ differ from the original values by a factor

$$\frac{\alpha_{tk}}{\alpha'_{tk}} = \prod_{i=1}^{t} c_i = \frac{p(x_{1:t}, z_t | \theta)}{p(z_t | x_{1:t}, \theta)} = p(x_{1:t} | \theta). \tag{70}$$

Since this relationship must hold for any t, this in particular implies that we can recover the marginal likelihood from the normalizing constants

$$p(x_{1:T} | \theta) = \prod_{t=1}^{T} c_t. \tag{71}$$

We can also normalize the values $\beta_{t,k}$ on the backward pass. A particularly good choice is

$$\beta'_{tk} = \frac{1}{c_{t+1}} \sum_{l=1}^{K} \beta'_{t+1,l} f(x_{t+1}; \eta_l) A_{kl}. \tag{72}$$

This choice of normalization implies that

$$\frac{\beta_{tk}}{\beta'_{tk}} = \frac{p(x_{1:T} | \theta)}{p(x_{1:t} | \theta)} = \prod_{i=t+1}^{T} c_i, \qquad \frac{\alpha_{tk}\beta_{tk}}{\alpha'_{tk}\beta'_{tk}} = \prod_{i=1}^{T} c_i = p(x_{1:T} | \theta), \tag{73}$$

which in turn ensures that we will no longer have to perform any normalization upon completion of the forward and backward recursion

$$\gamma_{t,k} = \frac{\alpha_{tk}\beta_{tk}}{p(x_{1:T} | \theta)} = \alpha'_{tk}\beta'_{tk}. \tag{74}$$

Similarly, the expression for $\xi_{t,kl}$ in terms of α' and β' becomes

$$\xi_{t,kl} = \frac{1}{c_{t+1}} \beta'_{t+1,l} f(x_{t+1}; \eta_l) A_{kl} \alpha'_{tk}. \tag{75}$$