

Hidden Markov Models: Lecture for April 21

Søren Asmussen

April 21, 2006, 6:45am

1 Kullback-Leibler Distance and Entropy

Let X be a r.v. (possibly multidimensional) with density $f(x)$. The *cross-entropy* or *Kullback-Leibler distance* between f and a different density g is then defined as

$$\mathbb{E}_g \log \frac{g(X)}{f(X)} = \int g(x) \log g(x) dx - \int g(x) \log f(x) dx, \quad (1.1)$$

where $\mathbb{E}_g h(X)$ means the expectation of $h(X)$ where X is a r.v. with density $g(x)$.

A lower bound follows from Jensen's inequality:

$$\begin{aligned} \mathbb{E}_g \log \frac{g(X)}{f(X)} &= -\mathbb{E}_g \log \frac{f(X)}{g(X)} \geq -\log \mathbb{E}_g \frac{f(X)}{g(X)} \\ &= -\int \frac{f}{g} g = -\int f = -\log 1 = 0, \end{aligned}$$

with equality if and only if $f = g^1$. This motivates to view cross-entropy as a measure of closeness, such that the f in a given class of densities which is closest to g is the one with minimal cross-entropy or equivalently with highest value of

$$\int g(x) \log f(x) dx = \mathbb{E}_g \log f(X). \quad (1.2)$$

For the following, note that this last expression also makes sense even if \mathbb{P}_g is singular.

A basic property is that *minimizing cross-entropy is the same as maximum likelihood estimation*: if U_1, \dots, U_n are observations from an unknown density $f_\theta(x)$, the MLE $\hat{\theta}$ is obtained by maximizing the log likelihood

$$\sum_{i=1}^n \log f(U_i; \theta) = n \mathbb{E}_{F_n} \log f(U; \theta)$$

w.r.t. θ where F_n is the empirical distribution giving mass $1/n$ to each of the U_i .

¹or rather $f(x) = g(x)$ for $g(x)dx$ -a.a. x .

2 Finite HMM's: Examples

We consider here hidden Markov models with a fixed length (as opposed to the variable length models occurring, e.g., in speech recognition and DNA sequencing); a basic reference is Cappé, Moulines & Rydén [1].

Such models aim at describing observations Y_0, \dots, Y_n with a dependence structure governed by an underlying unobserved time-homogeneous Markov chain X_0, \dots, X_n . That is,

$$\mathbb{P}(Y_0 \in A_0, \dots, Y_n \in A_n \mid X_0 = x_0, \dots, X_n = x_n) = G(x_0, A_0) \dots G(x_n, A_n)$$

for suitable distributions $G(x, \cdot)$. The Markov chain X_0, \dots, X_n may be discrete with transition matrix $\mathbf{Q} = (q_{xx'})_{x, x'=1, \dots, r}$ or general with transition kernel say $Q(x, F)$. We will use notation like $\mathbf{Y}_{0:n} \stackrel{\text{def}}{=} (Y_0, \dots, Y_n)$, $\mathbf{X}_{0:n} \stackrel{\text{def}}{=} (X_0, \dots, X_n)$. The joint density of $(\mathbf{X}_{0:n}, \mathbf{Y}_{0:n})$ in the discrete case is

$$\nu_{x_0} q_{x_0 x_1} \dots q_{x_{n-1} x_n} g_{x_0}(y_0) \dots g(x_n, y_n) \quad (2.1)$$

where $g_x(y)$ is the density of $G(x, dy)$ w.r.t. some reference measure μ .

Hidden Markov models are one of the basic vehicles for modeling dependent observations and one gets quite far by just using a finite set E of Markov states. In fact, if the Y_k take values in say \mathbb{R}^d , then *any* distribution H on $\mathbb{R}^{(n+1)d}$ is the weak limit of distributions H_m corresponding to hidden Markov model (typically with the size of $E = E_m$ going to ∞ with m).

A classical statistical problem is *filtering* or *smoothing* where one asks for the distribution of $\mathbf{X}_{0:n}$ (or suitable marginals) given the observed values; precise definitions are given in Section 3. For example:

Example 2.1 Consider a change-point problem where E has only two states 1, 2 such that a transition $2 \rightarrow 1$ cannot occur. This means that the Y_k have density $g_1(\cdot)$ up to a certain point (the time of change of state from 1 to 2) and $g_2(\cdot)$ thereafter. A basic task is to assert whether such a change occurs and when, which basically amounts to giving statements on the conditional distribution of $\mathbf{X}_{0:n}$ given $\mathbf{Y}_{0:n} = \mathbf{y}_{0:n}$. In particular, one is interested in the (conditional) probabilities of the sequences $\mathbf{x}_{0:n}^{(0)}$ of all 1's and $\mathbf{x}_{0:n}^{(k)}$, $k = 0, \dots, n-1$, with 1's at the first k places and 2's at the rest.

A more complicated and realistic change-point problem (geological layers of different types) is given as a recurrent example in [1]. □

Example 2.2 A target moves on a lattice but the observation is blurred by noise. More precisely, let the lattice be $L \stackrel{\text{def}}{=} \{1, \dots, M\}^2$ and let the (known) movement mechanism be random walk with probability 1/8 for each of the 4 neighbours and 1/2 to stay at the same site. The Markov chain at time n is the position X_n of the target which without noise is observed as a 1 at site X_n whereas the remaining sites have a 0. The noise at time n at site ij is $\epsilon_{n,i,j}$ where the $\epsilon_{n,i,j}$ are i.i.d. $N(0, 1)$. Thus, the observation at time n is $\mathbf{Y}_n = (Y_{n,ij})_{ij \in L}$ where $Y_{n,ij} = \epsilon_{n,i,j} + \mathbb{1}\{X_n = ij\}$. One is interested in $\mathbb{P}(X_{k+1} = ij \mid \mathbf{Y}_{0:k})$ in order to decide at which ij to aim the next shot (or bomb). □

Example 2.3 In financial modeling of log-returns Y_0, \dots, Y_n , the simplest model (Black-Scholes) is just that the Y_k are i.i.d. $N(\mu, \sigma^2)$. However, often one observes phenomena such as periods with larger variation than typical, i.e. stochastic volatility. In a hidden Markov model, one takes instead $g_x(\cdot)$ as the $N(\mu_x, \sigma_x^2)$ density. A frequently used model for the underlying Markov chain is an autoregressive process $X_{k+1} = \rho X_k + V_k$ with the V_k i.i.d. $N(0, \omega^2)$, but as noted above, an appealing alternative is just a general finite-state Markov chain. \square

Example 2.4 Let $C_0, \dots, C_n \in \{-1, 1\}$ be a sequence of bits transmitted along a noisy channel. More precisely, assume the observed sequence is Y_0, \dots, Y_n where $Y_k = W_k C_k + V_k$ where the V_k are i.i.d. $N(0, \sigma^2)$ and the W_k are so-called *fading coefficients*, describing time-varying properties of the channel and often modelled as an autoregressive process $W_{k+1} = \rho W_k + V'_k$ with the V'_k i.i.d. $N(0, \omega^2)$. Thus the hidden Markov chain may be taken as $X_k = (C_k, W_k)$, and one is facing a filtering problem, to reconstruct the C_k from the Y_k (the W_k are of no direct interest in themselves, i.e. they constitute nuisance variables). \square

3 Filtering and Smoothing in Finite HMM's

The state space of X_0, X_1, \dots is taken as $\{1, \dots, r\}$ throughout. When writing $\mathbb{P}(Y_k = y_k)$, this is to be understood as the density of Y_k at y_k . Thus, e.g., $\mathbb{P}(Y_k = y_k | X_k = x_k) = g_{x_k}(y_k)$. We will further use notation like $\mathbf{y}_{0:k} \stackrel{\text{def}}{=} (y_0, \dots, y_k)$, $\mathbf{x}_{k:n} \stackrel{\text{def}}{=} (x_k, \dots, x_n)$ and $p(\mathbf{y}_{0:k}) \stackrel{\text{def}}{=} \mathbb{P}(\mathbf{Y}_{0:x} = \mathbf{y}_{0:k})$,

$$p(\mathbf{y}_{0:k}, x_k) \stackrel{\text{def}}{=} \mathbb{P}(\mathbf{Y}_{0:x} = \mathbf{y}_{0:k}, X_k = x_k), \quad p(\mathbf{y}_{0:k} | x_k) \stackrel{\text{def}}{=} \mathbb{P}(\mathbf{Y}_{0:x} = \mathbf{y}_{0:k} | X_k = x_k)$$

etc.

We are concerned with recursions for the following quantities, using a notation close to [1] (note that often, we suppress the dependence on the observed sequence $\mathbf{y}_{0:n}$):

ℓ_k the log likelihood of $\mathbf{y}_{0:k}$, that is, the logarithm of

$$\sum_{\mathbf{x}_{0:k}} \nu_{x_0} g_{x_0}(y_0) q_{x_0, x_1} g_{x_1}(y_1) q_{x_1, x_2} \cdots q_{x_{k-1}, x_k} g_{x_k}(y_k) \quad (3.1)$$

where $\boldsymbol{\nu} \stackrel{\text{def}}{=} (\nu_x)$ is the initial distribution (the distribution of X_0 , i.e. $\nu_x = \mathbb{P}(X_0 = x)$). That (3.1) is the likelihood means that it gives the marginal probability that $\mathbf{Y}_{0:k} = \mathbf{y}_{0:k}$, which follows since for each $\mathbf{x}_{0:k}$ the term under the sum sign is the probability that $\mathbf{X}_{0:k} = \mathbf{x}_{0:k}$, $\mathbf{Y}_{0:k} = \mathbf{y}_{0:k}$.

$c_k(y_k)$ the conditional probability $p(y_k | \mathbf{y}_{0:k-1})$ that $Y_k = y_k$ given $\mathbf{y}_{0:k-1}$.

$\phi_{k:\ell|n}(\mathbf{x}_{k:\ell})$ the conditional probability $p(\mathbf{x}_{k:\ell} | \mathbf{y}_{0:n})$ that $\mathbf{X}_{k:\ell} = \mathbf{x}_{k:\ell}$ given the observations $\mathbf{y}_{0:n}$. Of particular interest are:

$\phi_{0:n|n}(\mathbf{x}_{0:n})$. This is the *joint smoothing probability*, that is, the conditional probability of the whole sequence of Markov states $\mathbf{x}_{0:n}$ given $\mathbf{y}_{0:n}$.

$\phi_{k|n}(x_k) \stackrel{\text{def}}{=} \phi_{k:k|n}(x_k)$. This is the *marginal smoothing probability*, that is, the conditional probability that $X_k = x$ given $\mathbf{y}_{0:n}$.

$\phi_{k:k+1|n}(x_k, x_{k+1})$. This is the conditional probability that $X_k = x, X_{k+1} = x_{k+1}$ given $\mathbf{y}_{0:n}$, i.e. that a transition from x_k to x_{k+1} occurred between k and $k + 1$. It is a key tool for obtaining the joint smoothing probability, and is further used in Section 5.

$\phi_{n+1|n}(x_{n+1})$. This is the *prediction probability*, that is the conditional probability that the next Markov state is x_{n+1} given $\mathbf{y}_{0:n}$.

$\phi_{k|k}(x_k) \stackrel{\text{def}}{=} \phi_{k:k|k}(x_k)$. This is the *marginal filtering probability*, that is the conditional probability given $\mathbf{y}_{0:k}$ (not $\mathbf{y}_{0:n}$!) that the Markov state at time k is x_k

The computational difficulty in obtaining the above quantities is largely the curse of dimension, i.e. large values of n and/or k . For example, in (3.1) the sum extends over r^{n+1} values of $\mathbf{x}_{0:n}$ and each term is a product of $n + 1$ factors, so the complexity is $O(nr^n)$. Further, a naive computation of (3.1) will typically give rise to overflow or underflow when n is large, which motivates some of the normalizations to be used in the following.

3a The Forward Step

This step computes the conditional likelihoods $c_k(\mathbf{y}_k)$, the marginal filtering probabilities $\phi_{k|k}(x_k)$, the prediction probabilities $\phi_{k+1|k}(x_{k+1})$ and thereby the log likelihood ℓ_n , cf. (3.2) below, by a forward pass in k (to get the smoothing probabilities, the additional backward step presented below is needed).

The initialization is $\phi_{0|-1}(x_0) = \nu_{x_0}$ and the updating from $k - 1$ to k is

1. $c_k(\mathbf{y}_k) \leftarrow \sum_{x_k=1}^r \phi_{k|k-1}(x_k) g_{x_k}(\mathbf{y}_k);$
2. $\phi_{k|k}(x_k) \leftarrow \phi_{k|k-1}(x_k) g_{x_k}(\mathbf{y}_k) / c_k(\mathbf{y}_k);$
3. $\phi_{k+1|k}(x_{k+1}) \leftarrow \sum_{x_k=1}^r \phi_{k|k}(x_k) q_{x_k x_{k+1}}.$

The complexity of this algorithm is $O(nr^2)$. The validity follows by elementary conditioning arguments and the definitions of the $c_k, \phi_{k|k}, \phi_{k+1|k}$ as follows. First

$$\begin{aligned} c_k(\mathbf{y}_k) &= p(\mathbf{y}_k \mid \mathbf{y}_{0:k-1}) = \sum_{x_k=1}^r p(\mathbf{y}_k, x_k \mid \mathbf{y}_{0:k-1}) \\ &= \sum_{x_k=1}^r p(x_k \mid \mathbf{y}_{0:k-1}) p(\mathbf{y}_k \mid \mathbf{y}_{0:k-1}, x_k) \\ &= \sum_{x_k=1}^r \phi_{k|k-1}(x_k) p(\mathbf{y}_k \mid x_k) = \sum_{x_k=1}^r \phi_{k|k-1}(x_k) g_{x_k}(\mathbf{y}_k) \end{aligned}$$

which shows 1. Next the chain rule for conditional probabilities gives

$$p(x_k, \mathbf{y}_k \mid \mathbf{y}_{0:k-1}) = p(x_k \mid \mathbf{y}_{0:k}) p(\mathbf{y}_{0:k} \mid \mathbf{y}_{0:k-1}).$$

The l.h.s. is $g_{x_k}(y_k)p(x_k | \mathbf{y}_{0:k-1})$ so altogether, we get

$$g_{x_k}(y_k)\phi_{k|k-1}(x_k) = \phi_{k|k}(x_k)c_k(y_k)$$

which proves 2. Finally,

$$\begin{aligned} \phi_{k+1|k}(x_{k+1}) &= p(x_{k+1} | \mathbf{y}_{0:k}) = \sum_{x_k=1}^r p(x_k, x_{k+1} | \mathbf{y}_{0:k}) \\ &= \sum_{x_k=1}^r \mathbb{P}(x_k | \mathbf{y}_{0:k}) q_{x_k x_{k+1}} = \sum_{x_k=1}^r \phi_{k|k}(x_k) q_{x_k x_{k+1}}, \end{aligned}$$

proving 3.

For ℓ_n , it follows by writing $p(\mathbf{y}_{0:n})$ as

$$p(y_0)p(y_1 | y_0)p(y_2 | \mathbf{y}_{0:1}) \cdots p(y_n | \mathbf{y}_{0:n-1}) \quad (3.2)$$

and taking logarithms that

$$\ell_n = \sum_{k=0}^n c_k(y_k). \quad (3.3)$$

3b The Backward Step

This step allows to compute smoothing probabilities, first low dimensional ones like $\phi_{k|n}(x_k)$, $\phi_{k:k+1|n}(x_k, x_{k+1})$ and next the entire set $\phi_{0:n|n}(\mathbf{x}_{0:n})$. Again, the complexity is $O(nr^2)$.

The algorithm involves the auxiliary quantities

$$\beta_{k|n}(x_k) \stackrel{\text{def}}{=} p(\mathbf{y}_{k+1:n} | x_k), \quad \check{\beta}_{k|n}(x_k) \stackrel{\text{def}}{=} \frac{\beta_{k|n}(x_k)}{c_k \cdots c_n} = \frac{p(\mathbf{y}_{0:k-1})}{p(\mathbf{y}_{0:n})} \beta_{k|n}(x_k)$$

(for the last identity, use a decomposition similar to (??)). Here the $\beta_{k|n}$ satisfy the obvious recursion

$$\beta_{k|n}(x_k) = \sum_{x_{k+1}=1}^p q_{x_k x_{k+1}} g_{x_{k+1}}(y_{k+1}) \beta_{k+1|n}(x_{k+1}),$$

and hence

$$\check{\beta}_{k|n}(x_k) = \frac{1}{c_k} \sum_{x_{k+1}=1}^p q_{x_k x_{k+1}} g_{x_{k+1}}(y_{k+1}) \check{\beta}_{k+1|n}(x_{k+1}). \quad (3.4)$$

The initial conditions are $\beta_{n|n}(x_n) \stackrel{\text{def}}{=} 1$, $\check{\beta}_{n|n}(x_n) \stackrel{\text{def}}{=} 1/c_n$. From $\check{\beta}_{n|n}$, one then computes first $\check{\beta}_{n-1|n}$, next $\check{\beta}_{n-2|n}$ and so on (backward scan).

The implications are that combining with the filtering probabilities computed in the forward step, we can obtain the smoothing probabilities as

$$\phi_{k|n}(x_k) = \frac{\phi_{k|k}(x_k) \check{\beta}_{k|n}(x_k)}{\sum_{x'_k=1}^p \phi_{k|k}(x'_k) \check{\beta}_{k|n}(x'_k)}, \quad (3.5)$$

$$\phi_{k:k+1|n}(x_k, x_{k+1}) = \phi_{k|k}(x_k) q_{x_k x_{k+1}} g_{x_{k+1}}(y_{k+1}) \check{\beta}_{k+1|n}(x_{k+1}). \quad (3.6)$$

To see this, note first that up to a constant depending only on $\mathbf{y}_{0:n}$ but not x_k , the numerator $\phi_{k:k|k}(x_k)\check{\beta}_{k|n}(x_k)$ in (3.4) is proportional to

$$p(\mathbf{y}_{0:k}, x_k)p(\mathbf{y}_{k+1:n}|x_k) = p(\mathbf{y}_{0:n}, x_k).$$

Therefore the r.h.s. of (3.4) equals

$$\frac{p(\mathbf{y}_{0:n}, x_k)}{\sum_{x'_k=1}^p p(\mathbf{y}_{0:n}, x'_k)} = \frac{p(\mathbf{y}_{0:n}, x_k)}{p(\mathbf{y}_{0:n})} = \phi_{k|n}(x_k).$$

Next the r.h.s. of (3.5) equals

$$\begin{aligned} & \frac{p(\mathbf{y}_{0:k}, x_k)}{p(\mathbf{y}_{0:k})} q_{x_k x_{k+1}} g_{x_{k+1}}(\mathbf{y}_{k+1}) \frac{p(\mathbf{y}_{0:k})}{p(\mathbf{y}_{0:n})} p(\mathbf{y}_{k+2:n} | x_{k+1}) \\ &= \frac{1}{p(\mathbf{y}_{0:n})} p(\mathbf{y}_{0:k}, x_k) q_{x_k x_{k+1}} g_{x_{k+1}}(\mathbf{y}_{k+1}) p(\mathbf{y}_{k+2:n} | x_{k+1}) \\ &= \frac{p(\mathbf{y}_{0:n}, x_k, x_{k+1})}{p(\mathbf{y}_{0:n})} = \phi_{k:k+1|n}(x_k, x_{k+1}). \end{aligned}$$

To obtain the full smoothing probability $\phi_{0:n|n}(\mathbf{x}_{0:n})$,

Not implemented in this version

4 The EM Algorithm

The EM algorithm is one of the main tools for performing maximum likelihood (ML) estimation in the absence of full data information (incomplete observations, lost data etc.). The main example is exponential families with density

$$f_{\boldsymbol{\theta}}(\mathbf{v}) \stackrel{\text{def}}{=} e^{\boldsymbol{\theta}^T \mathbf{t}(\mathbf{v}) - \kappa(\boldsymbol{\theta})} \quad (4.1)$$

w.r.t. some reference measure $\mu(d\mathbf{v})$ where \mathbf{v} is the observation vector. The ML estimator is then often some nice explicit function $\hat{\boldsymbol{\theta}} \stackrel{\text{def}}{=} \hat{\boldsymbol{\theta}}(\mathbf{t}(\mathbf{v}))$ of $\mathbf{t}(\mathbf{v})$ where \mathbf{v} is the observed outcome of the r.v. \mathbf{V} with the prescribed density $f_{\boldsymbol{\theta}}$.

Example 4.1 Let V_1, \dots, V_n be i.i.d. exponential θ and $\mathbf{V} \stackrel{\text{def}}{=} (V_1 \dots V_n)^T$. Then (4.1) holds with $t(\mathbf{V}) = -V_1 - \dots - V_n$, $\kappa(\theta) = n \log \lambda$, and we have $\hat{\lambda} = n/(V_1 + \dots + V_n) = -n/t(\mathbf{V})$, as can be obtained by straightforward differentiation of the log likelihood $-\lambda(V_1 + \dots + V_n) + n \log \lambda$. \square

Example 4.2 In a normal mixture problem $\mathbf{Y} = (Y_1, \dots, Y_n)$ are i.i.d. with a common density $f_{\boldsymbol{\theta}}(y)$ which is a mixture

$$\sum_{i=1}^d \alpha_i \frac{1}{\sqrt{2\pi}} e^{-(y-\mu_i)^2/2}$$

of $N(\mu, 1)$ densities (for simplicity, we assume that the variance is known and equal to 1), so that the unknown parameters are $(\alpha_i, \mu_i)_{i=1, \dots, d}$ (constrained by $\alpha_1 + \dots + \alpha_d = 1$). One can interpret the model as a given observation Y being assigned type i w.p. α_i and then having distribution $N(\mu_i, 1)$.

To obtain the representation (4.1), one defines

$$\begin{aligned} \mathbf{V} &\stackrel{\text{def}}{=} (J_1, \dots, J_n, Y_1, \dots, Y_n), & \mathbf{t}(\mathbf{V}) &\stackrel{\text{def}}{=} (N_1, \dots, N_d, S_1, \dots, S_d), \\ \boldsymbol{\theta} &\stackrel{\text{def}}{=} (\theta_1, \dots, \theta_d, \theta_{d+1}, \dots, \theta_{2d}) &\stackrel{\text{def}}{=} (\log \alpha_1, \dots, \log \alpha_d, \mu_1, \dots, \mu_d), \end{aligned}$$

where J_k is the type of Y_k and

$$N_i \stackrel{\text{def}}{=} \sum_{k=1}^n \mathbb{1}_{\{J_k=i\}}, \quad S_i \stackrel{\text{def}}{=} \sum_{k: J_k=i} Y_k = \sum_{k=1}^n Y_k \mathbb{1}_{\{J_k=i\}}.$$

Further, μ is supported by d disjoint copies of \mathbb{R} , such that the restriction $\mu(dy_k)$ to the k th is $(\sqrt{2\pi})^{-1} e^{-y_k^2/2} dy_k$. Then (4.1) holds with $\kappa(\boldsymbol{\theta}) = (\mu_1^2 + \dots + \mu_d^2)/2$.

After conditioning upon the J_k , it is readily guessed that the MLE estimator $\widehat{\boldsymbol{\theta}}$ is given by

$$\widehat{\alpha}_i = \frac{N_i}{n}, \quad \widehat{\mu}_i = \frac{S_i}{N_i}, \quad i = 1, \dots, d, \quad (4.2)$$

i.e. the ML estimator of α_i is the empirical mean $N(i)/n$ of the Y_k with $J_k = i$, and similarly the ML estimator of μ_i is the empirical mean of the same set of Y_k . For a formal verification of (4.2), introduce a Lagrangian multiplier λ and consider the minimization of

$$\sum_{j=1}^{2d} \theta_j t_j(\mathbf{v}) + \lambda (e^{\theta_{d+1}} + \dots + e^{\theta_{2d}} - 1). \quad \square$$

Now consider the general exponential family setting, and assume that only \mathbf{Y} is observed and not the whole of \mathbf{V} . For example in Example 4.1, there could be censoring at t_0 so that only the $V_1 \wedge t_0, \dots, V_1 \wedge t_0$ are observed, and in Example 4.2 only the Y_k but not their types J_k could be observed. The statistical estimation problem remains explicitly tractable in the first example but not in the second where we are left with an $2d - 1$ -dimensional optimization problem without an explicit solution. The EM algorithm proceeds iteratively. When updating from $\boldsymbol{\theta}_n$ to $\boldsymbol{\theta}_{n+1}$, $\mathbf{t}(\mathbf{V})$ is replaced by its $\mathbb{P}_{\boldsymbol{\theta}_n}$ -conditional expectation \mathbf{t}_n given \mathbf{Y} :

$$\mathbf{t}_n \stackrel{\text{def}}{=} \mathbb{E}_{\boldsymbol{\theta}_n}[\mathbf{t}(\mathbf{X}) | \mathbf{Y}], \quad \boldsymbol{\theta}_{n+1} \stackrel{\text{def}}{=} \widehat{\boldsymbol{\theta}}(\mathbf{t}_n). \quad (4.3)$$

It can be shown that the $\boldsymbol{\theta}_n$ -likelihood is non-decreasing in n and hence that $\boldsymbol{\theta}_n \rightarrow \widehat{\boldsymbol{\theta}}$ under suitable regularity conditions.

Example 4.3 Consider the normal mixture example 4.2. Write

$$\boldsymbol{\theta}_n = (\log \alpha_{n,1}, \dots, \log \alpha_{n,d}, \mu_{n,1}, \dots, \mu_{n,d}).$$

Then

$$\begin{aligned} \mathbb{P}_{\boldsymbol{\theta}_n}(J_k = i | \mathbf{y}) &= \mathbb{P}_{\boldsymbol{\theta}_n}(J_k = i | y_k) \\ &= \frac{\alpha_{n,i} (2\pi)^{-1/2} e^{-(y_k - \mu_{n,i})^2/2}}{\sum_1^d \alpha_{n,j} (2\pi)^{-1/2} e^{-(y_k - \mu_{n,j})^2/2}} \stackrel{\text{def}}{=} \psi(\boldsymbol{\theta}_n, y_k), \\ \mathbf{t}_n &= (N_{n,1}, \dots, N_{n,d}, S_{n,1}, \dots, S_{n,d}) \end{aligned}$$

where

$$N_{n,i} \stackrel{\text{def}}{=} \mathbb{E}_{\boldsymbol{\theta}_n} [N_i | \mathbf{Y}] = \sum_{k=1}^n \psi(\boldsymbol{\theta}_n, y_k),$$

$$S_{n,i} \stackrel{\text{def}}{=} \mathbb{E}_{\boldsymbol{\theta}_n} [S_i | \mathbf{Y}] = \sum_{k=1}^n Y_i \psi(\boldsymbol{\theta}_n, y_k)$$

From (4.2), it therefore follows that $\boldsymbol{\theta}_{n+1} = \widehat{\boldsymbol{\theta}}(\mathbf{t}_n)$ is given by $\widehat{\alpha}_{n+1,i} = N_{n,i}/n$, $\widehat{\mu}_{n+1,i} = S_{n,i}/N_{n,i}$. \square

The difficulty in applying the EM algorithm is usually the computation of the conditional expectation \mathbf{t}_n (the E-step) rather than the computation of $\widehat{\boldsymbol{\theta}}(\mathbf{t}_n)$ (the M-step) which involve just the same calculations as when computing the MLE in the presence of full observations. It is therefore tempting to perform the E-step by Monte Carlo, which means that \mathbf{t}_n is redefined as

$$\mathbf{t}_n \stackrel{\text{def}}{=} \frac{1}{m} (\mathbf{t}(\mathbf{V}_{n,1}) + \cdots + \mathbf{t}(\mathbf{V}_{n,m})), \quad (4.4)$$

where $\mathbf{V}_{n,1}, \dots, \mathbf{V}_{n,m}$ are simulated replications from the conditional $\mathbb{P}_{\boldsymbol{\theta}_n}$ -distribution of \mathbf{V} given \mathbf{Y} . This algorithm is known under names such as the *Monte Carlo EM algorithm*, the *stochastic EM algorithm* etc. (outside the exponential family setting we have restricted ourselves to, these names may actually cover slightly different algorithms). The Monte Carlo EM algorithm obviously has the property that $\{\boldsymbol{\theta}_n\}_{n \in \mathbb{N}}$ becomes a time-homogeneous Markov chain with no state being absorbing (in particular not the MLE $\widehat{\boldsymbol{\theta}}!$), so that the best one can hope for is oscillations around $\widehat{\boldsymbol{\theta}}$ which are small given m has been large enough, not convergence in probability or a.s. To obtain this, one needs to let $m \stackrel{\text{def}}{=} m_n$ go to ∞ with n .

5 The EM Algorithm for Finite HMM's

If $\mathbf{X}_{0:n}$ is a discrete Markov chain with completely unknown transition probabilities q_{ij} and known initial distribution $\boldsymbol{\nu}$, it is easy to see that the maximum likelihood (ML) estimates are given by

$$\widehat{q}_{xx'} = \frac{N_{xx'}}{N_x} \quad (5.1)$$

where $N_{xx'} \stackrel{\text{def}}{=} \sum_0^{n-1} \mathbb{1}\{X_k = x, x_{k+1} = x'\}$ is the observed number of transitions $x \rightarrow x'$ and $N_x \stackrel{\text{def}}{=} \sum_0^{n-1} \mathbb{1}\{X_k = x\}$ the observed number of visits to x up to $n-1$. The intuition is of course that (5.2) gives the empirical frequency of transitions $x \rightarrow x'$ among all transitions out of x .

Now consider a HMM where the $q_{xx'}$ are completely unknown and the g_x have the form $g(y; \theta_x)$. To be specific, we will assume that θ_x is a normal mean as in Example 4.2. As is readily guessed, the MLE estimators when both $\mathbf{X}_{0:n}$ and $\mathbf{Y}_{0:n}$ are observed are then

$$\widehat{q}_{xx'} = \frac{N_{xx'}}{N_x}, \quad \widehat{\theta}_x = \frac{S_x}{N_x^*} \quad (5.2)$$

where $N_x^* \stackrel{\text{def}}{=} \sum_0^{n-1} \mathbb{1}\{X_k = x\}$ and $S_x \stackrel{\text{def}}{=} \sum_{k=0}^n Y_k \mathbb{1}\{X_k = x\}$. Again, the expression for $\widehat{\theta}(x)$ is intuitive: $\widehat{\theta}_x$ is the empirical mean of all Y_k for which $X_k = x$.

In the hidden situation, where only $Y_{0:n}$ is observed and we proceed via the EM algorithm, we have to plug in the conditional expectations of $N_{x'}(m), N_x(m), N_x^*(m), S_x(m)$ given $\mathbf{Y}_{0:n}$ in step m . Letting

$$\mathbf{Q}(m) = (q_{xx'}(m))_{x,x'=1,\dots,r}, \quad \boldsymbol{\theta}(m) = (\theta_x(m))_{x=1,\dots,r}$$

be the current estimates $q_{xx'}(m), \theta_x(m)$, this gives

$$\begin{aligned} N_x(m) &= \mathbb{E}_{\mathbf{Q}(m), \boldsymbol{\theta}(m)}[N_x \mid \mathbf{y}_{0:n}] \\ &= \sum_{k=0}^{n-1} \mathbb{P}_{\mathbf{Q}(m), \boldsymbol{\theta}(m)}(X_k = x \mid \mathbf{y}_{0:n}) = \sum_{k=0}^{n-1} \phi_{k|n}(x; m) \end{aligned}$$

where the $\phi_{k|n}(x; m)$ are the smoothing probabilities computed using $\mathbf{Q}(m), \boldsymbol{\theta}(m)$ as parameters. Similarly,

$$\begin{aligned} N_x^*(m) &= \sum_{k=0}^n \phi_{k|n}(x; m), \quad N_{xx'}(m) = \sum_{k=0}^{n-1} \phi_{k:k+1|n}(x, x'; m), \\ S_x(m) &= \sum_{k=0}^n \phi_{k|n}(x; m) Y_k, \end{aligned}$$

and the EM updating formulas are

$$q_{xx'}(m+1) = \frac{N_{xx'}(m)}{N_x(m)}, \quad \theta_x(m+1) = \frac{S_x(m)}{N_x^*(m)} \quad (5.3)$$

References

- [1] O. Cappé, E. Moulines & T. Rydén (2005) *Inference in Hidden Markov Models*. Springer-Verlag.