`osl-dynamics`: HMM Cost Function

C. Gohil, and R. Huang

OHBA, Department of Psychiatry, Warneford Hospital, Oxford, OX3 7JX

(March 13, 2023)

Abstract

We describe the calculation of the cost function used to update the observation model parameters (state means and covariances) in the `osl-dynamics` implementation of a Hidden Markov Model (HMM). We also describe the calculation of the variational free energy for this model.

1 Variational Free Energy

In variational Bayesian inference we learn a posterior distribution for model parameters, $q(.)$, by minimising the *variational free energy*, $\mathcal{F}$, given some data we have observed, x_t. For the HMM, our model parameters are:

- The hidden state at each time point, s_t.
- The state transition probability at each time point, $\mathbf{P}$.
- The initial state probability, π.
- The observation model parameters, θ_{obs}.

If we were being Bayesian on all of these model parameters, we would minimise the following variational free energy[1] [1]

$$\mathcal{F} = \iiiint q(s_{1:T})q(\mathbf{P})q(\pi)q(\theta_{\text{obs}}) \log \left[\frac{q(s_{1:T})q(\mathbf{P})q(\pi)q(\theta_{\text{obs}})}{p(x_{1:T}, s_{1:T}, \mathbf{P}, \pi, \theta_{\text{obs}})} \right] ds_{1:T} d\mathbf{P} d\pi d\theta_{\text{obs}}, \quad (1)$$

where $s_{1:T}$ and $x_{1:T}$ denote $s_1, ..., s_T$ and $x_1, ..., x_T$ respectively. However, in the `osl-dynamics` implementation of an HMM, we will only be Bayesian on the hidden states, $s_{1:T}$. We will learn point estimates for all the other parameters: θ_{obs}, $\mathbf{P}$ and π. We learn all of our model parameters by minimising the following variational free energy,

$$\mathcal{F} = \int q(s_{1:T}) \log \left[\frac{q(s_{1:T})}{p(x_{1:T}, s_{1:T})} \right] ds_{1:T}. \quad (2)$$

We will show that Eq. (2) implicitly depends on the point estimates for θ_{obs} below.

[1] We have used the mean field approximation.

2 Generative Model

The denominator in the log function, $p(.)$, is determined by our generative model. For the HMM, if we were being fully Bayesian this would be [1]

$$p(x_{1:T}, s_{1:T}, \mathbf{P}, \pi, \theta_{\text{obs}}) = p(x_1|s_1, \theta_{\text{obs}})p(s_1|\pi)p(\pi)p(\mathbf{P})p(\theta_{\text{obs}}) \prod_{t=2}^{T} p(x_t|s_t, \theta_{\text{obs}})p(s_t|s_{t-1}, \mathbf{P}). \quad (3)$$

However, because we are learning point estimates for most of these parameters $(\theta_{\text{obs}}, \mathbf{P}, \pi)$ their prior distributions disappear. We will use the following generative model,

$$p(x_{1:T}, s_{1:T}) = p(x_1|s_1, \theta_{\text{obs}})p(s_1) \prod_{t=2}^{T} p(x_t|s_t, \theta_{\text{obs}})p(s_t|s_{t-1}), \quad (4)$$

where θ_{obs} is a point estimate. We assume a multivariate normal distribution for the observed data,

$$p(x_t|s_t = k, \theta_{\text{obs}}) = \mathcal{N}(m_k, C_k), \quad (5)$$

where m_k and C_k are the mean and covariance for state k respectively. Our observation model parameters θ_{obs} are the set of state means and covariances, $\theta_{\text{obs}} = \{m_k, C_k\}$.

3 Cost Function for Learning $\theta_{\text{obs}} = \{m_k, C_k\}$

We update our point estimate for θ_{obs} by minimising Eq. (2). We separate Eq. (2) into the following terms[2]

$$\mathcal{F} = -\int q(s_{1:T}) \log\left[p(x_{1:T}, s_{1:T})\right] ds_{1:T} + \int q(s_{1:T}) \log\left[q(s_{1:T})\right] ds_{1:T}. \quad (6)$$

Only the first term depends on θ_{obs} so the second term can be ignored. Substituting Eq. (4) into the first term, we have

$$\mathcal{F} \propto -\int q(s_{1:T}) \log\left[p(x_1|s_1, \theta_{\text{obs}})p(s_1) \prod_{t=2}^{T} p(x_t|s_t, \theta_{\text{obs}})p(s_t|s_{t-1})\right] ds_{1:T}. \quad (7)$$

Again, only retaining the factors that depend on θ_{obs}, we have

$$\begin{aligned}
\mathcal{F} &\propto -\int q(s_{1:T}) \log\left[\prod_{t=1}^{T} p(x_t|s_t, \theta_{\text{obs}})\right] ds_{1:T} \\
&\propto -\sum_{t=1}^{T} \int q(s_{1:T}) \log\left[p(x_t|s_t, \theta_{\text{obs}})\right] ds_{1:T} \\
&\propto -\sum_{t=1}^{T} \int ... \int q(s_1)...q(s_T) \log\left[p(x_t|s_t, \theta_{\text{obs}})\right] ds_1...ds_T \\
&\propto -\sum_{t=1}^{T} \int q(s_t) \log\left[p(x_t|s_t, \theta_{\text{obs}})\right] ds_t = \mathcal{L}.
\end{aligned} \quad (8)$$

Here, we have defined the negative log-likelihood loss, $\mathcal{L}$, which is minimised via stochastic gradient descent to learn the parameters θ_{obs}. $q(s_t)$ is the marginal posterior calculated using

[2] We have used $\int q(\xi) d\xi = 1$ to evaluate some of the integrals.

the Baum-Welch algorithm, commonly denoted using the symbol γ. As $q(s_t)$ is a discrete probability distribution for the state, we can evaluate the integral as

$$\begin{aligned}
\mathcal{L} &= -\sum_{t=1}^{T}\sum_{k=1}^{K} q(s_t = k)\log\left[p(x_t|s_t = k, \theta_{\text{obs}})\right] \\
&= -\sum_{t=1}^{T}\sum_{k=1}^{K} \gamma_{kt}\log\left[p(x_t|s_t = k, \theta_{\text{obs}})\right],
\end{aligned} \tag{9}$$

where K is the number of states and $q(s_t = k) = \gamma_{kt}$ is the probability of state k at time t. Substituting Eq. (5) into this we have

$$\mathcal{L} = -\sum_{t=1}^{T}\sum_{k=1}^{K} \gamma_{kt}\log\left[\mathcal{N}(x_t|m_k, C_k)\right], \tag{10}$$

which is the log-likelihood loss function implemented in `osl-dynamics` for inferring the point estimates for the observation model parameters $\theta_{\text{obs}} = \{m_k, C_k\}$.

4 Calculation of the Variational Free Energy

Once we have trained an HMM we may want to evaluate the variational free energy, i.e. Eq.(2). This can be done with the `free_energy` method of the `hmm.Model` class. The method calculates Eq. (2) by first splitting it into three terms:

$$\begin{aligned}
\mathcal{F} &= \int q(s_{1:T})\log\left[\frac{q(s_{1:T})}{p(x_{1:T}, s_{1:T})}\right] ds_{1:T}, \\
&= \int q(s_{1:T})\log\left[\frac{q(s_{1:T})}{p(x_1|s_1)p(s_1)\prod_{t=2}^{T} p(x_t|s_t)p(s_t|s_{t-1})}\right] ds_{1:T}, \\
&= -\int q(s_{1:T})\log\left[\prod_{t=1}^{T} p(x_t|s_t)\right] ds_{1:T} + \int q(s_{1:T})\log\left[\frac{q(s_{1:T})}{p(s_1)\prod_{t=2}^{T} p(s_t|s_{t-1})}\right] ds_{1:T}, \\
&= -\int q(s_{1:T})\log\left[\prod_{t=1}^{T} p(x_t|s_t)\right] ds_{1:T} + \int q(s_{1:T})\log\left[q(s_{1:T})\right] ds_{1:T} \\
&\qquad - \int q(s_{1:T})\log\left[p(s_1)\prod_{t=2}^{T} p(s_t|s_{t-1})\right] ds_{1:T}, \\
&= -LL + E - P,
\end{aligned} \tag{11}$$

where LL is the posterior expected log-likelihood (same as Eq. (10)), E is the posterior entropy and P is the posterior expect prior probability. The evaluate the terms in the above equation we factorise the posterior as

$$q(s_{1:T}) = q(s_1)\prod_{t=2}^{T} q(s_t|s_{t-1}) = q(s_1)\prod_{t=2}^{T}\frac{q(s_{t-1}, s_t)}{q(s_{t-1})} = q(s_1)\prod_{t=1}^{T-1}\frac{q(s_t, s_{t+1})}{q(s_t)}. \tag{12}$$

The above factorisation is an assumption of the Baum-Welch algorithm. Let's first look at the entropy term,

$$\begin{aligned} E &= \int q(s_{1:T}) \log \left[q(s_{1:T}) \right] ds_{1:T}, \\ &= \int q(s_{1:T}) \log \left[q(s_1) \prod_{t=1}^{T-1} \frac{q(s_t, s_{t+1})}{q(s_t)} \right] ds_{1:T}, \\ &= \sum_{t=1}^{T-1} \int q(s_t, s_{t+1}) \log q(s_t, s_{t+1}) ds_t ds_{t+1} - \sum_{t=2}^{T-1} \int q(s_t) \log q(s_t) ds_t. \end{aligned} \tag{13}$$

This can be calculated using the marginal posterior, $\gamma(t) = q(s_t)$, and joint posterior, $\xi(t) = q(s_t, s_{t+1})$, provided by the Baum-Welch algorithm:

$$E = \sum_{t=1}^{T-1} \sum_{i,j=1}^{K} \xi_{ij}(t) \log \xi_{ij}(t) - \sum_{t=2}^{T-1} \sum_{i=1}^{K} \gamma_i \log \gamma_i(t). \tag{14}$$

Finally, we calculate the posterior expected prior probability as

$$\begin{aligned} P &= \int q(s_{1:T}) \log \left[p(s_1) \prod_{t=2}^{T} p(s_t | s_{t-1}) \right] ds_{1:T}, \\ &= \int q(s_1) \prod_{t=1}^{T-1} \frac{q(s_t, s_{t+1})}{q(s_t)} \log \left[p(s_1) \prod_{t=2}^{T} p(s_t | s_{t-1}) \right] ds_{1:T}, \\ &= \int q(s_1) \log p(s_1) ds_1 + \sum_{t=1}^{T-1} \int q(s_t, s_{t+1}) \log p(s_{t+1} | s_t) ds_t ds_{t+1}. \end{aligned} \tag{15}$$

Using the marginal and joint posterior provided by the Baum-Welch algorithm and the point estimates for the initial probabilities, π and transition probability matrix, $\mathbf{P}$, this is evaluated as

$$P = \sum_{i=1}^{K} \gamma_i(1) \log \pi_i + \sum_{t=1}^{T-1} \sum_{i,j=1}^{K} \xi_{ij}(t) \log \mathbf{P}_{ij}. \tag{16}$$

References

[1] I. Rezek and S. Roberts, Ensemble hidden Markov models with extended observation densities for biosignal analysis. Probabilistic modeling in bioinformatics and medical informatics. Springer, London, 419-450 (2005).