Supplemental Material: Stochastic variational inference for hidden Markov models

Nicholas J. Foti[†], **Jason Xu**[†], **Dillon Laird**, and **Emily B. Fox**
University of Washington
{nfoti@stat,jasonxu@stat,dillonl2@cs,ebfox@stat}.washington.edu

1 Introduction

In this document we present further details into the how to compute the quantities necessary for the SVIHMM algorithm. We also derive key equations necessary for the analysis of the algorithm, and present and prove the convergence theorem for stochastic gradient ascent using approximate noisy natural gradients. We then present specifics of the synthetic data that we use to evaluate SVIHMM. Last, we discuss the timing experiment in depth.

2 Model specification and variational approximation

Recall our model specification for a hidden Markov model with K latent states, Gaussian emissions $y_t \in \mathbb{R}^p$, and conjugate Dirichlet and normal-inverse-Wishart (NIW) priors on the rows of the transition matrix and emission parameters, respectively. Specifically, let $\alpha \in \mathbb{R}_+^K$, $\mu_0 \in \mathbb{R}^p$, $\Sigma_0 \in \mathbb{S}_{++}^p$ a symmetric positive definite matrix, $\kappa_0 > 0$, and $\nu_0 > p+2$. Then, the model is specified as:

$$\begin{aligned} A_{k:} &\sim \mathrm{Dir}(\alpha), \quad k = 1, \ldots, K \\ \phi_k = (\mu_k, \Sigma_k) &\sim \mathrm{NIW}(\mu_0, \kappa_0, \Sigma_0, \nu_0), \quad k = 1, \ldots, K \\ x_1 | \pi_0 &\sim \mathrm{Mult}(\pi_0) \\ x_t | x_{t-1} &\sim \mathrm{Mult}(A_{x_{t-1}}) \\ y_t | x_t, \{\phi_k\}_{k=1}^K &\sim \mathrm{N}(\mu_{x_t}, \Sigma_{x_t}), \quad t = 1, \ldots, T. \end{aligned} \tag{1}$$

The algorithms presented in the main paper use the natural parameterization of the Dirichlet and NIW distributions which we provide here. The natural parameters of a $\mathrm{Dir}(\alpha)$ distribution are given by $\mathbf{u}^A = \alpha - 1 \in \mathbb{R}^K$. The natural parameters for the $\mathrm{NIW}(\mu_0, \Sigma_0, \kappa_0, \nu_0)$ are denoted $\mathbf{u}^\phi = (u_1^\phi, u_2^\phi, u_3^\phi, u_4^\phi)$ where the components are given by

$$\begin{aligned} u_1^\phi &= \kappa_0 \mu_0 \\ u_2^\phi &= \kappa_0 \\ u_3^\phi &= \Sigma_0 + \kappa_0 \mu_0 \mu_0^T \\ u_4^\phi &= \nu_0 + 2 + p. \end{aligned} \tag{2}$$

In the HMM model in Eq. (1) each row of A is given a $\mathrm{Dir}(\alpha)$ prior so that there is a natural parameter for each row, $\mathbf{u}_k^A \in \mathbb{R}^K$. Similarly, there is a natural parameter corresponding to each emission distribution, $\mathbf{u}_k^\phi, k = 1, \ldots, K$.

Recall from the main paper that we approximate the posterior of Eq. (1) as $p(A, \{\phi_k\}, \mathbf{x}) \approx q(A)q(\{\phi_k\})q(\mathbf{x})$ governed by variational parameters $\mathbf{w}^A$ and $\mathbf{w}^\phi$, respectively, where $q(A)$ is a product of Dirichlet distributions (one per row of A) and $q(\{\phi_k\})$ is a product of NIW distributions

(one per emission distribution). The variational distribution over the local variables, $q(\mathbf{x})$, is represented by a $T \times K$ row stochastic matrix where the entry in row t and column k is $q(x_t = k)$. We describe how to compute $q(\mathbf{x})$ in Sec. 4 of the Supplement.

3 Expected sufficient statistics for a HMM with Gaussian emissions

As shown in the main paper, in order to perform batch VB (Eq. (6)) via coordinate-ascent or SVI (Eq. (14)) via stochastic gradient ascent on the model in Eq. (1), we must be able to compute the *sufficient statistics*, $t(\cdot)$, of the various distributions. In this section we derive the necessary sufficient statistics for the HMM with Gaussian emissions and conjugate priors described above [1].

In the batch setting, the sufficient statistics for the jth row of A are given by the number of transitions from state j to each other state over the entire observation sequence. In particular, the sufficient statistics corresponding to the transition from state j to k are given by:

$$t^A_{jk}(\mathbf{x}) = \sum_{t=2}^{T} \mathbb{1}_{x_{t-1}=j, x_t=k}, \tag{3}$$

where the indicator function $\mathbb{1}_A$ is 1 when event A occurs, and 0 otherwise. Note that the sufficient statistics for the rows of the transition matrix only depend on the latent state sequence and not on the actual observations. We then combine all sufficient statistics for the jth row into the vector of counts $t^A_j(\mathbf{x}) = (t^A_{j1}(\mathbf{x}), \ldots, t^A_{jK}(\mathbf{x}))$. In the main paper we suppress the j notation, however, the update for each row of A uses the sufficient statistics corresponding to that row.

For the SVI case where we only consider a subchain of observations, S, the sufficient statistics for the transition from state j to k is given by:

$$t^A_{jk}(\mathbf{x}) = \sum_{\ell=2}^{L} \mathbb{1}_{x^S_{\ell-1}=j, x^S_\ell=k}. \tag{4}$$

That is, we consider the number of times a transition from state j to k occurs in S ignoring the rest of the observations.

To compute both the batch VB and SVI updates for the emission distributions we need to compute the sufficient statistics of the NIW distribution. Recall that the natural parameterization of the NIW distribution corresponding to emission k is of the form $\mathbf{u}^\phi_k = (u^\phi_{k,1}, u^\phi_{k,2}, u^\phi_{k,3}, u^\phi_{k,4})$. There will be a sufficient statistic corresponding to each entry of $\mathbf{u}^\phi_k$, which in the batch setting are given by:

$$\begin{aligned}
t^\phi_{k,1}(\mathbf{x}, \mathbf{y}) &= \sum_{t=1}^{T} y_t \mathbb{1}_{x_t=k} \\
t^\phi_{k,2}(\mathbf{x}, \mathbf{y}) &= \sum_{t=1}^{T} \mathbb{1}_{x_t=k} \\
t^\phi_{k,3}(\mathbf{x}, \mathbf{y}) &= \sum_{t=1}^{T} y_t y_t' \mathbb{1}_{x_t=k} \\
t^\phi_{k,4}(\mathbf{x}, \mathbf{y}) &= \sum_{t=1}^{T} \mathbb{1}_{x_t=k}.
\end{aligned} \tag{5}$$

These sufficient statistics are identical to those obtained for a NIW prior for independent Gaussian observations since conditioned on the state sequence, $\mathbf{x}$, the observations are independent. As above,

the analogous NIW sufficient statistics for a subchain, S, are given by:

$$\begin{aligned}
t^{\phi}_{k,1}(\mathbf{x}, \mathbf{y}) &= \sum_{\ell=1}^{L} y^S_\ell \mathbb{1}_{x^S_\ell = k} \\
t^{\phi}_{k,2}(\mathbf{x}, \mathbf{y}) &= \sum_{\ell=1}^{L} \mathbb{1}_{x^S_\ell = k} \\
t^{\phi}_{k,3}(\mathbf{x}, \mathbf{y}) &= \sum_{\ell=1}^{L} y^S_\ell (y^S_\ell)' \mathbb{1}_{x_\ell = k} \\
t^{\phi}_{k,4}(\mathbf{x}, \mathbf{y}) &= \sum_{\ell=1}^{L} \mathbb{1}_{x^S_\ell = k}.
\end{aligned} \tag{6}$$

For both the batch VB and SVI algorithms we need to compute the expectations of the sufficient statistics with respect to the variational distribution $q(\mathbf{x})$ which by Eqs. (3) and (5) are given by:

$$\begin{aligned}
\mathbb{E}_{q(\mathbf{x})}[t^{A}_{jk}(\mathbf{x})] &= \sum_{t=2}^{T} q(x_{t-1} = j, x_t = k) \\
\mathbb{E}_{q(\mathbf{x})}[t^{\phi}_{k,1}(\mathbf{x}, \mathbf{y})] &= \sum_{t=1}^{T} y_t q(x_t = k) \\
\mathbb{E}_{q(\mathbf{x})}[t^{\phi}_{k,2}(\mathbf{x}, \mathbf{y})] &= \sum_{t=1}^{T} q(x_t = k) \\
\mathbb{E}_{q(\mathbf{x})}[t^{\phi}_{k,3}(\mathbf{x}, \mathbf{y})] &= \sum_{t=1}^{T} y_t y_t' q(x_t = k) \\
\mathbb{E}_{q(\mathbf{x})}[t^{\phi}_{k,4}(\mathbf{x}, \mathbf{y})] &= \sum_{t=1}^{T} q(x_t = k).
\end{aligned} \tag{7}$$

The expected sufficient statistics for a subchain S are computed analogously, restricting the computations in Eq. (7) to the observations in the subchain. In particular, they are computed as:

$$\begin{aligned}
\mathbb{E}_{q(\mathbf{x})}[t^{A}_{jk}(\mathbf{x})] &= \sum_{\ell=2}^{L} q(x^S_{\ell-1} = j, x^S_\ell = k) \\
\mathbb{E}_{q(\mathbf{x})}[t^{\phi}_{k,1}(\mathbf{x}, \mathbf{y})] &= \sum_{\ell=1}^{L} y_t q(x^S_\ell = k) \\
\mathbb{E}_{q(\mathbf{x})}[t^{\phi}_{k,2}(\mathbf{x}, \mathbf{y})] &= \sum_{\ell=1}^{L} q(x^S_\ell = k) \\
\mathbb{E}_{q(\mathbf{x})}[t^{\phi}_{k,3}(\mathbf{x}, \mathbf{y})] &= \sum_{\ell=1}^{L} y_t y_t' q(x^S_\ell = k) \\
\mathbb{E}_{q(\mathbf{x})}[t^{\phi}_{k,4}(\mathbf{x}, \mathbf{y})] &= \sum_{\ell=1}^{L} q(x^S_\ell = k).
\end{aligned} \tag{8}$$

We can then plug the expected sufficient statistics into Eqs. (6) or (14) in the main paper to determine coordinate-ascent or stochastic gradient updates, respectively. However, in order to compute the expected sufficient statistics in either the coordinate-ascent (batch VB) or stochastic gradient-ascent (SVI) algorithms we must first compute $q(\mathbf{x})$ for batch VB or $q(\mathbf{x}^S)$ for SVI. We describe how to do this in the next section.

4 Forward-backward algorithm for local variational update

The optimal distribution over the local variables, $q^*(\mathbf{x}) = q^*(x_1, \ldots, x_T)$ for batch VB and $q^*(x^S) = q^*_(x_1^S, \ldots, x_L^S)$ for SVI, is needed in order to compute the expected sufficient statistics that appear in the coordinate-ascent and gradient equations for the global parameters. In particular, looking at Eq. (7) we need to be able to compute the *marginal-beliefs* of each hidden state, i.e. $q^*(x_t)$, and the *pairwise-beliefs*, $q^*(x_{t-1}, x_t)$. Following [1] we use the forward-backward algorithm, a dynamic programming algorithm, to determine the marginal- and pairwise-beliefs in time $O(K^2T)$.

Recall Eq. (7) from the main paper which describes the form of the optimal variational distribution for the local parameters:

$$q^*(\mathbf{x}) \propto \exp\left(E_{q(A)}\left[\ln \pi(x_1)\right] + \sum_{t=2}^{T} E_{q(A)}\left[\ln A_{x_{t-1},x_t}\right] + \sum_{t=1}^{T} E_{q(\phi)}\left[\ln p(y_t|x_t)\right] \right). \quad (9)$$

First, we define auxiliary parameters

$$\widetilde{A}_{j,k} := \exp\left[E_{q(A)} \ln(A_{j,k})\right] \quad \widetilde{p}(y_t|x_t = k) := \exp\left[E_{q(\phi)} \ln p(y_t|x_t = k)\right] \quad (10)$$

which we then use in the forward-backward algorithm as follows. Note $\widetilde{A} = (\widetilde{A}_{j,k})$ and $\widetilde{p}(y_t|x_t = k)$ can be loosely interpreted as the expected sufficient statistics of the global parameters. For the HMM defined in Eq. (1) we have that

$$\widetilde{A}_{j,k} = \exp\left[\psi\left(w^A_{jk}\right) - \psi\left(\sum_{l=1}^{K} w^A_{jl}\right)\right], \quad j,k \in 1,\ldots,K \quad (11)$$

where $\psi(\cdot)$ is the digamma function and $\log \widetilde{p}(y_t|x_t = k, \phi)$ is given by the expectation under the NIW variational distribution of the log-probability density of a Gaussian distribution, the details of which can be found in [2](Ch. 10.2.1).

In the batch VB case we use the auxiliary parameters to propagate a set of *forward messages*, $\alpha = (\alpha_{t,k}), t \in 1,\ldots,T, k \in 1,\ldots,K$, starting at $t = 1$ according to:

$$\alpha_{1,k} = \pi_{0,k}, \qquad \alpha_{t,k} = \sum_{j=1}^{K} \alpha_{t-1,j} \widetilde{A}_{j,k} \widetilde{p}(y_t|x_t = k), \quad (12)$$

where $\pi_{0,k} = p(x_1 = k)$ is the initial distribution. We then propagate a set of *backward messages*, $\beta = (\beta_{t,k}), t \in 1,\ldots,T, k \in 1,\ldots,K$, starting at $t = T$ and going backwards as:

$$\beta_{T,k} = 1, \qquad \beta_{t,k} = \sum_{j=1}^{K} \widetilde{A}_{k,j} \widetilde{p}(y_{t+1}|x_{t+1}) \beta_{t+1,k}. \quad (13)$$

The forward messages perform a filtering pass by propagating information forwards in time, while the backwards messages perform a smoothing pass by taking into account the information that future observations provide. The use of the auxiliary parameters is necessary since in Eq. (9) the expectation and logarithm are not interchangeable. For an in depth derivation of the forward and backward recursions see [1].

Given the forward and backward messages we can compute the quantities of $q^*(\mathbf{x})$ necessary for the global step. In particular, the marginal beliefs are given by

$$q^*(x_t = k) \propto \alpha_{t,k} \beta_{t,k} \quad (14)$$

and the pairwise beliefs by

$$q^*(x_{t-1} = j, x_t = k) \propto \alpha_{t-1,j} A_{j,k} p(y_t|x_t = k) \beta_{t,k}. \quad (15)$$

For SVI, the forward-backward algorithm remains largely the same. The major difference is that only observations and local variables in the subchain are considered. The corresponding modifications to the above equations are straight forward. Additionally, since in the SVI setting we cannot learn the initial state distribution, π_0, we initialize the forward messages as $\alpha_{1,k} = \hat{\pi}_k$, where as described in the main paper, $\hat{\pi}$ is the leading eigenvector of $E_{q(A)}[A]$.

5 Batch variational Bayes global udpate

The batch VB global update for the model in Eq. (1) is given by:

$$\begin{aligned} \mathbf{w}^A_{jk} &= \mathbf{u}^A_k + \sum_{t=2}^{T} q(x_{t-1}=j, x_t=k), \quad j,k \in 1,\ldots,K \\ w^\phi_{k,r} &= \mathbf{u}^\phi_{k,r} + \mathbb{E}_{q(\mathbf{x})}[t^\phi_{k,r}(\mathbf{x},\mathbf{y})], \quad k \in 1,\ldots,K,\ r \in 1,\ldots,4 \end{aligned} \tag{16}$$

where the expectations with respect to $q(\mathbf{x})$ are given in Eq. (7) and where quantities of $q(\mathbf{x})$ are computed via the forward-backward algorithm described previously. The index r indexes the sufficient statistics of the emission distributions, of which there are four in the case of the NIW.

6 Stochastic natural gradients for SVIHMM

The natural gradients (Eq. (14) in the main paper) for the model in Eq. (1) are given by:

$$\begin{aligned} \left[\widetilde{\nabla}_{w^A}\mathcal{L}^S\right]_{jk} &= u^A_{jk} + c^A \sum_{\tau=2}^{L} q(x^S_{\tau-1}=j, x^S_\tau=k) - w^A_{jk} \\ \left[\widetilde{\nabla}_{w^{\phi_k}}\mathcal{L}^S\right]_r &= u^\phi_r + c^\phi_r \sum_{\tau=1}^{L} E_{q(x^S)}[t^\phi_{k,r}(x^S,y^S)] - w^\phi_{k,r}, \quad k \in 1,\ldots,K,\ r \in 1,\ldots,4,. \end{aligned} \tag{17}$$

Quantities involving $q(\mathbf{x}^S)$ are computed using the forward-backward algorithm in Sec. 4 and the expected sufficient statistics are derived in Sec. 3. The gradients in Eq. (17) are then used in a Robbins-Monro averaging procedure to update the global variational parameters.

7 Batch factor

As described in Sec. 3.2 of the main paper, in order to obtain an unbiased estimate of the natural gradient of the $\mathcal{L}$ (Eq. (12) in the main paper) we must scale the terms of $\mathcal{L}^S$ to match the size of the original data set. Here we derive Eq. (15) from the main paper which allows us to read off the necessary factors to scale the natural gradient. As in the paper, we assume that a subchain, S, of length L is sampled according to $p(S) = \frac{1}{T-L+1}$ which results in:

$$\begin{aligned} E_S\left[E_q \ln p(\mathbf{y}^S,\mathbf{x}^S|\theta)\right] &= \frac{1}{T-L+1} E_q\Bigg[\ln \pi(x_1) + \sum_{t=2}^{L} \ln A_{x_{t-1},x_t} + \sum_{t=1}^{L} \ln p(y_t|x_t) \\ &\quad + \ln \pi(x_2) + \sum_{t=3}^{L+1} \ln A_{x_{t-1},x_t} + \sum_{t=2}^{L+1} \ln p(y_t|x_t) + \ldots \\ &\quad + \ln \pi(x_{T-L+1}) + \sum_{t=T-L+2}^{T} \ln A_{x_{t-1},x_t} + \sum_{t=T-L+1}^{T} \ln p(y_t|x_t)\Bigg] \\ &\approx \frac{1}{T-L+1} E_q\Bigg[\sum_{t=1}^{T-L+1} \ln \pi(x_t) + (L-1)\sum_{t=2}^{T} \ln A_{x_{t-1},x_t} + L\sum_{t=1}^{T} p(y_t|x_t)\Bigg]. \end{aligned} \tag{18}$$

The approximation arises because the observations near the endpoints of the observation sequence appear in fewer subchains than those in the middle of the sequence, e.g. x_1 and x_T only appear in one subchain. However, the error introduced from this approximation becomes negligible as the length of the sequence increases which is the case we are interested in. From Eq. (18) we can read off the batch factors as $\mathbf{c} = (c^A, c^\phi)$, where $c^A = (T-L+1)/(L-1)$, and $c^\phi = (T-L+1)/L$. More general choices for $p(S)$ may be used resulting in different batch factors.

8 Preservation of ascent direction with approximate local messages

Theorem 1. *If the noisy gradient with respect to the "true" messages*

$$\hat{\nabla}_{\mathbf{w}}\mathcal{L}^S = \mathbf{u} + E_{q^*}\left[\mathbf{c}^T t(\mathbf{x}^S, \mathbf{y}^S)\right] - \mathbf{w}$$

lies in the same half plane as the noisy gradient with respect to approximate messages

$$\bar{\nabla}_{\mathbf{w}}\mathcal{L}^S = \mathbf{u} + E_{q_\epsilon}\left[\mathbf{c}^T t(\mathbf{x}^S, \mathbf{y}^S)\right] - \mathbf{w},$$

then $\bar{\nabla}_{\mathbf{w}}\mathcal{L}^S$ is an ascent direction for $\mathcal{L}$ so that SVIHMM will converge to a local maximum of the ELBO [3, 4]. To ensure the gradients are in the same half-plane, it suffices to choose

$$\epsilon \leq \frac{M^S(\mathbf{w})}{||\mathbf{c}^T t(\mathbf{x}, \mathbf{y})||_2},$$

where

$$M^S(\mathbf{w}) := \max\left(\left|\left|\hat{\nabla}_{\mathbf{w}}\mathcal{L}^S\right|\right|_2, ||\bar{\nabla}_{\mathbf{w}}\mathcal{L}^S||_2\right)$$

Proof. Let $\mathbf{y}^S = (y_1^S, \ldots, y_L^S)$ be a subchain of observations where $L << T$ and $\mathbf{x} = (x_1, \ldots, x_L)$ denote any configuration of latent states corresponding to $\mathbf{y}^S$. Also assume we have an approximation $q_\epsilon(\mathbf{x})$ such that

$$\max_{\mathbf{x}} |q_\epsilon(\mathbf{x}) - q^*(\mathbf{x})| < \epsilon$$

where $q^*(\mathbf{x})$ again denotes the "true" distribution as if a full message pass were performed on the entire dataset of length T. In our setting, q^* is a discrete distribution (of dimension $K \times L$) over the latent state sequence, and t is some d-dimensional sufficient statistic function that we assume is bounded. The proof follows analogously in the continuous case as long as q^* and q_ϵ are absolutely continuous with respect to the same measure– one simply substitutes the summations over $\mathbf{x}$ below with integration.

To show that $\bar{\nabla}_{\mathbf{w}}\mathcal{L}^S$ lies in the same half-plane as $\hat{\nabla}_{\mathbf{w}}\mathcal{L}^S$, it is sufficient that

$$\left|\left|\hat{\nabla}_{\mathbf{w}}\mathcal{L}^S - \bar{\nabla}_{\mathbf{w}}\mathcal{L}^S\right|\right|_2 < \max\left(\left|\left|\hat{\nabla}_{\mathbf{w}}\mathcal{L}^S\right|\right|_2, ||\bar{\nabla}_{\mathbf{w}}\mathcal{L}^S||_2\right) \equiv M^S(\mathbf{w}).$$

Since $\mathbf{w}$ and $\mathbf{u}$ are independent of $q^*(\mathbf{x})$, we may translate the gradient vectors by $\mathbf{u} - \mathbf{w}$ and equivalently seek to show that

$$\left|\left|E_{q_\epsilon}[\mathbf{c}^T t(\mathbf{x}, \mathbf{y})] - E_{q^*}[\mathbf{c}^T t(\mathbf{x}, \mathbf{y})]\right|\right|_2 < M^S(\mathbf{w}).$$

Considering the difference component-wise, we have

$$\begin{aligned}
\left|\left|E_{q_\epsilon}[\mathbf{c}^T t(\mathbf{x}, \mathbf{y})] - E_{q^*}[\mathbf{c}^T t(\mathbf{x}, \mathbf{y})]\right|\right|_2^2 &= \sum_{j=1}^{d}\left(c_j \sum_{\mathbf{x}} t_j(\mathbf{x}, \mathbf{y}) q_\epsilon(\mathbf{x}) - c_j \sum_{\mathbf{x}} t_j(\mathbf{x}, \mathbf{y}) q^*(\mathbf{x})\right)^2 \\
&= \sum_{j=1}^{d}\left(c_j \sum_{\mathbf{x}} t_j(\mathbf{x}, \mathbf{y})(q_\epsilon(\mathbf{x}) - q^*(\mathbf{x}))\right)^2 \\
&\leq \sum_{j=1}^{d}\left(c_j \sum_{\mathbf{x}} |t_j(\mathbf{x}, \mathbf{y})||q_\epsilon(\mathbf{x}) - q^*(\mathbf{x})|\right)^2 \\
&\leq \epsilon^2 \sum_{j=1}^{d}\left(c_j \sum_{\mathbf{x}} |t_j(\mathbf{x}, \mathbf{y})|\right)^2 = \epsilon^2 \left|\left|\mathbf{c}^T t(\mathbf{x}, \mathbf{y})\right|\right|_2^2.
\end{aligned}$$

Finally, since we want this quantity to be bounded above by $M^S(\mathbf{w})^2$, we choose

$$\epsilon \leq \frac{M^S(\mathbf{w})}{||\mathbf{c}^T t(\mathbf{x}, \mathbf{y})||_2}$$

□

As one would expect, ascent direction is preserved in the limit as $\epsilon \to 0$ as long as $t(\cdot,\cdot)$ is a bounded sufficient statistic. Also, we note that while the upper bound is not easy to evaluate to guide our choice of ϵ since true messages are unavailable, we show empirically that setting small values $\epsilon = 1 \times 10^{-6}$ in `GrowBuf` leads to noticeable performance gains empirically in the experiments section.

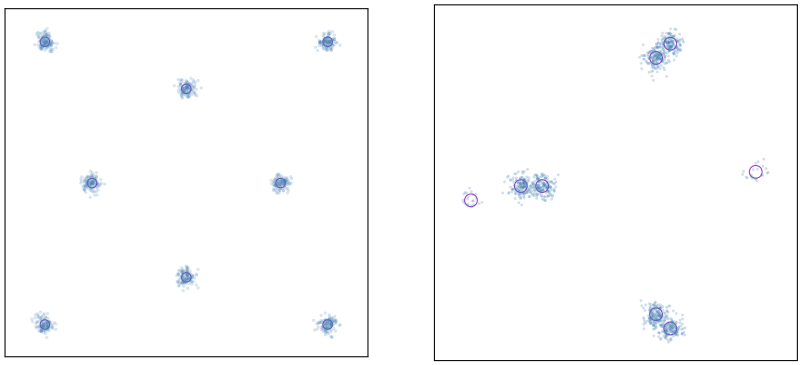

Figure 1: Observations generated from the diagonally dominant (left) and reversed cycles (right) examples. Ellipses indicate true covariance matrices of underlying components.

9 Synthetic data sets

In this section we present the *diagonally dominant* and *reversed cycles* synthetic data sets in detail.

The diagonally dominant data set uses the following transition matrix:

$$A = \begin{pmatrix} .999 & .001 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & .999 & .001 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & .999 & .001 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & .999 & .001 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .999 & .001 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .999 & .001 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & .999 & .001 \\ .001 & 0 & 0 & 0 & 0 & 0 & 0 & .999 \end{pmatrix}.$$

We see that there is a large probability that the observation sequence remains in the same state. The component means are given by

$$\boldsymbol{\mu} = \{(0,20);(20,0);(-90,-30);(30,-30);(-20,0);(0,-20);(30,30);(-30,30)\},$$

where all component covariances are given by the 2×2 identity matrix, I_2. The emission distributions and simulated data are depicted in Fig. 1 (left) and are meant to be highly identifiable so that learning is largely likelihood-dominated. This illustrates the importance of sampling disparate sections of the observation sequence in order for the global updates to contain sufficient information to obtain accurate estimates.

The reversed cycles data set consists of two 3-state cycles with essentially deterministic dynamics. The two cycles are connected by two bridge states that the process visits rarely to switch between the cycles. The state dynamics correspond to the following transition matrix:

$$A = \begin{pmatrix} .01 & .99 & 0 & 0 & 0 & 0 & 0 & 0 \\ 9 & .01 & .99 & 0 & 0 & 0 & 0 & 0 \\ .85 & 0 & 0 & .15 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .01 & .99 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .01 & .99 & 0 \\ 0 & 0 & 0 & 0 & .85 & 0 & 0 & .15 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

The emission means are set to

$$\boldsymbol{\mu} = \{(-50, 0); (30, -30); (30, 30); (-100, -10); (40, -40); (-65, 0); (40, 40); (100, 10)\},$$

with covariance matrices given by $20 * I_2$. Observations generated from this model and the emission distributions are shown in Fig. 1 (right). The means of emissions 1 and 5, states 2 and 6, and states 3 and 7 have indistinguishable means, but the cycles $1 \to 2 \to 3$ and $5 \to 6 \to 7$ visit the means in reverse orders. The emission means of the bridge states are far from the two cycles so that they are identifiable. Learning the transition dynamics in this case is key in order to learn the overlapping emissions.

10 Discussion of timing experiment

Here we explain our choice of settings for the timing comparison between SVIHMM and batch VB in Sec. 4 of the main paper. We implemented both the SVIHMM and batch VB algorithms in Python except that the forward-backward algorithm was written in C++. Additionally, since SVIHMM operates on shorter sequences than batch VB it does not benefit as much from the optimized forward-backward algorithm. The gradient computations for SVIHMM were not optimized and are subject to Python overhead, however, the coordinate-ascent update for bath VB are vectorized using Numpy. Therefore, in order to compare the batch VB and SVIHMM algorithms fairly we set $T = 3$ million and $M = 1$ as increasing M results in higher overhead due to the interpreted nature of Python which could be mitigated in C++. Since M is small, L must be chosen relatively large in order to obtain informative gradients. For large L the `growBuf` routine negligibly affects the predictive log-likelihood and the running time of the algorithm since the length of the subchain causes the message error to be small and thus few observations are added as a buffer.

References

[1] M. J. Beale. *Variational Algorithms for Approximate Bayesian Inference*. Ph.D. thesis, University College London, 2003.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer Verlag, 2006.

[3] J. Nocedal and S. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, 2006.

[4] B. T. Polyak and Y. Tsypkin. Pseudo-gradient adaptation and learning algorithms. *Automatics and Telemechanics*, 3:45–68, 1973.