Supplementary material

Konstantina Palla, David A. Knowles, Zoubin Ghahramani

May 12, 2014

1 Introduction

This paper contains supplementrary material to the main paper "*A reversible infinite HMM using normalised random measures*".

2 Relation to the Hierarchical Gamma and Dirichlet processes

This section expands on the analysis of Section 3 in the original manuscript regarding the relation of the SHGP to Hierarchical Gamma (HGP) and Hierarchical Dirichlet processes (HDP) as seen in Table 1. The SHGP is a prior over the weight matrix J as opposed to the HDP which is a prior over the transition matrix P. This difference is crucial, since it allows for direct manipulation of the weights, enabling us to enforce symmetry and thereby make the Markov chain reversible. The SHGP can be viewed as a HGP where symmetry is imposed on the produced weight matrix J. However, there are subtle differences in the construction of the weight matrix. Looking at the Table 1, both processes, the HGP and SHGP, use the Gamma process in a hierarchical way. The HGP constructs each row j in the weight matrix by sampling from the same Gamma process $\Gamma P(\tilde{\alpha}, G_0), \forall j$, as opposed to the SHGP where each row is sampled by a Gamma process with a different shape parameter dependent on the corresponding base weight w_j. This is a modelling choice and by choosing the base measure μ of the weight matrix to be the G_0 rather than the product $G_0 \times G_0$ the SHGP (with no symmetrization imposed) becomes identical to the HGP.

Table 1: HDP, HGP and SHGP

HDP	HGP	SHGP
$G_0' \sim \mathrm{DP}(\alpha_0 \mu_0)$	$G_0 \sim \Gamma P(\alpha_0, \mu_0)$	$G_0 \sim \Gamma P(\alpha_0, \mu_0)$
$P_j \sim \mathrm{DP}(\alpha' G_0')$	$\tilde{J}_j \sim \Gamma P(\tilde{\alpha}, G_0)$	$J_j \sim \Gamma P(\alpha w_j, G_0)$

The HGP itself is closely related to the HDP. With an appropriate choice of mixing measure over the second level concentration parameter α', the HDP is equal to the normalised HGP. This follows from the Dirichlet process being representable as a normalised underlying Gamma process. To obtain this equivalence we require (using the notation of Table 1)

$$\alpha' G_0'(\mathcal{X}) \stackrel{d}{=} \tilde{\alpha} G_0(\mathcal{X}) \Leftrightarrow \alpha' \stackrel{d}{=} \tilde{\alpha} \frac{G_0(\mathcal{X})}{G_0'(\mathcal{X})} \Rightarrow \alpha' \stackrel{d}{=} \tilde{\alpha} G_0(\mathcal{X})$$

$$\alpha' \sim \text{Gamma}(\alpha_0 \mu_0(\mathcal{X}), \frac{\alpha_0}{\tilde{\alpha}})$$

$$G_0(\mathcal{X}) \sim \text{Gamma}(\alpha_0 \mu_0(\mathcal{X}), \alpha_0), \tag{1}$$

where we used the fact that $G_0'(\mathcal{X}) = 1$ a.s., since $G_0'(\mathcal{X})$ is a sample from a DP. By the independence of $G_0/G_(\mathcal{X})$ and $G_0(\mathcal{X})$ we have that $\alpha' G_0' \stackrel{d}{=} \tilde{\alpha} G_0$. Finally $P_j \stackrel{d}{=} \tilde{J}_j / \tilde{J}_j(\mathcal{X})$ by a second application of the equivalence between the DP and a normalised ΓP. In other words, an HGP with normalised weights in order to produce the transition matrix P, is equivalent to a HDP in which the shape parameter α' is sampled as in Equation 1.

In addition, we can now see that the SHGP modulo symmetrisation is equivalent to an HDP where a different second level concentration parameter is sampled for each row of P, i.e.

$$\alpha_j' \sim \text{Gamma}(\alpha_0 \mu_0(\mathcal{X}), \frac{\alpha_0}{\alpha \omega_j}) \tag{2}$$

$$P_j \sim \text{DP}(\alpha_j' G_0') \tag{3}$$

3 Observed data Y

In hidden Markov models, the observations Y are supposed to be generated from a Markov chain of hidden states X. In other words, the Y consist of emissions as a result of the system jumping through hidden states. The way the hidden sequence determines the emissions is defined by the emission matrix E which connects the current state with the emission. The emission matrix E can take different forms, based on the dataset at hand, resulting in different forms of outputs Y. In this work, we considered three different forms: multinomial, multivariate Poisson and univariate Gaussian. In what follows, the number of hidden states is K, while the observations Y and hidden sequence X have length T.

Multinomial. In this case, the observation sequence Y is a T-length vector that consists of symbols. The process jumps from state to state based on the transition matrix and emits a symbol among L possible ones with probability defined by the $K \times L$ emission matrix E. $E(k, l)$ is the probability of emitting symbol l while being in state k and the random variable Y_t conditioned on X_t

has a L-multinomial distribution. The likelihood is given by:

$$p(Y|X,E) = \prod_{t=1}^{T} E(X_t, Y_t) = \prod_{kl} E(k,l)^{m_{kl}} \tag{4}$$

where $k, l \in \mathcal{S}$ and m_{kl} is the number of times that state k emitted symbol l. This choice of likelihood allows the conjugate Dirichlet prior, $\mathrm{Dir}(q_1, \ldots, q_K)$ to be used. We set $q_k = 1 \forall k \in [1, \ldots, K]$.

Multivariate Poisson. Here we define a multivariate Poisson hidden Markov model (PHMM). In a PHMM we consider Y to be a matrix of discrete observations (counts) and of $L \times T$ dimensions. The random variable Y_t is a vector of length L and conditioned on X_t each element Y_{lt} has a Poisson distribution with rate parameter $E(X_t, l)$, where E is a $K \times L$ matrix. The likelihood is given by

$$p(Y|X,E) = \prod_{t=1}^{T} f(Y_t; E(X_t, :)) = \prod_{t=1}^{T} \prod_{l=1}^{L} \frac{E(X_t,l)^{Y_t(l)} e^{-E(X_t,l)}}{Y_t(l)!} \tag{5}$$

where f is the Poisson p.m.f. We put a gamma prior $Gamma(\alpha_\lambda, \beta_\lambda)$ over each element of E, where α_λ and β_λ are the shape and rate hyper parameters. For the chipSeq experiment we set $\alpha_\lambda = \beta_\lambda = 1$.

Gaussian. We consider Y to be a sequence of real observations, that is $Y_t \in \mathbb{R}$. Each random variable Y_t has a Gaussian distribution with mean $\mu = E(X_t, 1)$ and standard deviation $\sigma = E(X_t, 2)$. The Gaussian distribution the random variable Y_t is sampled from depends on the current state X_t. The emission matrix E is a $K \times 2$ matrix that stores the hyper parameters that define the corresponding Gaussian distributions. The likelihood is given by

$$p(Y|X,E) = \prod_{t=1}^{T} \mathcal{N}(Y_t; E(X_t,1), E(X_t,2)) = \prod_{t=1}^{T} \frac{1}{E(X_t,2)\sqrt{2\pi}} e^{-\frac{(Y_t - E(X_t,1)^2)}{2E(X_t,2)^2}} \tag{6}$$

We set an normal-inverse-gamma prior over each of the K pairs of means and standard deviations, that is $(\mu, \sigma^2) \sim N\Gamma^{-1}(\mu_0, k_0, a_0, b_0)$. In the Alamethecin dataset we set $\mu_0 = 0, k_0 = a_0 = b_0 = 1$.

4 Prediction

A principled way to evaluate a generative model is by its ability to predict missing data values given some observations. For SHGP we collect M samples from the posterior $\{\{E^{(1)}, X^{(1)}\}, \ldots, \{E^{(M)}, X^{(M)}\}\}$ and estimate the predictive distribution of a missing entry in the dataset Y as the average of the predictive distributions for each of the collected samples. For the experiments we ran, we

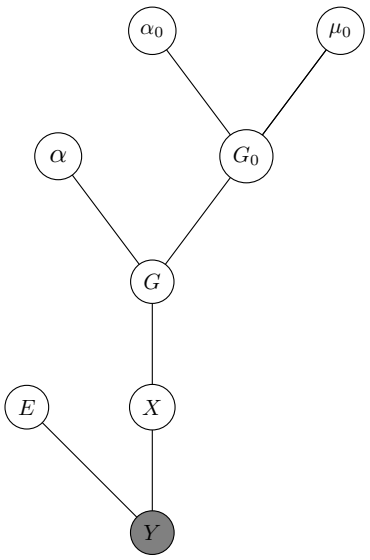

Figure 1: Graphical model for SHGP

used two different likelihoods, a Poisson and a Gaussian. For the Poisson model the approximate predictive distribution is

$$P(Y_{lt}|Y_{train}) \approx \frac{1}{M}\sum_{m=1}^{M}\frac{E(X_t^{(m)},l)^{Y_t(l)} - e^{E(X_t^{(m)},l)}}{Y_t(l)!},$$

while for the Gaussian is

$$P(Y_t|Y_{train}) \approx \frac{1}{M}\sum_{m=1}^{M}\frac{1}{E(X_t^{(m)},2)\sqrt{2\pi}}e^{-\frac{(Y_t - E(X_t^{(m)},1)^2)}{2E(X_t^{(m)},2)^2}}$$

5 Inference

This section describes the sampling steps for the SHGP finite model (see Figure 1).

Sampling the conentration parameters, α_0 and α. We used slice sampling to infer the parameters $\alpha 0$ and α using Gamma priors $\alpha_0 \sim \text{Gamma}(s_0, r_0)$ and $\alpha \sim \text{Gamma}(s, r)$, where $\{s_0, r_0\}$ and $\{s, r\}$ are the pairs of shape and rate parameters for α_0 and α respectively. The posterior distributions are:

$$\begin{aligned} p(\alpha_0|G_0, \mu_0) &\propto p(G_0|\alpha_0, \mu_0)p(\alpha_0) \\ p(\alpha|G, G_0) &\propto p(G|\alpha, G_0)p(\alpha) \end{aligned} \tag{7}$$

The likelihood terms expand to

$$p(G_0|\alpha_0,\mu_0) = \prod_{k=1}^{K} \text{Gamma}(w_k|\alpha_0\mu_k)$$

$$p(G|\alpha,G_0) = \prod_{i=1,j=m}^{K} \text{Gamma}(J_{ij}|\alpha w_i w_j), \tag{8}$$

where $m = 1$ in the irreversible case and $m = i$ in the reversible case. Moreover, μ_k is the mass assigned by the base measure μ_0 to each one of the K atoms. Here we assumed that $\mu(\mathcal{X}) = 1$ and $\mu_k = \frac{\mu(\mathcal{X})}{K}$.

Sampling the weight vector, G_0 The vector G_0 is the vector of the base weights $G_0 = [w_1, \ldots, w_K]$ in the corresponding random measure $G_0 = \sum_k w_k \delta_{x_k}$. Looking at the graphical model, the posterior over w_k is:

$$p(w_k|G_{0\neg k}, G, \mu_0, \alpha, \alpha_0) \propto p(G|G_0,\alpha)p(w_k|\alpha_0,\mu_0)$$

where $G_{0\neg k}$ is the vector of the base weights excluding the weight w_k. The likelihood term $p(G|G_0,\alpha)$ is given in Equation 7 and the prior is $p(w_k|\alpha_0,\mu_0) = \text{Gamma}(w_k|\alpha_0\mu_k)$. We used slice sampler to sample each weight w_k.

Sampling the weight matrix, G The weight matrix G contains the edge weights $\{J_{ij}\}$. The posterior over the whole matrix is

$$p(G|X,\alpha,G_0) \propto p(X|G)p(G|\alpha,G_0)$$

We used hybrid Monte Carlo (Neal, 2011) to sample the elements of the matrix G at once instead of sampling each element at a time using slice sampling. In our implementation, we consider G to be a K^2 vector containing all the weights (elements) of the weight matrix. In the reversible case the vector G is of length $\frac{K(K+1)}{2}$ since symmetry is imposed. Hybrid Monte Carlo is a Metropolis method, applicable to continuous state spaces, that makes use of gradient information to reduce random walk behaviour. The aim is to sample from the posterior distribution $p(G|X,\alpha,G_0)$. Two terms are introduced in HMC; the potential energy $\mathcal{E}(G) = -\log p(G|X,\alpha,G_0) + C$ and the momentum, an auxiliary vector of the same length as vector G. The momentum will be changed from iteration to iteration, and within iterations it will change as the G configuration explores the parameter space. The momentum vector changes over the course of an iteration, according to the gradient of the potential energy (or, equivalently, the log posterior). So, we need to be able to evaluate the vector of partial derivatives of the log posterior $\log p(G|X,\alpha,G_0)$. Before this, we use the change of variables $r_{ij} = \log(J_{ij})$ and the prior over the weights is:

$$p_r(r_{ij}) = p_J(J_{ij})J_{ij} \Rightarrow p(R) = \begin{cases} \prod_{i,j\geq i} p_J(e^{r_{ij}})e^{r_{ij}}, & G \text{ symmetric} \\ \prod_{ij} p_J(e^{r_{ij}}e^{r_{ij}}, & \text{if otherwise .} \end{cases}$$

The prior over each of the weights J_{ij} is $p_J(J_{ij}) = \text{Gamma}(\alpha w_i w_j, \alpha)$ and as such the joint log prior over all the variables r_{ij} is:

$$\log p(R) = \begin{cases} \sum_{i,j \geq i} \alpha w_i w_j r_{ij} - \sum_{i,j \geq i} \alpha e^{r_{ij}}, & \text{if } G \text{ symmetric} \\ \sum_{i,j} \alpha w_i w_j r_{ij} - \sum_{i,j} \alpha e^{r_{ij}}, & \text{otherwise.} \end{cases} \tag{9}$$

where we have omitted the terms that do not depend on r_{ij}. The log likelihood is given by:

$$\begin{aligned} \mathcal{L}(R) = \log \prod_{ij} P_{ij}^{n_{ij}} &= \sum_{ij} n_{ij} \log P_{ij} \\ &= \sum_{ij} n_{ij} \log \frac{J_{ij}}{\sum_k J_{ik}} = \sum_{ij} n_{ij} \log \frac{e^{r_{ij}}}{\sum_k e^{r_{ik}}} \end{aligned} \tag{10}$$

where P_{ij} is the transition probability from i to j. The potential energy can now be written as:

$$\mathcal{E}(R) = -\log p(R|X, \alpha, G_0) = -\mathcal{L}(\mathcal{R}) - \log p(R) \tag{11}$$

where we have omitted the conditional dependence on α and G_0 for simplicity. Using Equations 9 and 10, Equation 11 can be rewritten as follows:

$$\mathcal{E}(R) = \begin{cases} \sum_i \big(-\alpha w_i^2 r_{ii} + \alpha e^{r_{ii}} - n_{ii}(r_{ii} - \log \sum_k e^{r_{ik}}) \\ + \sum_{j>i} (-\alpha w_i w_j r_{ij} + \alpha e^{r_{ij}} - n_{ij}(r_{ij} - \log \sum_k e^{r_{ik}})\big), & \text{if } G \text{ symmetric} \\ \sum_{i,j} \big(-\alpha w_i w_j r_{ij} + \alpha e^{r_{ij}} \\ - n_{ij}(r_{ij} - \log \sum_k e^{r_{ik}}) - n_{ji}(r_{ji} - \log \sum_k e^{r_{jk}})\big), & \text{otherwise} \end{cases} \tag{12}$$

where we have omitted the terms that do not depend on r_{ij}'s. Calculating the derivatives of the energy with respect to each r_{ij} is now straightforward and for the symmetric case is:

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}r_{st}} = \begin{cases} -\alpha w_s w_t + \alpha e^{r_{st}} - n_{st} - n_{ts} + \sum_j n_{sj} \sigma_t(r_{s:}) + \sum_i n_{ti} \sigma_s(y_{t:}) & \text{if } s \neq t \\ -\alpha w_s^2 + \alpha e^{r_{ss}} - n_{ss} + \sum_j n_{sj} \sigma_s(r_{s:}) & \text{if otherwise.} \end{cases} \tag{13}$$

where we have used that $\frac{\mathrm{d}(\log \sum_k e^{r_{ik}})}{\mathrm{d}r_{ij}} = \sigma_j(r_{i:}) := \frac{e^{r_{ij}}}{\sum_k e^{r_{ik}}}$.

Sampling the state sequence X. We use the forward-backward algorithm to sample the latent state sequence X given the current state of all other variables in the model. This is a dynamic programming algorithm that efficiently computes the state posteriors over all the hidden state variables X_t.

Sampling the emission matrix E The posterior over the emission matrix is

$$p(E|Y,X) \propto p(Y|E,X)p(E)$$

The explicit form of the posterior depends on the output, the observed Y, that is multinomial, Poisson or Gaussian. In all cases, due to conjugacy, the emission matrix is sampled exactly. In particular:

- Multinomial output:

$$\begin{aligned} \text{prior: } p(E) &= Dir(q_1,\ldots,q_K),\ q_k = 1 \forall k \in [1,\ldots,K] \\ \text{posterior : } p(E_k|Y) &= Dir(q'_1,\ldots,q'_K),\ q'_k = q_k + em_{kl} \end{aligned}$$

  where E_k refers to the k-th row of the $k \times L$ matrix E and em_{kl} is the number of times the system was in state K and emitted symbol l

- Poisson output:

$$\begin{aligned} \text{prior: } p(E) &= \text{Gamma}(\alpha_\lambda, \beta_\lambda) \\ \text{posterior : } p(E_{kl}|Y) &= \text{Gamma}(\alpha_\lambda + em_k, \beta_\lambda + c_{kl}) \end{aligned}$$

  where em_k is the number of times the system was in state K and c_{kl} is the sum of the elements of Y that correspond to state k and row l.

- Gaussian output:

$$\begin{aligned} \text{prior: } p(E_k) = p(\mu,\sigma^2) &\sim N\Gamma^{-1}(\mu_0, k0, a0, b0) \\ \text{posterior : } p(E_k|Y) &= N\Gamma^{-1}(\mu'_0, k0', a0', b0') \end{aligned}$$

  where E_k is a 2-element row for each state k in the $K \times 2$ matrix E constaing the mean and the standard deviation for each state.

6 Joint distribution tests

We give evidence for the correctness of our algorithm using the joint distribution testing methodology of Geweke (2004). There are two ways to sample from the joint distribution, $P(Y,\theta)$ over parameters, $\theta = \{G_0, G, X, E, \alpha, \alpha_0\}$ and data, Y defined by a probabilistic model such as SHGP. The first we will refer to as "marginal-conditional" sampling, shown in Algorithm 1. Both steps here are straightforward: sampling from the prior followed by sampling from the likelihood model. The second way, referred to as "successive-conditional" sampling, is shown in Algorithm 2, where Q represents a single (or multiple) iteration(s) of our MCMC sampler. To validate our sampler we can then check, either informally or using hypothesis tests, whether the samples drawn from the joint $P(Y,\theta)$ in these two different ways appear to have come from the same distribution. We apply this method to our SHGP sampler with chain length $T = 200$. We draw 10^4 samples using both the marginal-conditional and

Algorithm 1 Marginal conditional

1: **for** $m = 1$ to M **do**
2: $\quad \theta^{(m)} \sim P(\theta)$
3: $\quad Y^{(m)} \sim P(Y|\theta^{(m)})$
4: **end for**

Algorithm 2 Successive conditional

1: $\theta^{(1)} \sim P(\theta)$
2: $Y^{(1)} \sim P(Y|\theta^{(1)})$
3: **for** $m = 2$ to M **do**
4: $\quad \theta^{(m)} \sim Q(\theta|\theta^{(m-1)}, Y^{(m-1)})$
5: $\quad Y^{(m)} \sim P(Y|\theta^{(m)})$
6: **end for**

successive-conditional procedures and look at various characteristics of the samples including α_0, α, the base weight vector W and the weight matrix J. The distribution of the number of features under the successive-conditional sampler matches that under the marginal-conditional sampler almost perfectly as shown in Figure 6.The histogram plots show the similarity of the two distributions. Under the successive-conditional sampler the average value of α 3.01 while under the marginal-conditional is 3.02 with standard deviations 0.99 and 1.00 respectively. For the mean value of the edge weights J_{ij}'s, the successive conditional sampler gave 0.95 with standard deviation 0.43, while the marginal conditional 0.93 with standard deviation 0.42 : a hypothesis test for both parameters did not reject the null hypothesis that the means of the two distributions are equal. While this cannot completely guarantee correctness of the algorithm and code, 10^4 samples is a large number for such a small model and thus provides strong evidence that our algorithm is correct.

References

Geweke, J. (2004). Getting it right. *JASA*.

Neal, R. (2011). *MCMC Using Hamiltonian Dynamics*. CRC Press.

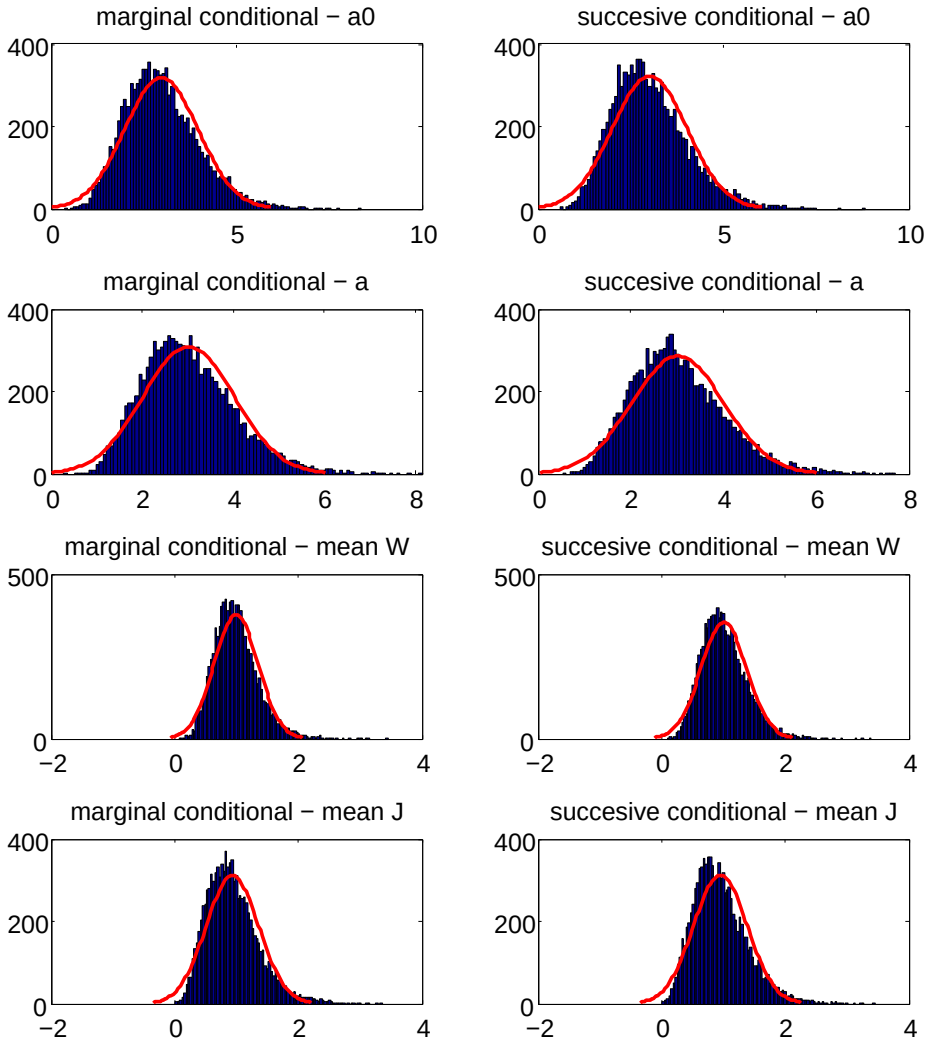

Figure 2: Geweke plots for SHGP using simple Hybrid Monte Carlo