Biometrika (2016), **103**, 2, *pp.* 319–335
Printed in Great Britain

doi: 10.1093/biomet/asw005
Advance Access publication 6 May 2016

Data augmentation for models based on rejection sampling

By VINAYAK RAO

Department of Statistics, Purdue University, West Lafayette, Indiana 47907, U.S.A.
varao@purdue.edu

LIZHEN LIN

Department of Statistics and Data Science, The University of Texas, Austin, Texas 78712, U.S.A.
lizhen.lin@austin.utexas.edu

AND DAVID B. DUNSON

Department of Statistical Science, Duke University, Durham, North Carolina 27708, U.S.A.
dunson@duke.edu

Summary

We present a data augmentation scheme to perform Markov chain Monte Carlo inference for models where data generation involves a rejection sampling algorithm. Our idea is a simple scheme to instantiate the rejected proposals preceding each data point. The resulting joint probability over observed and rejected variables can be much simpler than the marginal distribution over the observed variables, which often involves intractable integrals. We consider three problems: modelling flow-cytometry measurements subject to truncation; the Bayesian analysis of the matrix Langevin distribution on the Stiefel manifold; and Bayesian inference for a nonparametric Gaussian process density model. The latter two are instances of doubly-intractable Markov chain Monte Carlo problems, where evaluating the likelihood is intractable. Our experiments demonstrate superior performance over state-of-the-art sampling algorithms for such problems.

Some key words: Bayesian inference; Density estimation; Gaussian process; Intractable likelihood; Markov chain Monte Carlo; Matrix Langevin distribution; Rejection sampling; Truncation.

1. Introduction

Rejection sampling allows sampling from a probability density $p(x)$ by constructing an upper bound to $p(x)$, and accepting or rejecting samples from a density proportional to the bounding envelope. The envelope is usually much simpler than $p(x)$, with the number of rejections determined by how closely it matches the true density.

In typical applications, the probability density of interest is indexed by a parameter θ, and we write it as $p(x \mid \theta)$. A Bayesian analysis places a prior on θ, and, given observations from the likelihood $p(x \mid \theta)$, studies the posterior over θ. An intractable likelihood, often with a normalization constant depending on θ, precludes straightforward Markov chain Monte Carlo inference over θ: calculating a Metropolis–Hastings acceptance probability involves evaluating the ratio of two such likelihoods, and is itself intractable. This class of problems is called doubly-intractable (Murray et al., 2006), and existing approaches require the ability to draw exact samples from $p(x \mid \theta)$, or to obtain positive unbiased estimates of $p(x \mid \theta)$.

We describe an approach that is applicable when $p(x \mid \theta)$ has an associated rejection sampling algorithm. Our idea is to instantiate the rejected proposals preceding each observation, resulting in an augmented state-space on which we run a Markov chain. Including the rejected proposals can eliminate any intractable terms, and allows the application of standard techniques (Adams et al., 2009). We show that, conditioned on the observations, it is straightforward to independently sample the number and values of the rejected proposals: this just requires running the rejection sampler to generate as many acceptances as there are observations, with all rejected proposals kept. The ability to produce a conditionally independent draw of these variables is important when posterior updates of some parameters are intractable while others are simple. In such a situation, we introduce the rejected variables only when we need to carry out the intractable updates, after which we discard them and carry out the simpler updates.

A particular application of our algorithm is parameter inference for probability distributions truncated to sets like the positive orthant, the simplex, or the unit sphere. Such distributions correspond to sampling proposals from the untruncated distribution and rejecting those outside the domain of interest. We consider an application from flow cytometry where this representation is the actual data collection process. Truncated distributions also arise in applications like measured time-to-infection (Goethals et al., 2009), where times larger than a year are truncated, mortality data (Alai et al., 2013), annuity valuation for truncated lifetimes (Alai et al., 2013), and stock price changes (Aban et al., 2006). One approach for such problems was proposed in Liechty et al. (2009), through their algorithm samples from an approximation to the posterior distribution of interest. Our algorithm provides a simple and general way to apply the machinery of Bayesian inference to such problems.

2. Rejection sampling

Consider a probability density $p(x \mid \theta) = f(x, \theta)/Z(\theta)$ on some space $\mathbb{X}$, with the parameter θ taking values in Θ. We assume that the normalization constant $Z(\theta)$ is difficult to evaluate, so that naïve sampling from $p(x \mid \theta)$ is not easy. We also assume there exists a second, simpler density $q(x \mid \theta) \geqslant f(x, \theta)/M$ for all x and some positive M.

Rejection sampling generates samples distributed as $p(\cdot \mid \theta)$ by first proposing samples from $q(\cdot \mid \theta)$. A draw y from $q(\cdot \mid \theta)$ is accepted with probability $f(y, \theta)/\{Mq(y \mid \theta)\}$. Let there be r rejected proposals preceding an accepted sample x, and denote them by $\mathcal{Y} = \{y_1, \ldots, y_r\}$ where r itself is a random variable. Write $|\mathcal{Y}| = r$, so that the joint probability is

$$p(\mathcal{Y}, x) = \left[\prod_{i=1}^{|\mathcal{Y}|} q(y_i \mid \theta) \left\{1 - \frac{f(y_i, \theta)}{Mq(y_i \mid \theta)}\right\}\right] q(x \mid \theta) \left\{\frac{f(x, \theta)}{Mq(x \mid \theta)}\right\}$$

$$= \frac{f(x, \theta)}{M} \prod_{i=1}^{|\mathcal{Y}|} \left\{(q(y_i \mid \theta) - \frac{f(y_i, \theta)}{M}\right\}. \tag{1}$$

This procedure recovers samples from $p(x \mid \theta)$, so that (1) has the correct marginal distribution over x (Robert & Casella, 2005, p. 51). Later, we will need to sample the rejected variables $\mathcal{Y}$ given an observation x drawn from $p(\cdot \mid \theta)$. Simulating from $p(\mathcal{Y} \mid x, \theta)$ involves the two steps in Algorithm 1, which relies on Proposition 1 about $p(\mathcal{Y} \mid x, \theta)$; see the Appendix.

Algorithm 1. Algorithm to sample from $p(\mathcal{Y} \mid x, \theta)$

```
Input:  A sample $x$, and the parameter value $\theta$.
Output: The set of rejected proposals $\mathcal{Y}$ preceding $x$.

Sample $y_i$ independently from $q(\cdot \mid \theta)$ until a point $\hat{x}$ is accepted.
Discard $\hat{x}$, and treat the preceding rejected proposals as $\mathcal{Y}$.
```

Proposition 1. *The set of rejected samples $\mathcal{Y}$ preceding an accepted sample x is independent of x: $p(\mathcal{Y} \mid \theta, x) = p(\mathcal{Y} \mid \theta)$.*

3. Bayesian inference

3·1. *Sampling by introducing rejected proposals*

Given observations $X = \{x_1, \ldots, x_n\}$, and a prior $p(\theta)$, Bayesian inference typically uses Markov chain Monte Carlo simulation to sample from an intractable posterior $p(\theta \mid X)$. Split θ as (θ_1, θ_2) so that the normalization constant factors as $Z(\theta) = Z_1(\theta_1)Z_2(\theta_2)$, with Z_1 simple to evaluate, and Z_2 intractable. Updating θ_1 with θ_2 fixed is easy, and there are situations where we can place a conjugate prior on θ_1. Inference for θ_2 is a doubly-intractable problem.

We assume that $p(x \mid \theta)$ has an associated rejection sampling algorithm with proposal density $q(x \mid \theta) \geqslant f(x, \theta)/M$. For the ith observation x_i, write the preceding set of rejected samples as $\mathcal{Y}_i = \{y_{i1}, \ldots, y_{i|\mathcal{Y}_i|}\}$. The joint density of all samples, both rejected and accepted, is

$$p(x_1, \mathcal{Y}_1, \ldots, x_n, \mathcal{Y}_n) = \prod_{i=1}^{n} \frac{f(x_i, \theta)}{M} \prod_{j=1}^{|\mathcal{Y}_i|} \left\{ q(y_{ij} \mid \theta) - \frac{f(y_{ij}, \theta)}{M} \right\}.$$

This involves no intractable terms, so standard techniques can be applied to update θ. To introduce the rejected proposals $\mathcal{Y}_i$, we simply follow Algorithm 1: draw proposals from $q(\cdot \mid \theta)$ until we have n acceptances, with the ith batch of rejected proposals forming the set $\mathcal{Y}_i$.

The ability to produce conditionally independent draws of $\mathcal{Y}$ is important when, for instance, there exists a conjugate prior $p_1(\theta_1)$ on θ_1 for the likelihood $p(x \mid \theta_1, \theta_2)$. Introducing the rejected proposals $\mathcal{Y}_i$ breaks this conjugacy, and the resulting complications in updating θ_1 can slow down mixing, especially when θ_1 is high-dimensional. A much cleaner solution is to sample θ_1 from its conditional posterior $p(\theta_1 \mid X, \theta_2)$, introducing the auxiliary variables only when needed to update θ_2. After updating θ_2, they can be discarded. Algorithm 2 describes this.

Algorithm 2. An iteration of the Markov chain for posterior inference for $\theta = (\theta_1, \theta_2)$

```
Input:  The observations $X$, and the current parameter values $(\theta_1, \theta_2)$.
Output: New parameter values $(\tilde{\theta}_1, \tilde{\theta}_2)$.

Run Algorithm 1 $|X|$ times, keeping all the rejected proposals $\mathcal{Y} = \cup_{i=1}^{|X|} \mathcal{Y}_i$.
Update $\theta_2$ to $\tilde{\theta}_2$ with a Markov kernel having $p(\theta_2 \mid X, \mathcal{Y}, \theta_1)$ as stationary distribution.
Discard the rejected proposals $\mathcal{Y}$.
Sample a new value of $\tilde{\theta}_1$ from the conditional $p(\theta_1 \mid X, \tilde{\theta}_2)$.
```

3·2. *Related work*

One of the simplest and most widely applicable Markov chain Monte Carlo algorithms for doubly-intractable distributions is the exchange sampler of Murray et al. (2006). Simplifying an earlier idea by Møller et al. (2006), this algorithm effectively amounts to the following: given the current parameter θ_{curr}, propose a new parameter θ_{new} according to some proposal distribution. Additionally, generate a dataset of n pseudo-observations $\{\hat{x}_i\}$ from $p(x \mid \theta_{\text{new}})$. The exchange algorithm then proposes to exchange parameters associated with datasets. Murray et al. (2006) show that all intractable terms cancel out in the resulting acceptance probability, and that the resulting Markov chain has the correct stationary distribution.

While the exchange algorithm is applicable whenever one can sample from the likelihood $p(x \mid \theta)$, it does not exploit the mechanism used to produce these samples. When the latter is a rejection sampling algorithm, each pseudo-observation is preceded by a sequence of rejected proposals. These are all discarded, and only the accepted proposals are used to evaluate the new parameter θ_{new}. By contrast our algorithm explicitly instantiates these rejected proposals, so that they can be used to make good proposals. In our experiments, we use a Hamiltonian Monte Carlo sampler on the augmented space and exploit gradient information to make nonlocal moves with a high probability of acceptance. For reasonable acceptance probabilities under the exchange sampler, one must make local updates to θ, or resort to complicated annealing schemes. Of course, the exchange sampler is applicable when no efficient rejection sampling scheme exists, such as when carrying out parameter inference for a Markov random field.

Another framework for doubly-intractable distributions is the pseudo-marginal approach of Andrieu & Roberts (2009). The idea here is that even if we cannot exactly evaluate the acceptance probability, it is sufficient to use a positive, unbiased estimator: this will still result in a Markov chain with the correct stationary distribution. In our case, instead of requiring an unbiased estimate, we bound $Z(\theta)$ by choosing $f(x, \theta) \leqslant Mq(x)$. Additionally, like the exchange sampler, the pseudo-marginal method provides a mechanism to evaluate a proposed θ_{new}; making good proposals (Dahlin et al., 2015) is less obvious. Other papers are Beskos et al. (2006), based on a rejection sampling algorithm for diffusions, and Walker (2011).

Most closely related to our ideas is a sampler from Adams et al. (2009); see also § 7. Their problem also involved inferences on the parameters governing the output of a rejection sampling algorithm. Like us, they augment the state space to include the rejected proposals $\mathcal{Y}$, and like us, given these auxiliary variables, they use Hamiltonian Monte Carlo to efficiently update parameters. However, rather than generating independent realizations of $\mathcal{Y}$ when needed, Adams et al. (2009) outlined a set of Markov transition operators to perturb the current configuration of $\mathcal{Y}$, while maintaining the correct stationary distribution. With prespecified probabilities, they proposed adding a new variable to $\mathcal{Y}$, deleting a variable from $\mathcal{Y}$ and perturbing the value of an existing element in $\mathcal{Y}$. These local updates to $\mathcal{Y}$ can slow down Markov chain mixing, require the user to specify a number of parameters, and also involve calculating Metropolis–Hastings acceptance probabilities for each local step. Furthermore, the Markov nature of their updates require them to maintain the rejected proposals at all times; this can break any conjugacy, and complicate inference for other parameters.

4. Convergence properties

Write the Markov transition density of our chain as $k(\hat{\theta} \mid \theta)$, and the m-fold transition density as $k^m(\hat{\theta} \mid \theta)$. The Markov chain is uniformly ergodic if constants $\rho < 1$ and C exist such that for all m and θ, $\int_\Theta |p(\hat{\theta} \mid X) - k^m(\hat{\theta} \mid \theta)| \mathrm{d}\hat{\theta} \leqslant C\rho^m$. The term to the left is twice the total variation distance between the desired posterior and the state of the Markov chain initialized at θ after m iterations. Small values of ρ imply faster mixing. The following minorization condition is sufficient for uniform ergodicity (Jones & Hobert, 2001): there exists a probability density $h(\hat{\theta})$

and a $\delta > 0$ such that for all $\theta, \hat{\theta} \in \Theta$,

$$k(\hat{\theta} \mid \theta) \geqslant \delta h(\hat{\theta}). \tag{2}$$

When this holds, the mixing rate $\rho \leqslant 1 - \delta$, so that a large δ implies rapid mixing.

Our Markov transition density first introduces the rejected proposals $\mathcal{Y}$, and then conditionally updates θ. The set $\mathcal{Y}_i$ preceding the ith observation takes values in the union space $\mathbb{U} \equiv \cup_{r=0}^{\infty} \mathbb{X}^r$. The output of the rejection sampler, including the ith observation, lies in the product space $\mathbb{U} \times \mathbb{X}$ with density given by equation (1), so that any $(\mathcal{Y}, x) \in \mathbb{U} \times \mathbb{X}$ has probability

$$p(\mathcal{Y}, x \mid \theta) = \frac{f(x, \theta)}{M} \lambda(\mathrm{d}x) \prod_{i=1}^{|\mathcal{Y}|} \left\{ q(y_i \mid \theta) - \frac{f(y_i, \theta)}{M} \right\} \lambda(\mathrm{d}y_i). \tag{3}$$

Here, λ is the measure with respect to which the densities f and q are defined, and it is easy to see that equation (3) integrates to 1. From Bayes' rule, the conditional density over $\mathcal{Y}$ is

$$p(\mathcal{Y} \mid x, \theta) = \frac{Z(\theta)}{M} \prod_{i=1}^{|\mathcal{Y}|} \left\{ q(y_i \mid \theta) - \frac{f(y_i, \theta)}{M} \right\} \lambda(\mathrm{d}y_i). \tag{4}$$

The fact that the right-hand side does not depend on x is another proof of Proposition 1. Equation (4) also motivates the use of our algorithm outside the context of rejection sampling: we can view $\mathcal{Y}$ as convenient auxiliary variables that are independent of x, and whose density is such that $Z(\theta)$ cancels when evaluating the joint density of $(x, \mathcal{Y})$.

The density from equation (4) characterizes the data augmentation step of our sampling algorithm. In practice, we need as many draws from this density as there are observations. The next step involves updating θ given $(\mathcal{Y}, X, \theta)$, and depends on the problem at hand. We simplify matters by assuming that we can sample from $p(\theta \mid \mathcal{Y}, X)$ independently of the old θ: this is the classical data augmentation algorithm. We also assume that the functions $f(\cdot, \theta)$ and $q(\cdot \mid \theta)$ are uniformly bounded from above and below by finite, positive quantities (B_f, b_f) and (B_q, b_q) respectively, and that $\int_{\mathbb{X}} \lambda(\mathrm{d}x) < \infty$. It follows that there exist positive numbers r and R that minimize $1 - f(x, \theta)/\{MZ(\theta)\}$ and $Z(\theta)/M$. We can now state our result.

THEOREM 1. *Assume that $\int_{\mathbb{X}} \lambda(\mathrm{d}x) < \infty$ and that positive bounds b_f, B_f, b_q, B_q exist with r and R as defined earlier. Further assume we can sample from the conditional $p(\theta \mid \mathcal{Y}, X)$. Then our data augmentation algorithm is uniformly ergodic with mixing rate ρ bounded above by $\rho = 1 - [b_f/\{B_f(\beta + R^{-1})\}]^n$, where $\beta = b_q r / B_q$ and n is the number of observations.*

Despite our assumptions, our theorem has a number of useful implications. The ratio b_f/B_f is a measure of how flat the function f is, and the closer it is to unity, the more efficient rejection sampling for f can be. From our result, the smaller the ratio, the larger the bound on ρ, suggesting slower mixing. This is consistent with more rejected proposals $\mathcal{Y}$ increasing the coupling between successive θs in the Markov chain. On the other hand, a small b_q/B_q suggests a proposal distribution tailored to f, and our result shows that this implies faster mixing. The numbers r and $1/R$ are measures of mismatch between the target and proposal density, with small values giving better mixing. Finally, more observations n result in slower mixing. We suspect that this last property holds for most exact samplers for doubly-intractable distributions, though we are unaware of any such result.

Even without assuming we can sample from $p(\theta \mid \mathcal{Y}, X)$, our ability to sample $\mathcal{Y}$ independently means that the marginal chain over θ is Markovian. By contrast, existing approaches (Adams et al., 2009; Walker, 2011) only produce dependent updates in the complicated auxiliary space: they sample from $p(\hat{\mathcal{Y}} \mid \theta, \mathcal{Y}, X)$ by making local updates to $\mathcal{Y}$. Consequently, these chains are Markovian only in the complicated augmented space, and the marginal processes over θ have long-term dependencies. Besides affecting mixing, this can also complicate analysis.

5. Flow cytometry data

We apply our algorithm to a dataset of flow cytometry measurements from patients subjected to bone-marrow transplant (Brinkman et al., 2007). This graft-versus-host disease dataset has 6809 control and 9083 positive observations, corresponding to whether donor immune cells attack host cells. Each observation consists of four biomarker measurements truncated between 0 and 1024, though more complicated truncation rules are often used according to operator judgement (Lee & Scott, 2012). We normalize and plot the first two dimensions, markers CD4 and CD8b, in Fig. 1. Truncation complicates the clustering of observations into homogeneous groups, an important step in the flow-cytometry pipeline called gating. Consequently, Lee & Scott (2012) propose an expectation-maximization algorithm for truncated Gaussian mixture models, which must be adapted if different mixture components or truncation rules are used.

We model the untruncated distribution for each group as a Dirichlet process mixture of Gaussian kernels (Lo, 1984), with points outside the four-dimensional unit hypercube discarded to form the normalized dataset. The Dirichlet process mixture model is a flexible nonparametric prior over densities parameterized by a concentration parameter α and a base probability measure. We set $\alpha = 1$, and for the base measure, which gives the distribution over cluster parameters, we use a normal-inverse-Wishart distribution. Given the rejected variables, we can use standard techniques to update a representation of the Dirichlet process. We follow the blocked-sampler of Ishwaran & James (2001) based on the stick-breaking representation of the Dirichlet process, using a truncation level of 50 clusters. This corresponds to updating θ, step 2 in Algorithm 2. Having done this, we discard the old rejected samples, and produce a new set by drawing from a 50-component Gaussian mixture model, corresponding to step 1 in Algorithm 2.

Figure 1 shows the log mean posterior densities for the first two dimensions from 10 000 iterations. While the control group has three clear modes, these are much less pronounced in the positive group. Directly modelling observations with a Gaussian mixture model obscured this by forcing modes away from the edges. One can use components with bounded support in the mixture model, such as a Dirichlet process mixture of Beta densities; however, these do not reflect the underlying data generation process, and are unsuitable when different groups have different truncation levels. By contrast, it is easy to extend our modelling ideas to allow groups to share components, allowing better identification of disease predictors.

Our sampler took less than two minutes to run 1000 iterations, not much longer than a typical Dirichlet process sampler for datasets of this size. The average number of augmented points was 3960 and 4608 for the two groups. We study our sampler more systematically in the next section, but this application demonstrates the flexibility and simplicity of our main idea.

6. Bayesian inference for the matrix Langevin distribution

6·1. *The matrix Langevin distribution on the Stiefel manifold*

The Stiefel manifold $V_{p,d}$ is the space of all $d \times p$ orthonormal matrices, that is, $d \times p$ matrices X such that $X^{\mathrm{T}} X = I_p$, where I_p is the $p \times p$ identity matrix. When $p = 1$, this is the

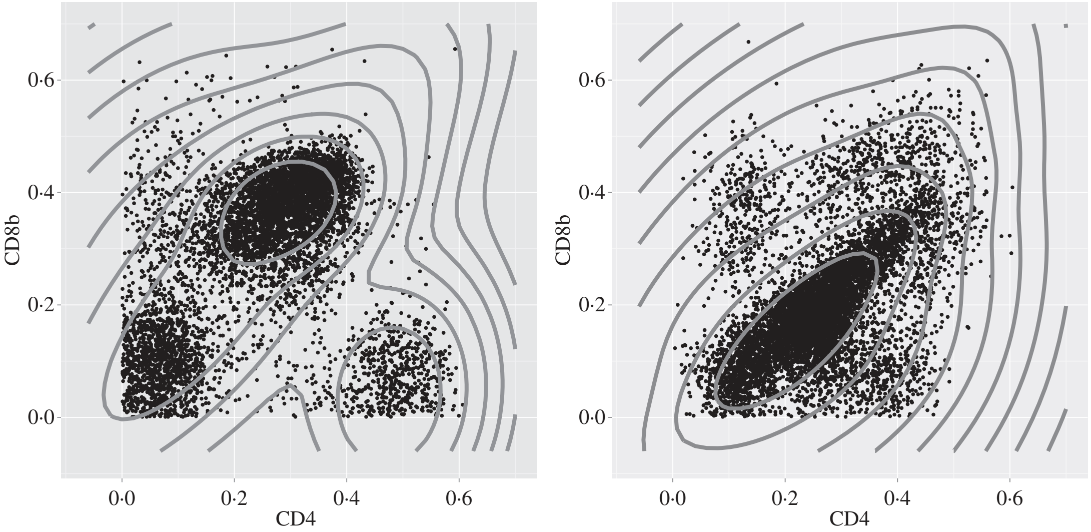

Fig. 1. Scatterplots of the first two dimensions for the control (left) and positive (right) group. Contours represent log posterior-mean densities under a Dirichlet process mixture.

$d-1$ hypersphere S^{d-1}, and when $p=d$, this is the space of all orthonormal matrices $O(d)$. Probability distributions on the Stiefel manifold play an important role in statistics, signal processing and machine learning, with applications ranging from studies of orientations of orbits of comets and asteroids to principal components analysis to the estimation of rotation matrices. The simplest such distribution is the matrix Langevin distribution, an exponential-family distribution whose density with respect to the invariant Haar volume measure (Edelman et al., 1998) is $p_{\mathrm{ML}}(X \mid F) = \mathrm{etr}(F^{\mathrm{T}}X)/Z(F)$. Here etr is the exponential-trace, and F is a $d \times p$ matrix. The normalization constant $Z(F) = {}_0F_1(d/2, F^{\mathrm{T}}F/4)$ is the hypergeometric function with matrix arguments, evaluated at $F^{\mathrm{T}}F/4$ (Chikuse, 2003). Let $F = G\kappa H^{\mathrm{T}}$ be the singular value decomposition of F, where G and H are $d \times p$ and $p \times p$ orthonormal matrices, and κ is a positive diagonal matrix. We parameterize p_{ML} by (G, κ, H), and one can think of G and H as orientations, with κ controlling the concentration in directions determined by these orientations. Large values of κ imply concentration along the associated directions, while setting κ to zero gives the uniform distribution on the Stiefel manifold. It can be shown (Khatri & Mardia, 1977) that ${}_0F_1(d/2, F^{\mathrm{T}}F/4) = {}_0F_1(d/2, \kappa^{\mathrm{T}}\kappa/4)$, so that this depends only on κ. We write it as $Z(\kappa)$. In our Bayesian analysis, we place independent priors on κ, G and H. The last two lie on the Stiefel manifolds $V_{p,d}$ and $V_{p,p}$, and we place matrix Langevin priors $p_{\mathrm{ML}}(\cdot \mid F_0)$ and $p_{\mathrm{ML}}(\cdot \mid F_1)$ on these: we will see below that these are conditionally conjugate. We place independent Gamma priors on the diagonal elements of κ. However, the difficulty in evaluating the normalization constant $Z(\kappa)$ makes posterior inference for κ doubly intractable. Thus, in a 2006 University of Iowa PhD thesis, Camano-Garcia keeps κ constant, while Hoff (2009a) uses a first-order Taylor expansion of the intractable term to run an approximate sampling algorithm. Below, we show how fully Bayesian inference can be carried out for this quantity as well.

6·2. *A rejection sampling algorithm*

We first describe a rejection sampling algorithm from Hoff (2009b) to sample from p_{ML}. For simplicity, assume H is the identity matrix. In the general case, we simply rotate the resulting draw by H, since if $X \sim p_{\mathrm{ML}}(\cdot \mid F)$, then $XH \sim p_{\mathrm{ML}}(\cdot \mid FH^{\mathrm{T}})$. At a high level, the algorithm sequentially proposes vectors from the matrix Langevin on the unit sphere: this is also called the von Mises–Fisher distribution and is easy to simulate (Wood, 1994). The mean of the rth vector is column r of G, $G_{[:r]}$, projected onto the nullspace of the earlier vectors, N_r. This sampled vector is then projected back onto N_r and normalized, and the process is repeated p times. Call the resulting distribution p_{seq}; for more details, see Algorithm 3 and Hoff (2009b).

Algorithm 3. Proposal $p_{\text{seq}}(\cdot \mid G, \kappa)$ for the matrix Langevin distribution (Hoff, 2009b)

```
Input:   Parameters $G, \kappa$; write $G_{[:i]}$ for column $i$ of $G$, and $\kappa_i$ for element $(i, i)$ of $\kappa$.
Output:  An output $X \in V_{p,d}$; write $X_{[:i]}$ for column $i$ of $X$.

Sample $X_{[:1]} \sim p_{\text{ML}}(\cdot \mid \kappa_1 G_{[:1]})$.
For $r \in \{2, \cdots p\}$
    Construct $N_r$, an orthogonal basis for the nullspace of $\{X_{[:1]}, \cdots X_{[:r-1]}\}$.
    Sample $z \sim p_{\text{ML}}(\cdot \mid \kappa_r N_r^{\mathrm{T}} G_{[:r]})$.
    Set $X_{[:r]} = z^{\mathrm{T}} N_r / \|z^{\mathrm{T}} N_r\|$.
```

Letting $I_k(\cdot)$ be the modified Bessel function of the first kind, p_{seq} is a density on the Stiefel manifold with

$$p_{\text{seq}}(X \mid G, \kappa) = \left\{ \prod_{r=1}^{p} \frac{\|\kappa_r N_r^{\mathrm{T}} G_{[:r]}/2\|^{(d-r-1)/2}}{\Gamma(\frac{d-r+1}{2}) I_{(d-r-1)/2}(\|\kappa_r N_r^{\mathrm{T}} G_{[:r]}\|)} \right\} \operatorname{etr}(\kappa G^{\mathrm{T}} X).$$

Write $D(X, \kappa, G)$ for the reciprocal of the term in braces. Since $I_k(x)/x^k$ is an increasing function of x, and $\|N_r^{\mathrm{T}} G_{[:r]}\| \leqslant \|G_{[:r]}\| = 1$, we have the following bound $D(\kappa)$ for $D(X, \kappa, G)$:

$$D(X, \kappa, G) \leqslant \prod_{r=1}^{p} \frac{\Gamma(\frac{d-r+1}{2}) I_{(d-r-1)/2}(\|\kappa_r\|)}{\|\kappa_r/2\|^{(d-r-1)/2}} = D(\kappa).$$

This implies that $\operatorname{etr}(\kappa G^{\mathrm{T}} X) \leqslant D(\kappa) p_{\text{seq}}(X \mid G, \kappa)$, allowing the following rejection sampler: sample X from $p_{\text{seq}}(\cdot)$, and accept with probability $D(X, \kappa, G)/D(\kappa)$. The accepted proposals come from $p_{\text{ML}}(\cdot \mid G, \kappa)$, and for samples from $p_{\text{ML}}(\cdot \mid G, \kappa, H)$, post multiply these by H.

6·3. *Posterior sampling*

Given a set of n observations $\{X_i\}$, and writing $S = \sum_{i=1}^{n} X_i$, we have:

$$p(G, \kappa, H \mid X_i\}) \propto \operatorname{etr}(H\kappa G^{\mathrm{T}} S) p(H) p(G) p(\kappa)/Z(\kappa)^n.$$

At a high level, our approach is a Gibbs sampler that sequentially updates H, G and κ. The pair of matrices (H, G) correspond to the tractable θ_1 in Algorithm 2, while κ corresponds to θ_2. Updating the first two is straightforward, while the third requires our augmentation scheme.

1. Updating G and H: With a matrix Langevin prior on H, the posterior is

$$p(H \mid X_i, \kappa, G) \propto \operatorname{etr}\left\{ (S^{\mathrm{T}} G \kappa + F_0)^{\mathrm{T}} H \right\}.$$

This is just the matrix Langevin distribution over rotation matrices, and one can sample from this following § 6·2. From here onwards, we will rotate the observations by H, allowing us to ignore this term. Redefining S as SH, the posterior over G is also matrix Langevin,

$$p(G \mid X_i\}, \kappa) \propto \operatorname{etr}\left\{ (S\kappa + F_1)^{\mathrm{T}} G \right\}.$$

2. Updating κ: Here, we exploit the rejection sampler scheme of the previous section, and instantiate the rejected proposals using Algorithm 1. From § 6·2, the joint probability is

$$p(\{X_i, \mathcal{Y}_i\} \mid G, \kappa) = \frac{\mathrm{etr}\left\{\kappa G^{\mathrm{T}}\left(S + \sum_{j=1}^{|\mathcal{Y}_i|} Y_{ij}\right)\right\}}{D(\kappa)^{1+|\mathcal{Y}|}} \prod_{i=1}^{n} \prod_{j=1}^{|\mathcal{Y}|} \frac{\{D(\kappa) - D(Y_{ij}, G, \kappa)\}}{D(Y_{ij}, G, \kappa)}. \quad (5)$$

All terms in (5) can be evaluated easily, allowing a simple Metropolis–Hastings algorithm in this augmented space. In fact, we can calculate gradients to run a Hamiltonian Monte Carlo algorithm (Neal, 2010) that makes significantly more efficient proposals than a random-walk sampling algorithm. In particular, let $N = n + \sum_{i=1}^{n} |\mathcal{Y}_i|$, and $S = \sum_{i=1}^{n}(X_i + \sum_{j=1}^{|\mathcal{Y}_i|} Y_{ij})$. The log joint probability $L \equiv \log\{p(\{X_i, \mathcal{Y}_i\})\}$ is

$$L = \mathrm{trace}(G^{\mathrm{T}} \kappa S) + \sum_{i=1}^{n} \sum_{j=1}^{|\mathcal{Y}_i|} \left[\log\left\{D(\kappa) - D(Y_{ij}, \kappa)\right\} - \log D(Y_{ij}, \kappa)\right] - n \log \{D(\kappa).$$

In the Appendix, we give an expression for the gradient of this loglikelihood. We use this to construct a Hamiltonian Monte Carlo sampler (Neal, 2010) for κ. Here, it suffices to note that a proposal involves taking L leapfrog steps of size ϵ along the gradient, and accepting the resulting state with probability proportional to the product of equation (5), and a simple Gaussian momentum term. The acceptance probability depends on how well the ϵ-discretization approximates the continuous dynamics of the system, and choosing a small ϵ and a large L can give global moves with high acceptance probability. A large L however costs a large number of gradient evaluations. We study this trade-off in § 6·5.

6·4. *Vectorcardiogram dataset*

The vectorcardiogram is a loop traced by the cardiac vector during a cycle of the heart beat. The two directions of orientation of this loop in three dimensions form a point on the Stiefel manifold. The dataset of Downs et al. (1971) includes 98 such recordings, and is displayed in Fig. 2(a). We represent each observation with a pair of orthonormal vectors, with the cone of lines to the right forming the first component. This empirical distribution possesses a single mode, so that the matrix Langevin distribution seems a suitable model.

We place independent exponential priors with mean 10 and variance 100 on the scale parameter κ, and a uniform prior on the location parameter G. We restrict H to be the identity matrix. Inferences were carried out using the Hamiltonian sampler to produce 10 000 samples, with a burn-in period of 1000. For the leapfrog dynamics, we set a step size of 0·3, with the number of steps equal to 5. We fix the mass parameter to the identity matrix. We implemented all algorithms in R (R Development Core Team, 2016), building on the rstiefel package of Hoff (2009b). Simulations were run on an Intel Core 2 Duo 3 Ghz CPU. For comparison, we include the maximum likelihood estimates of κ and G. For κ_1 and κ_2, these were 11·9 and 5·9, and we plot these in Fig. 2(b) as circles.

The bold straight lines in Fig. 2(a) show the maximum likelihood estimates of the components of G, with the small circles corresponding to 90% Bayesian credible regions estimated from the Monte Carlo output. The dashed circles correspond to 90% predictive probability regions for the Bayesian model. For these, we generated 50 points on $V_{3,2}$ for each sample, with parameters specified by that sample. The dashed circles contain 90% of these points across all samples. Figure 2(b) shows the posterior over κ_1 and κ_2.

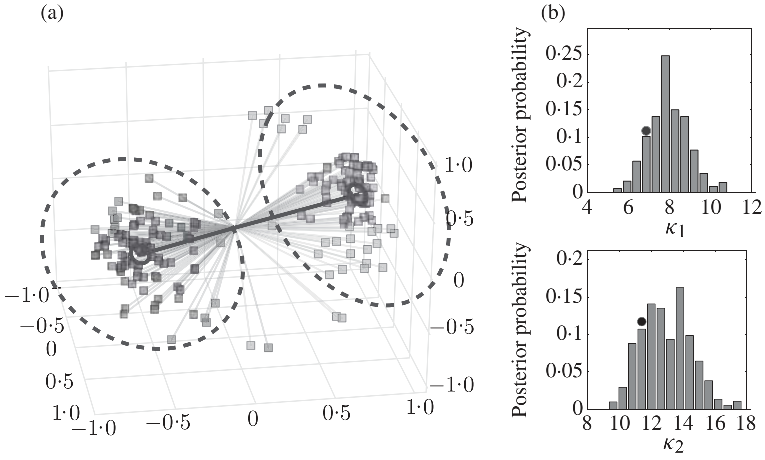

Fig. 2. (a) Vector cardiogram dataset with inferences. Bold solid lines are maximum likelihood estimates of G, and solid circles contain 90% posterior mass. Dashed circles are 90% predictive probability regions. (b) Posterior distribution over κ_1 and κ_2, circles are maximum likelihood estimates.

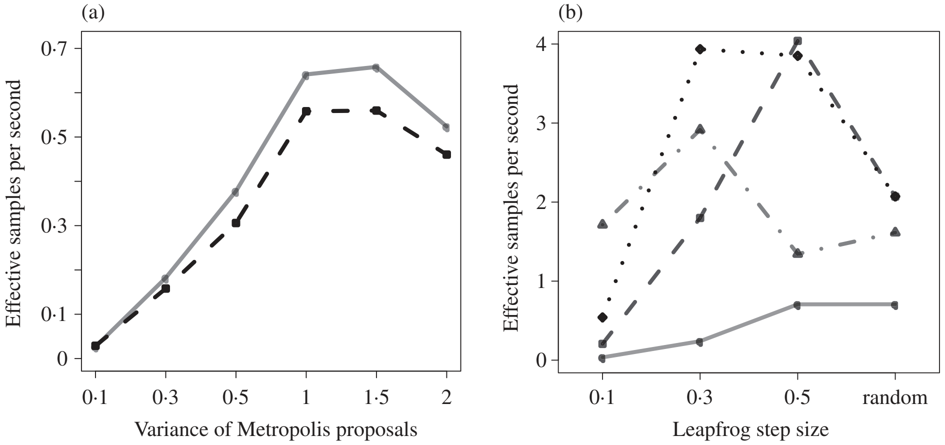

Fig. 3. Effective samples per second for (a) random walk and (b) Hamiltonian samplers. From bottom to top at abscissa 0·5: (a) Metropolis–Hastings data-augmentation sampler and exchange sampler, and (b) 1, 10, 5 and 3 leapfrog steps of the Hamiltonian sampler.

6·5. *Comparison of exact samplers*

To quantify sampler efficiency, we estimate the effective sample sizes produced per unit time. This corrects for correlation between successive Markov chain samples by estimating the number of independent samples produced; for this we used the rcoda package of Plummer et al. (2006).

Figure 3(a) shows the effective sample size per second for two Metropolis–Hastings samplers, the exchange sampler and our latent variable sampler on the vectorcardiogram dataset. Both perform a random walk in the κ-space, with the steps drawn for a normal distribution whose variance increases along the horizontal axis. The figure shows that both samplers' performance peaks when the proposals have a variance between 1 and 1·5, with the exchange sampler performing slightly better. However, the real advantage of our sampler is that introducing the latent variables results in a joint distribution with no intractable terms, allowing the use of more sophisticated sampling algorithms. Figure 3(b) studies the Hamiltonian Monte Carlo sampler described

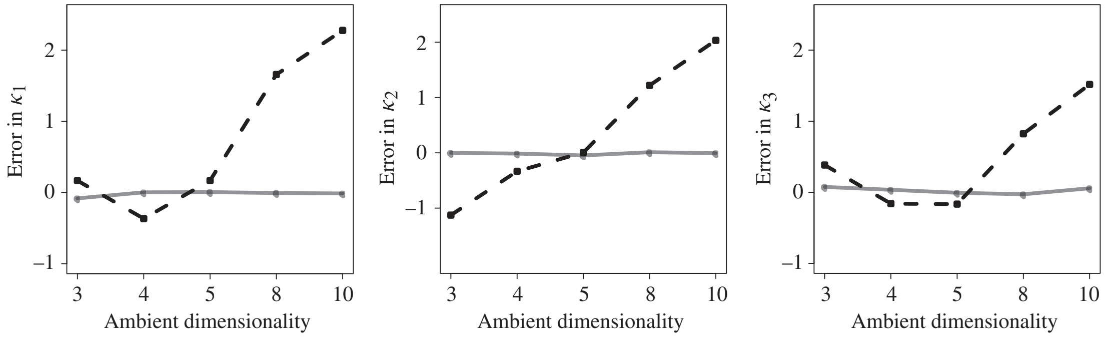

Fig. 4. Errors in the posterior mean for the vectorcardiogram dataset. Each panel is a different component of κ; solid/dashed lines are the Hamiltonian/approximate sampler.

at the end of § 3·1. Here we vary the size of the leapfrog steps along the horizontal axis, with the different curves corresponding to different numbers of such steps. This performs an order of magnitude better than either of the previous algorithms, with performance peaking with 3 to 5 steps of size 0·3 to 0·5, fairly typical values for this algorithm. This shows the advantage of exploiting gradient information in exploring the parameter space.

6·6. *Comparison with an approximate sampler*

In this section, we consider an approximate sampler based on an asymptotic approximation to $Z(\kappa) = {}_0F_1(d/2, \kappa^{\mathrm{T}}\kappa/4)$ for large values of $(\kappa_1, \ldots, \kappa_n)$ (Khatri & Mardia, 1977):

$$Z(\kappa) \simeq \left\{\frac{2^{-\frac{1}{4}p(p+5)+\frac{1}{2}pd}}{\pi^{\frac{1}{2}p}}\right\} \mathrm{etr}(\kappa) \prod_{j=1}^{p} \Gamma\left(\frac{d-j+1}{2}\right) \left[\left\{\prod_{j=2}^{p}\prod_{i=1}^{j-1}(\kappa_i+\kappa_j)^{\frac{1}{2}}\right\}\prod_{i=1}^{p}\kappa_i^{\frac{1}{2}(d-p)}\right]^{-1}.$$

We use this approximation in the acceptance probability of a Metropolis–Hastings algorithm; it can similarly be used to construct a Hamiltonian sampler. For a more complicated but accurate approximation, see Kume et al. (2013). In general however, using such approximate schemes involves the ratio of two approximations, and can have very unpredictable performance.

On the vectorcardiogram dataset, the approximate sampler is about forty times faster than the exact samplers. For larger datasets, this difference will be even greater, and the real question is how accurate the approximation is. Our exact sampler allows us to study this: we consider the Stiefel manifold $V_{d,3}$, with the three diagonal elements of κ set to 1, 5 and 10. With this setting of κ, and a random G, we generate datasets with 50 observations, with d taking values 3, 4, 5, 8, and 10. In each case, we estimate the posterior mean of κ by running the exchange sampler, and treat this as the truth. We compare this with posterior means returned by our Hamiltonian sampler, as well as the approximate sampler. Figure 4 shows these results. As expected, the two exact samplers agree, and the Hamiltonian sampler has almost no error. For values of d around 5, the estimated posterior mean for the approximate sampler is close to that of the exact samplers. Smaller values lead to an approximate posterior mean that underestimates the actual posterior mean, while in higher dimensions, the opposite occurs. Recalling that κ controls the concentration of the matrix Langevin distribution about its mode, this implies that in high dimensions, the approximate sampler underestimates uncertainty in the distribution of future observations.

7. The Gaussian process density sampler

7·1. *Nonparametric density modelling with a transformed Gaussian process*

Our next application is the Gaussian process density sampler of Adams et al. (2009), a nonparametric prior for probability densities induced by a logistic transformation of a random function from a Gaussian process. Letting $\sigma(\cdot)$ denote the logistic function, the random density is

$$g(x) \propto g_0(x)\sigma\{f(x)\}, \qquad f \sim \text{GP},$$

with $g_0(\cdot)$ a parametric base density and GP denoting a Gaussian process. The inequality $g_0(x)\sigma\{f(x)\} \leqslant g_0(x)$ allows a rejection sampling algorithm by making proposals from $g_0(\cdot)$. At a proposed location x^*, we sample the function value $f(x^*)$ conditioning on all previous evaluations, and accept the proposal with probability $\sigma\{f(x^*)\}$. Such a scheme involves no approximation error, and only requires evaluating the random function on a finite set of points. Algorithm 4 describes the steps involved in generating n observations.

Algorithm 4. Generate n new samples from the Gaussian process density sampler

```
Input:   A base probability density $g_0(\cdot)$.
         Previous accepted and rejected proposals $\tilde{X}$ and $\tilde{Y}$.
         Gaussian process evaluations $f_{\tilde{X}}$ and $f_{\tilde{Y}}$ at these locations.
Output:  $n$ new samples $X$, with the associated rejected proposals $Y$.
         Gaussian process evaluations $f_X$ and $f_Y$ at these locations.

Repeat
    Sample a proposal $y$ from $g_0(\cdot)$.
    Sample $f_y$, the Gaussian process evaluated at $y$, conditioning on $f_X$, $f_Y$, $f_{\tilde{X}}$ and $f_{\tilde{Y}}$.
    With probability $\sigma(f_y)$
        Accept $y$ and add it to $X$. Add $f_y$ to $f_X$.
    Else
        Reject $y$ and add it to $Y$. Add $f_y$ to $f_Y$.
Until $n$ samples are accepted.
```

7·2. *Posterior inference*

Given observations $X = \{x_1, \ldots, x_n\}$, we are interested in $p(g \mid X)$, the posterior over the underlying density. Since g is determined by the modulating function f, we focus on $p(f \mid X)$. While this quantity is doubly intractable, after augmenting the state space to include the proposals $\mathcal{Y}$ from the rejection sampling algorithm, $p(f \mid X, \mathcal{Y})$ has density $\prod_{i=1}^{n} \sigma\{f(x_i)\} \prod_{i=1}^{|\mathcal{Y}|} [1 - \sigma\{f(y_i)\}]$ with respect to the Gaussian process prior; see also Adams et al. (2009). In words, the posterior over f evaluated at $X \cup \mathcal{Y}$ is just the posterior from a Gaussian process classification problem with a logistic link-function, and with the accepted and rejected proposals corresponding to the two classes. Markov chain Monte Carlo methods such as Hamiltonian Monte Carlo or elliptical slice sampling (Murray et al., 2010) are applicable in such a situation. Given f on $X \cup \mathcal{Y}$, the Gaussian process can be evaluated anywhere else by conditionally sampling from a multivariate normal.

Sampling the rejected proposals $\mathcal{Y}$ given X and f is straightforward using Algorithm 1: run the rejection sampler until n accepts, and treat the rejected proposals generated along the way as $\mathcal{Y}$. In practice, we do not have access to the entire function f, only its values evaluated on X and $\mathcal{Y}_{old}$, the locations of the previous thinned variables. However, just as under the generative mechanism, we can retrospectively evaluate the function f where needed. After proposing from

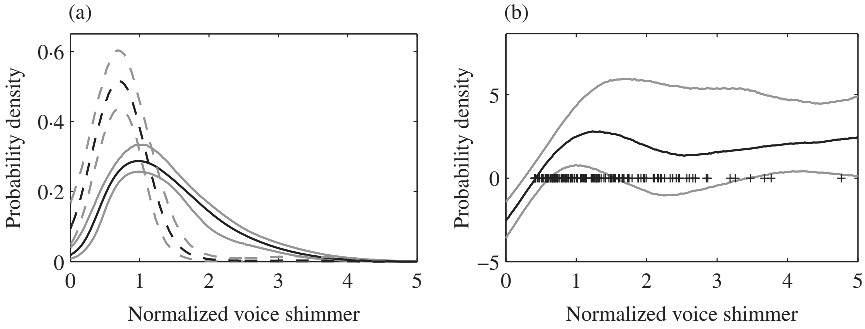

Fig. 5. Inferences for the Parkinson’s dataset: (a) posterior density for positive (solid) and control (dashed) groups, (b) posterior distribution of the Gaussian process function for positive group with observations. Both panels show the median with 80 percent credible intervals.

$g_0(\cdot)$, we sample the value of the function at this location conditioned on all previous evaluations, and use this value to decide whether to accept or reject. We outline the inference algorithm in Algorithm 5, noting that it is much simpler than that proposed in Adams et al. (2009). We also refer to that paper for limitations of the exchange sampler in this problem.

Algorithm 5. A Markov chain iteration for inference in the Gaussian process density sampler

```
Input:  Observations $X$ with corresponding function evaluations $\tilde{f}_X$.
        Current rejected proposals $\tilde{Y}$ with corresponding function evaluations $\tilde{f}_{\tilde{Y}}$.
Output: New rejected proposals $Y$.
        New Gaussian process evaluations $f_X$ and $f_Y$ at $X$ and $Y$.
        New hyperparameters.

Run Algorithm 4 to produce $|X|$ accepted samples, with $X$, $\tilde{Y}$, $\tilde{f}_X$ and $\tilde{f}_{\tilde{Y}}$ as inputs.
Replace $\tilde{Y}$ and $f_{\tilde{Y}}$ with values returned by the previous step; call these $Y$ and $\hat{f}_Y$.
Update $\tilde{f}_X$ and $\hat{f}_Y$ using for example, hybrid Monte Carlo, to get $f_X$ and $f_Y$.
Update Gaussian process and base-distribution hyperparameters.
```

7·3. *Experiments*

Voice changes are a symptom and measure of the onset of Parkinson’s disease, and one attribute is voice shimmer, a measure of variation in amplitude. We consider a dataset of such measurements for subjects with and without the disease (Little et al., 2007), with 147 measurements with, and 48 without the disease. We normalized these to vary from 0 to 5, and used the model of Adams et al. (2009) as a prior on the underlying probability densities. We set $g_0(\cdot)$ to a normal $\mathcal{N}(\mu, \sigma^2)$, with a normal-inverse-Gamma prior on (μ, σ). The latter had its mean, inverse-scale, degrees-of-freedom and variance set to 0,·1,1 and 10. The Gaussian process had a squared-exponential kernel, with variance and length-scale of 1. For each case, we ran a Matlab implementation of our data augmentation algorithm to produce 2000 posterior samples after a burn-in of 500 samples.

Figure 5(a) shows the resulting posterior over densities, corresponding to θ in Algorithm 2. The control group is fairly Gaussian, while the disease group is skewed to the right. Figure 5(b) focuses on the deviation from normality by plotting the posterior over the latent function f. We see that to the right of 0·5, this deviation is larger than its prior mean of zero, implying larger probability than under a Gaussian density. Figure 6 studies the distribution of the rejected proposals $\mathcal{Y}$. Figure 6(a) shows the distribution of their locations: most of these occured near the origin.

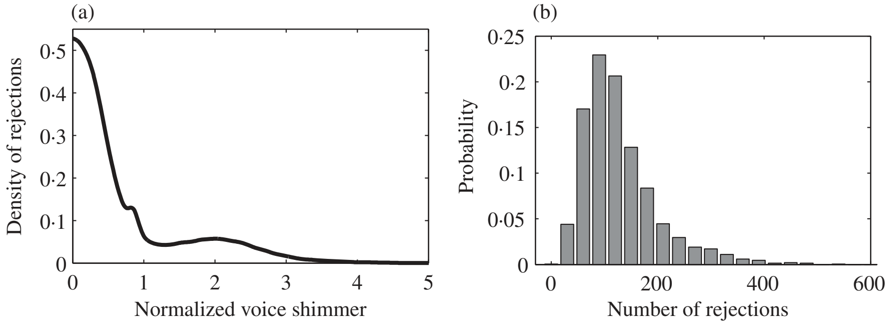

Fig. 6. Rejected proposals for the Parkinson's dataset: (a) kernel density estimate of locations of rejected proposals, and (b) histogram of the number of rejected proposals for the positive group.

Here, the disease density reverts to Gaussian or even sub-Gaussian density, with the intensity function taking small values. Figure 6(b) is a histogram of the number of rejected proposals: this is typically around 100 to 150, though the largest value we observed was 668. Since inference on the latent function involves evaluating it at the accepted as well as rejected proposals, the largest covariance matrix we had to deal with was about 600 × 600; typical values were around 100 × 100. Using the same set-up as § 6·5, it took a naïve Matlab implementation 26 and 18 minutes to run 2500 iterations for the disease and control datasets. One can imagine computations becoming unwieldy for a large number of observations, or when there is large mismatch between the true density and the base-measure $g_0(\cdot)$. In such situations, one might have to choose the Gaussian process covariance kernel more carefully, use one of many sparse approximation techniques, or use other nonparametric priors like splines instead. In all these cases, we can use our algorithm to recover the rejected proposals $\mathcal{Y}$, and given these, posterior inference for f can be carried out using standard techniques.

8. Future work

Our algorithm, while exact, also provides a framework for faster, approximate algorithms. A priori, the number of rejected proposals preceeding any observation is unbounded: one can bound the computational cost of an iteration by limiting the maximum number of rejected proposals. Similarly, one might share rejected proposals across observations. We leave the study of such approximate sampling algorithms for future research. Also left open is a more careful analysis of Markov mixing rates for the applications we considered. There are also a number of applications that we have not described here: particularly relevant are rejection samplers for diffusions (Beskos et al., 2006; Bladt & Sørensen, 2014).

Acknowledgement

This work was supported by the National Institute of Environmental Health Sciences of the National Institutes of Health. We are grateful to the editor and reviewers for valuable comments.

Appendix

Proofs

Proof of Proposition 1. Rejection sampling first proposes from $q(x|\theta)$, and then accepts with probability $f(x,\theta)/\{Mq(x|\theta)\}$. Conceptually, one can first decide whether to accept or reject, and then conditionally sample the location. The marginal acceptance probability is $Z(\theta)/M$, the area under $f(\cdot,\theta)$ divided by that under $Mq(\cdot\mid\theta)$. An accepted sample x is distributed as the target distribution $f(x,\theta)/Z(\theta)$, while rejected samples are distributed as $\{Mq(x\mid\theta)-f(x,\theta)\}/\{M-Z(\theta)\}$. This two-component mixture is just the proposal $q(x)$. While this mixture representation loses the computational benefits of the original algorithm, it shows that the location of an accepted sample is independent of the past, and consequently, that the number and locations of rejected samples preceding an accepted sample is independent of the location of that sample. Thus, one can use the rejected samples preceding any other accepted sample. □

Proof of Theorem 1. It follows from Bayes' rule and the assumed bounds that for an observation X,

$$p(\theta\mid X,\mathcal{Y})\geqslant p(\theta\mid X)\frac{b_f}{B_f}\left(\frac{b_q r}{B_q}\right)^{|\mathcal{Y}|}.$$

Let the number of observations $|X|$ be n. Then,

$$\begin{aligned}
k(\hat\theta\mid\theta)&=\int_{\mathbb{U}^n}p(\hat\theta\mid\mathcal{Y},X)p(\mathcal{Y}\mid\theta,X)\mathrm{d}\mathcal{Y}\\
&\geqslant\left(\frac{b_f}{B_f}\right)^n p(\hat\theta\mid X)\prod_{i=1}^{n}\int_{\mathbb{U}}\beta^{|\mathcal{Y}_i|}p(\mathcal{Y}_i\mid\theta,X)\mathrm{d}\mathcal{Y}_i\\
&=\left(\frac{b_f}{B_f}\right)^n p(\hat\theta\mid X)\prod_{i=1}^{n}\int_{\mathbb{U}}\beta^{|\mathcal{Y}_i|}\frac{Z(\theta)}{M}\prod_{j=1}^{|\mathcal{Y}_i|}\left\{q(y_{ji}\mid\theta)-\frac{f(y_{ji},\theta)}{M}\right\}\lambda(\mathrm{d}y_{ji})\\
&=\left\{\frac{b_f Z(\theta)}{B_f M}\right\}^n p(\hat\theta\mid X)\prod_{i=1}^{n}\sum_{|\mathcal{Y}_i|=0}^{\infty}\beta^{|\mathcal{Y}_i|}\prod_{j=1}^{|\mathcal{Y}_i|}\left\{1-\frac{Z(\theta)}{M}\right\}\\
&=p(\hat\theta\mid X)\left\{\frac{b_f Z(\theta)}{B_f M}\right\}^n\prod_{i=1}^{n}\frac{1}{1-\beta\left\{1-Z(\theta)/M\right\}}\\
&=\frac{p(\hat\theta\mid X)}{\delta_\theta^n},\qquad \delta_\theta=\frac{B_f}{b_f}\left[\frac{M}{Z(\theta)}-\beta\left\{\frac{M}{Z(\theta)}-1\right\}\right]=\frac{B_f}{b_f}\left\{\frac{M}{Z(\theta)}(1-\beta)+\beta\right\}\\
&\geqslant\delta p(\hat\theta\mid X),\qquad \delta=\left\{\frac{b_f}{B_f\left(\beta+R^{-1}\right)}\right\}^n.
\end{aligned}$$

Thus $k(\hat\theta\mid\theta)$ satisfies equation (2), with $\delta=[b_f\{B_f(\beta+R^{-1})\}]^n$, and $h(\hat\theta)=p(\hat\theta\mid X)$. □

Gradient information

For n pairs $\{X_i,\mathcal{Y}_i\}$, with $\tilde n=n+\sum_{i=1}^{n}|\mathcal{Y}_i|$, and $S=\sum_{i=1}^{n}(X_i+\sum_{j=1}^{|\mathcal{Y}_i|}Y_{ij})$, we have

$$\log\left[p(\{X_i,\mathcal{Y}_i\}|\kappa)\right]=\mathrm{trace}(\kappa G^{\mathrm{T}}S)+\sum_{i=1}^{n}\sum_{j=1}^{|\mathcal{Y}_i|}\log\left\{\frac{D(\kappa)-D(Y_{ij},\kappa)}{D(Y_{ij},\kappa)}\right\}-\tilde n\log D(\kappa).$$

Let $\tilde D(Y,\kappa)=\prod_{r=1}^{p}\|\kappa_r N_r^{\mathrm{T}}G_{[:r]}\|^{-(d-r-1)/2}I_{(d-r-1)/2}(\|\kappa_r N_r^{\mathrm{T}}G_{[:r]}\|)$, and $\tilde D(\kappa)=\prod_{r=1}^{p}\|\kappa_r\|^{-(d-r-1)/2}I_{(d-r-1)/2}(\|\kappa_r\|)$. Since $\mathrm{d}\{x^{-m}I_m(x)\}/\mathrm{d}x=x^{-m}I_{m+1}(x)$,

$$\frac{\mathrm{d}\tilde D(Y,\kappa)}{\mathrm{d}\kappa_j}=N_j^{\mathrm{T}}G_{[:j]}\tilde D(Y,\kappa)\frac{I_{(d-j+1)/2}}{I_{(d-j-1)/2}}(\kappa_j N_j^{\mathrm{T}}G_{[:j]}),\qquad\frac{\mathrm{d}\tilde D(\kappa)}{\mathrm{d}\kappa_j}=\tilde D(\kappa)\frac{I_{(d-j+1)/2}}{I_{(d-j-1)/2}}(\kappa_j).$$

Then, writing $L = \log p(\{X_i, \mathcal{Y}_i\} \mid \kappa)$, and $\tilde{D}'$ for $\mathrm{d}\tilde{D}/\mathrm{d}\kappa_k$, we have

$$\begin{aligned}\frac{\mathrm{d}L}{\mathrm{d}\kappa_k} &= G_{[:k]}^{\mathrm{T}} S_{[:k]} + \sum_{i=1}^{n} \sum_{j=1}^{|\mathcal{Y}_i|} \left\{ \frac{\tilde{D}'(\kappa) - \tilde{D}'(Y_{ij}, \kappa)}{\tilde{D}(\kappa) - \tilde{D}(Y_{ij}, \kappa)} - \frac{\tilde{D}'(Y_{ij}, \kappa)}{\tilde{D}(Y_{ij}, \kappa)} \right\} - \tilde{n} \frac{\tilde{D}'(\kappa)}{\tilde{D}(\kappa)} \\ &= G_{[:k]}^{\mathrm{T}} S_{[:k]} + \sum_{i=1}^{n} \sum_{j=1}^{|\mathcal{Y}_i|} \left\{ \frac{\frac{I_{(d-k+1)/2}(\kappa_k)}{I_{(d-k-1)/2}(\kappa_k)} - N_k^{\mathrm{T}} G_{[:k]} \frac{I_{(d-k+1)/2}(\kappa_k N_k^{\mathrm{T}} G_{[:k]})}{I_{(d-k-1)/2}(\kappa_k N_k^{\mathrm{T}} G_{[:k]})}}{1 - \tilde{D}(Y_{ij}, \kappa)/\tilde{D}(\kappa)} \right\} - \tilde{n} \frac{I_{(d-k+1)/2}(\kappa_k)}{I_{(d-k-1)/2}(\kappa_k)}.\end{aligned}$$

References

Aban, I. B., Meerschaert, M. M. & Panorska, A. K. (2006). Parameter estimation for the truncated Pareto distribution. *J. Am. Statist. Assoc.* **101**, 270–7.

Adams, R. P., Murray, I. & MacKay, D. J. C. (2009). The Gaussian process density sampler. In *Adv. Neural Info. Process. Syst. 21*, D. Koller, D. Schuurmans, Y. Bengio & L. Bottou, eds. Cambridge, MA: MIT Press, pp. 9–16.

Alai, D. H., Landsman, Z. & Sherris, M. (2013). Lifetime dependence modelling using a truncated multivariate Gamma distribution. *Insurance Math. Econom.* **52**, 542–9.

Andrieu, C. & Roberts, G. O. (2009). The pseudo-marginal approach for efficient Monte Carlo computations. *Ann. Statist.* **37**, 697–725.

Beskos, A., Papaspiliopoulos, O., Roberts, G. O. & Fearnhead, P. (2006). Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes. *J. R. Statist. Soc.* B **68**, 333–82.

Bladt, M. & Sørensen, M. (2014). Simple simulation of diffusion bridges with application to likelihood inference for diffusions. *Bernoulli* **20**, 645–75.

Brinkman, R. R., Gasparetto, M., Lee, S.-J. J., Ribickas, A. J., Perkins, J., Janssen, W., Smiley, R. & Smith, C. (2007). High-content flow cytometry and temporal data analysis for defining a cellular signature of graft-versus-host disease. *Biol. Blood Marrow Trans.* **13**, 691–700.

Chikuse, Y. (2003) *Statistics on Special Manifolds*. New York: Springer.

Dahlin, J., Lindsten, F. & Schön, T. B. (2015). Particle Metropolis–Hastings using gradient and Hessian information. *Statist Comp.* **25**, 81–92.

Downs, T. D., Liebman, J. & Mackay, W. (1971). Statistical methods for vectorcardiogram orientations. In *Vectorcardiography 2*: *Proc. XIth Intn. Symp. Vectorcardiography*, R. H. I. Hoffman & E. E. Glassman, eds. Amsterdam: North-Holland, pp. 216–22.

Edelman, A., Arias, T. A. & Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.* **20**, 303–53.

Goethals, K., Ampe, B., Berkvens, D., Laevens, H., Janssen, P. & Duchateau, L. (2009). Modeling interval-censored, clustered cow udder quarter infection times through the shared gamma frailty model. *J. Agric. Biol. Envir. Statist.* **14**, 1–14.

Hoff, P. D. (2009a). A hierarchical eigenmodel for pooled covariance estimation. *J. R. Statist. Soc.* B **71**, 971–92.

Hoff, P. D. (2009b). Simulation of the matrix Bingham–von Mises–Fisher distribution, with applications to multivariate and relational data. *J. Comp. Graph. Statist.* **18**, 438–56.

Ishwaran, H. & James, L. F. (2001). Gibbs sampling methods for stick-breaking priors. *J. Am. Statist. Assoc.* **96**, 161–73.

Jones, G. L. & Hobert, J. P. (2001). Honest exploration of intractable probability distributions via Markov chain Monte Carlo. *Statist. Sci.* **16**, 312–34.

Khatri, C. G. & Mardia, K. V. (1977). The von Mises–Fisher matrix distribution in orientation statistics. *J. R. Statist. Soc.* B **39**, 95–106.

Kume, A., Preston, S. P. & Wood, A. T. A. (2013). Saddlepoint approximations for the normalizing constant of Fisher–Bingham distributions on products of spheres and Stiefel manifolds. *Biometrika* **100**, 971–84.

Lee, G. & Scott, C. (2012). EM algorithms for multivariate Gaussian mixture models with truncated and censored data. *Comput. Statist. Data Anal.* **56**, 2816–29.

Liechty, M. W., Liechty, J. C. & Müller, P. (2009). The shadow prior. *J. Comp. Graph. Statist.* **18**, 368–83.

Little, M. A., McSharry, P. E., Roberts, S. J., Costello, D. A. E. & Moroz, I. M. (2007). Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *Biomed. Eng. Online*, **6**, 23.

Lo, A. (1984). On a class of Bayesian nonparametric estimates: I. Density estimates. *Ann. Statist.* **12**, 351–7.

Møller, J., Pettitt, A. N., Reeves, R. & Berthelsen, K. K. (2006). An efficient Markov chain Monte Carlo method for distributions with intractable normalising constants. *Biometrika* **93**, 451–8.

Murray, I., Ghahramani, Z. & MacKay, D. J. C. (2006). MCMC for doubly-intractable distributions. In *Proc. 22nd Conf. Uncert. Artif. Intell.* AUAI Press, pp. 359–66.

Murray, I., Adams, R. P. & MacKay, D. J. (2010). Elliptical slice sampling. *J. Mach. Learn. Res.* **9**, 541–8.

NEAL, R. M. (2010). MCMC using Hamiltonian dynamics. In *Handbook of Markov Chain Monte Carlo*, S. P. Brooks, A. Gelman, G. L. Jones, & X.-L. Meng, eds. Boca Raton, Florida: Chapman & Hall/CRC Press, pp. 113–62.

PLUMMER, M., BEST, N., COWLES, K. & VINES, K. (2006). CODA: Convergence diagnosis and output analysis for MCMC. *R News* **6**, 7–11.

ROBERT, C. P. & CASELLA, G. (2005). *Monte Carlo Statistical Methods*. New York: Springer, 2nd ed.

R DEVELOPMENT CORE TEAM, (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. http://www.R-project.org.

WALKER, S. G. (2011). Posterior sampling when the normalizing constant is unknown. *Commun. Statist.* B **40**, 784–92.

WOOD, A. T. A. (1994). Simulation of the von Mises–Fisher distribution. *Commun. Statist.* B **23**, 157–64.

[*Received June* 2014. *Revised January* 2016]