Riemann Manifold Langevin and Hamiltonian Monte Carlo

Mark Girolami

Department of Computing Science, University of Glasgow, UK

Department of Statistics, University of Glasgow, UK

Ben Calderhead

Department of Computing Science, University of Glasgow, UK

Siu A. Chin

Department of Physics, Texas A&M University, Texas, USA

Summary. This paper proposes Metropolis adjusted Langevin and Hamiltonian Monte Carlo sampling methods defined on the Riemann manifold to resolve the shortcomings of existing Monte Carlo algorithms when sampling from target densities that may be high dimensional and exhibit strong correlations. The methods provide fully automated adaptation mechanisms that circumvent the costly pilot runs required to tune proposal densities for Metropolis-Hastings or indeed Hamiltonian Monte Carlo and Metropolis Adjusted Langevin Algorithms. This allows for highly efficient sampling even in very high dimensions where different scalings may be required for the transient and stationary phases of the Markov chain. The proposed methodology exploits the Riemannian geometry of the parameter space of statistical models and thus automatically adapts to the local structure when simulating paths across this manifold providing highly efficient convergence and exploration of the target density. The performance of these Riemannian Manifold Monte Carlo methods is rigorously assessed by performing inference on logistic regression models, log-Gaussian Cox point processes, stochastic volatility models, and Bayesian estimation of dynamical systems described by nonlinear differential equations. Substantial improvements in the time normalised Effective Sample Size are reported when compared to alternative sampling approaches. Matlab code at `http://www.dcs.gla.ac.uk/inference/rmhmc` allows replication of all results reported.

1. Introduction

For an unnormalised probability density function, $\tilde{p}(\boldsymbol{\theta})$ where $\boldsymbol{\theta} \in \mathbb{R}^D$, the normalised density follows as $p(\boldsymbol{\theta}) = \tilde{p}(\boldsymbol{\theta}) / \int \tilde{p}(\boldsymbol{\theta}) d\boldsymbol{\theta}$, which for many statistical models is analytically intractable. Monte Carlo estimates of integrals with respect to $p(\boldsymbol{\theta})$, which commonly appear in Bayesian statistics, are therefore required. The predominant methodology for sampling from such a probability density is Markov chain Monte Carlo (MCMC) see e.g. (Robert, 2004; Gelman *et al.*, 2004; Liu, 2001). The most general algorithm defining a Markov process with invariant density $p(\boldsymbol{\theta})$ is the *Metropolis-Hastings* algorithm (Metropolis *et al.*, 1953; Hastings, 1970), which is arguably one of the *most successful and influential* Monte Carlo algorithms (Beichl and Sullivan, 2000) .

The Metropolis-Hastings algorithm proposes transitions $\boldsymbol{\theta} \mapsto \boldsymbol{\theta}^*$ with density $q(\boldsymbol{\theta}^*|\boldsymbol{\theta})$, which are then accepted with probability $\alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*) = \min\{1, p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta}|\boldsymbol{\theta}^*)/p(\boldsymbol{\theta})q(\boldsymbol{\theta}^*|\boldsymbol{\theta})\}$. This acceptance probability ensures that the Markov chain is reversible with respect to the stationary target density $p(\boldsymbol{\theta})$ and satisfies detailed balance, see for example Robert (2004); Neal (1993a, 1996); Liu (2001). Typically, the proposal distribution $q(\boldsymbol{\theta}^*|\boldsymbol{\theta})$ which drives the Markov chain takes the form of a random walk, e.g. $q(\boldsymbol{\theta}^*|\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}^*|\boldsymbol{\theta}, \boldsymbol{\Lambda})$ is a D-dimensional Normal distribution with mean $\boldsymbol{\theta}$ and covariance $\boldsymbol{\Lambda}$.

High acceptance rates can be achieved by proposing smaller transitions, however larger amounts of time will then be required to make long traversals of parameter space. In high dimensions, when D is large, the random walk becomes inefficient resulting in low acceptance rates, poor mixing of the chain and highly correlated samples. A consequence of this is a small effective sample size (ESS) from the chain, see Robert (2004); Neal (1996); Liu (2001). Whilst there have been a number of suggestions to overcome this inefficiency, guaranteeing detailed balance and ergodicity of the chain places constraints on what can be achieved in alleviating this problem (Andrieu and Thoms, 2008; Robert, 2004; Neal, 1993a). Design of a good general purpose proposal mechanism providing large proposal transitions that are accepted with high probability remains something of an engineering art-form.

Major steps forward in this regard were made when a proposal process derived from a discretised Langevin diffusion with a drift term based on the gradient information of the target density was suggested in the Metropolis Adjusted Langevin Algorithm (MALA) (Roberts and Stramer, 2003). Likewise the Hamiltonian Monte Carlo (HMC) method (Duane *et al.*, 1987) was proposed in the statistical physics literature as a means of efficiently simulating states from a physical system which was then applied to problems of statistical inference (Neal, 1993a,b, 1996; Liu, 2001). In HMC, a deterministic proposal process based on Hamiltonian dynamics is employed along with additional stochastic proposals that together provide an ergodic Markov chain capable of making large transitions that are accepted with high probability.

Despite the potential efficiency gains to be obtained in MCMC sampling from such proposal mechanisms inherent in MALA and HMC, the tuning of these MCMC methods remains a major issue especially for challenging inference problems. This paper seeks to address these issues in a systematic manner by adopting an overarching geometric framework for the overall development of MCMC methods such as these.

A brief review of MALA and HMC within the context of statistical inference are provided in the following two sections. In Section 4 differential geometric concepts employed in the study of asymptotic statistics are considered within the context of MCMC methodology. Section 5 proposes a generalisation of MALA that takes into account the natural geometry of the target density making use of the definition of a Langevin diffusion on a Riemann manifold. Likewise in Section 6 a generalisation of HMC, Riemann manifold HMC (RM-HMC) is presented, which takes advantage of the manifold structure of the parameter space and allows for more efficient proposal transitions to be made. Finally, in Sections 7 to 10, this new methodology is demonstrated and assessed on a number of interesting statistical problems, i.e. Bayesian logistic regression, stochastic volatility modeling, log-Gaussian Cox point processes, and parameter inference in dynamical systems.

2. Metropolis Adjusted Langevin Algorithm

Consider the random vector $\boldsymbol{\theta} \in \mathbb{R}^D$ with density $p(\boldsymbol{\theta})$ and denote the log density as $\mathcal{L}(\boldsymbol{\theta}) \equiv \log p(\boldsymbol{\theta})$, then the Metropolis Adjusted Langevin Algorithm (MALA) is based on a Langevin diffusion, with stationary distribution $p(\boldsymbol{\theta})$, defined by the stochastic differential equation (SDE)

$$d\boldsymbol{\theta}(t) = \frac{1}{2}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}(t))dt + d\mathbf{b}(t) \tag{1}$$

where $\mathbf{b}$ denotes a D-dimensional Brownian motion. A first-order Euler discretisation of the SDE gives the following proposal mechanism

$$\boldsymbol{\theta}^{n+1} = \boldsymbol{\theta}^n + \frac{\epsilon^2}{2}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}^n) + \epsilon\mathbf{z}^n \tag{2}$$

where $\mathbf{z} \sim \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I})$ and ϵ is the integration step size. Convergence to the invariant distribution, $p(\boldsymbol{\theta})$, is no longer guaranteed for finite step size ϵ due to the first-order integration error introduced. This discrepancy can be corrected by employing a Metropolis acceptance probability after each integration step thus ensuring convergence to the invariant measure. As $\mathbf{z}$ is an isotropic standardised Normal variate and denoting $\boldsymbol{\mu}(\boldsymbol{\theta}^n, \epsilon) = \boldsymbol{\theta}^n + \frac{\epsilon^2}{2}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}^n)$ then the discrete form of the SDE (2) defines a proposal density $q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^n) = \mathcal{N}(\boldsymbol{\theta}^*|\boldsymbol{\mu}(\boldsymbol{\theta}^n, \epsilon), \epsilon^2\mathbf{I})$ with acceptance probability of standard form $\min\{1, p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta}^n|\boldsymbol{\theta}^*)/p(\boldsymbol{\theta}^n)q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^n)\}$.

The optimal scaling, ϵ, for MALA has been theoretically analysed in the limit as $D \to \infty$ for factorisable $p(\boldsymbol{\theta})$, (Roberts and Rosenthal, 1998). Although the drift term in the proposal mechanism for MALA in (2) defines the direction for the proposal based on the gradient information (albeit the Euclidean form) it is clear that the isotropic diffusion will be inefficient for strongly correlated variables $\boldsymbol{\theta}$ with widely differing variances forcing the stepsize to accommodate the variate with smallest variance. This issue can be circumvented by employing a pre-conditioning matrix, $\mathbf{M}$, such that $\boldsymbol{\theta}^{n+1} = \boldsymbol{\theta}^n + \epsilon^2\mathbf{M}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}^n)/2 + \epsilon\sqrt{\mathbf{M}}\mathbf{z}^n$ (Roberts and Stramer, 2003). It is unclear how this should be defined in any principled manner, indeed a global level of pre-conditioning may well be inappropriate for differing transient and stationary regimes of the Markov process as demonstrated in (Christensen *et al.*, 2005).

3. Hamiltonian Monte Carlo

We now give a brief introduction to the Hamiltonian Monte Carlo method, for a detailed description and extensive review see (Neal, 2010). As in the previous section consider the random variable $\boldsymbol{\theta} \in \mathbb{R}^D$ with density $p(\boldsymbol{\theta})$. In HMC an independent auxiliary variable $\mathbf{p} \in \mathbb{R}^D$ with density $p(\mathbf{p}) = \mathcal{N}(\mathbf{p}|\mathbf{0}, \mathbf{M})$ is introduced. The joint density follows in factorised form as $p(\boldsymbol{\theta}, \mathbf{p}) = p(\boldsymbol{\theta})p(\mathbf{p}) = p(\boldsymbol{\theta})\mathcal{N}(\mathbf{p}|\mathbf{0}, \mathbf{M})$. Denoting the log of the desired density as $\mathcal{L}(\boldsymbol{\theta}) \equiv \log p(\boldsymbol{\theta})$, the negative joint log-likelihood is

$$H(\boldsymbol{\theta}, \mathbf{p}) = -\mathcal{L}(\boldsymbol{\theta}) + \frac{1}{2}\log\left((2\pi)^D|\mathbf{M}|\right) + \frac{1}{2}\mathbf{p}^{\mathsf{T}}\mathbf{M}^{-1}\mathbf{p} \tag{3}$$

The physical analogy of this negative joint log-likelihood is a Hamiltonian (Duane *et al.*, 1987; Leimkuhler and Reich, 2004), which describes the sum of a potential energy function $-\mathcal{L}(\boldsymbol{\theta})$ defined at the position $\boldsymbol{\theta}$, and a kinetic energy term $\mathbf{p}^{\mathsf{T}}\mathbf{M}^{-1}\mathbf{p}/2$ where the auxiliary variable $\mathbf{p}$ is interpreted as a momentum variable and the covariance matrix $\mathbf{M}$ denotes a mass matrix.

The score function with respect to $\boldsymbol{\theta}$ and $\mathbf{p}$, of the log joint density over the two random variables has a physical interpretation as the time evolution, with respect to a fictitious time τ, of the dynamic system as given by Hamilton's equations,

$$\frac{d\boldsymbol{\theta}}{d\tau} = \frac{\partial H}{\partial \mathbf{p}} = \mathbf{M}^{-1}\mathbf{p} \qquad \frac{d\mathbf{p}}{d\tau} = -\frac{\partial H}{\partial \boldsymbol{\theta}} = \nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}) \tag{4}$$

The solution flow for the differential equations, $(\boldsymbol{\theta}(\tau), \mathbf{p}(\tau)) = \Phi_\tau(\boldsymbol{\theta}(0), \mathbf{p}(0))$, (a) preserves the total energy i.e. $H(\boldsymbol{\theta}(\tau), \mathbf{p}(\tau)) = H(\boldsymbol{\theta}(0), \mathbf{p}(0))$ and hence the joint density $p(\boldsymbol{\theta}(\tau), \mathbf{p}(\tau)) = p(\boldsymbol{\theta}(0), \mathbf{p}(0))$, (b) preserves the volume element $d\boldsymbol{\theta}(\tau)d\mathbf{p}(\tau) = d\boldsymbol{\theta}(0)d\mathbf{p}(0)$, and (c) is time reversible (Leimkuhler and Reich, 2004). For practical applications of interest the differential equations (4) cannot be solved analytically and numerical methods are required. There is a class of numerical integrators for Hamiltonian systems which will fully satisfy the criteria (b) and (c), volume preservation and time reversibility, and approximately satisfy (a) energy conservation to a given order of error, see (Leimkuhler and Reich, 2004). The Stormer-Verlet or Leapfrog integrator was employed in the original paper of Duane *et al.* (1987), and in various statistical applications e.g. (Liu, 2001; Neal, 1993b, 2010) as described below,

$$\begin{aligned}\mathbf{p}(\tau+\epsilon/2) &= \mathbf{p}(\tau)+\epsilon\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}(\tau))/2 &(5)\\ \boldsymbol{\theta}(\tau+\epsilon) &= \boldsymbol{\theta}(\tau)+\epsilon\mathbf{M}^{-1}\mathbf{p}(\tau+\epsilon/2) &(6)\\ \mathbf{p}(\tau+\epsilon) &= \mathbf{p}(\tau+\epsilon/2)+\epsilon\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}(\tau+\epsilon))/2 &(7)\end{aligned}$$

Since the joint likelihood is factorisable (i.e. in physical terms, the Hamiltonian is separable), it is obvious by inspection that each complete Leapfrog step (equations (5), (6) and (7)) is reversible by the negation of the integration step-size, ϵ. Likewise as the Jacobians of the transformations $(\boldsymbol{\theta},\mathbf{p})\mapsto(\boldsymbol{\theta},\mathbf{p}+\epsilon\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta})/2)$ and $(\boldsymbol{\theta},\mathbf{p})\mapsto(\boldsymbol{\theta}+\epsilon\mathbf{M}^{-1}\mathbf{p},\mathbf{p})$ have unit determinant then volume is preserved. As total energy is only approximately conserved with the Stormer-Verlet integrator then a corresponding bias is introduced into the joint density which can be corrected by an accept-reject step. Due to the volume preserving property of the integrator the determinant of the Jacobian matrix for the mapping defined by Φ_τ does not need to be taken into account in the Hastings ratio of the acceptance probability. Therefore for a mapping $(\boldsymbol{\theta},\mathbf{p})\mapsto(\boldsymbol{\theta}+\delta\boldsymbol{\theta},\mathbf{p}+\delta\mathbf{p})=(\boldsymbol{\theta}^*,\mathbf{p}^*)$ obtained from a number of Stormer-Verlet integration steps the corresponding acceptance probability is $\text{min}[1,\exp\{-H(\boldsymbol{\theta}^*,\mathbf{p}^*)+\mathbf{H}(\boldsymbol{\theta},\mathbf{p})\}]$, and due to the reversibility of the dynamics the joint density and hence the marginals $p(\boldsymbol{\theta})$ and $p(\mathbf{p})$ are left invariant. If the integration error in the total energy is small then the acceptance probability will remain at a high level.

The Stormer-Verlet integration steps provide a deterministic proposal mechanism such that $\boldsymbol{\theta}^*=\boldsymbol{\theta}+\delta\boldsymbol{\theta}$ and $\mathbf{p}^*=\mathbf{p}+\delta\mathbf{p}$ and overall HMC sampling from the invariant density $p(\boldsymbol{\theta})$ can be considered as a Gibbs sampler where the momentum $\mathbf{p}$ acts simply as an auxiliary variable

$$\begin{aligned}\mathbf{p}|\boldsymbol{\theta} &\sim p(\mathbf{p}|\boldsymbol{\theta})=p(\mathbf{p})=\mathcal{N}(\mathbf{p}|\mathbf{0},\mathbf{M}) &(8)\\ \boldsymbol{\theta}^*|\mathbf{p} &\sim p(\boldsymbol{\theta}^*|\mathbf{p})\propto\exp\left(-H(\boldsymbol{\theta}^*,\mathbf{p}+\delta\mathbf{p})\right) &(9)\end{aligned}$$

where samples from $p(\boldsymbol{\theta}^*|\mathbf{p})$ are obtained by running the Stormer-Verlet integrator for a certain number of steps to give proposed moves $\boldsymbol{\theta}^*$ and $\mathbf{p}^*$ and accepting or rejecting with probability $\text{min}[1,\exp\{-H(\boldsymbol{\theta}^*,\mathbf{p}^*)+\mathbf{H}(\boldsymbol{\theta},\mathbf{p})\}]$. This Gibbs sampling scheme produces an ergodic, time reversible Markov chain satisfying detailed balance whose stationary marginal density is $p(\boldsymbol{\theta})$ (Duane *et al.*, 1987; Liu, 2001; Neal, 1996, 2010).

It should be noted that the combination of equations (5) and (6) in a single step of the integrator yields an update of the form

$$\boldsymbol{\theta}(\tau+\epsilon)=\boldsymbol{\theta}(\tau)+\frac{\epsilon^2}{2}\mathbf{M}^{-1}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}(\tau))+\epsilon\mathbf{M}^{-1}\mathbf{p}(\tau) \quad (10)$$

which is nothing more than a discrete pre-conditioned Langevin diffusion as employed in MALA (Roberts and Stramer, 2003) (see Neal (1993a, 1996, 2010) for further discussion on this point). Viewed in this form it is clear that the choice of the mass matrix, $\mathbf{M}$, as in MALA, is going to be critical for the performance of HMC, and like MALA there is no guiding principle as to how this should be chosen and tuned.

The demonstrated ability of HMC to overcome random walks in MCMC sampling suggests it should be a highly successful tool for Bayesian inference. A study suggests in excess of 300 citations of the original (Duane *et al.*, 1987) paper within the literature devoted to Molecular Modelling and Simulation, Physics and Chemistry. However there are a much smaller number of citations in the literature devoted to Statistical Methodology and Application, e.g. (Liu, 2001; Neal, 1996, 1993b;

Gustafson, 1997; Ishwaran, 1999; Husmeier *et al*, 1999; Hanson, 2001), indicating that it has not been widely adopted as a practical inference method.

Whilst the choice of the step size ϵ and number of integration steps can be tuned based on the overall acceptance rate of the HMC sampler, as already mentioned it is unclear how to select the values of the weight matrix $\mathbf{M}$ in any automated or principled manner that does not require some knowledge of the target density, similar to the situation with MALA. Although rules of thumb are suggested (Liu, 2001; Neal, 1993a, 1996, 2010) these typically rely on knowledge of the marginal variance of the target density, which is of course not known at the time of simulation and thus requires preliminary pilot runs of HMC, this is also the case for MALA although asymptotic settings are suggested in Christensen *et al.* (2005). The experimental sections of this paper will demonstrate how crucial this tuning is to obtain acceptable performance of HMC and MALA.

The potential of both the MALA and HMC methodology may be more fully realised by employing transitions that take into account the *local structure* of the target density when proposing moves to different likelihood regions, as this may improve the overall mixing of the chain. Therefore rather than employing a fixed global covariance matrix in the proposal density $\mathcal{N}(\mathbf{p}|\mathbf{0}, \mathbf{M})$, a position specific covariance could be adopted. Furthermore, the *deterministic* proposal mechanism of HMC, when viewed as the deterministic component of the discrete pre-conditioned Langevin diffusion, equation (10), relies on the likelihood gradient pre-conditioned by the inverse of a globally constant mass matrix. We turn our attention now to geometric concepts which will be shown to be of fundamental importance in addressing these shortcomings.

4. Exploiting Geometric Concepts in MCMC

The relationship between differential geometry and statistics has recently been employed in the development of, primarily asymptotic, statistical theory see e.g. (Amari and Nagaoka, 2000; Kass, 1989; Murray and Rice, 1993; Barndorff-Nielsen *et al*, 1986; Critchley *et al*, 1993; Lauritzen, 1987; Dawid, 1975; Efron, 1975). Geometric concepts of distance, curvature, manifolds, geodesics (shortest paths between two points), and invariants are of natural interest in statistical methodology and in the following we shall exploit some of these in the development of MCMC methods.

The formal definition of distance between two density functions first appeared in (Rao, 1945) with the same result appearing later in (Jeffreys, 1948). A distance metric based on a first order expression for the symmetric Kullback Liebler divergence between two densities p and q, $D_{\mathcal{S}}(p||q) = D(p||q) + D(q||p)$ was derived. Noting that to first order $p(\mathbf{y}; \boldsymbol{\theta} + \delta\boldsymbol{\theta}) = p(\mathbf{y}; \boldsymbol{\theta}) + \delta\boldsymbol{\theta}^{\mathsf{T}}\nabla_{\boldsymbol{\theta}} p(\mathbf{y}; \boldsymbol{\theta}) + \mathcal{O}(2)$ and as $\log(1 + \epsilon) \approx \epsilon$ then $D_{\mathcal{S}}(p(\mathbf{y}; \boldsymbol{\theta} + \delta\boldsymbol{\theta})||p(\mathbf{y}; \boldsymbol{\theta}))$ is

$$\delta\boldsymbol{\theta}^{\mathsf{T}} E_{\mathbf{y}|\boldsymbol{\theta}} \left\{ \nabla_{\boldsymbol{\theta}} \log p(\mathbf{y}; \boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \log p(\mathbf{y}; \boldsymbol{\theta})^{\mathsf{T}} \right\} \delta\boldsymbol{\theta} = \delta\boldsymbol{\theta}^{\mathsf{T}} \mathbf{G}(\boldsymbol{\theta}) \delta\boldsymbol{\theta} \tag{11}$$

where $\mathbf{G}(\boldsymbol{\theta})$ is the Fisher Information matrix. Rao noted that as the matrix $\mathbf{G}(\boldsymbol{\theta})$ is by definition positive definite it is a metric of a Riemannian manifold. Therefore the space of probability density functions is endowed with a natural, Riemannian, geometry. Given this geometry Rao went further and showed that expressions for the curvature of the manifold and shortest paths (geodesics) on the manifold between two densities could, in principle, be derived (Rao, 1945). These ideas have been formalised in the study of *Information Geometry* (Amari and Nagaoka, 2000).

It should be noted that the Fisher metric also emerges from purely geometric arguments (Skilling, 2006) and it is straightforward to show for a probability simplex, $p^i \geq 0$, $\sum_{i=1}^{D} p^i = 1$ the metric is $g_{ij} = \delta_{ij}/p^i$ where $\delta_{ij} = 1$ iff $i = j$. It then follows that a small displacement δl has length $(\delta l)^2 = \sum_{i,j} \delta p^i \delta p^j g_{ij} = \sum_i (\delta p^i)^2/p^i$ which is nothing more than the Fisher Information for a discrete probability distribution, suggesting this as the fundamental metric for probability spaces.

However it can be argued that the choice of metric is problem dependent, for example the requirement for asymmetry in statistical inference is captured in the Preferred Point metric and associated geometry (Critchley *et al*, 1993). As a Bayesian perspective is being adopted in this paper, the examples reported employ the joint likelihood of data and parameters when defining the metric tensor i.e. $-E_{\mathbf{y}|\boldsymbol{\theta}}\left\{\partial^2/\partial\boldsymbol{\theta}^2 \log p(\mathbf{y},\boldsymbol{\theta})\right\}$ which is the Fisher Information plus the negative Hessian of the log-prior. For further discussion on ways to capture prior informativeness in the metric tensor see e.g. (Tsutakawa, 1972; Ferreira, 1981). This freedom to choose the metric does however open up a new line of investigation regarding the intrinsic geometry obtained by the choice and design of metrics and the characteristics which make them appropriate for specific MCMC applications.

In summary the parameter space of a statistical model is a Riemannian manifold. Therefore the natural geometric structure of the density model $p(\boldsymbol{\theta})$ is defined by the Riemannian manifold and associated metric tensor. Given this geometric structure of the parameter space of statistical models the appropriate adoption of the position specific metric, $\mathbf{G}(\boldsymbol{\theta})$, within an MCMC scheme should yield more effective transitions in the overall algorithm. We now show how the Riemannian manifold structure may be exploited within a correct MCMC framework for the Metropolis Adjusted Langevin Algorithm.

5. Riemann Manifold Metropolis Adjusted Langevin Algorithm

Given the geometric structure for probability models a Langevin diffusion with invariant measure $p(\boldsymbol{\theta}), \boldsymbol{\theta} \in \mathbb{R}^D$ can be defined directly upon the Riemannian manifold with metric tensor $\mathbf{G}(\boldsymbol{\theta})$ (Roberts and Stramer, 2003; Chung, 1982; Kent, 1978). The stochastic differential equation defining the Langevin diffusion on the manifold is

$$d\boldsymbol{\theta}(t) = \frac{1}{2}\tilde{\nabla}_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}(t))dt + d\tilde{\mathbf{b}}(t) \tag{12}$$

where the natural gradient (Amari and Nagaoka, 2000) is $\tilde{\nabla}_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}(t)) = \mathbf{G}^{-1}(\boldsymbol{\theta}(t))\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}(t))$ and the Brownian motion on the Riemannian manifold follows as

$$d\tilde{\mathbf{b}}_i(t) = |\mathbf{G}(\boldsymbol{\theta}(t))|^{-1/2}\sum_{j=1}^{D}\frac{\partial}{\partial\boldsymbol{\theta}_j}\left(\mathbf{G}^{-1}(\boldsymbol{\theta}(t))_{ij}|\mathbf{G}(\boldsymbol{\theta}(t))|^{1/2}\right)dt + \left(\sqrt{\mathbf{G}^{-1}(\boldsymbol{\theta}(t))}d\mathbf{b}(t)\right)_i \tag{13}$$

Clearly in a Euclidean space where the metric tensor is an identity matrix then (12) reduces to the standard form of SDE (1). The first term on the right hand side of (13) relates to the local curvature of the manifold and reduces to zero if curvature is everywhere constant. The second right hand term provides a position specific axis alignment of the Brownian motion based on the local metric by transformation of the independent Brownian motion, $\mathbf{b}(t)$.

The discrete form of the above SDE employing a first order Euler integrator follows as

$$\begin{aligned}
\boldsymbol{\theta}_i^{n+1} &= \boldsymbol{\theta}_i^n + \frac{\epsilon^2}{2}\left(\mathbf{G}^{-1}(\boldsymbol{\theta}^n)\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}^n)\right)_i - \epsilon^2\sum_{j=1}^{D}\left(\mathbf{G}^{-1}(\boldsymbol{\theta}^n)\frac{\partial\mathbf{G}(\boldsymbol{\theta}^n)}{\partial\boldsymbol{\theta}_j}\mathbf{G}^{-1}(\boldsymbol{\theta}^n)\right)_{ij} \\
&+ \frac{\epsilon^2}{2}\sum_{j=1}^{D}\left(\mathbf{G}^{-1}(\boldsymbol{\theta}^n)\right)_{ij} Tr\left(\mathbf{G}^{-1}(\boldsymbol{\theta}^n)\frac{\partial\mathbf{G}(\boldsymbol{\theta}^n)}{\partial\boldsymbol{\theta}_j}\right) + \left(\epsilon\sqrt{\mathbf{G}^{-1}(\boldsymbol{\theta}^n)}\mathbf{z}^n\right)_i \\
&= \boldsymbol{\mu}(\boldsymbol{\theta}^n,\epsilon)_i + \left(\epsilon\sqrt{\mathbf{G}^{-1}(\boldsymbol{\theta}^n)}\mathbf{z}^n\right)_i
\end{aligned}$$

defining a proposal mechanism with density $q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^n) = \mathcal{N}(\boldsymbol{\theta}^*|\boldsymbol{\mu}(\boldsymbol{\theta}^n, \epsilon), \epsilon^2\mathbf{G}^{-1}(\boldsymbol{\theta}^n))$ and acceptance probability $\min\{1, p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta}^n|\boldsymbol{\theta}^*)/p(\boldsymbol{\theta}^n)q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^n)\}$ to ensure convergence to the invariant density $p(\boldsymbol{\theta})$. Immediately it is clear that the proposal mechanism makes moves approximately along the manifold embedded in $\mathbb{R}^D$ rather than the D-dimensional Euclidean space and these moves respect the curvature at each point of the manifold. Pseudo-code describing the full manifold MALA (mMALA) scheme is given in Appendix (D). For a flat manifold with constant curvature this reduces further to a position specific pre-conditioned MALA proposal.

$$\boldsymbol{\theta}^{n+1} \quad = \quad \boldsymbol{\theta}^n + \frac{\epsilon^2}{2}\mathbf{G}^{-1}(\boldsymbol{\theta}^n)\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}^n) + \epsilon\sqrt{\mathbf{G}^{-1}(\boldsymbol{\theta}^n)}\mathbf{z}^n$$

Of course even if the curvature of the manifold is not constant the above simplified proposal mechanism, used in conjunction with the acceptance probability, will still define a correct MCMC method which converges to the target measure. However dependent on the characteristics of the curvature the proposal process may not be so efficient in converging to the stationary distribution and this shall be explored further in the experimental evaluation. To illustrate this geometric approach and gain some insight into mMALA a simple example is now given.

5.1. An Illustrative Example: The Normal Distribution as Invariant Density

For N observations drawn from the Normal distribution $\mathcal{N}(x|\mu, \sigma)$ the metric tensor based on the Fisher Information is

$$\mathbf{G}(\mu, \sigma) = \begin{pmatrix} N/\sigma^2 & 0 \\ 0 & 2N/\sigma^2 \end{pmatrix} \tag{14}$$

and this defines a Riemann manifold with constant curvature which is a Hyperbolic space on the upper-half plane defined by the horizontal and vertical coordinates μ and σ (Amari and Nagaoka, 2000) . The distance between two densities $\mathcal{N}(x|\mu, \sigma)$ and $\mathcal{N}(x|\mu + \delta\mu, \sigma + \delta\sigma)$ as defined on this manifold is $(\delta\mu^2 + 2\delta\sigma^2)/\sigma^2$ indicating that as the value of σ increases the distance between the densities decreases. The first-order Euler approximations for the standard Langevin diffusion with invariant measure proportional to $\prod_l \mathcal{N}(x_l|\mu, \sigma)$ follows as

$$\mu_{n+1} \quad = \quad \mu_n + \frac{\epsilon^2}{2\sigma_n^2}\sum_l (x_l - \mu_n) + \epsilon z_n \tag{15}$$

$$\sigma_{n+1} \quad = \quad \sigma_n + \frac{\epsilon^2}{2\sigma_n^3}\sum_l (x_l - \mu_n)^2 - \frac{N\epsilon^2}{2\sigma_n} + \epsilon z \tag{16}$$

When the diffusion is defined on the Riemann manifold specified by the metric tensor (14) then the approximate diffusion follows as

$$\mu_{n+1} \quad = \quad \mu_n + \frac{\epsilon_m^2}{2N}\sum_l (x_l - \mu_n) + \frac{\epsilon_m \sigma_n}{\sqrt{N}} z_n \tag{17}$$

$$\sigma_{n+1} \quad = \quad \sigma_n + \frac{\epsilon_m^2}{4N\sigma_n}\sum_l (x_l - \mu_n)^2 - \frac{\epsilon_m^2 \sigma_n}{4} + \frac{\epsilon_m \sigma_n}{\sqrt{2N}} z_n \tag{18}$$

The discrete diffusion based on a Euclidean metric (15, 16) has a diffusion term ϵz_n whose scaling is fixed by the integration step size ϵ irrespective of position. On the other hand the approximate

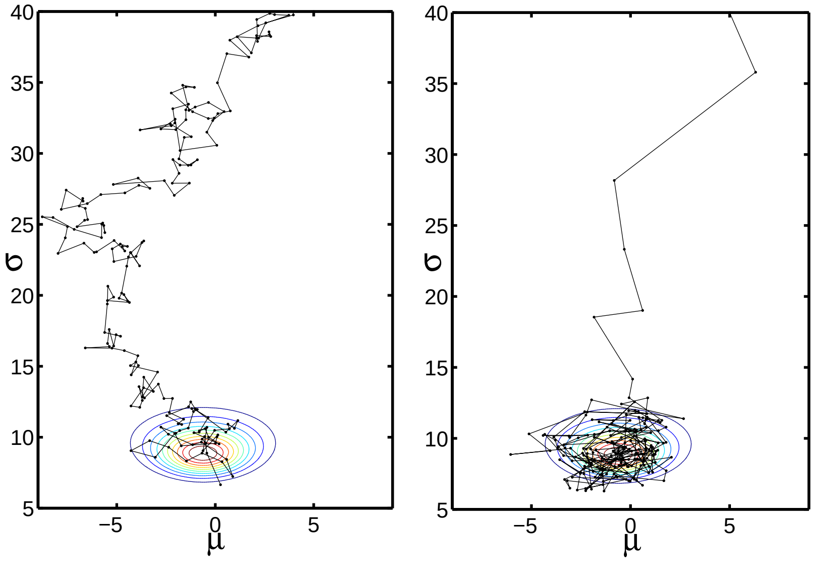

Fig. 1. The above contours represent the sample estimate of $p(\mu, \sigma|X)$ where a sample of size $N = 30$ was drawn from $\mathcal{N}(X|\mu = 0, \sigma = 10)$. Both MALA and mMALA discrete diffusions were forward simulated from initial points $\mu_0 = 5$ and $\sigma_0 = 40$ with a step size $\epsilon = 0.75$ for 200 steps. The left-hand panel shows a sample path of the MALA proposal process. As the space is hyperbolic and a Euclidean metric is employed the proposals take inefficient steps of almost equal length thoughout. On the other hand the mMALA proposals, right hand pane, are defined based on the metric for the Hyperbolic space with constant negative curvature and as such the distances covered by each step reflect the natural distances on the manifold resulting in much more efficient traversal of the space.

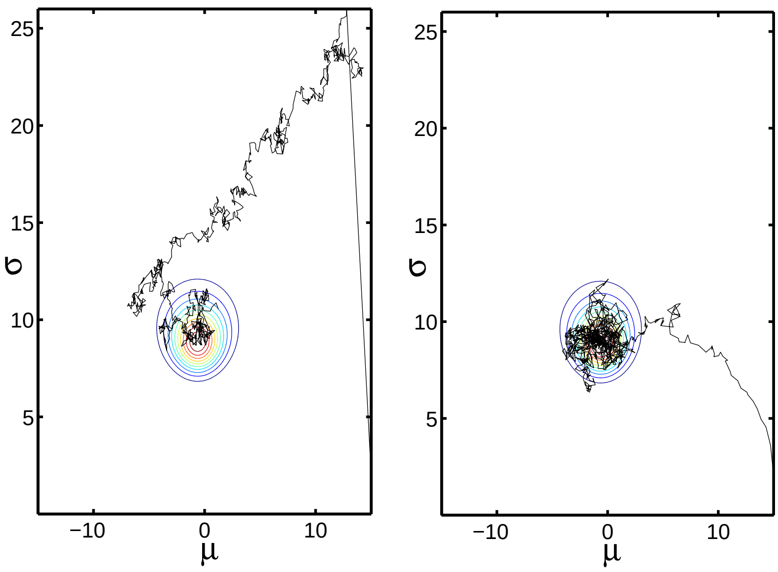

Fig. 2. In this example the same data sample is used and initial starting points are $\mu_0 = 15$ and $\sigma_0 = 2$. The step size is reduced to $\epsilon = 0.2$ in order that MALA converges and 1000 proposal steps are taken. As previously in the left hand panel it is clear that the Euclidean metric of MALA does not exploit the Hyerbolic geometry and overshoots dramatically at the start, whereas in the right hand panel it is clear that mMALA converges efficiently due to the exploitation of the metric.

Langevin diffusion obtained by employing the Riemannian metric tensor (17, 18) produces a term $\epsilon_m \sigma_n z_n / \sqrt{N}$ for the mean parameter and $\epsilon_m \sigma_n z_n / \sqrt{2N}$ for the variance which are position dependent thus ensuring appropriate scaling of the diffusion. The integration step size ϵ_m is effectively dimensionless whilst ϵ requires dimension proportional to σ_n thus indicating proposal inefficiency with ϵ set at a fixed value as demonstrated in Figures (1) and (2). Extensive detailed investigation of the performance of mMALA will be provided in the experimental sections.

6. Riemann Manifold Hamiltonian Monte Carlo

Following on from the previous section the Hamiltonian which forms the basis of HMC will now be defined in general form on a Riemann manifold. Zlochin and Baram (2001) originally attempted to exploit this manifold structure in HMC however their use of a numerical integration method that did not guarantee reversibility or volume preservation prevented them from developing a correct MCMC procedure.

The definition of the Hamiltonian on a Riemann manifold is straightforward and is a technique employed in geometric mechanics to solve partial differential equations (Calin and Chang, 2004). From equation (4), it follows that $\mathbf{p} = \mathbf{M}\dot{\boldsymbol{\theta}}$, so the norm of each $\dot{\boldsymbol{\theta}}$ under the metric $\mathbf{M}$ is $\|\dot{\boldsymbol{\theta}}\|^2_{\mathbf{M}} = \dot{\boldsymbol{\theta}}^{\mathsf{T}}\mathbf{M}\dot{\boldsymbol{\theta}} = \mathbf{p}^{\mathsf{T}}\mathbf{M}^{-1}\mathbf{p}$. In a more general form, as the statistical model is defined on a Riemannian manifold, the metric tensor defines the position specific norm such that $\|\dot{\boldsymbol{\theta}}\|^2_{\mathbf{G}(\boldsymbol{\theta})} = \dot{\boldsymbol{\theta}}^{\mathsf{T}}\mathbf{G}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} = \mathbf{p}^{\mathsf{T}}\mathbf{G}^{-1}(\boldsymbol{\theta})\mathbf{p}$ and thus the kinetic energy term can be defined via the inverse metric (Calin and Chang, 2004). In order to ensure that the Hamiltonian can be interpreted as a log-density and that the desired marginal density for $\boldsymbol{\theta}$ is obtained, the addition of the normalising constant for the Gaussian is included in the potential energy term. Therefore, the Hamiltonian defined on the Riemann manifold follows as

$$H(\boldsymbol{\theta}, \mathbf{p}) = -\mathcal{L}(\boldsymbol{\theta}) + \frac{1}{2}\log\left((2\pi)^D |\mathbf{G}(\boldsymbol{\theta})|\right) + \frac{1}{2}\mathbf{p}^{\mathsf{T}}\mathbf{G}(\boldsymbol{\theta})^{-1}\mathbf{p} \tag{19}$$

so that $\exp(-H(\boldsymbol{\theta}, \mathbf{p})) = p(\boldsymbol{\theta}, \mathbf{p}) = p(\boldsymbol{\theta})p(\mathbf{p}|\boldsymbol{\theta})$ and the marginal density

$$p(\boldsymbol{\theta}) \propto \int \exp(-H(\boldsymbol{\theta}, \mathbf{p}))d\mathbf{p} = \frac{\exp\{\mathcal{L}(\boldsymbol{\theta})\}}{\sqrt{2\pi^D|\mathbf{G}(\boldsymbol{\theta})|}} \int \exp\left\{-\frac{1}{2}\mathbf{p}^{\mathsf{T}}\mathbf{G}(\boldsymbol{\theta})^{-1}\mathbf{p}\right\} d\mathbf{p} = \exp\{\mathcal{L}(\boldsymbol{\theta})\}$$

is the desired target density.

Unlike the previous case for HMC this joint density is no longer factorisable and therefore the log-likelihood does not correspond to a separable Hamiltonian. The conditional distribution for momentum values given parameter values is a zero-mean Gaussian with the point specific metric tensor acting as the covariance matrix $p(\mathbf{p}|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{p}|\mathbf{0}, \mathbf{G}(\boldsymbol{\theta}))$, which will resolve the scaling issues associated with HMC, as will be demonstrated in the following sections. The dynamics are defined by Hamiltons equations as

$$\begin{aligned} \frac{d\theta_i}{d\tau} &= \frac{\partial H}{\partial p_i} = \left(\mathbf{G}(\boldsymbol{\theta})^{-1}\mathbf{p}\right)_i && (20) \\ \frac{dp_i}{d\tau} &= -\frac{\partial H}{\partial \theta_i} = \frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \theta_i} - \frac{1}{2}\mathrm{Tr}\left[\mathbf{G}(\boldsymbol{\theta})^{-1}\frac{\partial \mathbf{G}(\boldsymbol{\theta})}{\partial \theta_i}\right] + \frac{1}{2}\mathbf{p}^{\mathsf{T}}\mathbf{G}(\boldsymbol{\theta})^{-1}\frac{\partial \mathbf{G}(\boldsymbol{\theta})}{\partial \theta_i}\mathbf{G}(\boldsymbol{\theta})^{-1}\mathbf{p} && (21) \end{aligned}$$

The Hamiltonian dynamics on the manifold are simulated by solving the continuous time derivatives and it is straightforward to see that they satisfy Liouville's theorem of volume preservation (Leimkuhler and Reich, 2004). However, for the discrete integrator it is not so straightforward.

Naively employing the discrete Stormer-Verlet Leapfrog integrator (equations (5), (6) and (7)) gives transformations of the form $(\boldsymbol{\theta}, \mathbf{p}) \mapsto (\boldsymbol{\theta}, \mathbf{p} - \epsilon\varphi(\boldsymbol{\theta}, \mathbf{p}))$ and $(\boldsymbol{\theta}, \mathbf{p}) \mapsto (\boldsymbol{\theta} + \epsilon\phi(\boldsymbol{\theta}, \mathbf{p}), \mathbf{p})$, neither of which admits a Jacobian with unit determinant. In addition, it is straightforward to see that reversibility for $\boldsymbol{\theta}$ and $\mathbf{p}$ is not satisfied for finite step-size ϵ, as $\mathbf{G}(\boldsymbol{\theta}(\tau)) \neq \mathbf{G}(\boldsymbol{\theta}(\tau + \epsilon))$ and $\mathbf{p}(\tau)^{\mathsf{T}}\mathbf{F}(\boldsymbol{\theta})\mathbf{p}(\tau) \neq \mathbf{p}(\tau+\epsilon)^{\mathsf{T}}\mathbf{F}(\boldsymbol{\theta})\mathbf{p}(\tau+\epsilon)$. Therefore proposals generated from this integrator will not satisfy detailed balance in a Hybrid Monte Carlo scheme. What is required is a time reversible volume preserving numerical integrator for solving this non-separable Hamiltonian to ensure a correct MCMC algorithm. Such a second-order semi-explicit integrator can be formed by the use of first-order implicit Euler integrators. This is referred to as the Generalised Leapfrog algorithm, see Leimkuhler and Reich (2004) for details, and follows below.

$$\mathbf{p}^{n+\frac{1}{2}} = \mathbf{p}^{n} - \frac{\epsilon}{2}\nabla_{\boldsymbol{\theta}} H(\boldsymbol{\theta}^{n}, \mathbf{p}^{n+\frac{1}{2}}) \tag{22}$$

$$\boldsymbol{\theta}^{n+1} = \boldsymbol{\theta}^{n} + \frac{\epsilon}{2}\left[\nabla_{\mathbf{p}} H(\boldsymbol{\theta}^{n}, \mathbf{p}^{n+\frac{1}{2}}) + \nabla_{\mathbf{p}} H(\boldsymbol{\theta}^{n+1}, \mathbf{p}^{n+\frac{1}{2}})\right] \tag{23}$$

$$\mathbf{p}^{n+1} = \mathbf{p}^{n+\frac{1}{2}} - \frac{\epsilon}{2}\nabla_{\boldsymbol{\theta}} H(\boldsymbol{\theta}^{n+1}, \mathbf{p}^{n+\frac{1}{2}}) \tag{24}$$

If the Hamiltonian is separable then the Generalised Leapfrog reduces to the standard Stormer-Verlet Leapfrog integrator. For the case of interest where the Hamiltonian is non-seperable then (22) and (23) are defined implicitly. These require to be solved and we employ simple fixed point iterations run to convergence for this purpose, typically between 5 to 6 iterations were required in the experiments conducted. The repeated application of the above steps provides the means to obtain a deterministic proposal that is guided not only by the derivative information of the target density, as in HMC or MALA, but also exploits the local geometric structure of the manifold as determined by the metric tensor. Intuitively, comparing the two Hamiltonians (3) and (19) shows that the constant mass matrix $\mathbf{M}$, defining a globally constant metric, is now replaced with the position specific metric thus removing the requirement to tune the values of the elements of $\mathbf{M}$, which so dramatically affects the performance of HMC. Since the integration scheme detailed above is both time reversible and volume preserving employing it as a proposal process provides a correct MCMC scheme satisfying detailed balance and convergence to the desired target density. The overall Riemannian Manifold HMC (RM-HMC) scheme can once again be written as a Gibbs sampler

$$\mathbf{p}|\boldsymbol{\theta} \sim p(\mathbf{p}|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{p}|\mathbf{0}, \mathbf{G}(\boldsymbol{\theta})) \tag{25}$$

$$\boldsymbol{\theta}^{*}|\mathbf{p} \sim p(\boldsymbol{\theta}^{*}|\mathbf{p}) \propto \exp\left\{-H(\boldsymbol{\theta}^{*}, \mathbf{p} + \delta\mathbf{p})\right\} \tag{26}$$

where samples from $p(\boldsymbol{\theta}^{*}|\mathbf{p})$ are obtained by running the Generalised Leapfrog integrator for a certain number of steps to give proposed moves $\boldsymbol{\theta}^{*}$ and $\mathbf{p}^{*}$ and accepting or rejecting with probability $\mathsf{min}[1, \exp\{-H(\boldsymbol{\theta}^{*}, \mathbf{p}^{*}) + H(\boldsymbol{\theta}, \mathbf{p})\}]$. As for standard HMC this Gibbs sampling scheme produces an ergodic, time reversible Markov chain satisfying detailed balance and whose stationary marginal density is $p(\boldsymbol{\theta})$ (Duane *et al.*, 1987; Liu, 2001; Neal, 1996, 2010). However in this case there is no need to manually select and tune the mass matrix as it is defined at each step by the underlying geometry. Pseudo-code is provided in Appendix (D)

An interesting point to note is that the Hamiltonian flow (solutions of the differential equations) for a purely kinetic Hamiltonian i.e. in the absence of a potential energy term is a geodesic flow (Calin and Chang, 2004). In other words paths produced by the solution of Hamiltons equations follow the geodesics (paths of least distance between points) on the manifold. For the case that we consider where there also is a potential term then the flows are locally geodesic (McCord *et al*, 2002). This observation presents an interesting area for further theoretical analysis and characterisation of the properties of the RM-HMC method.

Table 1. Summary of datasets for logistic regression

Name	Covariates (D)	Data Points (N)	Dimension of $\boldsymbol{\beta}$ (b)
Pima Indian	7	532	8
Australian Credit	14	690	15
German Credit	24	1000	25
Heart	13	270	14
Ripley	2	250	7
Caravan	86	5822	87

Figures 3 and 4 provide an intuitive visual demonstration of the differences in HMC and RM-HMC when converging to and sampling from a target density. To illustrate the RM-HMC sampling scheme and evaluate performance against alternative MCMC methods, a number of example applications are now presented. We begin with posterior sampling for Logistic Regression models.

7. RM-HMC and mMALA for Bayesian Logistic Regression

Consider an $N \times D$ design matrix $\mathbf{X}$ comprising N samples each with D covariates and a binary response variable $\mathbf{t} \in \{0,1\}^N$. Denoting the logistic link function as $\sigma(\cdot)$, a Bayesian logistic regression model of the binary response (Gelman *et al.*, 2004; Liu, 2001) is obtained by the introduction of regression coefficients $\boldsymbol{\beta} \in \mathbb{R}^D$ with an appropriate prior, which for illustrative purposes is given as $\boldsymbol{\beta} \sim \mathcal{N}(\mathbf{0}, \alpha\mathbf{I})$ where α is given. Neglecting constants, the log joint-likelihood follows in standard form as

$$\log p(\mathbf{t}, \boldsymbol{\beta}|\mathbf{X}, \alpha) = \mathcal{L}(\boldsymbol{\beta}) - \frac{1}{2\alpha}\boldsymbol{\beta}^\mathsf{T}\boldsymbol{\beta} = \boldsymbol{\beta}^\mathsf{T}\mathbf{X}^\mathsf{T}\mathbf{t} - \sum_{n=1}^{N} \log(1 + \exp(\boldsymbol{\beta}^\mathsf{T}\mathbf{X}_{n,\cdot}^\mathsf{T})) - \frac{1}{2\alpha}\boldsymbol{\beta}^\mathsf{T}\boldsymbol{\beta} \quad (27)$$

where $\mathbf{X}_{n,\cdot}$ denotes the vector that is the n^{th} row of the $N \times D$ matrix $\mathbf{X}$. The derivative of the log joint-likelihood is $\nabla\mathcal{L}(\boldsymbol{\beta}) - \alpha^{-1}\boldsymbol{\beta}$ and its second derivative follows as $\nabla\nabla\mathcal{L}(\boldsymbol{\beta}) - \alpha^{-1}\mathbf{I}$ which is comprised of the matrix of second derivatives of the likelihood and the log-prior. As already mentioned throughout the practical examples to include the effect of the prior on the geometry we form the metric tensor based on the negative of the expectation of this second derivative, which is the Fisher Information plus the negative Hessian of the log-prior. The metric tensor therefore follows as

$$\mathbf{G}(\boldsymbol{\beta}) = E_{\mathbf{t}|\mathbf{X},\boldsymbol{\beta},\alpha}\left\{-\nabla\nabla\mathcal{L}(\boldsymbol{\beta}) + \alpha^{-1}\mathbf{I}\right\} = \mathbf{X}^\mathsf{T}\boldsymbol{\Lambda}\mathbf{X} + \alpha^{-1}\mathbf{I} \quad (28)$$

where the diagonal $N \times N$ matrix $\boldsymbol{\Lambda}$ has elements $\boldsymbol{\Lambda}_{n,n} = \sigma(\boldsymbol{\beta}^\mathsf{T}\mathbf{X}_{n,\cdot}^\mathsf{T})(1 - \sigma(\boldsymbol{\beta}^\mathsf{T}\mathbf{X}_{n,\cdot}^\mathsf{T}))$. Finally the derivative matrices of the metric tensor take the form $\partial\mathbf{G}(\boldsymbol{\beta})/\partial\beta_i = \mathbf{X}^\mathsf{T}\boldsymbol{\Lambda}\mathbf{V}^i\mathbf{X}$ where the $N \times N$ diagonal matrix $\mathbf{V}^i$ has elements $(1 - 2\sigma(\boldsymbol{\beta}^\mathsf{T}\mathbf{X}_{n,\cdot}^\mathsf{T}))X_{ni}$. The above identities are all that are required to define the RM-HMC and mMALA sampling methods, which will be illustrated in the following experimental section.

7.1. Experimental Results for Bayesian Logistic Regression

We present results from the analysis of 6 datasets (Michie *et al.*, 1994; Ripley, 1996), summarised in Table 1. These datasets exhibit a wide range of characteristics which provides a challenging test

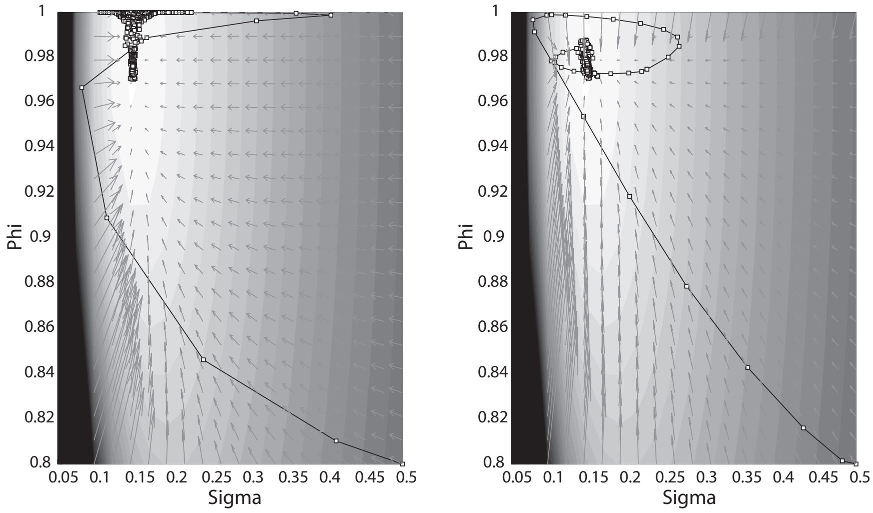

Fig. 3. The above contours were plotted from the stochastic volatility model investigated later in the paper. The latent volatilities and the parameter β are set to their true values, while the log-joint likelihood given different values of the parameters σ and ϕ is shown by the contour plot. The left hand plot shows the evolution of a Markov chain using HMC with a unit mass matrix, while the right hand plot shows the evolution of a chain from the same starting point using RM-HMC. Note how the use of the metric allows RM-HMC to converge much quicker to the target density.

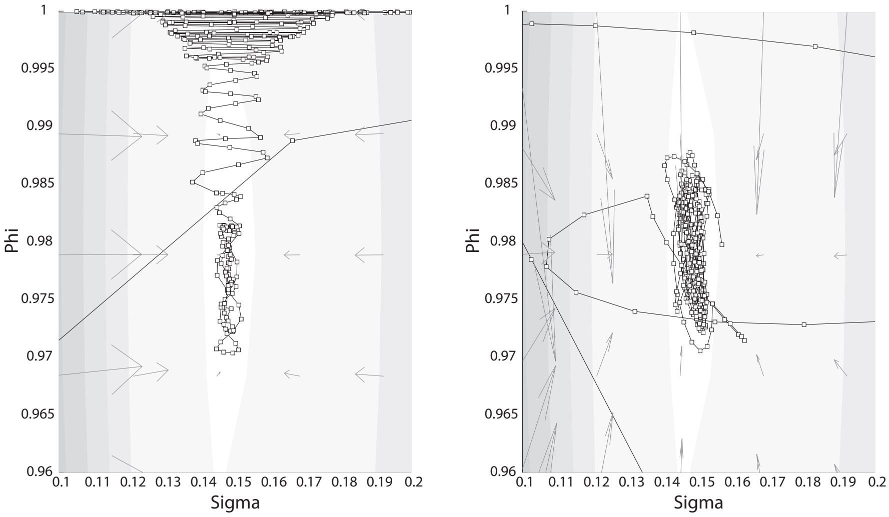

Fig. 4. Here we see a close-up of the Markov chain paths shown in Figure 3. It is clear that RM-HMC effectively normalises the gradients in each direction, whereas HMC, with a unit mass matrix, exhibits stronger gradients along the horizontal direction compared to the vertical direction, and therefore takes longer to explore the space fully. A carefully tuned mass matrix may improve HMC sampling, while RM-HMC deals with this automatically.

for any applied sampling method; the number of covariates ranges from 2 to 87, the number of data points ranges from 250 to 5877, and the standard deviations of the induced marginal posterior distributions range from 0.0004 to 9.9. We investigate the use of RM-HMC and mMALA applied to this problem and also implement the following sampling methods for comparison:-

(a) Component-Wise Adaptive Metropolis-Hastings (Robert, 2004) (Chapter. 7)
(b) Joint Updating Gibbs Sampler (Holmes and Held, 2005)
(c) Metropolis Adjusted Langevin Algorithm (Roberts and Stramer, 2003)
(d) Hybrid Monte Carlo (Duane *et al.*, 1987; Neal, 1993a; Liu, 2001)
(e) Iterated Weighted Least Squares (Gamerman , 1997)

Given each dataset we wish to sample from the posterior distribution over the regression coefficients $\boldsymbol{\beta}$, and in each experiment wide Gaussian prior distributions were employed such that $\pi(\beta_i) \sim \mathcal{N}(0, 100)$. A linear logistic regression model with intercept was used for each of the datasets with the exception of the Ripley dataset, for which a cubic polynomial regression model was employed.

Each method was run 10 times with every dataset and the average results were recorded. We reproduce the results of Holmes and Held (2005) by allowing 5000 burn in iterations so that each sampler reaches the stationary distribution and has time to adapt as necessary. The next 5000 iterations were used to collect posterior samples for each of the methods and the CPU time required to collect these samples was recorded. Each method was implemented in the interpreted language Matlab to ensure fair comparison. We compared the relative efficiency of these methods by calculating the effective sample size (ESS) using the posterior samples for each covariate, $ESS = N(1 + 2\sum_k \gamma(k))^{-1}$ where N is the number of posterior samples and $\sum_k \gamma(k)$ is the sum of the K monotone sample autocorrelations as estimated by the initial monotone sequence estimator (see Geyer (1992)). The standard error around the mean ESS was less than 2×10^{-2} for all results. Such an approach was also taken by Holmes and Held (2005), in which they report the *mean* ESS, averaged over each of the covariates. However, we feel this could give a rather inflated measure of the true ESS, since ideally we want a measure of the number of samples which are uncorrelated over *all* covariates. In this paper we therefore report the *minimum* ESS of the sampled covariates. This minimum ESS is then normalised relative to the CPU time by calculating the time taken to obtain 1 sample which is effectively uncorrelated across all covariates.

7.1.1. Metropolis-Hastings

We employed an adaptive Metropolis-Hastings (M-H) scheme, such that each covariate was updated individually with its stepsize being adapted in every 100 iterations during burn-in to achieve an acceptance rate of between 20% and 40%. The stepsize was then fixed at the end of the burn-in period. With Metropolis-Hastings it is sometimes useful to employ sub-sampling, in order to remove the autocorrelations in the posterior samples. Table 2 demonstrates that since our current measure of efficiency is time normalised, it automatically takes into account the trade-off between the additional computational cost of drawing more samples, and the improved ESS that results. We see that the computational effort required to take additional steps through parameter space is generally greater than the benefit of increased ESS that results, such that the time taken to produce one effectively independent sample increases as the number of discarded samples increases using subsampling. In the main experiments we therefore compare the best case scenario which results from not employing subsampling.

Table 2. Bayesian Logistic Regression with Metropolis Sampling - investigating the effect of subsampling on our time normalised efficiency measure

Subsample every:	1	2	5	10	20	50
Dataset						
Australian	0.59	0.91	0.99	1.03	0.97	0.97
German	2.02	2.87	3.96	4.70	4.97	4.84
Pima	0.29	0.35	0.34	0.36	0.39	0.38
Heart	0.65	0.86	1.20	1.53	1.33	1.44
Ripley	0.22	0.40	0.51	0.56	0.56	0.59

7.1.2. Auxiliary Variable Gibbs Sampler

The auxiliary variable Gibbs sampler of Holmes and Held (2005) was implemented with a joint update of $\{\mathbf{z}, \boldsymbol{\beta}\}$, where $\mathbf{z} \in \mathbb{R}^N$ is the auxiliary variable designed to improve mixing of the covariate samples. We implemented the algorithm based on the very detailed pseudo-code given in the appendix of their paper, and in contrast to the M-H algorithm this method has the advantage of requiring no tuning of parameters. The main computational expense however is in the repeated sampling from truncated normal distributions, for which we implemented code based on the efficient method defined in Johnson *et al.* (1999).

7.1.3. Metropolis Adjusted Langevin Algorithm

We implemented a MALA sampler with proposed covariates being drawn from the multivariate normal distribution $\mathcal{N}\left(\boldsymbol{\beta} + \nabla \log(\pi\{\boldsymbol{\beta}\})h/2, h\mathbf{I}_D\right)$, where $\mathbf{I}_D$ is the D-dimensional identity matrix and h controls the scaling of the proposal variance. We follow the advice of Roberts and Rosenthal (1998) by scaling h like $O(D^{-\frac{1}{3}})$, where D is the number of covariates, such that we achieve an acceptance rate of between 40% and 70%.

7.1.4. Hybrid Monte Carlo

Hybrid Monte Carlo has promised to offer more efficient sampling from high dimensional probability distributions by effectively reducing the amount of random walk present in the parameter values being proposed. This has indeed been shown to be the case for relatively simple, although high-dimensional, multivariate normal distributions, however there has been little application to more complex data models. We believe the reason for this lies in the amount of tuning required to obtain reasonable mixing and rates of acceptance, as will be highlighted in the following section. The two main parameters which require tuning are the number of leapfrog steps, N, and the size of each leapfrog step, ϵ. It has been suggested that choosing the leapfrog stepsize to be proportional to the marginal standard deviation of the target distribution along each dimension drastically improves mixing, particularly when such marginals are of greatly varying orders of magnitude. Setting different leapfrog stepsizes along different directions can be equivalently encoded in the so-called mass matrix (Neal, 1993a, 1996). However, this approach clearly requires advance knowledge of the distribution being sampled from, and in a practical setting this information is very rarely available. The use of exploratory runs of a Metropolis sampler to obtain initial estimates of the target distribution has been suggested (Hajian, 2007), however there is the obvious associated computational cost and the fact that this may not be feasible for very complex distributions.

Table 3. RM-HMC with generalised leapfrog integration scheme - investigating the effect of parameter settings on sampling efficiency with German Credit dataset

ϵL	Max ϵ	Mean Time (s)	Min ESS	s/Min ESS
1	1/2	149.5	637	0.234
2	1/2	224.7	2085	0.108
3	1/2	287.9	4791	0.060

Following the advice of Neal (1993a, 1996), we fix the size of each leapfrog step ϵ to a value slightly smaller than the smallest marginal standard deviation of the model parameter posteriors, and set the number of leapfrog steps L such that the maximum distance that can be travelled in a single move, ϵL, is larger than the largest standard deviation of the marginal parameter distributions. A larger step size would result in large rejection rates, while a smaller number of steps would result in very slow exploration of the target distribution.

In our experiments we make the, rather optimistic assumption, that this information is known when implementing HMC, presumably after a number of exploratory runs of the algorithm, and set ϵ small enough to obtain a high acceptance rate ($> 70\%$) and $\epsilon L \approx 3$ allowing the chain to traverse a distance larger than the standard deviation of the largest marginal posterior for all datasets, see Table 10. This approach works well for distributions in which the marginal standard deviations are of a similar magnitude, however the algorithm soon becomes computationally very expensive to run in situations where they greatly differ and the number of leapfrog steps required for adequate mixing consequently becomes very large.

7.1.5. Iterated Weighted Least Squares

We consider in addition the second order method Iterated Weighted Least Squares (IWLS) (Gamerman , 1997), which makes use of second derivatives in its Metropolis proposal steps. It should be noted that the term involving the second derivatives for IWLS is indeed different from the metric tensor expression employed in RM-HMC and mMALA, and we shall see how this impacts on the results shortly. This method is relatively straightforward to implement and has the advantage that it requires no tuning, similar to the auxiliary variable Gibbs sampler of Holmes and Held (2005).

7.2. Comparison of MCMC Methods

We begin by investigating the RM-HMC method in detail for one of the more challenging of our six datasets, German Credit, which consists of 24 covariates and 1000 datapoints. We then compare the results for all six datasets employing the alternative sampling methods described previously.

The maximum total distance which a chain may travel in a single proposed move is given by ϵL, and for any given value of ϵL we chose ϵ small enough such that the acceptance ratio was above 70% and then adjusted L appropriately. Table 3 shows the results of the generalised leapfrog integration scheme using a variety of choices for these parameters. We found that sampling generally became more efficient as the maximum total distance travelled by a chain, ϵL, was increased, i.e. when the chain was able to traverse a distance greater than the width of each marginal distribution.

Following these guidelines, we find that the RM-HMC and mMALA sampling methods work very well for a variety of datasets and RM-HMC is fairly robust to the choice of algorithm parameters. For comparison with the alternative sampling methods, we chose the settings for RM-HMC based on the above analysis. We employed the generalised leapfrog scheme, setting ϵ for each

Table 4. Australian Credit Dataset, $D = 14$, $N = 690$, 15 regression coefficients - Comparison of sampling methods

Method	Time	ESS (Min, Med, Max)	s/Min ESS	Rel. Speed
Metropolis	9.1	(15, 208, 691)	0.61	×27
Aux. Var.	757.2	(46, 1074, 1454)	16.5	×1
MALA	No Convergence	(-, -, -)	-	-
HMC	No Convergence	(-, -, -)	-	-
IWLS	4.9	(3.7, 8.7, 52.5)	1.32	×12.5
mMALA	11.8	(730, 872, 1033)	0.0162	×1019
mMALA Simp.	2.6	(459, 598, 726)	0.0057	×2895
RM-HMC	115.3	(4940, 5000, 5000)	0.023	×717
RM-HMC (Stud. t)	145.8	(1745, 1916, 2282)	0.084	×196

dataset equal to the smallest stepsize for which the acceptance rate was reasonably high ($> 70\%$), and the number of integration steps such that $\epsilon L \approx 3$. The scaling for mMALA was chosen to obtain an acceptance rate of around 70%. We repeated the sampling experiments 10 times and averaged the results, which are shown for each of the datasets in Tables 4 to 8. It is interesting to see that MALA generally performs poorly. Whereas all other methods converge within 5000 burn-in iterations for all datasets, MALA needs as many as 2 million iterations to converge due to the very small stepsize required to achieve an acceptance ratio above 40%. This is particularly the case for the Australian Credit and Heart datasets, which exhibit very large differences in scale between the largest and smallest marginal standard deviations (see Table 10), resulting in extremely slow exploration of the target distribution, indeed even after 2 million iterations the Langevin guided chains had still not reached their stationary distributions. Clearly some method of scaling the regression coefficients would improve the mixing, however this is again unfeasible unless information regarding the marginal posterior distributions is known in advance. Similarly the standard HMC method fails to converge for the Australian Credit dataset, since the stepsize is so small that the number of integration steps required becomes computationally impractical to implement. Figure 5 shows the trace and autocorrelation plots for 1000 posterior samples using the Heart dataset. The difference in autocorrelation is quite striking, both from inspection of the traces and from examination of the autocorrelation plots themselves. The autocorrelation of the RM-HMC samples drop towards zero far quicker than for any of the other methods.

As the number of covariates in the dataset increases, so the overall performance of RM-HMC and mMALA decreases due to the increased computational burden of calculating partial derivatives with respect to each of the covariates. Indeed we see that RM-HMC is only about twice as efficient as Metropolis with the Caravan dataset, with mMALA performing worse still. The simplified mMALA scheme (where the curvature terms are removed) on the other hand performs far better, employing an approximation of the local geometry with a much reduced computational cost.

We consider also an alternative second order method, IWLS, which makes use of terms involving second derivatives and therefore some measure of the curvature of the parameter space. IWLS performs fairly poorly, indeed in the examples it performs about the same as parameter-wise Metropolis. Although IWLS is a second order method, it makes use of a metric which appears to be significantly less efficient than employing the expected Fisher Information as in mMALA and RM-HMC. In addition, we note that IWLS runs into severe numerical problems with the Caravan dataset, due the fact that the second order derivatives it employs are not guaranteed to be positive semi-definite.

Table 5. German Credit Dataset, $D = 24$, $N = 1000$, 25 regression coefficients - Comparison of sampling methods

Method	Time	ESS (Min, Med, Max)	s/Min ESS	Rel. Speed
Metropolis	20.9	(10, 82, 601)	2.09	×1.1
Aux. Var.	1155.1	(1071, 2200, 2620)	1.08	×2.2
MALA	2.7	(3, 5, 130)	0.9	×2.6
HMC	3161.6	(2707, 4201, 5000)	1.17	×2
IWLS	9.37	(4, 9, 31)	2.34	×1
mMALA	36.2	(616, 769, 911)	0.059	×39.6
mMALA Simp.	4.1	(463, 611, 740)	0.009	×260
RM-HMC	287.9	(4791, 5000, 5000)	0.06	×39
RM-HMC (Stud. t)	360.5	(1665, 2412, 2942)	0.22	×10.6

Table 6. Pima Indian Dataset, $D = 7$, $N = 532$, 8 regression coefficients - Comparison of sampling methods

Method	Time	ESS (Min, Med, Max)	s/Min ESS	Rel. Speed
Metropolis	4.1	(14, 37, 201)	0.29	×1.9
Aux. Var.	565.4	(1176, 1877, 2340)	0.48	×1.1
MALA	1.63	(3, 10, 39)	0.54	×1
HMC	1499.1	(3149, 3657, 3941)	0.48	×1.1
IWLS	3.2	(6, 16, 34)	0.53	×1
mMALA	4.4	(1124, 1266, 1409)	0.0039	×138
mMALA Simp.	1.9	(1022, 1185, 1312)	0.0019	×284
RM-HMC	50.9	(5000, 5000, 5000)	0.01	×54
RM-HMC (Stud. t)	56.0	(2090, 2146, 3105)	0.027	×20

Table 7. Heart Dataset, $D = 13$, $N = 270$, 14 regression coefficients - Comparison of sampling methods

Method	Time	ESS (Min, Med, Max)	s/Min ESS	Rel. Speed
Metropolis	5.2	(8, 65, 530)	0.65	×8.4
Aux. Var.	281.5	(721, 1276, 1761)	0.39	×14.1
MALA	No Convergence	(-, -, -)	-	-
HMC	2018	(368, 2740, 2938)	5.48	×1
IWLS	2.9	(3, 6, 16)	0.97	×5.6
mMALA	6.4	(649, 793, 920)	0.01	×548
mMALA Simp.	1.7	(373, 486, 610)	0.004	×1191
RM-HMC	59.2	(4925, 5000, 5000)	0.012	×457
RM-HMC (Stud. t)	67.0	(936, 1144, 1822)	0.072	×76.1

Table 8. Ripley Dataset, $D = 2$, $N = 250$, 7 regression coefficients - Comparison of sampling methods

Method	Time	ESS (Min, Med, Max)	s/Min ESS	Rel. Speed
Metropolis	2.5	(11, 20, 251)	0.23	×15.6
Aux. Var.	258.6	(72, 374, 1967)	3.59	×1
MALA	1.1	(4, 8, 30)	0.28	×12.8
HMC	52.8	(1365, 1596, 1754)	0.039	×92.1
IWLS	1.7	(8, 26, 252)	0.21	×17.1
mMALA	3.0	(857, 975, 1098)	0.0035	×1026
mMALA Simp.	1.4	(682, 799, 927)	0.0021	×1710
RM-HMC	25.3	(4999, 5000, 5000)	0.0051	×704
RM-HMC (Stud. t)	27.9	(813, 1266, 1463)	0.034	×106

Table 9. Caravan Dataset, $D = 86$, $N = 5822$, 87 regression coefficients - Comparison of sampling methods

Method	Time	ESS (Min, Med, Max)	s/Min ESS	Rel. Speed
Metropolis	388.7	(3.8, 23.9, 804)	101.9	×6.7
Aux. Var.	4628	(6.7, 570, 4788)	687	×1
MALA	17.4	(2.8, 5.3, 17.2)	6.2	×110.8
HMC	12,519	(33.8, 4032, 5000)	369.7	×1.9
IWLS	N/A	N/A	N/A	N/A
mMALA	305.3	(7.5, 21.1, 50.7)	305.3	×2.25
mMALA Simp.	48.9	(7.5, 18.4, 44)	6.5	×105.7
RM-HMC	45,760	(877, 1554, 2053)	52.1	×13.2
RM-HMC (Stud. t)	45,877	(279, 477, 705)	164	×4.2

Table 10. Summary of standard deviations of the marginal posterior distributions for each dataset

Dataset	Smallest Marg. S.D.	Largest Marg. S.D.	Ratio
Pima Indian	0.0043	0.9646	225
Australian Credit	0.00017	1.0667	6404
German Credit	0.0038	1.1492	303
Heart	0.004	2.9221	739
Ripley	1.2575	7.556	6
Caravan	0.042	9.916	236

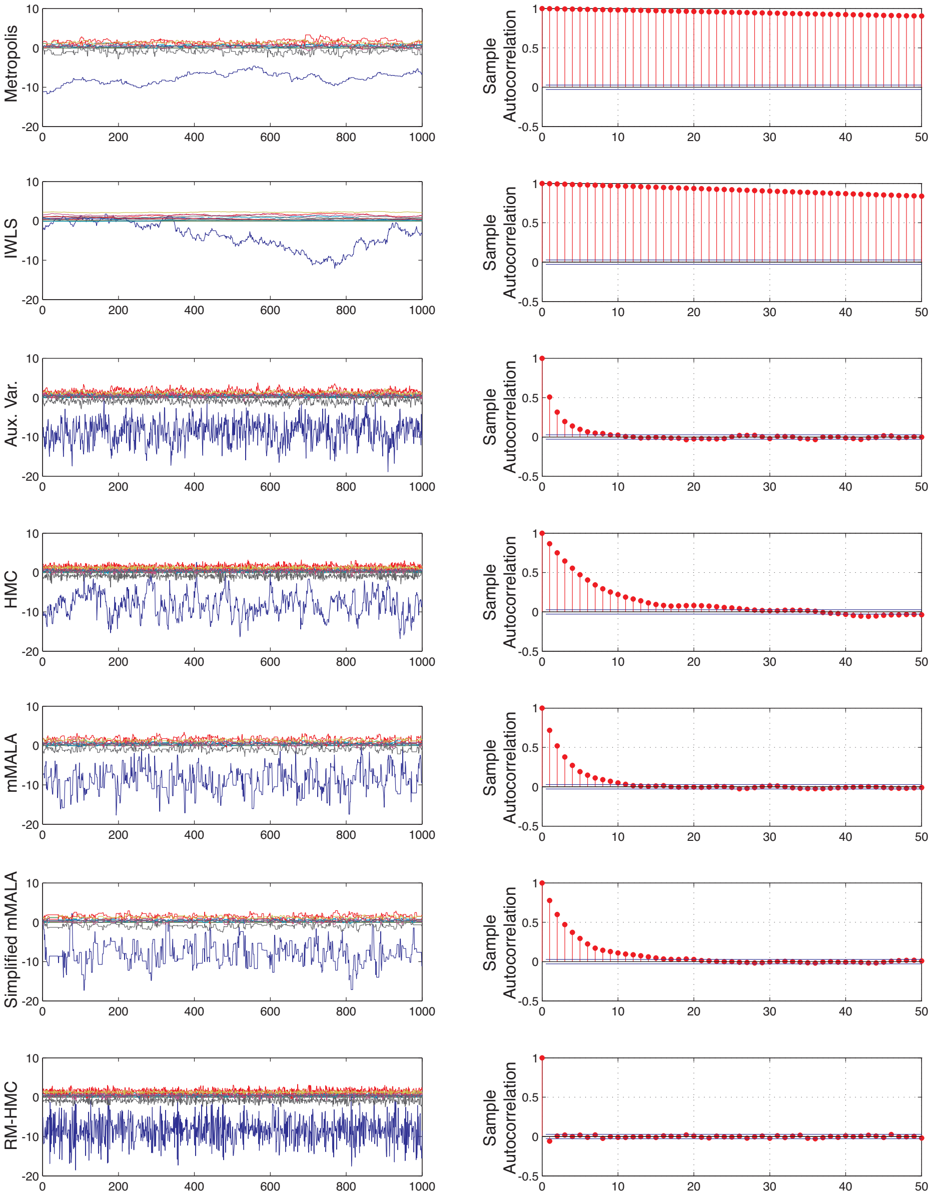

Fig. 5. Trace plots for 1000 posterior samples with the Heart dataset using (from top to bottom) Metropolis, IWLS, auxiliary variable sampler, standard HMC, mMALA, Simplified mMALA and RM-HMC. Autocorrelation plots are also shown for one of its parameters, which may be seen in the trace plots to have a mean of around -7.

7.3. Comparison of RM-HMC and mMALA Variants

We now investigate variants of RM-HMC and mMALA to see whether results may be improved based on slight alterations to the standard forms. We first consider a simplified version of mMALA, which assumes a locally flat metric tensor during each Metropolis step and will still converge to the stationary distribution due to the Metropolis adjustment. It is clear that this is computationally much less expensive than the full mMALA as it avoids the calculation of metric tensor derivatives. It is interesting that simplified mMALA has worse ESS than the complete mMALA, which intuitively makes sense since proposed steps across the manifold will have greater error by not taking into account any changes in curvature. The time normalised ESS however is much better, as the computational complexity is far less.

It is also interesting to investigate the use of an alternative kinetic energy function in RM-HMC†. This idea is also briefly mentioned in Liu (2001) although no example is given. We consider therefore the use of a Student-t kinetic energy term, with the idea that since the heavy tails might occasionally mean a larger momentum is sampled, this could plausibly result in less correlated samples of the target distribution. We note that since the multivariate Student-t distribution is symmetric, then the resulting Hamiltonian is still reversible. The equations describing the dynamics of such a Hamiltonian follow as

$$\begin{aligned}\frac{d\theta_i}{d\tau} &= \frac{\partial H}{\partial p_i} = \left(\frac{(v+d)\mathbf{G}(\boldsymbol{\theta})^{-1}\mathbf{p}}{v+\mathbf{p}^{\mathsf{T}}\mathbf{G}(\boldsymbol{\theta})^{-1}\mathbf{p}}\right)_i \\ \frac{dp_i}{d\tau} &= -\frac{\partial H}{\partial \theta_i} = \frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \theta_i} - \frac{1}{2}\mathsf{Tr}\left[\mathbf{G}(\boldsymbol{\theta})^{-1}\frac{\partial \mathbf{G}(\boldsymbol{\theta})}{\partial \theta_i}\right] + \frac{(v+d)}{2}\frac{\mathbf{p}^{\mathsf{T}}\mathbf{G}(\boldsymbol{\theta})^{-1}\frac{\partial \mathbf{G}(\boldsymbol{\theta})}{\partial \theta_i}\mathbf{G}(\boldsymbol{\theta})^{-1}\mathbf{p}}{v+\mathbf{p}^{\mathsf{T}}\mathbf{G}(\boldsymbol{\theta})^{-1}\mathbf{p}}\end{aligned}$$

The simulations take slightly longer to run than with standard Gaussian distributed momentum using the same integration time steps. This is due to the increased computation required to sample from a Student-t distribution, and also to the more involved computation required to calculate the dynamics of this new Hamiltonian. The results show that the ESS is actually significantly less than that of a Hamiltonian defined with Gaussian momentum. This is possibly a result of a higher concentration of mass producing momenta with values closer to zero, even though there will be occasional samples of momentum with much larger magnitude.

In our simulations, manifold based methods outperform all of the other methods using small to medium sized datasets (with the exception of when a Student-t distribution is employed in the kinetic energy term for RM-HMC). It is interesting to note that due to the dense matrix form of the metric tensor and its inverse, the computational cost of mMALA and RM-HMC on Bayesian logistic regression will not scale favourably and it can be seen that their time-normalised efficiency does indeed decrease as the number of regression coefficients in the dataset increases. This issue of scaling can however be eased somewhat by employing simplified mMALA sampling, which assumes a locally constant metric tensor and thus avoids expensive computation of the derivatives of the metric tensor. A further, more complex, example based on a stochastic volatility model is now considered where the metric tensor and its inverse are sparse, permitting scaling of RM-HMC to very high dimensions.

8. RM-HMC and mMALA for a Stochastic Volatility Model

A stochastic volatility model (SVM) studied in Liu (2001); Kim *et al* (1998) is defined with the latent volatilities taking the form of an AR(1) process such that $y_t = \epsilon_t \beta \exp(x_t/2)$ with $x_{t+1} =$

†as was suggested by one of the reviewers.

$\phi x_t + \eta_{t+1}$ where $\epsilon_t \sim \mathcal{N}(0,1)$, $\eta_t \sim \mathcal{N}(0,\sigma^2)$ and $x_1 \sim \mathcal{N}(0,\sigma^2/(1-\phi^2))$ having joint likelihood

$$p(\mathbf{y},\mathbf{x},\beta,\phi,\sigma) = \prod_{t=1}^{T} p(y_t|x_t,\beta)p(x_1)\prod_{t=2}^{T} p(x_t|x_{t-1},\phi,\sigma)\pi(\beta)\pi(\phi)\pi(\sigma). \tag{29}$$

We may split up the sampling procedure into two steps, which as we shall see allows the implementation of RM-HMC in a computationally efficient manner. Firstly we may simulate ϕ,σ,β from $p(\beta,\phi,\sigma|\mathbf{y},\mathbf{x})$, where the priors are chosen to be $p(\beta) \propto \exp(\beta)$, $\sigma^2 \sim$ Inv-$\chi^2(10, 0.05)$ and $(\phi+1)/2 \sim$ Beta$(20, 1.5)$. Secondly we may sample the latent volatilities by simulating from the conditional $p(\mathbf{x}|\mathbf{y},\beta,\phi,\sigma)$. We shall consider the use of mMALA, RM-HMC, MALA and HMC for the purpose of sampling both the parameters and latent volatilities.

8.1. mMALA and RM-HMC for SVM Parameters

We require the partial derivatives of the joint log likelihood with respect to the parameters to implement MALA and HMC, as well expressions for the metric tensor and its partial derivatives, in order to employ mMALA and RM-HMC. All of these quantities may be obtained straightforwardly (see Appendix A for details). In particular, the Fisher Information is given by

$$\begin{bmatrix} \frac{2T}{\beta^2} & 0 & 0 \\ 0 & \frac{2T}{\sigma^2} & \frac{2\phi}{\sigma^3(1-\phi^2)} \\ 0 & \frac{2\phi}{\sigma^3(1-\phi^2)} & \frac{2\phi^2}{(1-\phi^2)^2} + \frac{T-1}{1-\phi^2} \end{bmatrix}$$

where T is the number of observations. Prior information is incorporated into the metric tensor by adding this Fisher Information to the negative second partial derivatives of the log priors (see Appendix A for details). We may then use any of these methods to draw samples from the conditional posterior $p(\beta,\sigma,\phi|\mathbf{y},\mathbf{x},)$.

8.2. mMALA and RM-HMC for SVM Latent Volatilities

The gradient of the joint-log likelihood with respect to each of the latent volatilities is required. Defining the vectors $\mathbf{u} = (x_3,\cdots,x_T)^\mathsf{T}$, $\mathbf{v} = (x_2,\cdots,x_{T-1})^\mathsf{T}$, $\mathbf{w} = \frac{\phi}{\sigma^2}(\mathbf{u}-\phi\mathbf{v})$, $\mathbf{s} = (s_1,\cdots,s_T)^\mathsf{T}$ such that $s_i = 0.5(1-y_i^2\beta^{-2}\exp(-x_i))$, $\delta_1 = -\sigma^{-2}(x_1-\phi x_2)$, and $\delta_T = -\sigma^{-2}(x_T-\phi x_{T-1})$, we define the vector $\mathbf{r} = (\delta_1, \mathbf{w}^\mathsf{T}, \delta_2)^\mathsf{T}$ and the required gradient is $\nabla_\mathbf{x} \log p(\mathbf{y},\mathbf{x}|\beta,\phi,\sigma) \equiv \nabla_\mathbf{x}\mathcal{L} = \mathbf{s}-\mathbf{r}$.

To devise an mMALA and RM-HMC sampler for the latent volatilities, $\mathbf{x}$, we also require an expression for the metric tensor and its partial derivatives with respect to the latent volatilities. For the data likelihood of the model, $p(\mathbf{y}|\mathbf{x},\beta)$, the Fisher Information is a diagonal matrix with 0.5 for each element denoted as $\mathbf{I}_{0.5}$. The latent volatility is an AR(1) process having covariance matrix $\mathbf{C}$ with elements $E\{x_{t+n}x_t\} = \phi^{|n|}\sigma^2/(1-\phi^2)$ and as in the previous examples the metric tensor is defined as the sum of the Fisher Information and the negative Hessian of the log-prior, $\mathbf{G} = \mathbf{I}_{0.5} + \mathbf{C}^{-1}$, conditional on current values of σ,ϕ,β. Now the expression for the covariance matrix is completely dense and is therefore computationally expensive to manipulate. Fortunately, this AR(1) process admits a simple analytic expression for the precision matrix in the form of a sparse tridiagonal matrix, such that the diagonal elements are equal to $(1+\phi^2)/\sigma^2$, with the exception of the first and last diagonal elements which are equal to $1/\sigma^2$, and the super and sub diagonal elements are equal to $-\phi/\sigma^2$. Thus the metric tensor also has a tridiagonal form. For large numbers of observations this sparse structure allows great gains in computational efficiency, since the inverse of this tridiagonal metric tensor may be computed in $\mathcal{O}(n)$ as opposed to the usual

Table 11. 2000 simulated observations with $\beta = 0.65$, $\sigma = 0.15$ and $\phi = 0.98$ - Comparison of sampling the parameters β, σ and ϕ after 20,000 posterior samples averaged over 10 runs

Method	Mean Time	ESS (β,σ,ϕ)	S.E. (β,σ,ϕ)	s/(Min ESS)	Rel. Speed
MALA	41.7	(25.3, 12.5, 45.9)	(2.7,0.6,3.1)	3.34	×45.7
HMC	946.3	(177, 108, 270)	(4.5, 2.6, 7.9)	8.76	×17.4
mMALA	2547	(18.8, 16.7, 40.2)	(0.9, 0.7, 2.4)	152.5	×1
RM-HMC	381.9	(324, 113, 283)	(5.6, 3.1, 6.8)	3.37	×45.3

Table 12. 2000 simulated observations with $\beta = 0.65$, $\sigma = 0.15$ and $\phi = 0.98$ - Comparison of sampling the latent volatilities after 20,000 posterior samples averaged over 10 runs

Method	Mean Time	ESS (min, median, max)	s/(Min ESS)	Rel. Speed
MALA	41.7	(7.9, 15.1, 32.1)	5.28	×6.1
HMC	946.3	(566,903,1856)	1.67	×19.1
mMALA	2547	(79.7, 155.2, 344.5)	31.96	×1
RM-HMC	381.9	(963, 1723, 3412)	0.41	×77.9

$\mathcal{O}(n^3)$. We note that computationally efficient methods for manipulating tridiagonal matrices are automatically implemented by the standard routines in Matlab.

We notice that the metric tensor in this case is not a function of $\mathbf{x}$ and so the associated partial derivatives with respect to the latent volatilities are zero. In this case a one step RM-HMC integration scheme collapses to

$$\mathbf{x} = \mathbf{x}_0 + \frac{\epsilon^2}{2}\mathbf{G}^{-1}\nabla_{\mathbf{x}}\mathcal{L} + \epsilon\sqrt{\mathbf{G}^{-1}}\mathbf{p} \tag{30}$$

where $\mathbf{p} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ which is a discrete Langevin iteration that is preconditioned by the constant matrix $\mathbf{G}^{-1}$. It is clear that this preconditioning will improve both the mixing and overall ESS, see (Lambert and Eilers, 2009) for a recent application of this type of preconditioning in MALA. We point out that in the case of RM-HMC the preconditioning matrix emerges naturally from the underlying geometric principles of RM-HMC.

8.3. Experimental Results for Stochastic Volatility Model

We now compare the computational efficiency of RM-HMC, mMALA, HMC and MALA for sampling both the parameters and the latent variables of the stochastic volatility model as previously defined, Tables (11) and (12). 2000 observations were simulated from the model with the parameter values $\beta = 0.65, \sigma = 0.15$ and $\phi = 0.98$ as given in Liu (2001). Using this data, 20000 posterior samples were collected after a burn-in period of 10000 samples. This sampling procedure was repeated 10 times. The efficiency was compared in terms of time normalised ESS, as in the previous section, for the parameters and the latent volatilities. MALA was tuned such that the acceptance ratio was between 40% and 70%, and it was necessary to use a different tuning for the transient phase than for the stationary phase. HMC was implemented using a step size of 0.015 and 200 integration steps per parameter proposal, and a stepsize of 0.003 and 300 integration steps per volatility proposal. RM-HMC was implemented using a stepsize of 0.3 and 10 integration steps per parameter proposal, and a stepsize of 0.1 and 50 integration steps per volatility proposal.

In terms of sampling the hyperparameters, manifold methods offer little advantage over standard sampling approaches due to the small dimensionality of the problem. RM-HMC and MALA give

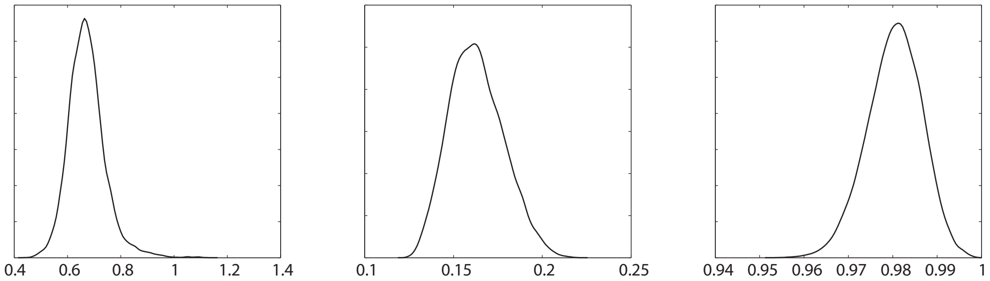

Fig. 6. Posterior marginal densities for β, σ and ϕ respectively, employing RM-HMC to draw 20,000 samples of the parameters and latent volatilities using a simulated dataset consisting of 2000 observations. The true values are $\beta = 0.65$, $\sigma = 0.15$ and $\phi = 0.98$.

the best performance in terms of time normalised ESS. MALA exhibits a very poor ESS, however the computation time is also extremely small compared to the other two methods. RM-HMC has the highest raw ESS, but has much more computational overhead compared to MALA. When we consider sampling the latent variable, RM-HMC offers greater advantages. In particular, it runs faster than HMC, partly because of the computationally efficient tridiagonal structure of the metric tensor and partly because RM-HMC follows the natural tensor gradient through the parameter space and requires significantly fewer leapfrog iterations to explore the target density. See Figure 3 and 4 for an illustration of the contrast between HMC and RM-HMC sampling of the parameters of this model. In this example, mMALA performs very badly due to the need to take a Cholesky decomposition of the inverse metric tensor of the latent variables, which is a dense matrix, compared to RM-HMC which only requires use of the tridiagonal metric tensor. It should be noted that RM-HMC again requires very little tuning compared to the other methods; unlike MALA it does not require different tuning in different parts of the parameter space, and unlike HMC it requires no manual setting of a mass matrix. It would be interesting to compare performance of mMALA and RM-HMC to the Particle MCMC methodology (Andrieu *et al*, 2010) for this particular model.

We now consider an example where the target density is extremely high dimensional, which is encountered when performing inference using spatial data modeled by a log-Gaussian Cox process.

9. RM-HMC and mMALA for Log-Gaussian Cox Point Processes

RM-HMC and mMALA are further studied using the example of inference in a log-Gaussian Cox point process as detailed in (Christensen *et al.*, 2005). This is a particularly useful example in that the target density is of high dimension with strong correlations and provides a severe test of MCMC capability. The data, model and experimental protocol as described in (Christensen *et al.*, 2005) is adopted here. A 64 × 64 grid is overlayed on the area $[0,1]^2$ with the number of points in each grid cell denoted by the random variables $\mathbf{Y} = \{Y_{i,j}\}$ which are assumed conditionally independent, given a latent intensity process $\Lambda(\cdot) = \{\Lambda(i,j)\}$, and are Poisson distributed with means $m\Lambda(i,j) = m\exp(X_{i,j})$, where m = 1/4096. The random variable $\mathbf{X} = \{X_{i,j}\}$ is a Gaussian process with mean $E\{\mathbf{x}\} = \mu\mathbf{1}$, where $\mathbf{x} = \mathsf{Vec}(\mathbf{X})$, $\mathbf{y} = \mathsf{Vec}(\mathbf{Y})$, and covariance function $\boldsymbol{\Sigma}_{(i,j),(i',j')} = \sigma^2\exp(-\delta(i,i',j,j')/64\beta)$, where $\delta(i,i',j,j') = \sqrt{(i-i')^2+(j-j')^2}$. The complete joint density is

$$p(\mathbf{y},\mathbf{x},\mu,\sigma,\beta) \propto \prod_{i,j}^{64} \exp\{y_{i,j}x_{i,j} - m\exp(x_{i,j})\}\exp(-(\mathbf{x}-\mu\mathbf{1})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\mu\mathbf{1})/2) \qquad (31)$$

We consider first the case where the hyperparameters of the covariance function are fixed and we are interested in inferring the latent field only. Denoting $\mathcal{L} \equiv \log p(\mathbf{y}, \mathbf{x}|\mu, \sigma, \beta)$ and $\mathbf{e} = \{m \exp(x_{i,j})\}$, then the derivative with respect to the latent variables follows straightforwardly as $\nabla_{\mathbf{x}}\mathcal{L} = \mathbf{y} - \mathbf{e} - \mathbf{\Sigma}^{-1}(\mathbf{x} - \mu\mathbf{1})$, and the Fisher Information (where the expectation is taken with respect to the complete likelihood) follows as $\mathbf{G}(\mathbf{x}) = -E_{\mathbf{y},\mathbf{x}|\boldsymbol{\theta}}\{\nabla_{\mathbf{x}}\nabla_{\mathbf{x}}\mathcal{L}\} = \mathbf{\Lambda} + \mathbf{\Sigma}^{-1}$, where the diagonal matrix $\mathbf{\Lambda}$, whose ith diagonal element is defined as $m \exp(\mu + (\mathbf{\Sigma})_{ii})$, follows from the expectation of the exponential of normal random variables.

We note that for fixed hyperparameters, the metric tensor describing the manifold for the random field $\mathbf{x}$ is constant. The generalised leapfrog method therefore collapses into a standard leapfrog algorithm, with no need to employ any fixed point iterations. The computational cost of calculating the required inverse of the metric tensor scales as $\mathcal{O}(N^3)$, however once this quantity has been calculated, a large number of leapfrog steps may be made with little additional overhead, which as we shall see results in very efficient sampling of the latent variables.

The second case we consider is where the hyperparameters are also inferred along with the latent variables. Considering them jointly, now with $\mathcal{L} \equiv \log p(\mathbf{y}, \mathbf{x}, \sigma, \beta|\mu)$, we see that the Fisher Information matrix is block diagonal with blocks $\mathbf{\Lambda} + \mathbf{\Sigma}^{-1}$ and $\mathbf{D}_{\boldsymbol{\theta}}^{-1}$ where the (l, m)th element of $\mathbf{D}_{\boldsymbol{\theta}}$ is $\frac{1}{2}\mathsf{trace}(\mathbf{\Sigma}^{-1}\frac{\partial\mathbf{\Sigma}}{\partial\theta_l}\mathbf{\Sigma}^{-1}\frac{\partial\mathbf{\Sigma}}{\partial\theta_m})$, and $\boldsymbol{\theta} = [\sigma, \beta]$. Unfortunately, jointly sampling the latent variables and the hyperparameters proves to be computationally too costly to implement, as the metric tensor is now no longer fixed and so the generalised leapfrog integration scheme must be implemented with fixed point iterations, during each of which the metric tensor and its inverse have to be recalculated. We therefore exploit the block diagonal structure of the metric tensor, and employ a Gibbs scheme in which we alternately sample from $p(\mathbf{x}|\mathbf{y}, \sigma, \beta, \mu)$ and $p(\sigma, \beta|\mathbf{y}, \mathbf{x}, \mu)$. A standard leapfrog integrator may then be used to generate samples of the latent variables, and a generalised leapfrog scheme for obtaining samples from the 2 dimensional hyperparameter space. The required partial derivatives of the metric tensor with respect to the hyperparameters follow straightforwardly and are given in Appendix B.

Noting that the metric tensor for the latent variables has dimension $N \times N$, where $N = 4096$ the $\mathcal{O}(N^3)$ operations required in the RM-HMC scheme are clearly going to be computationally costly. However, it should also be noted that in previous studies of this Log-Gaussian Cox process, (Christensen *et al.*, 2005), a transformation of the latent Gaussian field is necessary based on the Cholesky decomposition of $\mathbf{\Sigma}^{-1} + \mathsf{diag}(\mathbf{x})$, which will therefore also scale as $\mathcal{O}(N^3)$.

9.1. Experimental Results for Log-Gaussian Cox Processes

Following the example given by Christensen *et al.* (2005), we fix the parameters $\beta = 1/33$, $\sigma^2 = 1.91$ and $\mu = \log(126) - \sigma^2/2$. We generate a latent Gaussian field, **x**, from the Gaussian process and use these values to generate count data **y** from the latent intensity process $\mathbf{\Lambda}$. Given the generated data and the fixed hyperparameters, we infer **x** using mMALA, RM-HMC and the MALA method as in Christensen *et al.* (2005). The algorithms were run on a single AMD Opteron processor with 8GB of memory and were coded in Matlab for consistency.

In many settings MALA, like HMC, is particularly sensitive to the choice of scaling and very often a reparameterisation of the target density is required for these methods to be effective. Indeed this is seen to be the case with this particular example, where MALA is unable to sample **x** directly. We therefore follow Christensen *et al.* (2005) and employ the transformation $\mathbf{X} = \mu\mathbf{1} + \mathbf{L}\mathbf{\Gamma}$, where $\mathbf{L}$ is obtained by Cholesky factorisation such that $\{\mathbf{\Sigma} - \text{diag}(\mathbf{x})\}^{-1} = \mathbf{L}\mathbf{L}^{\mathsf{T}}$. Even after this reparameterisation, it is still necessary to carefully tune the scaling factor for this method to work at all. This challenging aspect of employing MALA has been investigated in detail by Christensen *et al.* (2005) who characterise the problem very well, advising great care in its implementation, but are

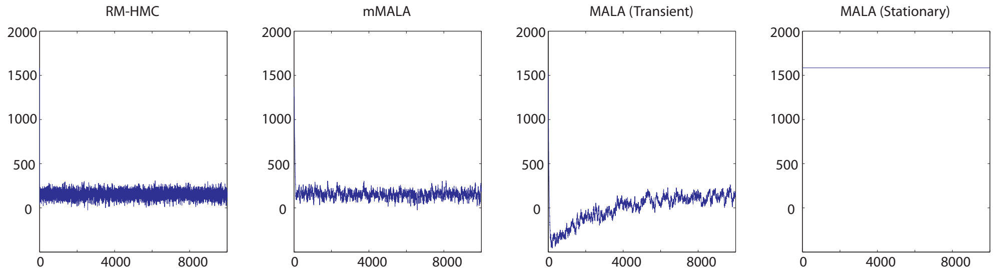

Fig. 7. Trace plots of the log joint-likelihood for the first 5000 samples of the latent variables of a log-Gaussian Cox process. The left hand plot shows the convergence of the RM-HMC scheme which is able to directly sample the latent variables **x** without the need for ad-hoc reparameterisations and pilot runs for fine-tuning. The left-middle plot shows the convergence of the mMALA scheme which, since it also uses information about the manifold in the form of the metric tensor, is able to directly sample without any reparameterisations. The right-middle plot shows the log joint-likelihood for samples drawn by MALA using a reparameterisation of the latent variables. The scaling was carefully tuned to allow traversal of the parameter space to the posterior mode. The right hand plot shows the trace of the MALA sampler tuned for optimally sampling from the posterior mode. We note that the algorithm is now unable to traverse the parameter space when initialised away from this mode. Such fine-tuning and reparameterisation is frequently necessary when employing MALA.

ultimately unable to offer any panacea. In contrast to the necessary transformation and fine-tuning required by MALA, both mMALA and RM-HMC allow us to directly sample the latent variables $\mathbf{x}$ *without* reparameterising the target density.

Figure 7 shows the traces of the log joint-likelihood for both methods using the starting position $x_{i,j} = \mu$ for $i, j = 1, \ldots, 64$. Note that for MALA these starting positions must be transformed into corresponding values for $\mathbf{\Gamma}$. The RM-HMC sampler quickly converges to the true mode after very minimal tuning of the integration stepsize based on the integration error, which corresponds directly to the acceptance rate. mMALA also converges very quickly to the true posterior mode. MALA converges in a similar number of iterations, but only for a suitable choice of scaling factor. The right-middle plot in Figure 7 shows convergence when the scaling factor is carefully tuned for the transient phase of the Markov chain, however the right hand plot demonstrates how it fails to converge at all given a scaling factor which is tuned for stationarity. Detailed results of the sampling efficiency of each method are given in table 13. In this example the RM-HMC method required just 1.5 seconds per effectively independent sample compared to more than 2 hours needed by MALA. In addition to taking far longer to sample, MALA also generates much more highly correlated samples and as a result has a far worse effective sample size. This can also be seen in figure 8 which shows the inferred posterior latent field, the posterior latent process and the variance associated with the Monte Carlo estimate. For RM-HMC, the variance in the estimates increases where there is little data, i.e. in the top right hand corner of the field. mMALA has slightly more variability, while the low ESS of the MALA methods methods manifests itself in patchy regions of high variability across the entire field. We note that MALA tuned for stationarity has slightly lower variance than MALA tuned for the transient phase, as one would expect.

Conditionally sampling the hyperparameters, in addition to the latent variables, using RM-HMC proves more costly, with 5000 posterior samples taking around 90 hours of computation time. However, the posterior estimates for the hyperparameters correspond extremely well to their true values, see Figure9.

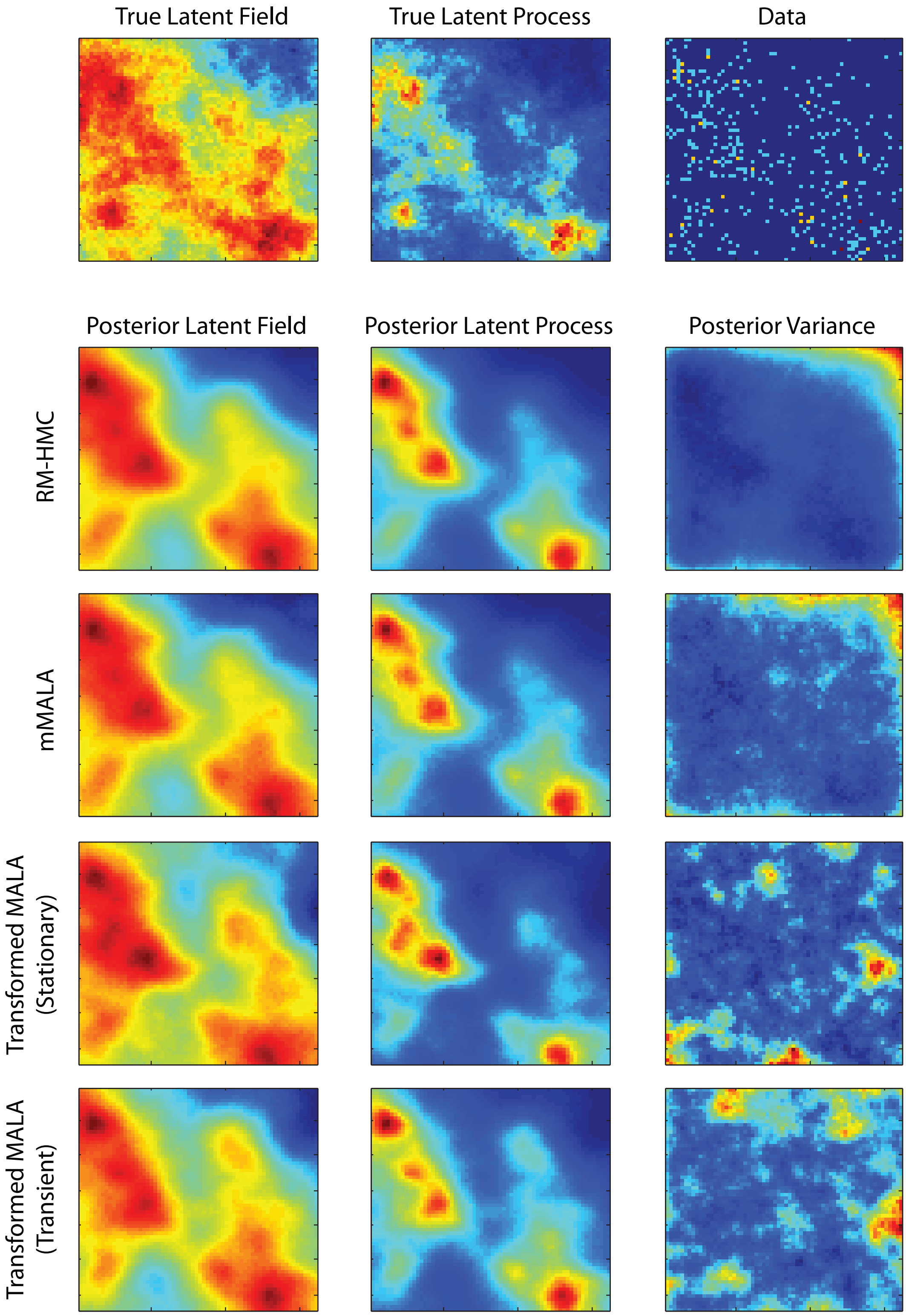

Fig. 8. Posterior latent fields and processes and associated variance, using each of the sampling methods, compared to the true latent field and process. The data employed to infer the latent field is also shown in the top left plot. RM-HMC produces the lowest variance estimates, which corresponds with it having the highest ESS. For RM-HMC there is higher variance where there is less data, however for the other methods there are patchy areas of high variance due to correlations in the collected samples.

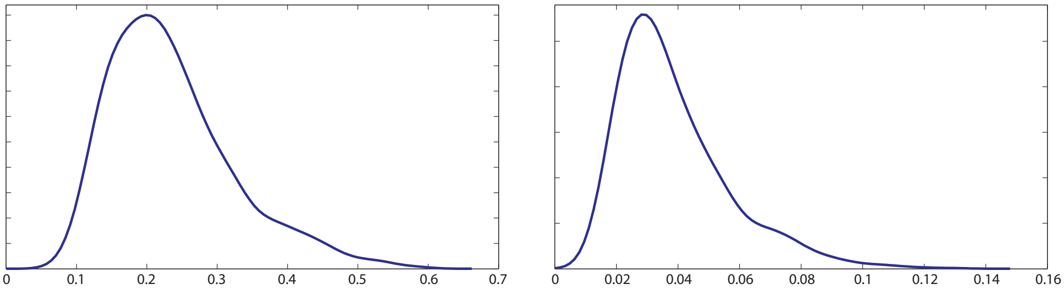

Fig. 9. Kernel density estimates of the hyperparameter samples obtained from Gibbs style sampling from the Log-Gaussian Cox model. The true values are $\sigma = 0.19$ (left hand plot) and $\beta = 0.03$ (right hand plot).

Table 13. Sampling the latent variables of a Log-Gaussian Cox Process - Comparison of sampling methods

Method	Time	ESS (Min, Med, Max)	s/Min ESS	Rel. Speed
MALA with Trans. (Transient)	31,577	(3, 8, 50)	10,605	×1
MALA with Trans. (Stationary)	31,118	(4, 16, 80)	7836	×1.35
mMALA	634	(26, 84, 174)	24.1	×440
RMHMC	2936	(1951, 4545, 5000)	1.5	×7070

Inferring the latent field of a log-Gaussian Cox process with a finely grained discretisation is clearly a very challenging problem due to the high dimensionality and strong spatial correlations present between the latent variables. The major challenges associated with employing MALA are firstly finding a suitable reparameterisation of the target density, and secondly making a suitable choice for the scaling factor according to whether the Markov chain is in a transient or stationary regime. In contrast, mMALA and RM-HMC do not exhibit such extreme technical difficulties. We have demonstrated that RM-HMC is able to sample the latent variables directly with minimal tuning and effort and without the need for reparameterisation. By employing a Gibbs style sampling scheme we were additionally able to sample the hyperparameters of the covariance function in a relatively computationally efficient manner. An investigation into the sparse approaches presented in (Vanhatalo and Vehtari, 2007; Rue *et al*, 2009) may provide further computational efficiencies. We will now turn our attention to the very topical application of statistical inference to nonlinear differential equations.

10. RM-HMC for Nonlinear Differential Equation Models

An important class of problems recently gaining attention is the statistical analysis of uncertainty in dynamical systems defined by a system of nonlinear differential equations (Ramsay *et al.*, 2007; Calderhead and Girolami, 2009; Vyshemirsky and Girolami, 2008). A dynamical system may be described by a collection of N nonlinear ordinary differential equations and model parameters $\boldsymbol{\theta}$ which define a functional relationship between the process state, $\mathbf{x}(\mathbf{t})$, and its time derivative such that $\dot{\mathbf{x}}(\mathbf{t}) = \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}, \mathbf{t})$. A sequence of process observations, $\mathbf{y}(\mathbf{t})$, are usually contaminated with some measurement error, which is modeled as $\mathbf{y}(\mathbf{t}) = \mathbf{x}(\mathbf{t}) + \boldsymbol{\epsilon}(\mathbf{t})$, where $\boldsymbol{\epsilon}(\mathbf{t})$ defines an appropriate multivariate noise process, e.g. a zero-mean Gaussian with variance σ_n^2 for each of the N states. If observations are made at T distinct time points, the $N \times T$ matrices summarise the overall observed system as $\mathbf{Y} = \mathbf{X} + \mathbf{E}$. In order to obtain values for $\mathbf{X}$, the system of ODEs must be solved, so that in the case of an initial value problem $\mathbf{X}(\boldsymbol{\theta}, \mathbf{x}_0)$ denotes the solution of the system of equations at the specified time points for the parameters $\boldsymbol{\theta}$ and initial conditions $\mathbf{x}_0$. The posterior density follows by employing appropriate priors such that $p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{x}_0, \boldsymbol{\sigma}) \propto \pi(\boldsymbol{\theta}) \prod_n \mathcal{N}(\mathbf{Y}_{n,\cdot}|\mathbf{X}(\boldsymbol{\theta}, \mathbf{x}_0)_{n,\cdot}, \boldsymbol{\Sigma}_n^{-1})$.

By considering the Gaussian noise model described above, where $\boldsymbol{\Sigma}_n = \mathbf{I}_T \sigma_n^2$, we straightforwardly obtain the following analytic expressions for the metric tensor and its derivatives in terms of the first and second order sensitivities of the states of the differential equations. The T-dimensional vectors of first order sensitivities for the n'th component of state relative to the i'th parameter are denoted as $\mathbf{s}_{ni} = \partial \mathbf{x}_n / \partial \theta_i$. The metric tensor and its derivatives follow as

$$\mathbf{G}(\boldsymbol{\theta})_{ij} = \sum_{n=1}^{N} \mathbf{s}_{ni} \boldsymbol{\Sigma}_n^{-1} \mathbf{s}_{nj}^{\mathsf{T}} \qquad \frac{\partial \mathbf{G}(\boldsymbol{\theta})_{ij}}{\partial \theta_k} = \sum_{n=1}^{N} \left(\frac{\partial \mathbf{s}_{ni}}{\partial \theta_k} \boldsymbol{\Sigma}_n^{-1} \mathbf{s}_{nj}^{\mathsf{T}} + \mathbf{s}_{ni} \boldsymbol{\Sigma}_n^{-1} \frac{\partial \mathbf{s}_{nj}^{\mathsf{T}}}{\partial \theta_k} \right)$$

One method of obtaining the required sensitivities at all time points, is to approximate them using finite differences, however for our purposes this may be inaccurate. For this example we differentiate the system of equations with respect to each of the parameters and directly solve the first order sensitivity equations defined as follows

$$\dot{\mathbf{s}}_{ni} = \frac{\partial \mathbf{f}_n(\mathbf{x}, \boldsymbol{\theta}, \mathbf{t})}{\partial \theta_i} = \sum_{l=1}^{N} \frac{\partial \mathbf{f}_n}{\partial x_l} \mathbf{s}_{li}^{\mathsf{T}} + \frac{\partial \mathbf{f}_n}{\partial \theta_i}$$

Note that we must take the total derivative with respect to $\boldsymbol{\theta}$, since the states $\mathbf{x}$ also depend on the parameter values. We may augment the original system with these new differential equations, such that we may solve to obtain both the states and the sensitivities of the states. Similarly we may augment the system with additional equations to solve for the second order sensitivities, which are required for calculating the partial derivatives of the metric tensor with respect to the model parameters. These equations follow as

$$\frac{\partial \dot{\mathbf{s}}_{ni}}{\partial \theta_k} = \sum_{l=1}^{N} \left[\left(\sum_{m=1}^{N} \frac{\partial^2 \mathbf{f}_n}{\partial x_l \partial x_m} \mathbf{s}_{mk}^{\mathsf{T}} + \frac{\partial^2 \mathbf{f}_n}{\partial x_l \partial \theta_k} \right) \mathbf{s}_{li}^{\mathsf{T}} + \frac{\partial \mathbf{f}_n}{\partial x_l} \frac{\partial \mathbf{s}_{li}^{\mathsf{T}}}{\partial \theta_k} \right] + \sum_{l=1}^{N} \frac{\partial^2 \mathbf{f}_n}{\partial \theta_i \partial x_l} \mathbf{s}_{lk}^{\mathsf{T}} + \frac{\partial^2 \mathbf{f}_n}{\partial \theta_i \partial \theta_k}$$

We now have everything required to implement RM-HMC and mMALA sampling schemes for dynamical system models defined by systems of nonlinear differential equations.

10.1. Experimental Results for Nonlinear Differential Equations

We present results comparing the sampling efficiency for the parameters of the Fitzhugh Nagumo differential equations (Ramsay *et al.*, 2007),

$$\dot{V} = c \left(V - \frac{V^3}{3} + R \right), \quad \dot{R} = - \left(\frac{V - a + bR}{c} \right) \tag{32}$$

We obtain samples from the posterior distribution $p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{x}_0, \boldsymbol{\sigma})$, and so in this example $\mathbf{X}_{1,\cdot} = \mathbf{V}$ and $\mathbf{X}_{2,\cdot} = \mathbf{R}$. The sampling schemes we employ are Metropolis-Hastings, MALA, HMC, mMALA, simplified mMALA and RM-HMC, as first described in the section on Bayesian logistic regression. We again compare the simulations by calculating the effective sample size (ESS) normalised by the computational time required to produce the samples.

Before proceeding we require the first and second partial derivatives of the Fitzhugh Nagumo equations in order to calculate the metric tensor for employing manifold sampling approaches to

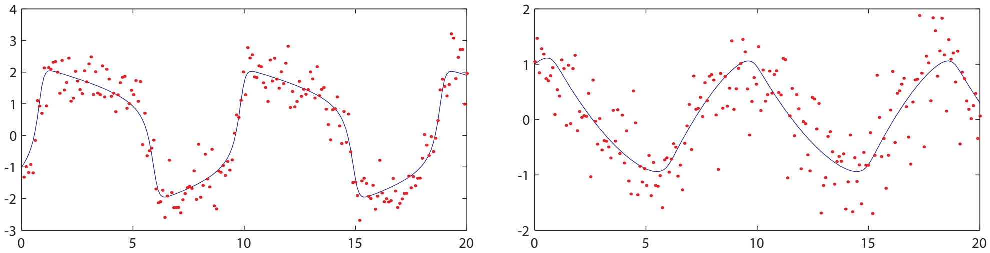

Fig. 10. Output for species V (left) and species R (right) of the Fitzhugh Nagumo model with parameters $a = 0.2$, $b = 0.2$, $c = 3$. An example noisy dataset is shown by the red points.

explore the posterior distribution, these are detailed in Appendix (C). In practice, all these expressions may be obtained automatically using symbolic differentiation and we supply Matlab code for this purpose.

10.1.1. Comparison of Sampling Schemes

We used 200 data points generated from the Fitzhugh Nagumo ODE model between $t = 0$ and $t = 20$ with the model parameters $a = 0.2$, $b = 0.2$, $c = 3$ and initial conditions $V(0) = -1$ and $R(0) = 1$. Gaussian distributed noise with standard deviation equal to 0.5 was then added to the data, see Figure 10.

Nonlinear ODEs generally induce corresponding nonlinearities in the target distribution, which may result in many local maxima. Careful attention must therefore be paid so that the Markov chains do not converge to the wrong mode, but rather sample from the correct distribution. All the sampling methods employed in this section may be embedded within a population MCMC framework to allow full exploration of and convergence to the target density (Calderhead *et al.*, 2009), however for the purpose of comparing sampling efficiency we employ a single Markov chain initialised on the true mode. We collected 5000 posterior samples and calculated the ESS for each parameter, using the minimum value to calculate the time per effectively independent sample. 10 simulations were run for each method, using the same dataset, and all methods were implemented in the interpreted language Matlab for consistency of comparison. All sampling methods were implemented in the same manner as previously described in Section 7.

The results of our simulations are shown in Table 14. Standard HMC takes the longest time for this problem due to the large number of leapfrog steps it needs to traverse the parameter space. RM-HMC on the other hand requires relatively few leapfrog steps, as it takes into account the local geometry to make better moves. We note however the additional computational cost of the leapfrog steps, during each of which it is necessary to solve the system of ODEs to evaluate the gradients and metric tensor. The first momentum update of RM-HMC is relatively quick since only a vector-matrix multiplication is necessary, however updating the parameter values requires the metric tensor to be evaluated for each fixed point iteration in the Generalised leapfrog algorithm as the parameter values converge, thus adding a considerable amount of computation to the overall algorithm. The mMALA methods offer the best performance for this particular example, as they have the benefit of using manifold information to guide the direction of the chain, but without the required fixed point iterations thus only requiring the ODEs to be numerically solved once per iteration. This suggests that mMALA is perhaps particularly suited for settings in which there is a non-flat metric tensor which is expensive to compute, as in this case.

Table 14. Fitzhugh Nagumo: Summary of results for 10 runs of the model parameter sampling scheme with 5000 posterior samples

Sampling Method	Time (s)	Mean ESS (a, b, c)	Total Time/ (Min mean ESS)	Relative Speed
Metropolis	18.5	132, 130, 108	0.17	$\times 3.9$
MALA	14.4	125, 21, 46	0.67	$\times 1$
HMC	815	4668, 3483, 3811	0.23	$\times 2.9$
mMALA	34.9	1057, 925, 956	0.037	$\times 18.1$
mMALA Simp.	14.9	1007, 479, 762	0.031	$\times 21.6$
RM-HMC	266	4302, 4202, 3199	0.083	$\times 8$

The Fitzhugh Nagumo model has only three parameters and we see that MALA and HMC perform adequately in this low dimensional setting, indeed the largest marginal parameter variance is only four times larger than the smallest marginal variance. We would expect MALA and HMC to perform worse in cases where there is a greater difference in the marginal variances, since the step size of each is restricted by the smallest marginal variance. Similarly, while component-wise Metropolis performs adequately in this setting, we would expect its performance to deteriorate in higher dimensions where there are greater correlations in the parameters.

11. Conclusions and Discussion

In this paper Riemannian Manifold Metropolis Adjusted and Hamiltonian Monte Carlo sampling methods have been proposed and evaluated in an attempt to improve upon existing MCMC methodology when sampling from target densities that may be of high dimension and exhibit strong correlations. It is argued that the methods are fully automated in terms of tuning the overall proposal mechanism to accommodate target densities which may exhibit strong correlations, widely varying scales in each dimension, and significant changes in the geometry of the manifold between the transitional and stationary phases of the Markov chain.

By exploiting the natural Riemannian structure of the parameter space of statistical models the proposed methods can be viewed as generalisations of both HMC and MALA methods and as such overcome the oftentimes complex manual tuning required of both methods. In high dimensional problems such as inferring the 4096 dimensional latent Gaussian field, MALA and HMC fail completely due to the high levels of spatial correlation in the latent field and can only proceed after a transformation is used to break those correlations. In contrast mMALA and RM-HMC proceed without the need for such a transformation or indeed any phase specific tuning.

Clearly there are two main overheads when employing mMALA or RM-HMC, the first being the ability to develop analytical expressions, or stable numerical alternatives, for the metric tensor (once it has been chosen) and the associated derivatives. The second is the worst case $\mathcal{O}(N^3)$ scaling of solving the linear systems when updating the parameter vectors i.e. inverting the metric tensor, especially for high dimensional problems. The issue of the $\mathcal{O}(N^3)$ scaling is something which deserves further consideration. In some statistical models there is a natural sparsity in the metric tensor, the SVM is a case in point where due to this structure RM-HMC was computationally more efficient than mMALA and HMC. In other models this is not the case, for example the logistic regression model and the Log-Gaussian Cox model. It should be noted that adaptive MCMC methods, see e.g. Andrieu and Thoms (2008), also incur the same level of cubic scaling. At the very high dimensional end of the scale a decorrelating transformation is required for MALA and HMC and this will also incur an $\mathcal{O}(N^3)$ scaling however further work to characterise the incurred computational costs at the intermediate dimensionality regime will be of value. As far as the computational issues are concerned automatic or adjoint differentiation methods may prove to be of use, and Hanson (2002) has proposed adjoint methods for HMC. There are clearly a number of numerical and computational avenues of investigation that may be followed in this regard.

In this paper all the examples that have been considered have had analytic expressions for the Fisher Information. However there are whole families of statistical models for which the Fisher Information is not available in closed analytic form, mixture models being an obvious example. In these cases it may be possible to either estimate the expected Fisher Information (Spall, 2005) or employ the observed Fisher Information, although numerical issues such as the loss of guaranteed positive-definiteness would require consideration. It is unclear what type of manifold structure this would induce so the theoretical and practical implications of the difference between the expected and observed information matrices would be worthy of further investigation. This leads onto the discussion about the particular choice of metric to be employed if one takes the view that the Fisher Information is but one possible metric that could be adopted. Alternatives have already been considered in the literature, e.g. the Preferred Point metric (Critchley *et al*, 1993) although not within the context of MCMC and this presents a new area of analysis and study to characterise the principles of optimality in appropriate metric design for MCMC.

A note of caution regarding the exploitation of the geometry induced by the Fisher Information metric in inference problems is spelled out in (Skilling, 2006). Two distributions may be a short distance apart on the probability simplex, however if the parameter sub-manifold (which we are interested in) is locally *rough* they may well be distantly separated and hence following small-scale detailed paths on the sub-manifold will be highly inefficient. This is not an observation made in this paper however there are many examples where this may well be a real problem, for example inference over dynamic systems that exhibit complex limit cycles is challenging due to the small scale structure induced in the likelihood (Calderhead *et al.*, 2009). Further theoretical and applied investigation will help to understand this issue more fully.

The work of (Christensen *et al.*, 2005; Roberts and Rosenthal, 1998; Roberts and Stramer, 2003) have provided theoretical analysis of limiting rates of convergence, egodicity, optimal step sizes and acceptance rates for MALA, and more recently HMC (Beskos *et al*, 2010). This type of theoretical study will be required for the mMALA and RM-HMC class of MCMC methods to characterise their theoretical properties in a rigorous manner. The highly promising performance reported in the experimental evaluation of mMALA and RM-HMC on challenging inference problems gives further motivation for this theoretical analysis.

From the experimental evaluation the raw ESS values for RM-HMC far exceeds that of mMALA despite both methods being based on geometric principles. There are a number of reasons for this, firstly the mMALA proposal is based on a single forward step of the Euler integrator whilst the proposal mechanism for RM-HMC can take multiple integration steps thus traveling further on the manifold (parameter space) for each proposal. Secondly the discrete version of the Langevin diffusion is being driven by a diffusion term defined by the metric tensor at the current point rather than the new one. Depending on the step size this will introduce further inefficiency based on deviation from the manifold of the effective path. Thirdly as has already been mentioned Hamiltonian flows of the form employed in RM-HMC are locally geodesic flows (Calin and Chang, 2004; McCord *et al*, 2002) suggesting a possible optimality, in terms of distance, in the paths simulated across the manifold by RM-HMC. This is an interesting point which requires further theoretical analysis to characterise the nature of these local geodesics and how they may be exploited further in this regard.

In summary the mMALA and RM-HMC methods provide novel MCMC algorithms whose performance has been assessed on a diverse range of statistical models and in all cases has been shown to be superior to similar MCMC methods. The adoption of this geometric viewpoint when designing MCMC algorithms provides a framework in which to further develop the theory, methodology, and application of this promising avenue of statistical inference.

12. Acknowledgements

M. Girolami is supported by an Engineering and Physical Sciences Research Council (EPSRC) Advanced Research Fellowship EP/E052029/1, and the Biotechnology and Biological Sciences Research Council (BBSRC) project grant BB/G006997/1. B. Calderhead is supported by a Microsoft Research European PhD Scholarship.

References

Amari. S. and Nagaoka. H. (2000) *Methods of Information Geometry*, Oxford University Press.

Andrieu. C. and Thoms. J. (2008). A Tutorial on Adaptive MCMC. *Statistics and Computing*, 18, pp. 343–373.

Andrieu, C. Doucet, A. and Holenstein, R. (2009). Particle Markov chain Monte Carlo (with discussion), *J. Royal Statist. Society Series B*, 72.

Barndorff-Nielsen, O. E, Cox, D.R, and Reid, N. (1986) The Role of Differential Geometry in Statistical Theory. *International Statistical Review*, 54, 83 - 96.

Beichl. I. and Sullivan. F. (2000). The Metropolis Algorithm. *Computing in Science and Engineering*. 2(1). pp 65–69.

Beskos. A. Pillai. N. Roberts. G. Serna. S. and Stuart. A. (2010) Optimal tuning of the Hybrid Monte-Carlo algorithm, *Technical Report* Department of Statistical Science, UCL.

Calderhead. B. Girolami. M and Lawrence. N. D. (2009). Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes, *Advances in Neural Information Processing*, 21, 217-224. MIT Press.

Calderhead. B. and Girolami. M. (2009) Estimating Bayes Factors via Thermodynamic Integration and Population MCMC, *Computational Statistics and Data Analysis*, 53, 4028 - 4045.

Calin. O. and Chang. D.C. (2004) *Geometric Mechanics on Riemannian Manifolds*. Birkhäuser.

Christensen. O.F. Roberts. G.O. and Rosenthal. J.S. (2005). Scaling Limits for the Transient Phase of Local Metropolis-Hastings Algorithms. *Journal of the Royal Statistical Society: Series B.* 67(2), pp. 253–268.

Chung. K.L. (1982) *Lectures from Markov Processes to Brownian Motion*, Springer.

Critchley, F., Marriot, P.K., and Salmon, M. (1993). Preferred Point Geometry and Statistical Manifolds. *The Annals of Statistics*, 21, 1197-1224.

Dawid, P. (1975) Invited Discussion of *Defining the Curvature of a Statistical Problem (with Applications to Second-Order Efficiency). The Annals of Statistics*, 3, 1231 - 1234.

Duane. S. Kennedy. A. D. Pendleton. B. J. and Roweth. D. (1987) Hybrid Monte Carlo, *Physics Letters. B.*, 55, pp. 2774–2777.

Efron, B. (1975) Defining the Curvature of a Statistical Problem (with Applications to Second-Order Efficiency). *The Annals of Statistics,* 3, 1189 - 1242.

Ferreira. P.E. (1981). Extending Fisher's Measure of Information. *Biometrika*. 68(3), pp. 695-698.

Gamerman. D. (1997). Sampling from the posterior distribution in generalized linear mixed models. *Statistics and Computing*. 7, pp.57-68.

Gelman. A. Carlin. J.B. Stern. H.S. and Rubin. D.B. (2004) *Bayesian Data Analysis*, Chapman & Hall.

Geyer. C. J. (1992) Practical Markov Chain Monte Carlo. *Statistical Science*. 7. pp 473 – 483.

Gustafson. P. (1997) Large Hierarchical Bayesian Analysis of Multivariate Survival Data. *Biometrics*. 53, pp 230 – 242.

Hajian. A. (2007) Efficient Cosmological Parameter Estimation with Hamiltonian Monte Carlo Technique. *Phys. Rev. D*. 75. 083525 – 1– 11.

Hanson. K. M. (2001) Markov Chain Monte Carlo posterior sampling with the Hamiltonian method, *Medical Imaging: Image Processing*, M. Sonka and K. M. Hanson, eds., Proc. SPIE 4322, 456-467.

Hanson. K. M. (2002) Use of Probability Gradients in Hybrid MCMC and a New Convergence Test. *Los Alamos Report LA-UR-02-4105*, summary of talk presented at 7th Valencia International Meeting on Bayesian Statistics.

Hastings. W.K. (1970) Monte Carlo Sampling Methods using Markov Chains and their Applications. *Biometrika* 57, pp 97-109.

Holmes. C.C. and Held. L. (2005). Bayesian Auxiliary Variable Models for Binary and Multinomial Regression, *Bayesian Analysis*, 1(1), pp. 145–168.

Husmeier D. , Penny W., and Roberts S.J. (1999). An Empirical Evaluation of Bayesian Sampling with Hybrid Monte Carlo for Training Neural Network Classifiers, *Neural Networks*, 12, 677-705.

Ishwaran. H. (1999) Applications of Hybrid Monte Carlo to Bayesian Generalised Linear Models: Quasicomplete Separation and Neural Networks. *Journal of Computational and Graphical Statistics*. 8, pp 779 – 799.

Jeffreys. H. (1948) *Theory of Probability*, 2nd Edition, Clarendon, Oxford.

Johnson. V. E. Krantz. S. G. and Albert. J. H. (1999) *Ordinal Data Modeling*. Springer Verlag.

Kass. R.E. (1989) The Geometry of Asymptotic Inference. *Statistical Science*. 4(3), pp 188–234.

Kent. J. (1978) Time reversible Diffusions. *Adv. Appl. Probab,* 10, 819-835.

Kim. S. Shephard. N. and Chib. S. (1998) Stochastic Volatility: Likelihood Inference and Comparison with ARCH Models. *The Review of Economic Studies,* 65(3), 361-393.

Lambert. P. and Eilers. P.H.C. (2009) Bayesian Density Estimation from Grouped Continuous Data *Computational Statistics and Data Analysis*. 53(4), pp 1388–1399.

Lauritzen. S.L. (1987) *Statistical Manifolds*. In: Differential Geometry in Statistical Inference, pp. 165-216. IMS Monographs, Vol. X., Hayward, CA.

Leimkuhler. B. and Reich. S. (2004) *Simulating Hamiltonian Dynamics*, Cambridge University Press.

Liu. J.S. (2001) *Monte Carlo Strategies in Scientific Computing*. New York: Springer

McCord. C., Meyer. K.R. and Offin. D. Are Hamiltonian Flows Geodesic Flows?. *Transactions of the American Mathematical Society*, 355(3), 1237-1250.

Metropolis. M. Rosenbluth. A.W. Rosenbluth. M.N. Teller.A.H. and Teller. E. (1953) Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics*. 21, pp 1087–1092.

Michie, D., Spiegelhalter, D. J., and Taylor, C. C. (1994). *Machine Learning, Neural and Statistical Classification*. Prentice Hall, Englewood Cliffs, N.J.

Murray. M.K. and Rice. J.W. (1993) *Differential Geometry and Statistics* Chapman and Hall, CRC.

Neal. R.M. (1993a). *Probabilistic Inference Using Markov Chain Monte Carlo Methods*. Technical. Report, University of Toronto, Canada.

Neal. R.M. (1996). *Bayesian Learing for Neural Networks*. Springer, Lecture Notes in Statistics, New York.

Neal. R.M. (1993b) Bayesian Learning via Stochastic Dynamics. *Advances in Neural Information Processing Systems*, 5. pp. 475–482.

Neal. R.M. (2010) *Handbook of Markov Chain Monte Carlo* edited by S. Brooks, A. Gelman, G. Jones, and Xiao-Li Meng, CRC Press.

Ramsay. J. Hooker. G. Campbell. D. J. and Cao. J. (2007) Parameter Estimation for Differential Equations: A Generalized Smoothing Approach, *Journal of the Royal Statistical Society: Series B*, 69 (5). pp 741–796.

Rao. C. R. Information and Accuracy Attainable in the Estimation of Statistical Parameters. *Bulletin of the Calcutta Mathematical Society*. 37. pp 81 – 91.

Ripley. B. D. (1996) *Pattern Recognition and Neural Networks*. Cambridge University press.

Robert. C. (2004). *Monte Carlo Statistical Methods*. Springer Verlag.

Roberts. G. and Rosenthal. J. S. (1998) Optimal Scaling of Discrete Approximations to Langevin Diffusions. *Journal of Royal Statistical Society, B*. 60. pp 255 –268.

Roberts. G. and Stramer. O. (2003) Langevin Diffusions and Metropolis-Hastings Algorithms. *Methodology and Computing in Applied Probability*. 4. pp 337–358.

Rue. H. Martino. S. Chopin. N. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71, 319-392.

Skilling. J. (2006) Probability and Geometry. *ESA-EUSC: Image Information Mining for Security and Intelligence* `http://earth.eo.esa.int/rtd/Events/ESA-EUSC_2006/Oral/Ar19_Skilling.pdf`

Spall, J. C. (2005). Monte Carlo Computation of the Fisher Information Matrix in Nonstandard Settings, *Journal of Computational and Graphical Statistics*, 14, 889-909.

Tsutakawa. R.K. (1972) Design of Experiment for Bioassay *Journal of the American Statistical Association*. 67(339). pp 584–590.

Vanhatalo. J. and Vehtari. A. (2007) Sparse Log Gaussian Processes via MCMC for Spatial Epidemiology, *JMLR Workshop and Conference Proceedings, Gaussian Processes in Practice*, 1, 73-89.

Vyshemirsky. V. and Girolami. M. (2008) Bayesian Ranking of Biochemical System Models, *Bioinformatics* 24, (2008), pp 833–839.

Zlochin. M. and Baram. Y. (2001) Manifold Stochastic Dynamics for Bayesian Learning. *Neural Computation*, 13. pp 2549–2572.

A. Required Expressions for Stochastic Volatility Model

The full joint target distribution can be written as

$$p(\mathbf{y}, \mathbf{x}, \beta, \sigma, \phi) = \prod_{t=1}^{T} p(y_t|x_t, \beta) p(x_1) \prod_{t=2}^{T} p(x_t|x_{t-1}, \sigma, \phi) \pi(\beta) \pi(\sigma) \pi(\phi) \tag{33}$$

where, similar to Liu (2001), we use the priors $p(\beta) \propto \exp(\beta)$, $\sigma^2 \sim \text{Inv-}\chi^2(10, 0.05)$ and $(\phi + 1)/2 \sim \text{Beta}(20, 1.5)$. The partial derivatives of joint log likelihood, $L = p(\mathbf{y}, \mathbf{x}|\beta, \sigma, \phi)$, are as follows

$$\begin{aligned}
\frac{\partial L}{\partial \beta} &= -\frac{T}{\beta} + \sum_{t=1}^{T} \frac{y_t^2}{\beta^3 \exp(x_t)} && (34)\\
\frac{\partial L}{\partial \sigma} &= -\frac{T}{\sigma} + \frac{x_1^2(1-\phi^2)}{\sigma^3} + \sum_{t=2}^{T} \frac{(x_t - \phi x_{t-1})^2}{\sigma^3} && (35)\\
\frac{\partial L}{\partial \phi} &= -\frac{\phi}{(1-\phi^2)} + \frac{\phi x_1^2}{\sigma^2} + \sum_{t=2}^{T} \frac{x_{t-1}(x_t - \phi x_{t-1})}{\sigma^2} && (36)
\end{aligned}$$

If we want to sample the parameters using mMALA or RM-HMC, then we also need expressions for the metric tensor and its partial derivatives with respect to β, σ and ϕ. We can obtain the following expressions for the individual components of the metric tensor for the likelihood

$$\begin{aligned}
E\left\{\frac{\partial L}{\partial \beta}\frac{\partial L}{\partial \beta}\right\} &= \frac{2T}{\beta^2}, \quad E\left\{\frac{\partial L}{\partial \sigma}\frac{\partial L}{\partial \sigma}\right\} = \frac{2T}{\sigma^2}, \quad E\left\{\frac{\partial L}{\partial \beta}\frac{\partial L}{\partial \sigma}\right\} = E\left\{\frac{\partial L}{\partial \beta}\frac{\partial L}{\partial \phi}\right\} = 0 && (37)\\
E\left\{\frac{\partial L}{\partial \sigma}\frac{\partial L}{\partial \phi}\right\} &= \frac{2\phi}{\sigma^3(1-\phi^2)}, \quad E\left\{\frac{\partial L}{\partial \phi}\frac{\partial L}{\partial \phi}\right\} = \frac{2\phi^2}{(1-\phi^2)^2} + \frac{T-1}{1-\phi^2} && (38)
\end{aligned}$$

Thus the metric tensor for the likelihood and partial derivatives follow as

$$\mathbf{G}(\phi,\sigma,\beta) = \begin{bmatrix} \frac{2T}{\beta^2} & 0 & 0 \\ 0 & \frac{2T}{\sigma^2} & \frac{2\phi}{\sigma^3(1-\phi^2)} \\ 0 & \frac{2\phi}{\sigma^3(1-\phi^2)} & \frac{2\phi^2}{(1-\phi^2)^2} + \frac{T-1}{1-\phi^2} \end{bmatrix}, \frac{\partial \mathbf{G}}{\partial \beta} = \begin{bmatrix} -\frac{4T}{\beta^3} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\frac{\partial \mathbf{G}}{\partial \sigma} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{4T}{\sigma^3} & -\frac{6\phi}{\sigma^4(1-\phi^2)} \\ 0 & -\frac{6\phi}{\sigma^4(1-\phi^2)} & 0 \end{bmatrix},$$

$$\frac{\partial \mathbf{G}}{\partial \phi} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{2}{\sigma^3(1-\phi^2)} + \frac{4\phi^2}{\sigma^3(1-\phi^2)^2} \\ 0 & \frac{2}{\sigma^3(1-\phi^2)} + \frac{4\phi^2}{\sigma^3(1-\phi^2)^2} & \frac{2\phi(1+T)}{(1-\phi^2)^2} + \frac{6\phi^3}{(1-\phi^2)^3} \end{bmatrix}$$

We therefore require expressions for the second order derivatives of the log priors, to get the metric tensor over the full target distribution, and also the third order derivatives of the log priors to calculate the partial derivatives of the metric tensor, these follow straightforwardly.

B. Required Expressions for Log Gaussian Cox Process Model

The Fisher Information matrix for inferring the hyperparameters of the Gaussian Process follows in standard form as

$$\mathbf{G}(\boldsymbol{\theta})_{ij} = \frac{1}{2}\text{trace}\left(\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_i}\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_j}\right) \tag{39}$$

Application of standard derivative of trace operators provides an analytic expression for the derivative of the metric tensor with respect to the parameters

$$\begin{aligned} \frac{\partial \mathbf{G}(\boldsymbol{\theta})_{ij}}{\partial \theta_k} &= \frac{\partial}{\partial \theta_k}\left[\frac{1}{2}\text{trace}\left(\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_i}\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_j}\right)\right] \\ &= -\frac{1}{2}\text{trace}\left(\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_k}\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_i}\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_j}\right) + \frac{1}{2}\text{trace}\left(\boldsymbol{\Sigma}^{-1}\frac{\partial^2 \boldsymbol{\Sigma}}{\partial \theta_i \partial \theta_k}\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_j}\right) \\ &\quad -\frac{1}{2}\text{trace}\left(\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_i}\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_k}\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_j}\right) + \frac{1}{2}\text{trace}\left(\boldsymbol{\Sigma}^{-1}\frac{\partial \boldsymbol{\Sigma}}{\partial \theta_i}\boldsymbol{\Sigma}^{-1}\frac{\partial^2 \boldsymbol{\Sigma}}{\partial \theta_j \partial \theta_k}\right) \end{aligned}$$

In our experiments we employ an infinitely differentiable stationary covariance function to calculate the $(i,j)^{th}$ entry of the covariance matrix,

$$\mathbf{K}_{i,j} = \varphi_1 \exp\left(-\frac{1}{2\varphi_2^2}(t_j - t_i)^2\right) + \sigma\delta_{ij} \tag{40}$$

The Fisher Information matrix above may therefore be obtained using the first and second partial derivatives of the covariance function. The first partial derivatives follow as,

$$\frac{\partial \mathbf{K}_{i,j}}{\partial \varphi_1} = \frac{1}{\varphi_1}(\mathbf{K}_{i,j} - \sigma\delta_{ij}), \quad \frac{\partial \mathbf{K}_{i,j}}{\partial \varphi_2} = \frac{1}{\varphi_2^3}(\mathbf{K}_{i,j} - \sigma\delta_{ij})(t_j - t_i)^2, \quad \frac{\partial \mathbf{K}_{i,j}}{\partial \sigma} = \delta_{ij}$$

The second partial derivatives may also be easily calculated, and indeed out of the nine second partial derivatives, only three of them are non-zero which eases their computation.

$$
\begin{aligned}
\frac{\partial^2 \mathbf{K}_{i,j}}{\partial \varphi_1^2} &= \frac{\partial^2 \mathbf{K}_{i,j}}{\partial \varphi_1 \partial \sigma} = \frac{\partial^2 \mathbf{K}_{i,j}}{\partial \varphi_2 \partial \sigma} = \frac{\partial^2 \mathbf{K}_{i,j}}{\partial \sigma \partial \varphi_1} = \frac{\partial^2 \mathbf{K}_{i,j}}{\partial \sigma \partial \varphi_2} = \frac{\partial^2 \mathbf{K}_{i,j}}{\partial \sigma^2} = 0 \\
\frac{\partial^2 \mathbf{K}_{i,j}}{\partial \varphi_1 \partial \varphi_2} &= \frac{1}{\varphi_1} \frac{\partial \mathbf{K}_{i,j}}{\partial \varphi_2}, \quad \frac{\partial^2 \mathbf{K}_{i,j}}{\partial \varphi_2^2} = \frac{\partial \mathbf{K}_{i,j}}{\partial \varphi_1} \frac{\varphi_1}{\varphi_2^6} (1 - 3\varphi_2^2)(t_j - t_i)^2
\end{aligned}
$$

C. Partial Derivatives for ODE Example

$$
\frac{\partial \dot{V}}{\partial a} = \frac{\partial \dot{V}}{\partial b} = 0, \frac{\partial \dot{V}}{\partial c} = \left(V - \frac{V^3}{3} + R \right), \frac{\partial \dot{R}}{\partial a} = \frac{1}{c}, \frac{\partial \dot{R}}{\partial b} = \frac{-R}{c}, \frac{\partial \dot{R}}{\partial c} = \left(\frac{V - a + bR}{c^2} \right)
$$

All of the second derivatives of $\dot{V}$ with respect to the model parameters are equal to zero, and the five non-zero second partial derivatives of $\dot{R}$ are as follows,

$$
\frac{\partial^2 \dot{R}}{\partial a \partial c} = -\frac{1}{c^2}, \quad \frac{\partial^2 \dot{R}}{\partial b \partial c} = \frac{R}{c^2}, \quad \frac{\partial^2 \dot{R}}{\partial c \partial a} = -\frac{1}{c^2}, \quad \frac{\partial^2 \dot{R}}{\partial c \partial b} = \frac{R}{c^2}, \quad \frac{\partial^2 \dot{R}}{\partial c^2} = 2 \left(\frac{-V + a - bR}{c^3} \right)
$$

In addition, the second partial derivatives with respect to all states and parameters are required for writing the differential equation describing the second order sensitivities. There are again five non-zero second partial derivatives with respect to the states and parameters as follows

$$
\frac{\partial^2 \dot{V}}{\partial V \partial c} = 1 - V^2, \quad \frac{\partial^2 \dot{V}}{\partial R \partial c} = 1, \quad \frac{\partial^2 \dot{R}}{\partial V \partial c} = \frac{1}{c^2}, \quad \frac{\partial^2 \dot{R}}{\partial R \partial b} = -\frac{1}{c}, \quad \frac{\partial^2 \dot{R}}{\partial R \partial c} = \frac{b}{c^2}
$$

D. Manifold MALA and RM-HMC Pseudocode

Algorithm 1 Manifold MALA

```
Initialise current $\boldsymbol{\theta}$
for IterationNum $= 1$ to NumSamples do
    Sample $\boldsymbol{\theta}^{\text{new}}$ based on Current $\boldsymbol{\theta}$ according to first order discretisation
    Calculate current log-likelihood $\mathcal{L}(\boldsymbol{\theta})$ and proposed log-likelihood $\mathcal{L}(\boldsymbol{\theta}^{\text{new}})$
    Calculate $\log(p(\boldsymbol{\theta}^{\text{new}}|\boldsymbol{\theta}))$, $\log(p(\boldsymbol{\theta}|\boldsymbol{\theta}^{\text{new}}))$, $\log(\text{Prior}(\boldsymbol{\theta}))$, $\log(\text{Prior}(\boldsymbol{\theta}^{\text{new}}))$
    Ratio $= \mathcal{L}(\boldsymbol{\theta}^{\text{new}}) + \log(\text{Prior}(\boldsymbol{\theta}^{\text{new}})) + \log(p(\boldsymbol{\theta}|\boldsymbol{\theta}^{\text{new}})) - \mathcal{L}(\boldsymbol{\theta}) - \log(\text{Prior}(\boldsymbol{\theta})) - \log(p(\boldsymbol{\theta}^{\text{new}}|\boldsymbol{\theta}))$
    % Accept or reject according to Metropolis ratio
    if Ratio $> 0$ OR Ratio $> \log(\text{rand})$ then
        Current $\boldsymbol{\theta} = \boldsymbol{\theta}^N$
    end if
end for
```

Algorithm 2 RMHMC with Generalised Leapfrog

```
Initialise current $\boldsymbol{\theta}$
for IterationNum $= 1$ to NumSamples do
  Sample new momentum $\mathbf{p}^1$
  Calculate current $H(\boldsymbol{\theta}, \mathbf{p}^1)$
  Randomise N (leapfrog steps)
  $\boldsymbol{\theta}^1 =$ Current $\boldsymbol{\theta}$
  for $n = 1$ to $N$ (leapfrog steps) do
    % Update the momentum with fixed point iterations
    $\hat{\mathbf{p}}^0 = \mathbf{p}^n$
    for $i = 1$ to NumOfFixedPointSteps do
      $\hat{\mathbf{p}}^i = \mathbf{p}^n - \frac{\epsilon}{2}\nabla_{\boldsymbol{\theta}} H(\boldsymbol{\theta}^n, \hat{\mathbf{p}}^{i-1})$
    end for
    $\mathbf{p}^{n+\frac{1}{2}} = \hat{\mathbf{p}}^i$
    % Update the parameters with fixed point iterations
    $\hat{\boldsymbol{\theta}}^0 = \boldsymbol{\theta}^n$
    for $i = 1$ to NumOfFixedPointSteps do
      $\hat{\boldsymbol{\theta}}^i = \boldsymbol{\theta}^n + \frac{\epsilon}{2}\nabla_{\mathbf{p}} H(\boldsymbol{\theta}^n, \mathbf{p}^{n+\frac{1}{2}}) + \frac{\epsilon}{2}\nabla_{\mathbf{p}} H(\hat{\boldsymbol{\theta}}^{i-1}, \mathbf{p}^{n+\frac{1}{2}})$
    end for
    $\boldsymbol{\theta}^{n+1} = \hat{\boldsymbol{\theta}}^i$
    % Update the momentum exactly
    $\mathbf{p}^{n+1} = \mathbf{p}^{n+1} - \frac{\epsilon}{2}\nabla_{\boldsymbol{\theta}} H(\boldsymbol{\theta}^{n+1}, \mathbf{p}^{n+\frac{1}{2}})$
  end for
  Calculate proposed $H(\boldsymbol{\theta}^N, \mathbf{p}^N)$
  Ratio $= -\log$(ProposedH) $+ \log$(CurrentH)
  % Accept or reject according to Metropolis ratio
  if Ratio $> 0$ OR Ratio $> \log$(rand) then
    Current $\boldsymbol{\theta} = \boldsymbol{\theta}^N$
  end if
end for
```