Adaptive MCMC for multimodal distributions

Chris Holmes*† Krzysztof Łatuszyński‡ Emilia Pompe§

September 21, 2017

Abstract

Poor performance of standard Monte Carlo techniques, such as the Metropolis-Hastings algorithm, on target distributions with isolated modes is a well-described problem in the MCMC literature. We propose a new Monte Carlo method, Adaptive MCMC for Multimodal Distributions, which can be used when the locations of the modes of the target distribution are known. Our algorithm relies on steps of two types: local ones, preserving the mode and jumps to a region associated with a different mode. The method we propose is adaptive to a significant extent, so it learns the optimal parameters while it runs, without requiring user intervention.

1 Introduction

MCMC algorithms are an extremely popular tool in statistics for all kinds of problems in which a researcher wants to obtain a sample from a complicated probability distribution. They are especially useful for the purposes of Bayesian inference by allowing to sample from posterior distributions. We will assume that the reader has a good understanding of MCMC methods (e.g. [Roberts et al., 2004]) and is familiar with the idea of Adaptive MCMC (e.g. [Roberts and Rosenthal, 2009]).

Despite their popularity, MCMC techniques have their drawbacks. For example, the classical MCMC methods, such as the Metropolis-Hastings algorithm or Hybrid Monte Carlo ([Duane et al., 1987]) are known to mix slowly between the modes if the target distribution is multimodal. The problem mentioned above is important in statistics, since multimodal distributions play a significant role in applications, e.g. in mixture models estimation ([Aitkin, 2001] and [Jasra et al., 2005]) or variable selection ([Lee et al., 2010]), to name a few. As for the variable selection problem, multimodal distributions arise in the Bayesian framework for regression model $y = X\beta + \epsilon$ as posteriors for β under some families of priors.

In Section 2 we introduce a new MCMC method, Adaptive MCMC for Multimodal Distributions, addressing the problem of obtaining a sample from a high-dimensional multimodal target distribution, in case when the locations of its modes are known. Section 3 illustrates an application of this algorithm to a toy example of a five-dimensional mixture of Gaussians. Section 4 is a discussion about the limitations of the method and ideas how the performance of the algorithm could potentially be enhanced. We outline a few possible future directions for development of the algorithm.

*Big Data Institute, University of Oxford
†Nuffield Department of Population Health, University of Oxford, (email cholmes@stats.ox.ac.uk).
‡Department of Statistics, University of Warwick, (email K.G.Latuszynski@warwick.ac.uk).
§Department of Statistics, University of Oxford, (email emilia.pompe@stats.ox.ac.uk).

2 Adaptive MCMC for Multimodal Distributions

In this note we report on our ongoing work on developing an MCMC algorithm suitable for multimodal distributions, especially in large dimensions. In this section we introduce our method of sampling from multimodal distributions, Adaptive MCMC for Multimodal Distributions.

Let $\{\mu_1, \ldots, \mu_N\}$ denote the set of modes of the target distribution π. We assume that their locations are known. They can be found using any algorithm for finding local maxima. We will address the issue of targeting distributions whose modes are unknown in our full paper, which will follow shortly.

2.1 Theoretical preliminaries

Before we proceed to present our method, we introduce a few concepts from the Adaptive MCMC literature that are crucial for our algorithm.

Adaptive MCMC algorithms attempt to tackle the problem of finding optimal parameters for MCMC algorithms, e.g. for the covariance matrix in the Random Walk Metropolis algorithm. They learn those parameters while they run and the adjustments of the parameters are based on the previous iterations of the algorithm. Development of the Adaptive MCMC was a considerable step in the direction of automating MCMC methods. Their key advantage is the user-friendliness, as manual tuning of the parameters is no longer necessary to ensure good mixing of the algorithm. Even though the Markov chain convergence theory does not apply to the Adaptive MCMC, there are conditions under which certain adaptive MCMC methods preserve stationarity ([Roberts and Rosenthal, 2007]). We present below two examples of such algorithms, which will later be used for construction of the main algorithm presented in this paper.

1. The first method is an adaptive version of the Random Walk Metropolis algorithm for a general covariance matrix of the proposal distribution, studied in [Haario et al., 2001]. Under some general assumptions, the optimal proposal distribution is given by $N\left(x, (2.38^2)\Sigma/d\right)$, where d is the dimension of the state space, x is the current state and Σ is the covariance matrix of π. Since in general Σ is not known, the following proposal provides the best approximation:

$$Q_{n+1}(y|X_n = x) = N\left(x, (2.38)^2 \Sigma_n/d\right),$$

where Σ_n is the estimate of the covariance matrix based on iterations from 1 to n (see [Roberts et al., 1997]). However, in practice a slightly modified version is often used:

$$Q_{n+1}(y|X_n = x) = (1-\beta)N\left(x, (2.38)^2 \Sigma_n/d\right) + \beta N\left(x, (0.1)^2 I_d/d\right),$$

where $0 < \beta << 1$ is a constant. Adding the other part of the mixture ensures that the proposed values do not belong to some lower-dimensional subspace of the state space, which may happen when the estimate for Σ_n is poor.

2. The second method is an adaptive version of the Random Walk Metropolis algorithm for a covariance matrix of the form $\sigma^2\Sigma$. Then the adaptive scheme may be given as follows:

$$\log(\sigma_{n+1}) = \log(\sigma_n) + n^{\gamma}\left(\alpha_n - \text{optimal acceptance rate}\right), \tag{1}$$

where σ_n is the value if σ used in iteration n, α_n is the probability of acceptance in iteration n, $0 < -\gamma \leq 1$ and the optimal acceptance rate is the value of the acceptance rate that was proved to ensure the best mixing. In this context the optimal acceptance rate is usually assumed to be equal to 0.44 in a one-dimensional case and 0.234 in higher dimensions (see [Roberts and Rosenthal, 2009]).

2.2 The algorithm

Let $\mathcal{X}$ be the space on which π is defined. Let $\mathcal{I}$ be the space of modes of π, i.e. $\mathcal{I} = \{\mu_1, \ldots, \mu_N\}$. For the purpose of designing a good MCMC algorithm that takes into account the knowledge we have about the locations of the modes, we consider an augmented state space $\mathcal{X} \times \mathcal{I}$. We construct an MCMC algorithm targeting the following distribution:

$$\tilde{\pi}(dx, i) = \pi(dx) \frac{w_i Q_i(\mu_i, \Sigma_i)(x)}{\sum_{j \in \mathcal{I}} w_j Q_j(\mu_j, \Sigma_j)(x)} \tag{2}$$

where $Q_i(\mu_i, \Sigma_i)$ is a symmetric distribution centred at mode μ_i with the covariance matrix Σ_i and w_i is a weight, i.e. $w_i \geq 0$ for $i \in \mathcal{I}$ and $\sum_{i \in \mathcal{I}} w_i = 1$. Note that the marginal distribution of $\tilde{\pi}$ with respect to x is equal to

$$\sum_{i \in \mathcal{I}} \tilde{\pi}(dx, i) = \sum_{i \in \mathcal{I}} \pi(dx) \frac{w_i Q_i(\mu_i, \Sigma_i)(x)}{\sum_{j \in \mathcal{I}} w_j Q_j(\mu_j, \Sigma_j)(x)} = \pi(dx).$$

For simplicity, we assume here that $w_i = \frac{1}{N}$ for $i \in \mathcal{I}$ and that Q_i is the normal distribution $N(\mu_i, \Sigma_i)$. Our algorithm makes steps of two types:

- **Local Move**: Assuming the current state of the chain is (x, i), a Random Walk Metropolis step is performed using the proposal distribution $Q_i(x, \Sigma_i)$. This move preserves the mode.
- **Jump Move**: A new mode j is proposed with probability a_j, where $\sum_{j \in \mathcal{I}} a_j = 1$. Then a new point y is proposed using the proposal distribution $Q_j(\mu_j, \Sigma_j)$.

Using the standard Metropolis-Hastings formula for the acceptance probability we obtain the following result for the Local Moves:

$$\alpha\left((x, i) \to (y, i)\right) = \min\left[1, \frac{\tilde{\pi}(dy, i)}{\tilde{\pi}(dx, i)}\right] = \min\left[1, \frac{\pi(dy) Q_i(\mu_i, \Sigma_i)(y)}{\pi(dx) Q_i(\mu_i, \Sigma_i)(x)} \frac{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(x)}{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(y)}\right]. \tag{3}$$

Similarly, for the Jump Moves we get:

$$\begin{aligned}
\alpha\left((x, i) \to (y, k)\right) &= \min\left[1, \frac{\tilde{\pi}(dy, k)}{\tilde{\pi}(dx, i)} \frac{a_i Q_i(\mu_i, \Sigma_i)(x)}{a_k Q_k(\mu_k, \Sigma_k)(y)}\right] = \\
&= \min\left[1, \frac{\pi(dy) Q_k(\mu_k, \Sigma_k)(y)}{\pi(dx) Q_i(\mu_i, \Sigma_i)(x)} \frac{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(x)}{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(y)} \frac{a_i Q_i(\mu_i, \Sigma_i)(x)}{a_k Q_k(\mu_k, \Sigma_k)(y)}\right] = \\
&= \min\left[1, \frac{\pi(dy)}{\pi(dx)} \frac{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(x)}{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(y)} \frac{a_i}{a_k}\right].
\end{aligned}$$

We perform the Local Moves with probability $1-\epsilon$ and the Jump Moves with probability ϵ. Hence, the algorithm may be treated as a mixture of transition kernels with the stationary distribution $\tilde{\pi}$. Therefore, by Proposition 3 in [Tierney, 1994] the Markov chain constructed as outlined above converges to $\tilde{\pi}$ in total variation.

Note that all the derivations above hold if $\mathcal{I}$ is any finite set of values in $\mathcal{X}$, not necessarily the modes. This is important, because in practice we will not normally know the exact locations of the local maxima, but only their approximations. The MCMC scheme proposed here is summarized by Algorithm 1.

Algorithm 1 MCMC for Multimodal Distributions (iteration $n+1$)

Input: a list of modes $\{\mu_1, \dots \mu_N\}$, a list of covariance matrices $\{\Sigma_1, \dots, \Sigma_N\}$, a vector of probabilities $a = (a_1, \dots, a_N)$, a starting point (x_n, i_n), a positive constant ϵ.
Generate $u_1 \sim U[0,1]$.
if $u_1 > \epsilon$ **then**
    Local Move:
    Propose a new value $y \sim Q_{i_n}(x_n, \Sigma_{i_n})$.
    Accept y with probability $\min\left[1, \frac{\pi(dy) Q_{i_n}(\mu_{i_n}, \Sigma_{i_n})(y)}{\pi(dx_n) Q_{i_n}(\mu_{i_n}, \Sigma_{i_n})(x_n)} \frac{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(x_n)}{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(y)}\right]$.
    if y accepted **then**
        $(x_{n+1}, i_{n+1}) = (y, i_n)$.
    else
        $(x_{n+1}, i_{n+1}) = (x_n, i_n)$.
    end if
else
    Jump Move:
    Propose a new mode $k \sim (a_1, \dots, a_N)$.
    Propose a new value $y \sim Q_k(\mu_k, \Sigma_k)$.
    Accept (y, k) with probability $\min\left[1, \frac{\pi(dy)}{\pi(dx_n)} \frac{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(x_n)}{\sum_{j \in \mathcal{I}} Q_j(\mu_j, \Sigma_j)(y)} \frac{a_{i_n}}{a_k}\right]$.
    if (y, k) accepted **then**
        $(x_{n+1}, i_{n+1}) = (y, k)$.
    else
        $(x_{n+1}, i_{n+1}) = (x_n, i_n)$.
    end if
end if
return A new sample (x_{n+1}, i_{n+1}).

The concept of this algorithm is shown in a schematic form in Figure 1. Note that such construction of the target distribution makes it unlikely for the algorithm to escape via local steps far away from the mode it is assigned to. Indeed, if a proposed point y is very distant from the current mode i, the acceptance probability becomes very small due to the expression $Q_i(\mu_i, \Sigma_i)(y)$ in the numerator in (3), which will typically be tiny in such case. This allows retaining control over 'from which mode a given state was drawn'.

While Algorithm 1 is proven to converge to the target distribution $\tilde{\pi}$ (and therefore, its marginal distribution with respect to x converges to π), it relies on an unrealistic assumption of knowing the covariance matrices $\Sigma_1, \dots, \Sigma_N$. Choosing suboptimal parameters of the algorithm, i.e. matrices $\Sigma_1, \dots, \Sigma_N$, may lead to very slow mixing. To circumvent this problem, we propose an adaptive version of the algorithm, summarised in Algorithm 2. It relies on incorporating the ideas from Section 2.1 into Algorithm 1. Let AC_1 and AC_2 be positive integers. As long as the number of samples in mode i is smaller than AC_1, we adapt Σ_i only in case of the Local Move within mode i, using Method 2 described in Section 2.1. What is more, following Method 1 in Section 2.1 in case of the Local Moves we do not propose y from $Q_{i_n}(x_n, \Sigma_{i_n})$, but from a mixture $y \sim (1-\beta) Q_{i_n}(x_n, \Sigma_{i_n}) + \beta N\left(x_n, (0.1)^2 I_d/d\right)$ for some $0 < \beta < 1$.

The reason why we do not adapt Σ_i in case of the Jump Moves to mode i is that in practice it is often difficult to jump into a given mode for the first time, so a sequence of consecutive rejections would lead to obtaining extremely small values of σ in (1). This would, in turn, distort heavily the

target distribution by making mode i very spiky and as a result, it would decrease significantly the acceptance probability of the Jump Move to mode i.

When the number of samples in the proposed mode i is larger than AC_1, it is assumed that the empirical covariance matrix approximates reasonably well the real covariance matrix around mode i, so we switch to using Method 1 from Section 2.1. We believe that it is more appropriate in this context than Method 2, since it makes use of the knowledge we have about the correlation structure. We update the matrices every AC_2 iterations.

Algorithm 2 Adaptive MCMC for Multimodal Distributions (iteration $n+1$)

Input: a list of modes $\{\mu_1, \dots \mu_N\}$, a list of covariance matrices $\{\Sigma_1, \dots, \Sigma_N\}$, a vector of probabilities $a = (a_1, \dots, a_N)$, a starting point (x_n, i_n), a constant ϵ such that $0 < \epsilon < 1$, a constant β such that $0 < \beta < 1$, empirical means and covariances of the samples from each mode, positive integers AC_1 and AC_2, a constant γ such that $0 < -\gamma \leq 1$, a constant for the optimal acceptance rate opt acc such that $0 <$ opt acc < 1, the dimension of the state space d.
Generate $u_1 \sim U[0,1]$.
if $u_1 > \epsilon$ **then**
    Local Move:
    Propose a new value $y \sim (1-\beta)Q_{i_n}(x_n, \Sigma_{i_n}) + \beta N\left(x_n, (0.1)^2 I_d/d\right)$.
    Accept y with probability min $\left[1, \frac{\pi(dy)Q_{i_n}(\mu_{i_n},\Sigma_{i_n})(y)}{\pi(dx_n)Q_{i_n}(\mu_{i_n},\Sigma_{i_n})(x_n)} \frac{\sum_{j\in\mathcal{I}} Q_j(\mu_j,\Sigma_j)(x_n)}{\sum_{j\in\mathcal{I}} Q_j(\mu_j,\Sigma_j)(y)}\right]$.
    if y accepted **then**
        $(x_{n+1}, i_{n+1}) = (y, i_n)$.
    else
        $(x_{n+1}, i_{n+1}) = (x_n, i_n)$.
    end if
else
    Jump Move:
    Propose a new mode $k \sim (a_1, \dots, a_N)$.
    Propose a new value $y \sim Q_k(\mu_k, \Sigma_k)$.
    Accept (y,k) with probability min $\left[1, \frac{\pi(dy)}{\pi(dx_n)} \frac{\sum_{j\in\mathcal{I}} Q_j(\mu_j,\Sigma_j)(x_n)}{\sum_{j\in\mathcal{I}} Q_j(\mu_j,\Sigma_j)(y)} \frac{a_{i_n}}{a_k}\right]$.
    if (y,k) accepted **then**
        $(x_{n+1}, i_{n+1}) = (y, k)$.
    else
        $(x_{n+1}, i_{n+1}) = (x_n, i_n)$.
    end if
end if
Update the empirical mean and covariance matrix in mode i_{n+1}.
if number of samples in mode $i_n < AC_1$ **then**
    if Local Move then
        $\Sigma_{i_n} = \exp\left((\text{number of samples in mode } i_n)^{\gamma}(\text{the last acceptance probability - opt acc})\right)\Sigma_{i_n}$
    end if
else
    if number of samples in mode i_{n+1} is divisible by AC_2 **then**
        $\Sigma_{i_{n+1}} = 2.38^2/d \cdot$ empirical covariance matrix in mode i_{n+1}.
    end if
end if
return A new sample (x_{n+1}, i_{n+1}).

The problem that the algorithm may struggle with jumping for the first time into a certain mode, has not been solved so far. It may be a challenging research direction to follow this lead and improve Algorithm 2 in this respect.

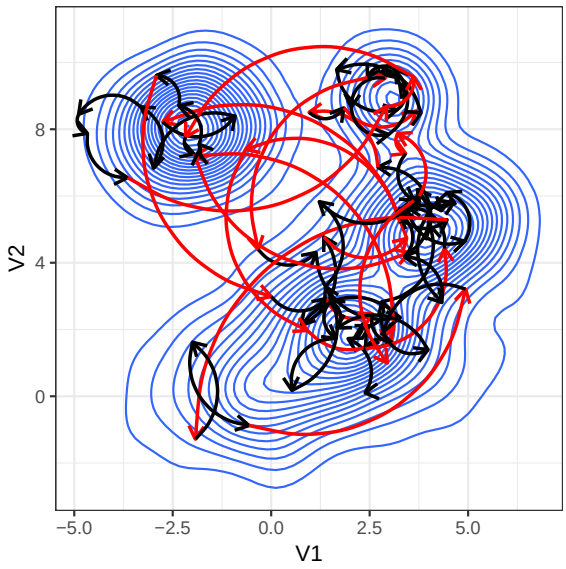


Figure 1: The blue lines represent contour lines of some two-dimensional target distributions π. Panel b) illustrates how Algorithm 1 works on a two-dimensional toy example. Red arrows represent Jump Moves and the black ones represent Local Moves.

3 Empirical results for a toy example

In this section we demonstrate the performance of Algorithm 2 on a five-dimensional toy example. Let the target distribution π be given by

$$0.2N(\mu_1, \Sigma_1) + 0.2N(\mu_2, \Sigma_2) + 0.2N(\mu_3, \Sigma_3) + 0.3N(\mu_4, \Sigma_4) + 0.1N(\mu_5, \Sigma_5),$$

where the locations of the modes μ_i for $i \in \{1, \ldots, 5\}$ are given in the rows of the table below.

1.27	0.52	-1.75	-0.59	-0.12
6.65	2.86	-2.61	3.21	0.50
9.13	-3.14	-9.29	8.45	4.53
-41.27	3.03	15.45	1.27	7.92
1.22	0.84	2.33	-0.17	-0.21

Furthermore, Σ_1 and Σ_2 are the 5×5 identity matrices, whereas Σ_3, Σ_4 and Σ_5 were generated from the Wishart distribution. To find the local maxima of the density function, we used an algorithm based on Simulated Annealing ([Kirkpatrick et al., 1983]), which returned the following approximations of the modes.

1.08	0.55	-1.57	-0.89	-0.18
6.43	3.05	-2.66	3.05	0.34
9.01	-2.87	-9.42	8.58	4.37
-41.31	3.00	15.49	1.17	7.92
1.72	1.02	2.63	-0.22	-0.17

Using these approximations of the modes, we performed 1000000 iterations of Algorithm 2 with $AC_1 = 2000$, $AC_2 = 500$, $\epsilon = 0.3$, $\beta = 0$ and $\alpha = -0.5$. The optimal acceptance rate was assumed to be equal to 0.234 and the vector a was equal to $(0.2, 0.2, 0.2, 0.2, 0.2)$. The initial matrices $\Sigma_1, \ldots, \Sigma_5$ were the 5×5 identity matrices.

Even though the numerical approximations of the modes are far from perfect, the MCMC sample seems to remain unaffected. Figure 2 shows that the Markov chain converged to the target distribution. Furthermore, both the traceplot and the autocorrelation provide an illustration of good mixing of the Markov chain. The traceplot reveals that mixing becomes faster after a certain number of iterations, which might be due to adapting the covariance matrices. Additionally, the percentage of the time spent by the Markov chain in each mode represents the vector of weights $(0.2, 0.2, 0.2, 0.3, 0.1)$ with almost absolute precision. To conclude, this toy example confirms that the results delivered by our method seem highly promising.

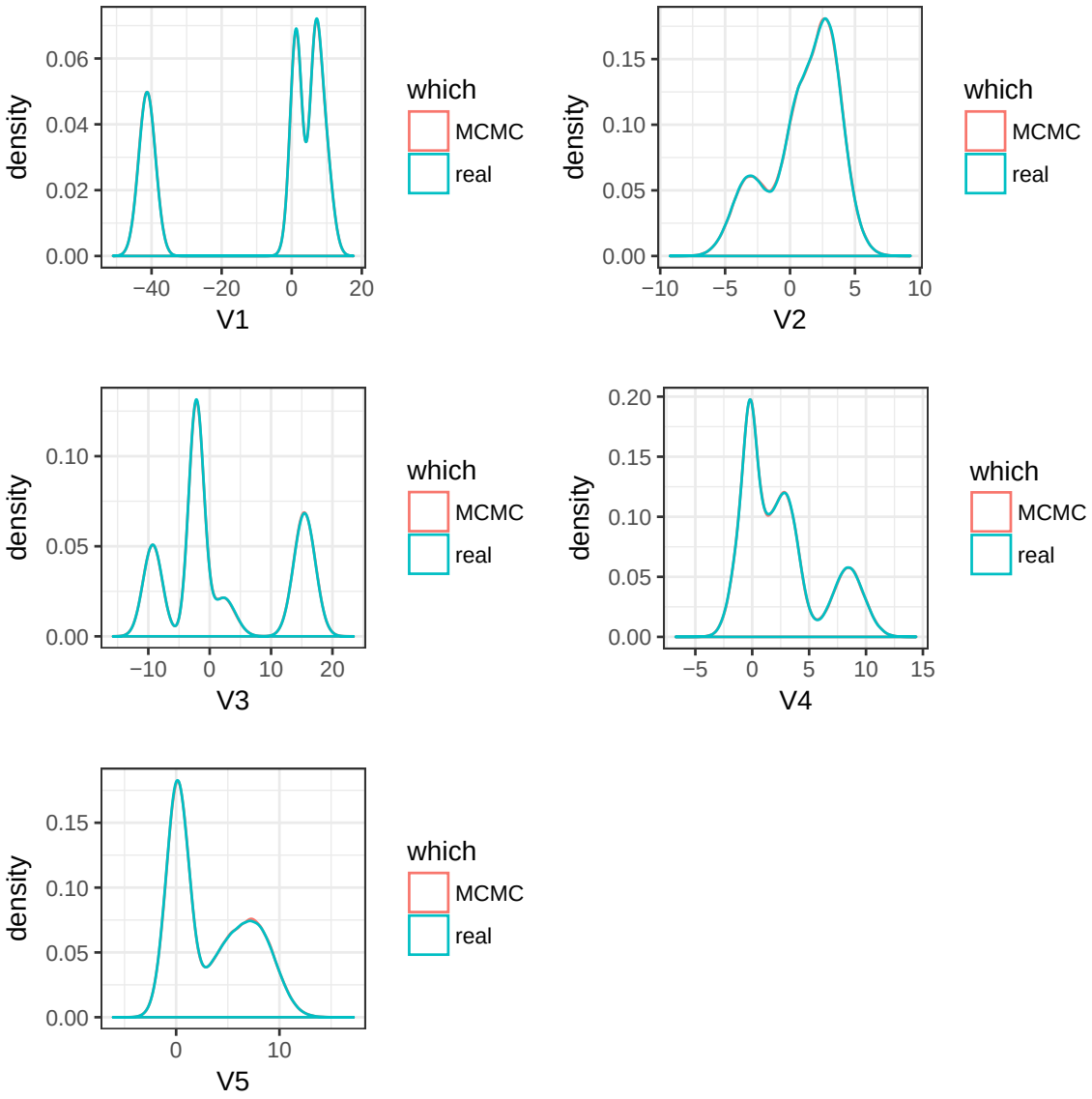
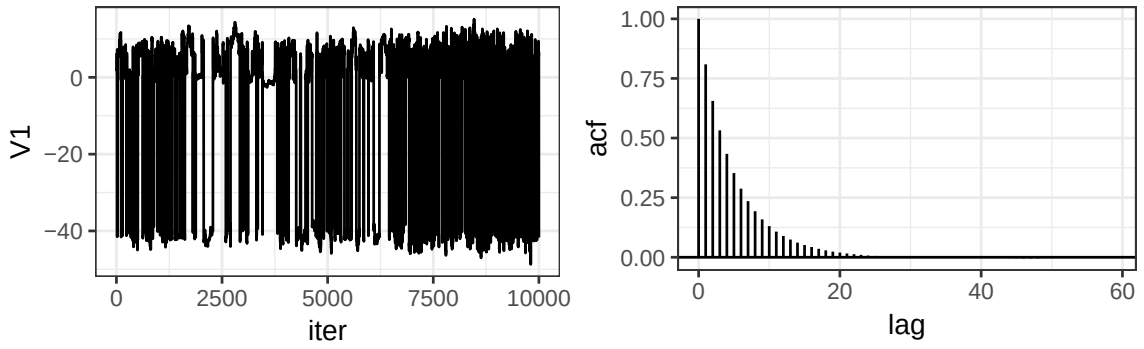


Figure 2: Histograms of each coordinate of the MCMC sample obtained via Algorithm 2; the target distribution was the five-dimensional mixture of Gaussians studied in 3. The histograms are based on 1000000 iterations with a 10% burn-in period. The MCMC samples look almost indistinguishable from the true distribution.

4 Discussion and future research directions

In this paper we show an alternative algorithm for MCMC sampling from multimodal distributions. The approach presented in Section 2 has several advantages.



(a) (b)

Figure 3: Panel a) shows a traceplot of 10000 first iterations of the algorithm for the first coordinate for the example studied in Section 3. Panel b) presents the autocorrelation function for the first coordinate.

Firstly, we define the target distribution $\tilde{\pi}$ in (2) on the augmented state space, which allows for controlling the mode at each iteration. Additionally, the construction of $\tilde{\pi}$ prevents the value x_n from escaping from its corresponding mode i_n. Consequently, the empirical estimates of the matrices $\Sigma_1, \ldots, \Sigma_N$ manage to capture well the correlation structure around a given mode and thanks to this, it is reasonable to include the idea of Method 1 in Section 2.1 in the algorithm. As a result, we obtain a fully adaptive algorithm that requires little intervention from the user.

However, there are several aspect of the method that need further investigation. We will address them in our full paper following the note. Firstly, Algorithm 2 is limited to the case when the locations of the modes are known. However, in practise the problem of finding the modes is often challenging itself, thus, we will propose an algorithm well-suited for the problem of finding local maxima of density functions.

Moreover, we will tackle the problem of 'not jumping into a given mode', mentioned in Section 2.2. Additionally, we will extend our algorithm, so that it takes into account the fact that Method 1 in Section 2.1 is known to perform well for distributions that are similar to normal, and other methods are more efficient when dealing with e.g. heavy-tailed distributions.

Furthermore, we will address the issue of the convergence properties of Algorithm 2. It is important to outline that since the target distributions $\tilde{\pi}$ keeps being modified as the algorithm runs, our method goes beyond the framework of Adaptive MCMC. However, we will provide a proof that the stochastic process resulting from Algorithm 2 converges to the desired distribution.

Even though the performance of the method on mixtures of Gaussians seems promising, it is necessary to test it on more challenging distributions. An example of an applied problem, on which our algorithm could be used, is the variable selection for Bayesian linear regression, outlined in Section 1. Finally, the performance of our technique should be compared with other MCMC methods designed for multimodal distributions.

References

[Aitkin, 2001] Aitkin, M. (2001). Likelihood and Bayesian analysis of mixtures. *Statistical Modelling*, 1(4):287–304.

[Duane et al., 1987] Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid Monte Carlo. *Physics letters B*, 195(2):216–222.

[Haario et al., 2001] Haario, H., Saksman, E., and Tamminen, J. (2001). An adaptive Metropolis algorithm. *Bernoulli*, pages 223–242.

[Jasra et al., 2005] Jasra, A., Holmes, C. C., and Stephens, D. A. (2005). Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statistical Science*, pages 50–67.

[Kirkpatrick et al., 1983] Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., et al. (1983). Optimization by simulated annealing. *science*, 220(4598):671–680.

[Lee et al., 2010] Lee, A., Caron, F., Doucet, A., and Holmes, C. (2010). A hierarchical Bayesian framework for constructing sparsity-inducing priors. *arXiv preprint arXiv:1009.1914*.

[Roberts et al., 1997] Roberts, G. O., Gelman, A., Gilks, W. R., et al. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *The annals of applied probability*, 7(1):110–120.

[Roberts and Rosenthal, 2007] Roberts, G. O. and Rosenthal, J. S. (2007). Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms. *Journal of applied probability*, 44(2):458–475.

[Roberts and Rosenthal, 2009] Roberts, G. O. and Rosenthal, J. S. (2009). Examples of adaptive MCMC. *Journal of Computational and Graphical Statistics*, 18(2):349–367.

[Roberts et al., 2004] Roberts, G. O., Rosenthal, J. S., et al. (2004). General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1:20–71.

[Tierney, 1994] Tierney, L. (1994). Markov chains for exploring posterior distributions. *the Annals of Statistics*, pages 1701–1728.