

ADVANCES IN MARKOV CHAIN MONTE CARLO

Jim E. Griffin

David A. Stephens

7.1 Introduction

7.1.1 *Markov chain Monte Carlo: 1985-1995*

The early work on Markov chain Monte Carlo (MCMC) throughout the 1980s, culminating with the publication of [5], led to the widespread use of Bayesian statistical inference in a broad range of application fields. The first half of the next decade saw many specific algorithmic developments, including the extension of the simple Gibbs sampling approaches of Gelfand and Smith to the use of the more general Metropolis-Hastings (MH) algorithm, and hybrid (mixture kernel) approaches in applications (see [31] for a summary of key theoretical concepts). There was also a huge growth in applications of Bayesian inference to challenging statistical problems that had previously proved intractable due to limitations of numerical approaches to integration.

By the mid 1990s, much was known about how MCMC should be implemented for optimal performance in standard statistical problems; see, for example, [11] for a summary of the state of knowledge at that time. Interest turned to addressing more complicated problems, such as model selection, and mixture problems, and a more in depth theoretical study of classical algorithms such as the Metropolis algorithm. Model selection remains to this day an incompletely resolved issue in Bayesian inference, but in the mid 1990s specific focus fell on problems where a sequence of models was indexed by a discrete random variable, where models corresponding to different indices had different dimensions, mixture models with different numbers of components being a canonical example. Research into the Metropolis algorithm led to one of the most familiar results of MCMC folklore, The Goldilocks Principle, which relates to the optimal choice of the scale of Metropolis proposals ([29]). This result, that suggests an optimal Metropolis acceptance rate of 0.234, continues to inform MCMC practitioners today, alerting them to the fact that they should focus on tuning their algorithm so that the acceptance rate is neither too high nor too low. It appeals to the logic that the MCMC kernels might be chosen in an adaptive fashion, in light of the observed performance of the algorithm. However, the result of [29] does not strictly relate to the construction of a truly adaptive algorithm.

In this chapter, we trace some of the key developments that further developed the underpinning theory, and potential applications, of

MCMC. In particular, we review reversible jump or transdimensional MCMC, population MCMC methods, and adaptive MCMC.

7.1.2 Notation

Following the notation introduced in the previous chapter, we initially consider the target distribution π^* over a parameter vector $\theta \in \Theta \subseteq \mathcal{R}^d$ derived for observed data $\mathbf{y}$. Denote generic Markov transition kernel $Q(\theta, d\theta'|\mathbf{y}) = \Pr(\theta \in d\theta'|\mathbf{y}, \theta)$, and modified transition kernel $P(\theta, d\theta'|\mathbf{y})$ which has π^* as its unique stationary distribution. Let π and q denote the densities of π^* and Q wrt to Lebesgue measure. In the MH algorithm let $P(\theta, d\theta'|\mathbf{y})$ denote the reversible Markov transition kernel derived from $Q(\theta, d\theta'|\mathbf{y})$ with π^* as the stationary distribution. By the usual arguments, the acceptance probability for values generated from q is

$$\alpha(\theta, \theta'|\mathbf{y}) = \min \left\{ 1, \frac{\pi(\theta'|\mathbf{y}) q(\theta', \theta|\mathbf{y})}{\pi(\theta|\mathbf{y}) q(\theta, \theta'|\mathbf{y})} \right\}$$

Under mild conditions on $Q(\cdot, \cdot|\mathbf{y})$, variates generated from this Markov chain with transition kernel $P(\cdot, \cdot|\mathbf{y})$ form a dependent sample from π , and the ergodicity of the chain permits Monte Carlo estimation of functionals of π .

7.2 Reversible Jump MCMC

After the foundational work on MCMC theory and applications, the next principal advance in MCMC theory was achieved by [14]. Around this time, interest in model selection via MCMC, and Markov chain methods operating on more complex parameter spaces, had become a major research field, and several authors had made significant contributions in this direction. In particular, the work of [15], [3] and [27] are notable. In [15] a particular continuous time Markov process, a Langevin jump diffusion, is constructed to sample the posterior distribution in a complex geometric and image analysis setting; a similar approach is used in [27], again in an image analysis setting. The use of a continuous time (birth-death) Markov process was, in itself, a novel contribution to Bayesian computation, although apart from a few specific areas of application (see for example [9, 30]), continuous time algorithms have remained under-utilized. In [3] a different approach is adopted; a more standard discrete time MCMC algorithm is constructed on an extended state space that encompasses all models considered simultaneously, with all parameters in all models being updated at each iteration, with model selection achieved during the MCMC run by sampling a discrete model index.

In the contributed Discussion of [15], Green laid out the basic construction that later formed the content of his paper [14]. He pointed out

that whereas the Gibbs sampler construction utilized by [15] might only be rarely of use, the Metropolis-Hastings algorithm they also suggested could provide a general mechanism for jumping between subspaces of different dimension. The suggested general algorithm retained the reversibility and detailed balance properties of the usual MH algorithm, but required certain “dimension-matching” terms to account for the differences in dimension of different subspaces.

We now study the reversible jump MCMC algorithm of [14]. Note that a formulation of MH algorithms on general state spaces was provided by [32], which includes Green’s formulation as a special case.

7.2.1 Reversible Jump MCMC: Formulation

We consider a countable collection of Bayesian models, $\{M_k, k = 1, 2, \dots\}$, where model M_k is parameterized via parameter θ_k with parameter space $\Theta_k \subset \mathbb{R}^{d_k}$. We consider both the cases of a finite collection, where we present a series of possibly non-nested models and search for the most appropriate explanation of the data, and also cases of an infinite collection; the latter situation might arise if the index k corresponds to a non-negative integer-valued parameter. For data $\mathbf{y}$, the full posterior can be written

$$\pi(M_k, \theta_k | \mathbf{y}) = \frac{f(\mathbf{y} | \theta_k, M_k) p(\theta_k | M_k) p(M_k)}{\sum_j \left\{ \int f(\mathbf{y} | \theta_j, M_j) p(\theta_j | M_j) d\theta_j \right\} p(M_j)} \quad (7.1)$$

Our objective is to construct a Metropolis-Hastings Markov chain to that is aperiodic and irreducible on the union parameter space $\Theta = \cup_k \Theta_k$. As usual, this requires the specification of a proposal transition density, $q(\cdot, \cdot | \mathbf{y})$, but, in contrast to the usual fixed-dimension case, we face the difficulty that the arguments of q are potentially of different dimensions, rendering the reversibility requirement difficult to meet. We now discuss the usual reversible jump solution.

At a specific iteration, suppose that the chain is in model M with parameter value θ having dimension d , and the proposal is to move to model M' with parameter value θ' having dimension d' . We envisage this move as first selecting a move between models, $M \rightarrow M'$, and then the proposal of a θ' possibly dependent on the current θ . Suppose that, when in model M , the move between the two models is selected with probability $r(M, M')$. To retain the possibility of a reversible proposal mechanism, we require equally dimensioned arguments to $q(\cdot, \cdot | \mathbf{y})$, so consider the introduction of collections of latent variables $\mathbf{u}$ and $\mathbf{v}$ of dimension d_u and d_v respectively, so that $d + d_u = d' + d_v$. The proposal density q is then considered for the extended parameter vectors $(\theta, \mathbf{u})$ and $(\theta', \mathbf{v})$ such that $q((\theta, \mathbf{u}), (\theta', \mathbf{v}) | \mathbf{y})$ is reversible; this is most easily constructed using a bijective differentiable mapping, that is

$$(\boldsymbol{\theta}', \mathbf{v}) = \mathbf{g}(\boldsymbol{\theta}, \mathbf{u}) \quad \Longleftrightarrow \quad (\boldsymbol{\theta}, \mathbf{u}) = \mathbf{h}(\boldsymbol{\theta}', \mathbf{v}).$$

Note that standard fixed-dimension moves also fall under this general proposal procedure; for example, for the ordinary Metropolis move, we might set

$$\boldsymbol{\theta}' = \boldsymbol{\theta} + \mathbf{u} \quad \mathbf{u} \sim N(0, \Sigma)$$

and for Metropolis-Hastings moves, we typically use a conditional generation $q(\boldsymbol{\theta}'|\boldsymbol{\theta}, \mathbf{y})$ taking the conditioning variable as a constant parameter in a suitably chosen density; the stochastic elements $\mathbf{u}$ in the MH move can be thought of as the *Uniform*(0, 1) variates used to perform basic random number generation from this conditional density.

To establish the acceptance probability for the reversible jump move, Green considers a “hybrid” MH algorithm comprising a mixture of move types, with move indexed by m with transition proposal Q_m selected with probability r_m , some of which may be trans-dimensional, but each of which retains the detailed balance property. If ψ and ψ' represent the latent-augmented parameter vectors, and let the augmented posterior density be denoted

$$\tilde{\pi}_m(\psi|\mathbf{y}) = \tilde{\pi}_m(M, \boldsymbol{\theta}, \mathbf{u}|\mathbf{y}) = \pi(M, \boldsymbol{\theta}|\mathbf{y})p_u(\mathbf{u})$$

Let $\alpha_m(\psi, \psi'|\mathbf{y})$ denote the acceptance probability for move type m , so that for arbitrary sets A and B

$$\int_A \tilde{\pi}_m^*(d\psi|\mathbf{y}) \int_B \alpha_m(\psi, \psi'|\mathbf{y}) Q_m(\psi, d\psi'|\mathbf{y}) = \int_B \tilde{\pi}_m^*(d\psi'|\mathbf{y}) \int_A \alpha_m(\psi', \psi|\mathbf{y}) Q_m(\psi', d\psi|\mathbf{y})$$

which implies as usual that

$$\alpha_m(\psi, \psi'|\mathbf{y}) \tilde{\pi}_m(\psi|\mathbf{y}) q_m(\psi, \psi'|\mathbf{y}) = \alpha_m(\psi', \psi|\mathbf{y}) \tilde{\pi}_m(\psi'|\mathbf{y}) q_m(\psi', \psi|\mathbf{y})$$

or, in Green’s notation

$$\alpha_m(\psi, \psi'|\mathbf{y}) f_m(\psi, \psi'|\mathbf{y}) = \alpha_m(\psi', \psi|\mathbf{y}) f_m(\psi', \psi|\mathbf{y})$$

where f_m is defined with respect to a common, symmetric measure on the product space. Consider first the forward move; in this case $f_m(\psi, \psi'|\mathbf{y})$ is given by

$$f_m(\psi, \psi'|\mathbf{y}) = \tilde{\pi}_m(\psi|\mathbf{y}) q_m(\psi, \psi'|\mathbf{y}) = \pi(M, \boldsymbol{\theta}|\mathbf{y}) p_u(\mathbf{u}) \vec{r}_m$$

say. For the reverse move, to preserve symmetry, we must set

$$f_m(\psi', \psi|\mathbf{y}) = \tilde{\pi}_m(\psi'|\mathbf{y}) q_m(\psi', \psi|\mathbf{y}) = \pi(M', \boldsymbol{\theta}'|\mathbf{y}) p_v(\mathbf{v}) \left| \frac{\partial(\boldsymbol{\theta}', \mathbf{v})}{\partial(\boldsymbol{\theta}, \mathbf{u})} \right| \overleftarrow{r}_m$$

where the term

$$\left| \frac{\partial(\boldsymbol{\theta}', \mathbf{v})}{\partial(\boldsymbol{\theta}, \mathbf{u})} \right| = \left| \frac{\partial \mathbf{g}(\mathbf{t}_1, \mathbf{t}_2)}{\partial(\boldsymbol{\theta}, \mathbf{u})} \right|_{\mathbf{t}_1=\boldsymbol{\theta}, \mathbf{t}_2=\mathbf{u}}$$

is the Jacobian associated with the bijection $\mathbf{g} : (\boldsymbol{\theta}, \mathbf{u}) \mapsto (\boldsymbol{\theta}', \mathbf{v})$. This term arises as in the augmented posterior, we have

$$\tilde{\pi}(M', \boldsymbol{\theta}', \mathbf{v} | \mathbf{y}) = \tilde{\pi}(M, \mathbf{h}(\boldsymbol{\theta}', \mathbf{v}) | \mathbf{y}) |J(\boldsymbol{\theta}', \mathbf{v})| = \tilde{\pi}(M, \boldsymbol{\theta}, \mathbf{u} | \mathbf{y}) |J(\boldsymbol{\theta}, \mathbf{u})|^{-1}$$

under the bijection; here

$$|J(\boldsymbol{\theta}', \mathbf{v})| = \left| \frac{\partial(\boldsymbol{\theta}, \mathbf{u})}{\partial(\boldsymbol{\theta}', \mathbf{v})} \right| = \left| \frac{\partial(\boldsymbol{\theta}', \mathbf{v})}{\partial(\boldsymbol{\theta}, \mathbf{u})} \right|^{-1} = |J(\boldsymbol{\theta}, \mathbf{u})|^{-1}$$

is the Jacobian. In the expressions above, the two terms $\overrightarrow{r}_m$ and $\overleftarrow{r}_m$ represent the probabilities of choosing to make move m (from M to M') and the probability of the reverse move (from M' to M). Thus, the acceptance probability for move type m is

$$\alpha_m((M, \boldsymbol{\theta}), (M', \boldsymbol{\theta}') | \mathbf{y}) = \min \left\{ 1, \frac{\pi(M', \boldsymbol{\theta}' | \mathbf{y}) p_v(\mathbf{v}) \overleftarrow{r}_m}{\pi(M, \boldsymbol{\theta} | \mathbf{y}) p_u(\mathbf{u}) \overrightarrow{r}_m} \left| \frac{\partial(\boldsymbol{\theta}', \mathbf{v})}{\partial(\boldsymbol{\theta}, \mathbf{u})} \right| \right\}.$$

Often, these general calculations simplify as one of $\mathbf{u}$ or $\mathbf{v}$ is a null vector. If $\dim(\boldsymbol{\theta}) < \dim(\boldsymbol{\theta}')$, and the proposed move attempts to increase the dimension of the current model, then only the augmenting variables $\mathbf{u}$ are needed to match dimension, that is, the bijection can be constructed by setting $\boldsymbol{\theta}' = \mathbf{g}(\boldsymbol{\theta}, \mathbf{v})$. In this case

$$\alpha_m((M, \boldsymbol{\theta}), (M', \boldsymbol{\theta}') | \mathbf{y}) = \min \left\{ 1, \frac{\pi(M', \boldsymbol{\theta}' | \mathbf{y}) \overleftarrow{r}_m}{\pi(M, \boldsymbol{\theta} | \mathbf{y}) p_u(\mathbf{u}) \overrightarrow{r}_m} \left| \frac{\partial(\boldsymbol{\theta}')}{\partial(\boldsymbol{\theta}, \mathbf{u})} \right| \right\}.$$

Conversely, if $\dim(\boldsymbol{\theta}) > \dim(\boldsymbol{\theta}')$, then only the augmenting variables $\mathbf{v}$ are needed, and

$$\alpha_m((M, \boldsymbol{\theta}), (M', \boldsymbol{\theta}') | \mathbf{y}) = \min \left\{ 1, \frac{\pi(M', \boldsymbol{\theta}' | \mathbf{y}) p_v(\mathbf{v}) \overleftarrow{r}_m}{\pi(M, \boldsymbol{\theta} | \mathbf{y}) \overrightarrow{r}_m} \left| \frac{\partial(\boldsymbol{\theta}', \mathbf{v})}{\partial(\boldsymbol{\theta})} \right| \right\}.$$

As pointed out by [14, p. 717], these proposal mechanisms can be generalized by allowing the generation for $\mathbf{u}$ or $\mathbf{v}$ to depend on the values of $\boldsymbol{\theta}$ or $\boldsymbol{\theta}'$ respectively. Under this generalization, $p_u(\mathbf{u})$ and $p_v(\mathbf{v})$ are replaced by $p_u(\mathbf{u} | \boldsymbol{\theta})$ and $p_v(\mathbf{v} | \boldsymbol{\theta}')$ in the acceptance probabilities. This merely corresponds to an alternative construction of the augmented posterior $\tilde{\pi}$.

It is evident from the construction that each move type m comprises a pair of moves operating in each direction between models M and M' . Thus, if move m is a move from M to M' , constructed by generation of augmenting variables $\mathbf{u}$ and transformation, there should exist in our

collection of potential moves the reverse move, indexed m' , say, which utilizes the augmenting variables $\mathbf{v}$, the two moves being selected with probabilities $\overrightarrow{r}_m$ and $\overleftarrow{r}_m$.

In some situations, the reversible jump acceptance probability simplifies even further. For example, in a move that increases dimension by d_u through the variables $\mathbf{u}$, it may be feasible to set the new parameters precisely equal to $\mathbf{u}$, that is, the proposed parameter vector $\boldsymbol{\theta}'$ is formed by concatenating $\boldsymbol{\theta}$ and $\mathbf{u}$. In this case, the Jacobian of the transformation is 1. Furthermore, it may be possible to generate $\mathbf{u}$ from a prior distribution, which facilitates cancellation in the Hastings ratio.

Example 7.1 Let $k = 1, 2, \dots$ index models $M_1, M_2, \dots$ under consideration, where k represents the dimension of $\boldsymbol{\theta}_k$. Let the elements of $\boldsymbol{\theta}_k$ be $(\theta_{k1}, \dots, \theta_{kk})$. Suppose that the model specification is such that identical, independent priors are used for the components of the parameter vector

$$p(\boldsymbol{\theta}_k) = \prod_{l=1}^k p_0(\theta_{kl}).$$

For a proposed move from M_j to M_k with $j < k$, suppose that the elements of $\mathbf{u}$ are generated independently from p_0 , so that

$$p_u(\mathbf{u}) = \prod_{l=1}^{k-j} p_0(u_l),$$

and suppose that $\boldsymbol{\theta}_k = (\boldsymbol{\theta}_j, \mathbf{u})$. Then

$$\begin{aligned} \alpha_j((M_j, \boldsymbol{\theta}_j), (M_k, \boldsymbol{\theta}_k) | \mathbf{y}) &= \min \left\{ 1, \frac{\pi(M_k, \boldsymbol{\theta}_k | \mathbf{y}) \overleftarrow{r}_j}{\pi(M_j, \boldsymbol{\theta}_j | \mathbf{y}) p_u(\mathbf{u}) \overrightarrow{r}_j} \left| \frac{\partial(\boldsymbol{\theta}_k)}{\partial(\boldsymbol{\theta}_j, \mathbf{u})} \right| \right\} \\ &= \min \left\{ 1, \frac{f(\mathbf{y} | M_k, \boldsymbol{\theta}_k) p(\boldsymbol{\theta}_k | M_k) p(M_k) \overleftarrow{r}_j}{f(\mathbf{y} | M_j, \boldsymbol{\theta}_j) p(\boldsymbol{\theta}_j | M_j) p(M_j) p_u(\mathbf{u}) \overrightarrow{r}_j} \left| \frac{\partial(\boldsymbol{\theta}_k)}{\partial(\boldsymbol{\theta}_j, \mathbf{u})} \right| \right\} \\ &= \min \left\{ 1, \frac{f(\mathbf{y} | M_k, \boldsymbol{\theta}_k) p(M_k) \overleftarrow{r}_j}{f(\mathbf{y} | M_j, \boldsymbol{\theta}_j) p(M_j) \overrightarrow{r}_j} \right\} \end{aligned}$$

as

$$p(\boldsymbol{\theta}_k | M_k) = \prod_{l=1}^k p_0(\theta_{kl}) = \left\{ \prod_{l=1}^j p_0(\theta_{jl}) \right\} \left\{ \prod_{l=1}^{k-j} p_0(u_l) \right\} = p(\boldsymbol{\theta}_j | M_j) p_u(\mathbf{u})$$

For the reverse move that attempts to decrease the model complexity by moving from model k to model j , the deterministic proposal that sets the last $k - j$ components of $\boldsymbol{\theta}_k$ to zero is used.

The approach described in this example can be useful in some applications, but can also lead to low acceptance rates.

7.2.2 Reversible Jump MCMC: Examples

Example 7.2. (Moves between 1-d and 2-d models) Consider two models M_1 and M_2 with parameters $\theta^{(1)} = \theta_1$ and $\theta^{(2)} = (\theta_{21}, \theta_{22})$, with all parameters taking values on $\mathbb{R}$. We consider four move types:

1. $m = 1$: move **within** Model M_1 ,
2. $m = 2$: move **within** Model M_2 ,
3. $m = 3$: move **from** Model M_1 **to** Model M_2 ,
4. $m = 4$: move **from** Model M_2 **to** Model M_1 .

For the within model moves, standard MH acceptance calculations proceed as usual. Moves 3 and 4 are a forward/reverse move pair. Clearly, if the current state of the chain is in M_1 , only moves 1 or 3 can be selected, and similarly for M_2 . We need only consider the relative magnitudes of the reversible jump moves selection probabilities; for example, if the probability of selecting move 3 is 0.3 and the probability of selecting move 4 is 0.1, then

$$\frac{\overleftarrow{r}_3}{\overrightarrow{r}_3} = \frac{0.1}{0.3} \quad \quad \frac{\overleftarrow{r}_4}{\overrightarrow{r}_4} = \frac{0.3}{0.1}$$

are the ratios that enter into the acceptance probability calculations for move types 3 and 4 respectively.

When the current state of the chain is in M_1 , to propose a move to M_2 after selecting move 3, we must introduce a single random variate u ; suppose that $u \sim \text{Normal}(0, 1)$, and let

$$\theta_{21} = \theta_1 + u \quad \theta_{22} = \theta_1 - u \quad \Longleftrightarrow \quad \theta_1 = \frac{\theta_{21} + \theta_{22}}{2}.$$

In this case the acceptance probability takes the form

$$\alpha_3((M_1, \theta_1), (M_2, (\theta_{21}, \theta_{22})) | \mathbf{y}) = \min \left\{ 1, \frac{\pi(M_2, (\theta_{21}, \theta_{22}) | \mathbf{y})}{\pi(M_1, \theta_1 | \mathbf{y}) \phi(u)} \frac{\overleftarrow{r}_3}{\overrightarrow{r}_3} \left| \frac{\partial(\theta_{21}, \theta_{22})}{\partial(\theta_1, u)} \right| \right\}.$$

where $\phi(\cdot)$ is the standard normal pdf, and

$$\left| \frac{\partial(\theta_{21}, \theta_{22})}{\partial(\theta_1, u)} \right| = \left| \begin{array}{cc} \frac{\partial \theta_{21}}{\partial \theta_1} & \frac{\partial \theta_{21}}{\partial u} \\ \frac{\partial \theta_{22}}{\partial \theta_1} & \frac{\partial \theta_{22}}{\partial u} \end{array} \right| = \left| \begin{array}{cc} 1 & 1 \\ 1 & -1 \end{array} \right| = 2.$$

For the reverse move 4, we have

$$\alpha_4((M_2, (\theta_{21}, \theta_{22})), (M_1, \theta_1) | \mathbf{y}) = \min \left\{ 1, \frac{\pi(M_1, \theta_1 | \mathbf{y}) \phi(u)}{\pi(M_2, (\theta_{21}, \theta_{22}) | \mathbf{y})} \frac{\overleftarrow{r}_4}{\overrightarrow{r}_4} \left| \frac{\partial(\theta_1, u)}{\partial(\theta_{21}, \theta_{22})} \right| \right\}.$$

where

$$\left| \frac{\partial(\theta_1, u)}{\partial(\theta_{21}, \theta_{22})} \right| = \frac{1}{2}.$$

Example 7.3. (Comparing non-linear regression models) Consider two competing pharmacokinetic (PK) models for scalar response data $y(t)$ measured at different time points $t_1, \dots, t_n$:

1. **Model M_1 :** One compartment, elimination only;

$$E[Y(t)] = A_1 \exp\{-\lambda_1 t\} \quad t \geq 0$$

2. **Model M_2 :** One compartment, absorption and elimination;

$$E[Y(t)] = A_2 (\exp\{-\lambda_{21}t\} - \exp\{-(\lambda_{21} + \lambda_{22})t\}) \quad t \geq 0$$

where (A_1, λ_1) and $(A_2, \lambda_{21}, \lambda_{22})$ are positive parameters. The data are displayed in Figure 7.1.

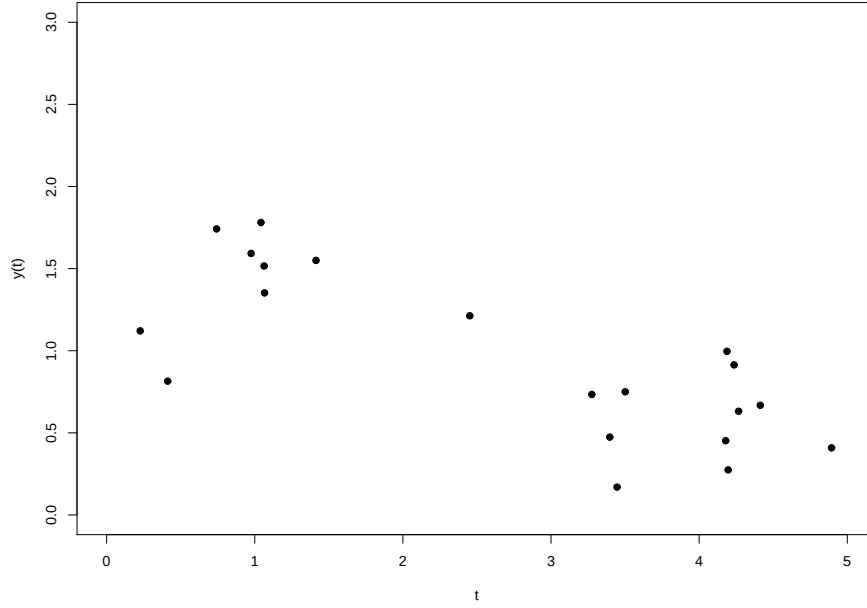


FIG. 7.1. PK data from one compartment model.

Under an assumption of additive, heteroscedastic Normal errors, we have two competing explanations for the observed data; both models can be fitted using ordinary least-squares, but model comparison is not straightforward as the models are nested but the nesting structure is complicated, as we require $\lambda_{22} \rightarrow \infty$, that is, a boundary point in the parameter space, which leads to non-regular frequentist asymptotic theory. Bayesian model comparison can be carried out using Bayes factors, but this requires numerical integration.

For illustration, we consider a reversible jump MCMC solution. First, we use a log-scale parameterization and set

$$\boldsymbol{\theta}_1 = (\log A_1, \log \lambda_1) \quad \boldsymbol{\theta}_2 = (\log A_2, \log \lambda_{21}, \log \lambda_{22}).$$

We place equal prior probabilities on M_1 and M_2 , and then place independent $N(0, \tau)$ priors on the components of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$. The prior on residual error variance σ^2 is inverse Gamma with parameters 10 and 4. The ML estimates $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$ can be computed easily, as can the Hessian matrices $\hat{\mathbf{I}}_1$ and $\hat{\mathbf{I}}_2$; these likelihood-based results yield reasonable approximations to the posterior distributions that can be used to produce independence MH algorithms. Specifically, at the ML estimates for σ under the two models, we may approximate the conditional posterior for $\boldsymbol{\theta}$ by the Normal density

$$p(\boldsymbol{\theta}_k | \hat{\sigma}, \mathbf{y}) \simeq N(\hat{\boldsymbol{\theta}}_k, n^{-1} \hat{\sigma}_k^2 \hat{\mathbf{I}}_k^{-1}) \quad (7.2)$$

On fitting using ML, the estimates of σ under the two models are found to be quite similar ($M_1 : \hat{\sigma}_1 = 0.252$, $M_2 : \hat{\sigma}_2 = 0.329$).

A reversible jump MCMC algorithm can be constructed as follows: we again consider four move types:

1. $m = 1$: move **within** M_1 ; update $\boldsymbol{\theta}_1$ from $p(\boldsymbol{\theta}_1 | M_1, \sigma, \mathbf{y})$
2. $m = 2$: move **within** M_1 ; update $\boldsymbol{\theta}_2$ from $p(\boldsymbol{\theta}_2 | M_2, \sigma, \mathbf{y})$
3. $m = 3$: move **from** M_1 **to** M_2 ; propose a new $\boldsymbol{\theta}_2$, and carry out an accept/reject step.
4. $m = 4$: move **from** Model M_2 **to** Model M_1 ; propose a new $\boldsymbol{\theta}_1$, and carry out an accept/reject step.

with the remaining parameter σ^2 being updated in a Gibbs sampler algorithm at each iteration. Moves $m = 3, 4$ are a forward/reverse move pair. For move 3, several options are available; for example, we could adopt the strategy of section 7.1, and generate a new variate u from the prior for the additional parameter, and then merely use the mapping

$$(\theta_{11}, \theta_{12}, u) \mapsto (\theta_{21} = \theta_{11}, \theta_{22} = \theta_{12}, \theta_{23} = u)$$

with reverse move setting $\theta_{23} = 0$. This approach may be adequate, but more probably would not facilitate good mixing across the models. A perhaps better strategy is to consider a different augmentation, where we generate $\mathbf{u} = (u_1, u_2, u_3)$ from the model in (7.2) for $k = 2$, and use

$$(\theta_{11}, \theta_{12}, u_1, u_2, u_3) \mapsto (\theta_{21} = u_1, \theta_{22} = u_2, \theta_{23} = u_3, v_1 = \theta_{11}, v_2 = \theta_{12})$$

with the paired reverse move being to generate $\mathbf{v} = (v_1, v_2)$ from the model in (7.2) for $k = 1$. This guarantees that the proposed value $\boldsymbol{\theta}_2$ lies in a region with reasonably high posterior support under model M_2 ,

although it does not guarantee that the move will be accepted with high probability. In the Hastings ratio, the Jacobian of the transformation is 1, and under equal probabilities of forward/reverse moves, we have

$$\frac{\pi(M_2, \boldsymbol{\theta}_2 | \mathbf{y}) p_v(v_1, v_2)}{\pi(M_1, \boldsymbol{\theta}_1 | \mathbf{y}) p_u(u_1, u_2, u_3)} = \frac{f(\mathbf{y} | M_2, \boldsymbol{\theta}_2, \sigma) \left\{ \prod_{j=1}^3 \phi(\theta_{2j}/\tau)/\tau \right\} \phi_2(\theta_{11}, \theta_{12}; \hat{\boldsymbol{\theta}}_1, \hat{\mathbf{I}}_1)}{f(\mathbf{y} | M_1, \boldsymbol{\theta}_1, \sigma) \left\{ \prod_{j=1}^2 \phi(\theta_{1j}/\tau)/\tau \right\} \phi_3(\theta_{21}, \theta_{22}, \theta_{23}; \hat{\boldsymbol{\theta}}_2, \hat{\mathbf{I}}_2)}$$

where τ is the prior variance for the regression parameters. The logic of this construction is that numerically

$$f(\mathbf{y} | M_1, \boldsymbol{\theta}_1, \sigma) \approx \phi_2(\theta_{11}, \theta_{12}; \hat{\boldsymbol{\theta}}_1, \hat{\mathbf{I}}_1) \quad f(\mathbf{y} | M_2, \boldsymbol{\theta}_2, \sigma) \approx \phi_3(\theta_{21}, \theta_{22}, \theta_{23}; \hat{\boldsymbol{\theta}}_2, \hat{\mathbf{I}}_2).$$

A different approximation that incorporates the Normal asymptotic likelihood in equation (7.2) as well as the Normal prior distribution for the parameters can be constructed.

The algorithm was run for 100000 iterations. In this run with $\tau = 4$, the chain spent about 66 % of the time in model M_1 , indicating the posterior probabilities are

$$p(M_1 | \mathbf{y}) \approx 0.66 \quad p(M_2 | \mathbf{y}) \approx 0.34.$$

The model posterior probabilities vary with the choice of τ ; this is as expected, as the model probabilities are closely related to the marginal likelihood, or prior predictive distribution, which is the expected value of the likelihood for the observed data with respect to the prior distribution. It is evident from the discussion that the prior specification acts as a penalty for complexity. For illustration, if $\tau = 1$, the model probabilities change to (0.80, 0.20); if $\tau = 10$, the model probabilities are (0.26, 0.74).

Conditional on M_1 or M_2 being true, we can perform inference about the parameters of the two models, and also reconstruct estimates and posterior credible intervals for $E[Y]$. Figure 7.2 displays the reconstructed posterior intervals for the two models.

It is evident that both models are plausible explanations of the observed data, but offer potentially different predictions of future responses, especially near $t = 0$. Note that the possible nesting structure, incorporated by the assumption that $\lambda_{22} \rightarrow \infty$, is not supported by the priors studied. Finally, note that the BIC values for the two models, computed as

$$\text{BIC}_k = -2 \log f(\mathbf{y} | M_k, \hat{\boldsymbol{\theta}}_k, \hat{\sigma}) + d_k \log n$$

where d_k is the total number of parameters fitted, are 25.553 and 33.227 respectively for models 1 and 2, indicating strong support for model 1.

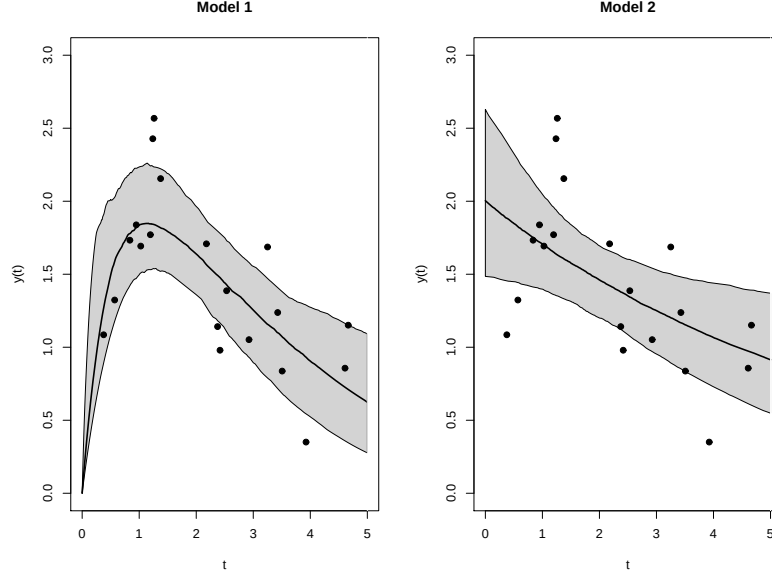


FIG. 7.2. Example 7.3: Pointwise 95% credible intervals for $E[Y]$ for Model 1 (left) and Model 2 (right).

Example 7.4. (Finite Mixture Model) The finite mixture model with K components has density

$$f(y|\omega, \theta, K) = \sum_{k=1}^K \omega_k f_k(y|\theta_k)$$

where θ_k are the parameters for component density k , and

$$0 < \omega_k < 1, \forall k \quad \& \quad \sum_{k=1}^K \omega_k = 1.$$

Most typically, although not exclusively, the component densities are from the same parametric location-scale family (say Gaussian) which differ in location and scale. For independent and identically distributed data $\mathbf{y} = (y_1, \dots, y_n)$ from this model, the likelihood arising from this model does not permit analytical calculation of the posterior distribution, so MCMC is commonly used. If K is known, MCMC is typically implemented using a collection of auxiliary variables $z_1, \dots, z_n$, and a completed data model

$$f(y, z|\omega, \theta, K) = \Pr[Z = z] f_z(y|\theta_z) \quad z \in \{1, 2, \dots, K\}$$

and a Gibbs sampler strategy updating the $\mathbf{z}$ and (ω, θ) from their full conditional posterior distributions. Conditional on (ω, θ) , the posterior

distribution for each z_i is a discrete distribution on $\{1, 2, \dots, K\}$, whereas conditional on the $\mathbf{z}$, the posterior for the mixture weights ω is a density on the K dimensional simplex, and the posterior for the components of θ_k proceeds using only those y_i for which $z_i = k$, for $k = 1, \dots, K$. Under a proper prior specification, the posterior is also proper, although inference is complicated by a permutation non-identifiability in the model specification (that is, the labels on the components may be permuted without changing the likelihood). Non-identifiability issues in inference can be resolved by, say, imposing constraints on the location parameters of the component densities, although this is not always completely satisfactory. See [16]. Perhaps the most satisfactory solution to MCMC mixing problems induced by lack of permutation identifiability is to introduce an independent random permutation of the component labels to supplement each MCMC iteration.

In addition to the difficulties of standard Bayesian analysis for this model, if K is also treated as an unknown parameter, then transdimensional MCMC is also needed; see [28] for an early influential paper. A discrete prior on K , typically on some finite set, completes the posterior specification, and yields a posterior distribution $\pi(K, \theta^{(K)}, \omega^{(K)} | \mathbf{y})$, where $\theta^{(K)} = (\theta_{K1}, \dots, \theta_{KK})$ and $\omega^{(K)} = (\omega_{K1}, \dots, \omega_{KK})$. In this model, dimension-changing moves correspond to changing the value of K , and in order to construct an efficient and tunable MCMC algorithm, moves which perturb K by ± 1 are typically considered. For example, moves of type $K \rightarrow K + 1$ are termed “birth” moves, and $K \rightarrow K - 1$ are termed “death” moves. The forward/reverse move pairs are then births from K and deaths from $K + 1$.

We consider the case of a Normal mixture model for simplicity. If y is one dimensional, each component density has two parameters, so a birth/death pair requires the introduction of three latent variables $\mathbf{u}$, a new mean and variance, and also a parameter to introduce a new component weight. If y is D -dimensional, $D + D(D + 1)/2 + 1$ new scalar parameters are needed, that is, a new mean and covariance matrix, and a new component weight parameter.

- **New location/scale parameters:** Two strategies are used to propose a new set of location and scale parameters; either these new parameters are generated independently of the current model parameters from proposal distribution, or a subset of the current components are selected and used to generate a new additional component. In the latter case, a common strategy involves “splitting” a currently existing component, that is, taking the mean and variance from component k , say, and by generating latent variables $\mathbf{u}$, and combining them with the parameters θ_{Kk} using the approach described in the previous section. Perturbing a location parameter using a Normal increment is generally straightforward

and may be sufficient to produce a reasonably functioning chain. Perturbing a scalar variance parameter is also reasonably easy, but perturbing a covariance matrix is more complex, although Wishart/Inverse Wishart generation can usually be implemented successfully in low dimensions. For example, if Σ_{Kk} is the covariance matrix for a selected component k , and A is a $D \times D$ positive definite symmetric matrix variate generated from a Wishart density, then $A\Sigma_{Kk}$ is also a positive definite symmetric matrix, so we may use it as the covariance matrix for a newly birthed component, and the Jacobian for this transform is straightforward to compute. The reverse “merge” move can be defined using the rules outlined above.

- **New weight parameters:** Generating the component weight for a new component is straightforward, using a random split or rescaling of the currently existing weights.

In typical mixture applications, inference about the parameter K may be of primary interest in a given application, but perhaps more commonly the mixture model offers flexible modelling in the presence of heterogeneity. The Normal mixture case is the most commonly studied, but other mixtures of other parametric models have also been used.

It should be noted that MCMC computation for varying dimension mixture models is notoriously hard to implement successfully using single chain methods, even if the multiple modes caused by permutation invariance can be overcome. The likelihood surface is complicated and multi-modal, so it is easy for the chain to get trapped in local high probability regions of the parameter space. To overcome this, multi-chain (or population) MCMC methods are used; see section 7.3

The flexible modelling of multivariate density functions using finite mixture approximations has a long history in Bayesian inference, and has been extended to the use of nonparametric or infinite mixture models based on the Dirichlet Process. These Bayesian nonparametric approaches address dimension-changing issues from a different perspective, and typically through Polyá urn schemes; they essentially utilize a similar auxiliary variable scheme, but identify the “clusters” of data having the same z value as observations from the same component mixture. *A priori*, the Polyá urn facilitates a countably infinite mixture model representation, although *a posteriori*, in light of a data set of size n , a maximum of n components are supported. Working on this finite space of cluster allocations avoids the need to dimension-changing moves.

Example 7.5. (Flexible curve-fitting via step functions) Suppose the regression relationship between a scalar covariate x and scalar covariate y

$$y = g(x) + \epsilon$$

is the focus of interest, for independent homoscedastic Normal errors $\{\epsilon\}$. Typically, we seek the prediction $\hat{y} = \hat{g}(x)$. Suppose we seek to approximate g locally on a finite domain $\mathcal{X}$ by a series of step functions:

$$g_K(x; \beta_K, \kappa_K) = \sum_{k=1}^K \beta_{Kk} \mathbb{1}_{B_k}(x)$$

where $(B_1, \dots, B_K)$ form a partition of $\mathcal{X}$ defined by a series of knots $\kappa_0, \kappa_1, \dots, \kappa_{K-1}, \kappa_K$ so that $B_k \equiv [\kappa_{k-1}, \kappa_k)$. Without loss of generality, assume $\mathcal{X} \equiv [0, 1)$, with $\kappa_0 = 0$, $\kappa_K = 1$. In this step-function approximation, $\beta_K = (\beta_{K1}, \dots, \beta_{KK})$ are the parameters defining the piecewise constant levels of the function.

The likelihood for data $(x_i, y_i), i = 1, \dots, n$ is a standard Normal likelihood

$$f(\mathbf{y}|M_K, \beta_K, \sigma, \kappa_K) = \prod_{i=1}^n \phi\left(\frac{y_i - g_K(x_i; \beta_K, \kappa_K)}{\sigma}\right)$$

Typically, within model M_K , a conjugate prior specification on (β_K, σ) is used so that the marginal likelihood $f(\mathbf{y}|M_K, \kappa_K)$ can be computed analytically by integrating out (β_K, σ) . The conjugate prior takes the form

$$p(\beta_K, \sigma) = p(\beta_K|\sigma)p(\sigma)$$

and $p(\beta_K|\sigma)$ can be chosen to be some multivariate Normal density; a suitable choice might be to have a zero mean Gaussian process structure, where the covariance matrix depends on some hyperparameters and κ_K .

A reversible jump MCMC algorithm can be constructed to explore the posterior $\pi(M_K, \kappa_K|\mathbf{y})$ which, under the conjugate specification for (β_K, σ) , depends on the marginal likelihood $f(\mathbf{y}|M_K, \kappa_K)$, and the prior $p(M_K, \kappa_K)$. One approach to the specification of this prior is to assume a Poisson process model for the change-in-level or jump locations of the step function. This amounts to setting $p(M_K)$ to be a Poisson distribution with rate parameter λ say, and then within model M_K , to regard κ_K as the order statistics derived from a Uniform random sample on $(0, 1)$, so that

$$p(\kappa_{K1}, \dots, \kappa_{K,K-1}|M_K) = (K-1)! \quad 0 < \kappa_1 < \dots < \kappa_{K,K-1} < 1.$$

Note that this prior depends only on the value of K and not the individual parameter values. A variant on this prior is given in [14], where a prior model based on the odd order statistics derived from a Uniform sample of size $2K-1$ is considered. This prior results in a larger expected separation between the κ values.

Dimension-changing moves in this model correspond to the addition or removal of a knot or knots. There are several mechanisms for doing

this; a new knot can be proposed uniformly on $(0,1)$, or uniformly in an interval selected in light of the current knot positions. In the former case, the reverse move corresponds to removing a knot uniformly at random from the current collection; this forward/reverse pair corresponds to the situation in Example 7.1, where the latent augmenting variable u is generated from the prior distribution, so the relevant terms cancel in the Hastings ratio. Model complexity in this model is controlled by the Poisson-model hyperparameter λ , and the covariance structure in the prior model for β_K .

After the marginalized posterior $\pi(M_K, \kappa_K | \mathbf{y})$ has been sampled at each iteration, it is trivial to sample from the conditional posterior on the remaining parameters, $\pi(\beta_K, \sigma | M_K, \kappa_K, \mathbf{y})$, which is available in closed form in the conjugate model. This then allows posterior samples of the function $g(x; \beta_K, \kappa_K)$ to be reconstructed. This methodology was exploited extensively for flexible modelling in the latter half of the 1990s to perform Bayesian versions of classical machine learning procedures such as classification and regression trees (CART), as it can be extended readily to the case of multiple predictors. In that case, the response function is modelled as a series of constants within hyper-rectangles defined by cut-points that partition predictor space. For a survey of the key techniques and literature, and examples, see [4].

Example 7.6. (Flexible Regression Modelling and Feature Selection)

In the extended linear regression model, response y is modelled as a linear combination of parameters d β plus error, that is

$$y = \mathbf{x}\beta + \epsilon \quad (7.3)$$

where $\mathbf{x} = (x_1, \dots, x_d)$ is a vector of predictor values. These predictors can correspond to the levels of a factor predictor, continuous covariates, nonlinear transforms of covariates, or functions combining any of these terms. With a fixed predictor set, there are $K = 2^d$ models that could be compared, and, although reversible jump MCMC could be used to explore the model space, it is not strictly necessary unless the marginal likelihood $f(\mathbf{y} | M_k)$ is not available analytically - this might be the case if ϵ is presumed to have a non-Normal distribution for example.

Using the basic regression model from equation (7.3), it is possible to broaden the modelling viewpoint to encompass flexible predictors. Flexible constructions include predictors based on splines, wavelets or other basis functions; these predictors allow the regression function to represent complicated regression relationships, and typically in such models it is the prediction, $\hat{y} = \mathbf{x}\hat{\beta}$, that is of more interest than inference about the parameters.

To illustrate the use of reversible jump methodology in this field, consider the derived predictor, $\varphi(x)$, based on a truncated polynomial basis

$$\varphi(x) = (g(x) - \kappa)_+^\nu = \begin{cases} (g(x) - \kappa)^\nu & x > \kappa \\ 0 & x \leq \kappa \end{cases}$$

for some function $g(\cdot)$ and parameters ν, κ . The simplest version is the linear form

$$\varphi(x) = (x - \kappa)_+ = \begin{cases} (x - \kappa) & x > \kappa \\ 0 & x \leq \kappa \end{cases}$$

which produces a piecewise linear continuous response function.

A series of such basis functions $\varphi_k(x), k = 1, \dots, K$, with corresponding $\kappa_1, \dots, \kappa_K$ can be used to form the flexible regression model

$$y = \varphi(x)\beta_K + \epsilon = \sum_{k=1}^K \varphi_k(x)\beta_{Kk} + \epsilon$$

which may be augmented using standard polynomial terms, say

$$y = \mathbf{x}\alpha + \varphi(x)\beta_K + \epsilon = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \sum_{k=1}^K \varphi_k(x)\beta_{Kk} + \epsilon$$

In this latter case, setting $K = 0$, leaving only the polynomial terms, is legitimate. This model is a generalization of the model described in Example 7.5; the model, albeit non-linear in x in general, is still linear in the parameters (α, β_K) , so much of the previous conjugate analysis goes through as before.

Another strategy for flexible modelling is to use wavelets as the basis function set. These basis functions have compact support, are orthogonal, and typically do not rely on the specification of separate knot positions, as these are specified in an automatic fashion depending on the type of wavelet used. The standard classical procedure for implementing wavelet-based models is to select a wavelet family, which automatically specifies the discrete wavelet transform matrix $\mathbf{W}(\mathbf{x})$ such that $\mathbf{y} = \mathbf{W}(\mathbf{x})\mathbf{c}$ say, where $\mathbf{c}$ are the estimated wavelet coefficients. The Bayesian version of this model uses the $\mathbf{W}(\mathbf{x})$ matrix as the design matrix in a linear model; variable dimension MCMC then proceeds to simplify this model by omitting sets of rows of this matrix.

7.2.3 Reversible Jump MCMC: Summary

The reversible jump MCMC algorithm introduced by Green was an important development as it allowed facilitated greater flexibility in Bayesian modelling. The algorithm itself relies on what are, in fact, standard Metropolis-Hastings moves that utilize latent or auxiliary variables and an extension to the state space on which the Markov chain operates. It is similar in spirit to the contemporaneous approach of [3], but does not retain the auxiliaries at all steps of the algorithm as in the Carlin and

Chib method. Good mixing of the reversible jump chain can sometimes be hard to achieve in practice without expert tuning, but the approach offers an important contribution to Bayesian model selection.

7.3 Population MCMC

7.3.1 Motivation

In early statistical applications of MCMC, a common approach to implementation was to run multiple (and usually short) chains independently, and to combine the samples from the chains in order to produce posterior summaries (see for example the applications in [5]), with, on occasion, just a single draw from the posterior being obtained from each separate chain, and more commonly, the output from each chain being heavily thinned, so as to produce an independent Monte Carlo sample. In simple applications, short runs can be effective, as the chain quickly reaches its stationary distribution, but in more complicated multi-parameter settings there is a danger that the chain remains in its transient phase for many iterations. The *multi-chain* approach offers some protection against the chain getting trapped in localized high probability regions, but it can also be considered wasteful, as it prizes the (at equilibrium) independent draws from the target posterior above everything else. Once MCMC methods came into the statistical mainstream, most practitioners realized the utility of *single-chain* approaches, that could be used effectively to perform standard Monte Carlo calculations using dependent draws from the posterior, albeit at the cost of increased variance of the Monte Carlo estimators. The difficulty in implementing single-chain algorithms is that the chain may not explore the entire model space effectively; the usual Metropolis-Hastings algorithm is essentially local in nature, as proposed moves are usually highly dependent on the current value of the chain. Therefore, the practical application of MCMC methods using single chains can be challenging for complicated models.

An early multi-chain method, *adaptive direction sampling*, was devised by [12]. In this Markov chain algorithm, a population of chains with identical target distribution f is maintained, and values are updated by proposing new values in the direction connecting two randomly selected population members. Specifically, in d dimensions, if the population is $\{\mathbf{x}_1, \dots, \mathbf{x}_N\}$, and $\mathbf{x}_j$ and $\mathbf{x}_k$ are selected at random, a change to $\mathbf{x}_j$ is proposed by setting

$$\mathbf{x}'_j = \mathbf{x}_j + u(\mathbf{x}_j - \mathbf{x}_k)$$

where u is a scalar random variate drawn from the density

$$f_U(u : \mathbf{x}_j, \mathbf{x}_k) \propto |u|^{d-1} f(\mathbf{x}_j + u(\mathbf{x}_j - \mathbf{x}_k))$$

7.3.2 Simulated Annealing

An early work that was influential on the statistical MCMC literature was the paper by Geman and Geman [7], which essentially popularized the Gibbs sampler, but which also utilized an optimization algorithm, *simulated annealing* [19], which operates as follows: to minimize the non-negative scalar function $g(\cdot)$ of argument $\mathbf{x}$, consider the unnormalized probability density

$$p_T(\mathbf{x}) \propto \exp \left\{ -\frac{1}{T} g(\mathbf{x}) \right\}$$

for constant T . Simulated annealing proceeds by running a Markov chain sampling algorithm at a decreasing sequence of T values $T_1 > T_2 > \dots$, with T_1 large. As T decreases, the function $p_T(\mathbf{x})$ becomes increasingly peaked around its mode. When T is large, $p_T(\mathbf{x})$ is relatively flat, so traversing the support is relatively straightforward, allowing the sampled points to locate the regions of high probability. In practice, T should be decreased slowly, so that the sampled points have sufficient chance to become concentrated around the mode.

Although simulated annealing was devised as an optimization algorithm, the core idea of running a chain or chains at different “temperatures” was subsequently used to develop multiple-chain MCMC methods in which the chains, rather than operating in parallel, independently and under the same transition dynamics, instead have potentially different stationary distributions, but are allowed to interact and exchange information.

7.3.3 Parallel Tempering

The key ideas in this section were developed by Geyer ([8, 10]), and form the basis of most current implementations of population MCMC methods. Consider a collection of related unnormalized (posterior) densities $\pi_1, \dots, \pi_N$ defined for simplicity on a common support within $\mathbb{R}^d$. Consider also a related collection of MH proposal kernels $P_1, \dots, P_N$ with proposal densities $q_1, q_2, \dots, q_N$ that facilitate the implementation of MCMC for each density. Assume that the usual conditions of irreducibility and aperiodicity are met.

Consider the augmented posterior

$$\tilde{\pi}(\theta_1, \dots, \theta_N) = \prod_{j=1}^N \pi_j(\theta_j) \tag{7.4}$$

defined on the Cartesian product of the individual parameter spaces. An MCMC algorithm constructed by taking a fixed cycle (or random scan) across $P_1, \dots, P_N$ is irreducible and aperiodic and has $\tilde{\pi}$ as its stationary

distribution; of course, marginally, samples of θ_j derived from the chain are distributed according to π_j .

In addition to the usual within-chain moves, consider the following “exchange” move; in addition to the fixed scan, select two chains j and k at random from $\{1, \dots, N\}$, and propose to exchange their values, that is, for chain j , propose to set θ'_j equal to θ_k , the current value in chain k , and similarly propose $\theta'_k = \theta_j$. This move is then be accepted with probability

$$\alpha(\theta_j, \theta_k) = \left\{ 1, \frac{\pi_j(\theta_k)\pi_k(\theta_j)}{\pi_j(\theta_j)\pi_k(\theta_k)} \right\}$$

This additional reversible move leaves the chain Markov on the product space, and does not affect the stationary distribution. However, the exchange move is likely to have a low acceptance rate unless the densities π_i and π_j are similar.

Combining the exchange move with temperature changing ideas inspired by simulated annealing, we have a mechanism for allowing parallel chains to exchange information in a useful fashion. Suppose that in a Bayesian inference problem, the target posterior $\pi(\cdot)$ is indexed $j = 1$, and that

$$\pi_j(\theta) \propto \{\pi(\theta)\}^{1/T_j}$$

say, for temperature constants $1 = T_1 < T_2 < \dots$. Note that this construction does not necessarily leave π_j a proper density, so in the Bayesian setting the alternate construction

$$\pi_j(\theta) \equiv \pi(\theta_j | \mathbf{y}) \propto \{f(\mathbf{y} | \theta)\}^{1/T_j} p(\theta)$$

which ensure a proper distribution might be considered. Using this approach, we ensure that if T_j and T_k are not separated by a large amount, π_j and π_k should be quite be similar; it should be possible to propose and accept frequently if $k = j \pm 1$, say, and although moves between π_1 and π_N changes might only be infrequently accepted, the temperature “ladder” allows for eventual passage of information across all chains. Applying the exchange move strategy produces an effective way of traversing the support of the target posterior, especially in the case of multi-modal densities, as densities that are severely multi-modal, with modes separated by low probability regions will be considerably more uniform (although still multi-modal) in their powered form.

At convergence, the stored MCMC samples represent a sample from the augmented joint posterior in equation (7.4), and consequently the samples of θ_1 values are variates generated from the corresponding marginal, that is, the target posterior.

Example 7.7 Tempering a mixture density Consider the two component mixture density

$$f(x) = \frac{3}{4}\phi(x+5) + \frac{1}{4}\phi(x-4)$$

with modes at -5 and 4 . The effect of the tempering is evident in Figure 7.3; the two well-separated modes in the target ($T = 1$) density are linked by an intervening region of increasingly large probability content. Thus MH moves from one modal region to the other, which are difficult when $T = 1$, are much easier when $T = 10$.

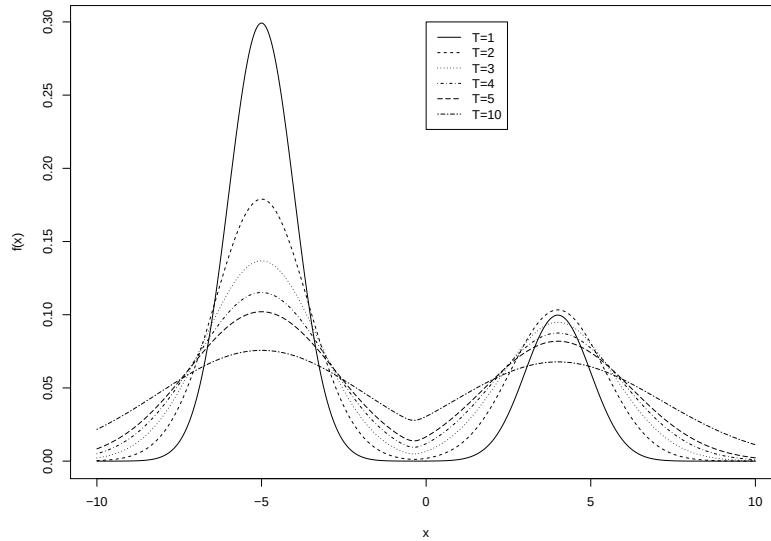


FIG. 7.3. Example 7.7: Tempered mixture density with tempering parameters $T = 1, 2, 3, 4, 5, 10$

Stochastic exchange moves: Simple deterministic exchange methods can be modified to exploit several different chains in a single move. For example, suppose a move within chain 1 (with stationary distribution $\pi_1 \equiv \pi$, the target posterior) is proposed as follows.

Algorithm 1 Stochastic Exchange Proposal

Current value of θ_1 is $\mathbf{u}_1$
for j in $2 : N$ **do**
 Propose $\mathbf{u}_j$ from $q_j(\mathbf{u}_{j-1}, \cdot)$
end for
Propose $\mathbf{v}_N$ from $q_N(\mathbf{u}_N, \cdot)$
for j in $(N - 1) : 1$ **do**
 Propose $\mathbf{v}_j$ from $q_{j+1}(\mathbf{v}_{j+1}, \cdot)$
end for
Take $\mathbf{v}_1$ as final proposed value for θ'_1 .
Accept θ'_1 with probability

$$\alpha(\theta_1, \theta'_1) = \min \left\{ 1, \frac{\pi_1(\theta'_1)}{\pi_1(\theta_1)} q(\mathbf{u}, \mathbf{v}) \right\}$$

where $q(\mathbf{u}, \mathbf{v})$ is the Hastings ratio for the proposed move.

It is straightforward to establish that

$$q(\mathbf{u}, \mathbf{v}) = \frac{\left\{ \prod_{j=2}^N q_j(\mathbf{v}_{j-1}, \mathbf{v}_j) \right\} q_N(\mathbf{v}_N, \mathbf{u}_N) \left\{ \prod_{j=1}^{N-1} q_{N-j+1}(\mathbf{u}_{N-j+1}, \mathbf{u}_{N-j}) \right\}}{\left\{ \prod_{j=2}^N q_j(\mathbf{u}_{j-1}, \mathbf{u}_j) \right\} q_N(\mathbf{u}_N, \mathbf{v}_N) \left\{ \prod_{j=1}^{N-1} q_{N-j+1}(\mathbf{v}_{N-j+1}, \mathbf{v}_{N-j}) \right\}}.$$

For example, if $N = 3$, we have

$$q(\mathbf{u}, \mathbf{v}) = \frac{q_2(\mathbf{v}_1, \mathbf{v}_2) q_3(\mathbf{v}_2, \mathbf{v}_3) q_3(\mathbf{v}_3, \mathbf{u}_3) q_3(\mathbf{u}_3, \mathbf{u}_2) q_2(\mathbf{u}_2, \mathbf{u}_1)}{q_2(\mathbf{u}_1, \mathbf{u}_2) q_3(\mathbf{u}_2, \mathbf{u}_3) q_3(\mathbf{u}_3, \mathbf{v}_3) q_2(\mathbf{v}_3, \mathbf{v}_2) q_3(\mathbf{v}_2, \mathbf{v}_1)}.$$

Recall that each $q_j(s, t)$ proposes a value t on the same space as s . If the proposal density is symmetric in its arguments, as in a standard Metropolis proposal, the expression simplifies to $q(\mathbf{u}, \mathbf{v}) = 1$.

The stochastic exchange approach is similar in spirit to the path and bridge sampling approaches to numerical integration in the computation or normalizing constants [6]. It has been used widely in Bayesian inference (see for example [25, 16] for examples in the context of mixture models).

Choosing the temperature values: It is clear that good choice of the number and values of the temperatures T_j is crucial to the adequate performance of the algorithm. In general, there are few results to give guidance as to the optimal choice for complex models where the parallel tempering may be necessary. The most common strategy adopted is equal or geometric spacing, although several formulations have been

presented suggestion alternative unequal spacing (see for example [10, 13, 1, 2]). Algorithms have also been developed that allow the temperatures to vary stochastically throughout the Markov chain runs, that is, the $\{T_j\}$ are treated as parameters in a global probability model. *Simulated tempering* [24, 10] is a parallel chain method that allows each T_j to vary according to stochastic updates, and allow the choice of a pseudoprior for the temperatures. This can allow favourable choices of the temperatures to be discovered by the algorithm.

7.3.4 Evolutionary Monte Carlo

Evolutionary Monte Carlo (EMC) was introduced by [22]. It combined techniques from the literature on genetic algorithms with population Monte Carlo. A genetic algorithm is a general optimization approach that maintains a diverse collection of “chromosomes” (parameter vectors) and searches for optimal configurations by quantifying and updating “fitness” using pseudo-evolutionary modifications. The two basic genetic algorithm evolutionary steps acting on a set of chromosomes $\mathbf{x}_1, \dots, \mathbf{x}_N$ include

- **Mutation:** Update $\mathbf{x}_j \rightarrow \mathbf{x}'_j$
- **Crossover:** Recombine $\mathbf{x}_j$ and $\mathbf{x}_k$ by exchanging a portion of each chromosome. That is, if $\mathbf{x}_j = (\mathbf{x}_{j1}, \mathbf{x}_{j2})$ and $\mathbf{x}_k = (\mathbf{x}_{k1}, \mathbf{x}_{k2})$, set

$$\mathbf{x}'_j = (\mathbf{x}_{j1}, \mathbf{x}_{k2}) \quad \mathbf{x}'_k = (\mathbf{x}_{k1}, \mathbf{x}_{j2}).$$

These two moves, in conjunction with the **Exchange** move from section 7.3.3, form the basis of a stochastic version of the genetic algorithm, which, in line with the Metropolis-Hastings philosophy, proposes new candidate chromosomes, and then accepts or rejects the proposed move with the usual MH acceptance probability. In EMC, the chromosomes are parameter vectors in the joint probability model from equation 7.4 formed by combining tempered marginal densities $\pi_j, j = 1, \dots, N$ at temperatures $T_j, j = 1, \dots, N$.

It can be readily seen that stochastic versions of all three move types are reversible, and have standard accept/reject dynamics, and the acceptance probability can be computed in a straightforward fashion. For the crossover move, this can be carried out with a deterministic proposal (hence becoming a partial exchange move) or stochastically. To carry out a stochastic crossover move, an adaptive direction sampling or “snooker” proposal [23] may be used; this proposes a new value for $\mathbf{x}_j$ in the direction connecting $\mathbf{x}_j$ and $\mathbf{x}_k$.

7.3.5 The Equi-Energy Sampler

The equi-energy sampler [20] is another population approach which utilizes other aspects of the tempered posterior distributions. Suppose that the target posterior is

$$\pi(\boldsymbol{\theta}) \propto \exp \{-\psi(\boldsymbol{\theta})\}$$

and that the temperature ladder is given by $1 = T_1 < T_2 < \dots < T_N < T_{N+1} = \infty$. Consider also a sequence of “energy levels” $-\infty = E_1 < E_2 < \dots < E_N$, and define the tempered distributions $\pi_1, \dots, \pi_N$ by

$$\pi_j(\boldsymbol{\theta}) \propto \exp \left\{ -\frac{1}{T_j} \max\{E_j, \psi(\boldsymbol{\theta})\} \right\} \quad j = 1, \dots, N.$$

Consider a partition of the parameter space $B_1, \dots, B_N$ where B_j is the set $B_j \equiv \{\boldsymbol{\theta} : E_j \leq \psi(\boldsymbol{\theta}) < E_{j+1}\}$. The equi-energy sampler utilizes this partition to develop effective moves in the parameter space.

Algorithm 2 Equi-Energy Sampler

- 1: Chain N : Run an MCMC algorithm to sample π_N at the highest temperature; after a suitable period, use the samples to define the initial partition, $\hat{B}_j^{(N)}, j = 1 \dots, N$.
- 2: After R iterations of Chain N , initiate a second chain, Chain $N - 1$, to sample from π_{N-1} (whilst continuing to update Chain N) using two types of move:
 - **Move 1:** MH move within Chain $N - 1$ (selected with probability $1 - p$),
 - **Move 2:** Equi-Energy Jump into Chain $N - 1$ (selected with probability p): suppose that the current value in Chain $N - 1$, $\boldsymbol{\theta}_{N-1}$, lies within partition component $\hat{B}_j^{(N)}$ say. Then propose a value $\boldsymbol{\theta}'_{N-1}$ uniformly from $\hat{B}_j^{(N)}$, and accept the move with probability

$$\alpha(\boldsymbol{\theta}_{N-1}, \boldsymbol{\theta}'_{N-1}) = \min \left\{ 1, \frac{\pi_{N-1}(\boldsymbol{\theta}'_{N-1})\pi_N(\boldsymbol{\theta}_{N-1})}{\pi_{N-1}(\boldsymbol{\theta}_{N-1})\pi_N(\boldsymbol{\theta}'_{N-1})} \right\}$$

otherwise remain at the same point.

After a suitable period, use the samples from Chain $N - 1$ to define a new partition, $\hat{B}_j^{(N-1)}, j = 1 \dots, N$.

- 3: Repeat Steps 1 and 2, for new Chains $N - 2, N - 3, \dots, 1$ etc, allowing equi-energy jumps between adjacent chains.
-

Under fairly general conditions, the equi-energy sampler can be shown to be ergodic at each level of the temperature ladder, such that Chain j has stationary distribution π_j [20]. The method is ingenious as it exploits both the temperature and location in the parameter space information.

7.3.6 Annealed Importance Sampling

In a related but distinct vein, Neal [26] developed a parallelized version of importance sampling that operated using a sequence of tempered distributions as in parallel tempering, and Markov transition kernels q_j with stationary distributions π_j . Instead of performing iterative sampling and exchanging information between chains, *annealed importance sampling* uses the tempered distributions to produce importance sampling weights that can be used to perform Monte Carlo integration.

Suppose that π_j is defined in terms of $\pi_1 \equiv \pi$, the (unnormalized) target posterior, and some (diffuse) density p_N that can be sampled in a straightforward manner, by

$$\pi_j(\boldsymbol{\theta}) = \{\pi_1(\boldsymbol{\theta})\}^{\eta_j} \{p_N(\boldsymbol{\theta})\}^{1-\eta_j}$$

for parameters $1 = \eta_1 > \eta_2 > \dots > \eta_N = 0$. The annealed importance sampling algorithm proceeds as follows to estimate the quantity

$$\mathbb{E}[g(\boldsymbol{\theta})] = \int g(\boldsymbol{\theta}) \pi_1(\boldsymbol{\theta}) d\boldsymbol{\theta}$$

Algorithm 3 Annealed Importance Sampling

- 1: **for** i in $1 : M$ **do**
- 2: Sample $\boldsymbol{\theta}_N$ from $\pi_N \equiv p_N$
- 3: **for** j in $(N - 1) : 1$ **do**
- 4: Sample $\boldsymbol{\theta}_j$ from $q_{j+1}(\boldsymbol{\theta}_{j+1}, \cdot)$
- 5: **end for**
- 6: Retain $\boldsymbol{\theta}_i = \boldsymbol{\theta}_1$, and form the importance sampling weight

$$w_i = \prod_{j=1}^{N-1} \frac{\pi_j(\boldsymbol{\theta}_j)}{\pi_{j+1}(\boldsymbol{\theta}_j)} \quad (7.5)$$

- 7: **end for**
- 8: Form the Monte Carlo estimate

$$\hat{\mathbb{E}}[g(\boldsymbol{\theta})] = \frac{1}{M} \sum_{i=1}^M w_i g(\boldsymbol{\theta}_i)$$

Neal [26] demonstrates that this Monte Carlo estimator is unbiased for the estimated as the importance sampling weights are correctly defined. Again, he used an augmented state space construction to demonstrate this. The first important note is that, for any π with associated Markov

transition density q that has π as its invariant distribution, we have by definition of the invariance that

$$\pi(d\boldsymbol{\theta}) = \int q(\mathbf{u}, \boldsymbol{\theta}) \pi(d\mathbf{u}).$$

Now, using the importance sampling identity with importance density p

$$\begin{aligned} \mathbb{E}_\pi[g(\boldsymbol{\theta})] &= \int g(\boldsymbol{\theta}) \pi(\boldsymbol{\theta}) d\boldsymbol{\theta} = \int \int g(\boldsymbol{\theta}) q(\mathbf{u}, \boldsymbol{\theta}) \pi(\mathbf{u}) d\mathbf{u} d\boldsymbol{\theta} \\ &= \int \int \frac{g(\boldsymbol{\theta}) \pi(\mathbf{u})}{p(\mathbf{u})} p(\mathbf{u}) q(\mathbf{u}, \boldsymbol{\theta}) d\mathbf{u} d\boldsymbol{\theta} \end{aligned}$$

which suggests the estimator

$$\hat{\mathbb{E}}_\pi[g(\boldsymbol{\theta})] = \frac{1}{M} \sum_{i=1}^M \frac{g(\boldsymbol{\theta}_i) \pi(\mathbf{u}_i)}{p(\mathbf{u}_i)}$$

where $\{u_i\}$ are independent draws from $p(\cdot)$, and $\{\boldsymbol{\theta}_i\}$ are independent draws from $q(\mathbf{u}_i, \cdot)$ for $i = 1, \dots, n$. The annealed importance sampling algorithm adapts this result on the augmented space using the augmented posterior equation (7.4). We have that the augmented density can be written

$$\tilde{\pi}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_N) = \pi_1(\boldsymbol{\theta}_1) \prod_{j=2}^N \tilde{q}_j(\boldsymbol{\theta}_{j-1}, \boldsymbol{\theta}_j)$$

where

$$\tilde{q}_j(\mathbf{x}, \mathbf{y}) = \frac{q_j(\mathbf{y}, \mathbf{x}) \pi_j(\mathbf{y})}{\pi_j(\mathbf{x})}$$

is the so-called reverse transition density. The proposal density is

$$p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_N) = \pi_N(\boldsymbol{\theta}_N) \prod_{j=2}^N q_j(\boldsymbol{\theta}_j, \boldsymbol{\theta}_{j-1})$$

and thus the importance weight is given by

$$\frac{\tilde{p}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_N)}{p(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_N)}$$

which reduces to (7.5).

7.3.7 *Population MCMC: Summary*

Population MCMC methods facilitate effective exploration of the support of complicated and potentially multi-modal posterior distributions that single-chain methods often find difficult to explore. Usually population MCMC relies upon a series of temperature modulated versions of the target posterior, and on designed MH moves across the collection of Markov chains working with each version separately, allowing for exchange of sampled values between them. In this section, we have discussed the fixed-dimension case, but the methodology applies also to populations of variable-dimension chains [18]. Also, population approaches with exchange of information can be adopted in sequential Monte Carlo (see, for example, [17] for a comparison of population MCMC and sequential Monte Carlo). For a comprehensive survey of population MCMC methods, and other recent advances, see [21].

REFERENCES

- [1] Atchadé, Y. F., Roberts, G. O., and Rosenthal, J. S. (2011). Towards optimal scaling of Metropolis-coupled Markov chain Monte Carlo. *Stat. Comp.*, **21**(4), 555–568.
- [2] Behrens, G., Friel, N., and Hurn, M. (2012). Tuning tempered transitions. *Stat. Comp.*, **22**, 65–78.
- [3] Carlin, B. P. and Chib, S. (1995). Bayesian model choice via Markov chain Monte Carlo. *J. R. Statist. Soc. B*, **57**, 473–84.
- [4] Denison, D. G. T., Holmes, C. C., Mallick, B. K., and Smith, A. F. M. (2002). *Bayesian Methods for Nonlinear Classification and Regression*. John Wiley.
- [5] Gelfand, A.E. and Smith, A.F.M. (1990). Sampling based approaches to calculating marginal densities. *J. Amer. Statist. Assoc.*, **85**, 398–409.
- [6] Gelman, A. and Meng, X-L (1998). Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical Science*, **13**(2), 163–185.
- [7] Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **6**, 721–741.
- [8] Geyer, C. J. (1991). Monte Carlo maximum likelihood for dependent data. In *Comp. Sci. and Statis.: Proc. 23rd Symp. Interface*, (ed. E. Keramidas), pp. 156–163.
- [9] Geyer, C. J. and Møller, J. (1994). Simulation procedures and likelihood inference for spatial point processes. *Scand. J. Stat.*, **21**(4), pp. 359–373.
- [10] Geyer, C. J. and Thompson, E. A. (1995). Annealing Markov chain Monte Carlo with applications to ancestral inference. *J. Amer. Statist. Assoc.*, 909–920.
- [11] Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1995). *Markov Chain Monte Carlo in Practice: Interdisciplinary Statistics (Chapman & Hall/CRC Interdisciplinary Statistics)*. Chapman and Hall/CRC.
- [12] Gilks, W. R., Roberts, G. O., and George, E. I. (1994). Adaptive direction sampling. *J. R. Statist. Soc. D (The Statistician)*, **43**(1), pp. 179–189.
- [13] Goswami, G. and Liu, J. S. (2007). On learning strategies for evolutionary Monte Carlo. *Stat. Comp.*, **17**(1), 23–38.
- [14] Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, **82**(4),

- 711–732.
- [15] Grenander, U. and Miller, M. (1994). Representations of knowledge in complex systems (with discussion). *J. R. Statist. Soc. B*, **56**(4), 549–603.
 - [16] Jasra, A., Holmes, C. C., and Stephens, D. A. (2005). MCMC and the label switching problem in Bayesian mixture models. *Statistical Science*, **20**, 50–67.
 - [17] Jasra, A., Stephens, D. A., and Holmes, C. C. (2007). On population-based simulation for static inference. *Stat. Comp.*, **17**, 263–279. 10.1007/s11222-007-9028-9.
 - [18] Jasra, A., Stephens, D. A., and Holmes, C. C. (2007). Population-Based Reversible Jump Markov Chain Monte Carlo. *Biometrika*, **94**(4), 787–807.
 - [19] Kirkpatrick, S., Jr., C.D. Gelatt, and Vecchi, M.P. (1983). Optimization by simulated annealing. *Science*, **220**, 671–680.
 - [20] Kou, S. C., Zhou, Q., and Wong, W. H. (2006). Equi-energy sampler with applications in statistical inference and statistical mechanics. *Ann. Stat.*, **34**(4), pp. 1581–1619.
 - [21] Liang, F., Liu, C., and Carroll, R. J. (2010). *Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples*. Wiley Series in Computational Statistics. John Wiley & Sons.
 - [22] Liang, FM and Wong, WH (2001). Real-parameter evolutionary Monte Carlo with applications to Bayesian mixture models. *J. Amer. Statist. Assoc.*, **96**(454), 653–666.
 - [23] Liu, J. S., Liang, F., and Wong, W. H. (2000). The multiple-try method and local optimization in Metropolis sampling. *J. Amer. Statist. Assoc.*, **95**(449), 121–134.
 - [24] Marinari, E. and Parisi, G. (1992). Simulated Tempering: A New Monte Carlo Scheme. *Europhys. Lett.*, **19**(6), 451.
 - [25] Neal, R. M. (1996). Sampling from multimodal distributions using tempered transitions. *Stat. Comp.*, **6**, 353–366. 10.1007/BF00143556.
 - [26] Neal, R. M. (2001). Annealed importance sampling. *Stat. Comp.*, **11**(2), 125–139.
 - [27] Phillips, D. B. and Smith, A. F. M. (1995). Bayesian model comparison via jump diffusions. In *Markov Chain Monte Carlo in Practice*, (ed. W. R. Gilks, S. R. Richardson, and D. J. Spiegelhalter). Chapman and Hall, London.
 - [28] Richardson, S. and Green, P. J. (1997). On Bayesian Analysis of Mixtures with an Unknown Number of Components (with discussion). *J. R. Statist. Soc. B*, **59**(4), 731–792.
 - [29] Roberts, G.O., Gelman, A., and Gilks, W.R. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Prob.*, **7**, 110–120.

- [30] Stephens, M. (2000). Bayesian analysis of mixture models with an unknown number of components- an alternative to reversible jump methods. *Ann. Stat.*, **28**(1), 40–74.
- [31] Tierney, L. (1994). Markov chains for exploring posterior distributions. *Ann. Stat.*, **22**, 1701–1762.
- [32] Tierney, L. (1998). A note on metropolis-hastings kernels for general state spaces. *Ann. Appl. Prob.*, **8**(1), pp. 1–9.