Markov Chain Monte Carlo Methods for Bayesian Analysis

Joseph B. Kadane
Department of Statistics, Carnegie Mellon University, Pittsburgh, Pennsylvania

Brian Diggs and Christopher Genovese
Physics Department, Carnegie Mellon University, Pittsburgh, Pennsylvania

Shing-te Li and Robert Swendsen
Department of Statistics, Carnegie Mellon University, Pittsburgh, Pennsylvania

Abstract. This paper reviews the ways that statisticians use Markov Chain Monte Carlo (MCMC) methods. These techniques have made it possible to attack problems that would previously have been intractable. The application of MCMC methods has grown tremendously in recent years as this potential has been recognized. The impact on statistical practice, particularly on Bayesian computation, has been profound.

1. Introduction

Kadane [1993] made a case for Bayesian analysis as a way of thinking about statistical problems in general and physical oceanographic problems in specific. This paper discusses a popular method for doing the required calculations that relies heavily on methods borrowed from theoretical physics.

Bayesian inference centers on a posterior distribution for a parameter $\theta$ (which may be a scalar, a vector, a matrix or whatever) given data $x$ (which again can take many forms). There are two basic ingredients to a Bayesian inference: a likelihood $f(x \mid \theta)$ which relates the data $x$ to the parameters $\theta$, and a prior $\pi(\theta)$ which is a probability distribution representing what is already known or believed about $\theta$ before the data $x$ are collected and analyzed. The likelihood function is the density function of the data $x$ viewed as a function of the parameter $\theta$. Both the likelihood and the prior are statements of belief on the part of the author(s); the point of making them explicit is that the reader's beliefs can be compared with those of the author(s).

The posterior distribution $g(\theta \mid x)$ can be expressed in terms of the prior distribution $\pi(\theta)$ and the likelihood function $f(x \mid \theta)$ as

$$g(\theta \mid x) = \frac{f(x \mid \theta)\pi(\theta)}{\int_{\Theta} f(x \mid \theta)\pi(\theta) d\theta}.$$  

(1)

To implement this paradigm, it is necessary to specify the prior $\pi(\theta)$ and the likelihood $f(x \mid \theta)$, and to compute the integral in the denominator. The specifications are part of elicitation; the computation of the integral is what Markov Chain Monte Carlo (MCMC) methods do. MCMC methods are based on a computer simulation. The output of an MCMC simulation is a sample of parameter values $\theta_1, \ldots, \theta_N$ drawn (approximately) from the posterior distribution $g(\theta \mid x)$. Practical issues arise in deciding when the approximation to the posterior is good enough and in coping with the fact that the sample values are not independent.

A general form of Markov chain sampler in use in statistics is the Metropolis-Hastings algorithm [Metropolis et al., 1953; Hastings, 1970]. Suppose the chain is at the current state $\theta^t$ at iteration $t$. A new candidate observation $\theta^{\text{cond}}$ is generated from a proposal distribution $q(\cdot \mid \theta^t)$. The candidate $\theta^{\text{cond}}$ is accepted with probability

$$
\alpha(\theta^t, \theta^{\text{cond}}) = \min \left(1, \frac{f(x \mid \theta^{\text{cond}}) \pi(\theta^{\text{cond}}) q(\theta^t \mid \theta^{\text{cond}})}{f(x \mid \theta^t) \pi(\theta^t) q(\theta^{\text{cond}} \mid \theta^t)} \right).
$$

(2)

Note that after $\theta^{\text{cond}}$ is chosen, $\alpha$ can be computed because the likelihood $f$, the prior $\pi$ and the proposal distribution $q$ are each available. The integral in (1) is unnecessary because of the ratio form in (2). If the candidate $\theta^{\text{cond}}$ is accepted, $\theta^{t+1} = \theta^{\text{cond}}$. Otherwise, $\theta^{t+1} = \theta^t$.

A more general chain is defined by applying Metropolis-Hastings ideas component-wise. Here the parameter $\theta$ is divided into $h$ components, each of which may have an arbitrary structure (scalar, vector, matrix, etc.). Then $\theta$ can be written as $\theta = (\theta_1, \theta_2, \ldots, \theta_h)$. Let $\theta_{t,i}$ denote the state of $\theta_i$ after the $t^{th}$ iteration. For step $i$ of iteration $t + 1$, $\theta_i$ is updated using a M-H step. The candidate $\theta^{\text{cond}}_i$ is generated from the proposal distribution

$$q_i(\theta^{\text{cond}}_i \mid \theta_{t,i}, \theta_{t,i-1}),$$

41
where \( \theta_{t-i} = \{ \theta_{t+1,i}, \ldots, \theta_{t+1,i-1}, \theta_{t+1,i}, \ldots, \theta_{t+1} \} \) (so the updated values are used for the first \( i - 1 \) components, and unupdated values are used for the last \( h - i \) components.) The candidate is accepted with probability
\[
\alpha(\theta_{t-i}, \theta_{t-i}, \theta_{i}^{\text{cand}}) = \min \left( 1, \frac{f(x | \theta_{i}^{\text{cand}}, \theta_{t-i}) \pi(\theta_{i}^{\text{cand}} | \theta_{t-i}) q(\theta_{i} | \theta_{t-i}, \theta_{i}^{\text{cand}}, \theta_{t-i})}{f(\theta_{t-i} | \theta_{t-i}, \theta_{i}^{\text{cand}}) \pi(\theta_{t-i} | \theta_{t-i}) q(\theta_{i}^{\text{cand}} | \theta_{t-i}, \theta_{t-i})} \right)
\]
(3)

If \( \theta_{i}^{\text{cand}} \) is accepted, \( \theta_{t+1,i} = \theta_{i}^{\text{cand}} \); otherwise \( \theta_{t+1,i} = \theta_{t,i} \).

There are several important special cases of the Metropolis-Hastings algorithm, as given in Table 1. The most widely used of these, the Gibbs or heat bath algorithm, entered statistics through work in bitmap graphics. It has power only when used component-wise. The conditional distributions required to use the Gibbs or Heat bath algorithm are computed analytically [Gilks, 1996]. In a component-wise algorithm, each step may be chosen to be one of the special cases; thus the sampling strategy may be chosen to take advantage of the particular nature of the likelihood and prior involved.

How can a sample from the posterior distribution \( g(\theta | x) \) be used for Bayesian inference? A common approach is to approximate the expectation of some function \( h(\theta) \), i.e.
\[
\int_{\theta} h(\theta) g(\theta | x) d\theta.
\]
(4)

A natural way to do this is with a sample average \( \frac{1}{n} \sum_{i=1}^{n} h(\theta^{i}) \), or perhaps with a truncated sample average
\[
\frac{1}{n-m} \sum_{i=m+1}^{n} h(\theta^{i})
\]
(5)

Under what conditions will (5) approach (4), and can anything be said about the error distribution of the difference?

My treatment of these issues is extremely sketchy, but I hope the references I give will help those who may wish a careful treatment of the issues. In ascending order of generality and difficulty, see Roberts [1996], Tierney [1996], and Tierney [1994]. Under very general conditions, (5) does approach (4) as \( n \to \infty \), and the difference has a normal (or Gaussian) distribution.

Using the spectral decomposition of the Markov chain [Kartin, 1966], the posterior distribution \( g(\theta | x) \) is the eigenvector corresponding to the eigenvalue 1. The next largest eigenvalue in absolute value, \( \lambda^* \), satisfies \( | \lambda^* | < 1 \). (Many of the regularity conditions have to do with assuring that the eigenvalue 1 has multiplicity 1). The variance of the limiting Gaussian distribution depends on \( \lambda^* \), which is related to the correlation time \( \tau \) in physics by \( \tau = -1 / \log \lambda^* \).

There are many choices to be made in implementing a Markov Chain Monte Carlo simulation. Surprisingly, perhaps, the choice of parameterization is among the more critical. If the values of two components of \( \theta \) are highly correlated, component-wise methods will not mix well, i.e., will be slow to visit the full support of \( g(\theta | x) \). However, a simple linear transformation can create new variables that are independent, and which will mix well. Gilks and Roberts [1996] have a good discussion of various reparameterization methods.

Another choice is what proposal distribution to use. The general advice is to choose a method (i.e., from Table 1), and then to choose specific distributions that have somewhat wider (or thicker) tails than the posterior distribution. Often the t-family (univariate or multivariate) is used for this purpose, as posterior distributions tend (as the sample size gets large) toward normal, or Gaussian distributions. The literature suggests that an acceptance rate of 0.15 to 0.5 is desirable; often the proposal distribution has a tunable parameter that can be manipulated to achieve acceptance rates in this range.

There is often a natural way to choose components in a component-wise simulation, but joining possibly highly correlated parameters into a single component may be wise.

**Table 1. Special Cases of the General Component-Wise Metropolis-Hastings Algorithm**

<table>
<thead>
<tr>
<th>Name</th>
<th>Proposal Distribution</th>
<th>Acceptance Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Metropolis</td>
<td>( q(\theta^{\text{cand}}</td>
<td>\theta) = q(\theta</td>
</tr>
<tr>
<td>1a. Random-Walk Metropolis</td>
<td>( q(\theta^{\text{cand}}</td>
<td>\theta) = q(\theta - \theta^{\text{cand}}</td>
</tr>
<tr>
<td>2. Independence</td>
<td>( q(\theta^{\text{cand}}</td>
<td>\theta) = q(\theta^{\text{cand}}) )</td>
</tr>
<tr>
<td>3. Gibbs/Heat Bath</td>
<td>( q(\theta^{\text{cand}}</td>
<td>\theta) = f(X</td>
</tr>
</tbody>
</table>

42
In a component-wise simulation, it is not necessary that the order in which components are considered be fixed. It is possible to have random choice of components for updating, or to systematically update some components more than others to achieve better mixing of the chain [Zeger and Karim, 1991].

The issues of the choice of a starting value and the length of the burn-in period (i.e., the choice of \( m \) in (5)), are closely related. If the starting value could be sampled from the target posterior distribution, the chain would have achieved its stationary distribution right away, and \( m \) could be taken to be zero. More practically, some initial exploration is often done, partly with the thought of finding a reasonable starting value, thus permitting a small \( m \) to be chosen. Often graphs of certain components of \( \theta \) against \( t \) are monitored to determine \( m \) by eye. (See Gelman [1996] for a discussion.)

Another question is when to stop. All the theory justifying Markov Chain Monte Carlo analysis is asymptotic, having to do with limits as the length of the chain increases without limit. For this reason, the theory is not very helpful in determining when to stop. The "gibbsit" program [Raftery and Lewis, 1996], available in Statlib, uses a preliminary sample to estimate how long a chain should be run. A recent review of convergence diagnostics is Cowles and Carlin [1996].

The question of whether to run one very long chain, several long chains, or many short ones, has been rather hotly debated in the statistical literature, respectively argued by Geyer [1992]; Gelman and Rubin [1992a,b]; and Gelfand and Smith [1990]. Generally statisticians who are comfortable with time series of dependent observations tend toward fewer, longer chains, while those used to independent observations tend toward more, shorter chains.

Finally, there is an issue of improper distributions, i.e., distributions whose integral is infinite. All too, often, statisticians use such distributions claiming that they represent ignorance, especially about aspects of the prior distribution. Yet impropriety anywhere in the probabilistic structure invalidates the theorems used to justify MCMC. In practice, a chain that does not converge often embeds an improper distribution as the cause. For one recent view of this matter, see Robert and Casella [1996].

Generally, progress is made on these issues by a combination of theory and experience. For simple problems, the implementation issues often do not matter much. However, for larger problems, they can be quite important.

One of the more influential ideas in contemporary Bayesian statistics is that of a hierarchical model. In general, in a hierarchical model the data \( x \) are related to a set of parameters \( \theta_1 \) by a likelihood \( f(x \mid \theta_1) \). The parameters \( \theta_1 \) are related probabilistically to parameters \( \theta_2 \) by a function \( \pi_1(\theta_1 \mid \theta_2) \) etc., until finally there is a prior \( \pi_p(\theta_p) \) for some \( p \). Then the posterior is

\[
g(\theta_1, \theta_2, \ldots, \theta_p \mid x) \propto f(x \mid \theta_1) \pi_{p-1} \pi_1(\theta_1 \mid \theta_{p-1}) \pi(\theta_p).
\]

Models of this kind can come up naturally in oceanography where the data \( x \) are measured observations, parameters \( \theta_1 \) might represent unobserved true values at particular times and places, \( \theta_2 \) might represent more aggregated quantities, etc. The point of modeling in this way is that it encourages careful consideration of each stage in the process. The first stage is about measurement error, and \( f(x \mid \theta_1) \) should incorporate (probabilistically) what is known about that process. Then \( \pi_1(\theta_1 \mid \theta_2) \) would model how time and place parameters are believed to relate, etc.

Hierarchical models are natural to explore using Markov Chain Monte Carlo methods. Each \( \theta_1 \) becomes one or more components. Because of the conditional independences built into a hierarchical model, each \( \theta_1 \) depends only on \( \theta_{i-1} \) and \( \theta_{i+1} \), and not on any of the other \( \theta \)’s.

The combination of Markov Chain Monte Carlo methods and hierarchical modeling makes this an exciting time for Bayesian statistics.

References


