THE POWER TO KNOW.

SAS/STAT® 14.2 User's Guide
Introduction to Bayesian Analysis Procedures

This document is an individual chapter from *SAS/STAT® 14.2 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2016. *SAS/STAT® 14.2 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/STAT® 14.2 User's Guide

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

November 2016

Chapter 7
Introduction to Bayesian Analysis Procedures

Contents

Overview

SAS/STAT software provides Bayesian capabilities in six procedures: BCHOICE, FMM, GENMOD, LIFEREG, MCMC, and PHREG. The FMM, GENMOD, LIFEREG, and PHREG procedures provide Bayesian analysis in addition to the standard frequentist analyses they have always performed. Thus, these procedures provide convenient access to Bayesian modeling and inference for finite mixture models, generalized linear models, accelerated life failure models, Cox regression models, and piecewise constant baseline hazard models (also known as piecewise exponential models). The BCHOICE procedure provides Bayesian analysis for discrete choice models. The MCMC procedure is a general procedure that fits Bayesian models with arbitrary priors and likelihood functions.

This chapter provides an overview of Bayesian statistics; describes specific sampling algorithms used in these four procedures; and discusses posterior inference and convergence diagnostics computations. Sources that provide in-depth treatment of Bayesian statistics can be found at the end of this chapter, in the section "A Bayesian Reading List" on page 153. Additional chapters contain syntax, details, and examples for the individual procedures BCHOICE(see Chapter 27, "The BCHOICE Procedure"), FMM (see Chapter 40, "The FMM Procedure"), GENMOD (see Chapter 45, "The GENMOD Procedure"), LIFEREG (see Chapter 70, "The LIFEREG Procedure"), MCMC (see Chapter 74, "The MCMC Procedure"), and PHREG (see Chapter 86, "The PHREG Procedure").

Introduction

The most frequently used statistical methods are known as *frequentist* (or *classical*) methods. These methods assume that unknown parameters are fixed constants, and they define probability by using limiting relative frequencies. It follows from these assumptions that probabilities are objective and that you cannot make probabilistic statements about parameters because they are fixed. Bayesian methods offer an alternative approach; they treat parameters as random variables and define probability as "degrees of belief" (that is, the probability of an event is the degree to which you believe the event is true). It follows from these postulates that probabilities are subjective and that you can make probability statements about parameters. The term "Bayesian" comes from the prevalent usage of Bayes' theorem, which was named after the Reverend Thomas Bayes, an eighteenth century Presbyterian minister. Bayes was interested in solving the question of inverse probability: after observing a collection of events, what is the probability of one event?

Suppose you are interested in estimating θ from data $\mathbf{y} = \{y_1, \ldots, y_n\}$ by using a statistical model described by a density $p(\mathbf{y}|\theta)$. Bayesian philosophy states that θ cannot be determined exactly, and uncertainty about the parameter is expressed through probability statements and distributions. You can say that θ follows a normal distribution with mean 0 and variance 1, if it is believed that this distribution best describes the uncertainty associated with the parameter. The following steps describe the essential elements of Bayesian inference:

1. A probability distribution for θ is formulated as $\pi(\theta)$, which is known as the *prior* distribution, or just the prior. The prior distribution expresses your beliefs (for example, on the mean, the spread, the skewness, and so forth) about the parameter before you examine the data.
2. Given the observed data $\mathbf{y}$, you choose a statistical model $p(\mathbf{y}|\theta)$ to describe the distribution of $\mathbf{y}$ given θ.
3. You update your beliefs about θ by combining information from the prior distribution and the data through the calculation of the *posterior* distribution, $p(\theta|\mathbf{y})$.

The third step is carried out by using Bayes' theorem, which enables you to combine the prior distribution and the model in the following way:

$$p(\theta|\mathbf{y}) = \frac{p(\theta, \mathbf{y})}{p(\mathbf{y})} = \frac{p(\mathbf{y}|\theta)\pi(\theta)}{p(\mathbf{y})} = \frac{p(\mathbf{y}|\theta)\pi(\theta)}{\int p(\mathbf{y}|\theta)\pi(\theta)d\theta}$$

The quantity

$$p(\mathbf{y}) = \int p(\mathbf{y}|\theta)\pi(\theta)d\theta$$

is the normalizing constant of the posterior distribution. This quantity $p(\mathbf{y})$ is also the marginal distribution of $\mathbf{y}$, and it is sometimes called the marginal distribution of the data. The likelihood function of θ is any function proportional to $p(\mathbf{y}|\theta)$; that is, $L(\theta) \propto p(\mathbf{y}|\theta)$. Another way of writing Bayes' theorem is as follows:

$$p(\theta|\mathbf{y}) = \frac{L(\theta)\pi(\theta)}{\int L(\theta)\pi(\theta)d\theta}$$

The marginal distribution $p(\mathbf{y})$ is an integral. As long as the integral is finite, the particular value of the integral does not provide any additional information about the posterior distribution. Hence, $p(\theta|\mathbf{y})$ can be written up to an arbitrary constant, presented here in proportional form as:

$$p(\theta|\mathbf{y}) \propto L(\theta)\pi(\theta)$$

Simply put, Bayes' theorem tells you how to update existing knowledge with new information. You begin with a prior belief $\pi(\theta)$, and after learning information from data $\mathbf{y}$, you change or update your belief about θ and obtain $p(\theta|\mathbf{y})$. These are the essential elements of the Bayesian approach to data analysis.

In theory, Bayesian methods offer simple alternatives to statistical inference—all inferences follow from the posterior distribution $p(\theta|\mathbf{y})$. In practice, however, you can obtain the posterior distribution with straightforward analytical solutions only in the most rudimentary problems. Most Bayesian analyses require sophisticated computations, including the use of simulation methods. You generate samples from the posterior distribution and use these samples to estimate the quantities of interest. PROC MCMC uses a self-tuning Metropolis algorithm (see the section "Metropolis and Metropolis-Hastings Algorithms" on page 130). The GENMOD, LIFEREG, and PHREG procedures use the Gibbs sampler (see the section "Gibbs Sampler" on page 131). The BCHOICE and FMM procedure use a combination of Gibbs sampler and latent variable sampler. An important aspect of any analysis is assessing the convergence of the Markov chains. Inferences based on nonconverged Markov chains can be both inaccurate and misleading.

Both Bayesian and classical methods have their advantages and disadvantages. From a practical point of view, your choice of method depends on what you want to accomplish with your data analysis. If you have prior information (either expert opinion or historical knowledge) that you want to incorporate into the analysis, then you should consider Bayesian methods. In addition, if you want to communicate your findings in terms of probability notions that can be more easily understood by nonstatisticians, Bayesian methods might be appropriate. The Bayesian paradigm can often provide a framework for answering specific scientific questions that a single point estimate cannot sufficiently address. Alternatively, if you are interested only in estimating parameters based on the likelihood, then numerical optimization methods, such as the Newton-Raphson method, can give you very precise estimates and there is no need to use a Bayesian analysis. For further discussions of the relative advantages and disadvantages of Bayesian analysis, see the section "Bayesian Analysis: Advantages and Disadvantages" on page 128.

Background in Bayesian Statistics

Prior Distributions

A prior distribution of a parameter is the probability distribution that represents your uncertainty about the parameter before the current data are examined. Multiplying the prior distribution and the likelihood function

together leads to the posterior distribution of the parameter. You use the posterior distribution to carry out all inferences. You cannot carry out any Bayesian inference or perform any modeling without using a prior distribution.

Objective Priors versus Subjective Priors

Bayesian probability measures the degree of belief that you have in a random event. By this definition, probability is highly subjective. It follows that all priors are *subjective priors*. Not everyone agrees with this notion of subjectivity when it comes to specifying prior distributions. There has long been a desire to obtain results that are objectively valid. Within the Bayesian paradigm, this can be somewhat achieved by using prior distributions that are "objective" (that is, that have a minimal impact on the posterior distribution). Such distributions are called *objective* or *noninformative* priors (see the next section). However, while noninformative priors are very popular in some applications, they are not always easy to construct. See DeGroot and Schervish (2002, Section 1.2) and Press (2003, Section 2.2) for more information about interpretations of probability. See Berger (2006) and Goldstein (2006) for discussions about objective Bayesian versus subjective Bayesian analysis.

Noninformative Priors

Roughly speaking, a prior distribution is noninformative if the prior is "flat" relative to the likelihood function. Thus, a prior $\pi(\theta)$ is noninformative if it has minimal impact on the posterior distribution of θ. Other names for the noninformative prior are *vague*, *diffuse*, and *flat* prior. Many statisticians favor noninformative priors because they appear to be more objective. However, it is unrealistic to expect that noninformative priors represent total ignorance about the parameter of interest. In some cases, noninformative priors can lead to *improper posteriors* (nonintegrable posterior density). You cannot make inferences with improper posterior distributions. In addition, noninformative priors are often not invariant under transformation; that is, a prior might be noninformative in one parameterization but not necessarily noninformative if a transformation is applied.

See Box and Tiao (1973) for a more formal development of noninformative priors. See Kass and Wasserman (1996) for techniques for deriving noninformative priors.

Improper Priors

A prior $\pi(\theta)$ is said to be improper if

$$\int \pi(\theta) d\theta = \infty$$

For example, a uniform prior distribution on the real line, $\pi(\theta) \propto 1$, for $-\infty < \theta < \infty$, is an improper prior. Improper priors are often used in Bayesian inference since they usually yield noninformative priors and proper posterior distributions. Improper prior distributions can lead to posterior impropriety (improper posterior distribution). To determine whether a posterior distribution is proper, you need to make sure that the normalizing constant $\int p(\mathbf{y}|\theta)p(\theta)d\theta$ is finite for all $\mathbf{y}$. If an improper prior distribution leads to an improper posterior distribution, inference based on the improper posterior distribution is invalid.

The GENMOD, LIFEREG, and PHREG procedures allow the use of improper priors—that is, the flat prior on the real line—for regression coefficients. These improper priors do not lead to any improper posterior distributions in the models that these procedures fit. PROC MCMC allows the use of any prior, as long as the

distribution is programmable using DATA step functions. However, the procedure does not verify whether the posterior distribution is integrable. You must ensure this yourself.

Informative Priors

An informative prior is a prior that is not dominated by the likelihood and that has an impact on the posterior distribution. If a prior distribution dominates the likelihood, it is clearly an informative prior. These types of distributions must be specified with care in actual practice. On the other hand, the proper use of prior distributions illustrates the power of the Bayesian method: information gathered from the previous study, past experience, or expert opinion can be combined with current information in a natural way. See the "Examples" sections of the GENMOD and PHREG procedure chapters for instructions about constructing informative prior distributions.

Conjugate Priors

A prior is said to be a conjugate prior for a family of distributions if the prior and posterior distributions are from the same family, which means that the form of the posterior has the same distributional form as the prior distribution. For example, if the likelihood is binomial, $y \sim \text{Bin}(n, \theta)$, a conjugate prior on θ is the beta distribution; it follows that the posterior distribution of θ is also a beta distribution. Other commonly used conjugate prior/likelihood combinations include the normal/normal, gamma/Poisson, gamma/gamma, and gamma/beta cases. The development of conjugate priors was partially driven by a desire for computational convenience—conjugacy provides a practical way to obtain the posterior distributions. The Bayesian procedures do not use conjugacy in posterior sampling.

Jeffreys' Prior

A very useful prior is Jeffreys' prior (Jeffreys 1961). It satisfies the local uniformity property: a prior that does not change much over the region in which the likelihood is significant and does not assume large values outside that range. It is based on the Fisher information matrix. Jeffreys' prior is defined as

$$\pi(\theta) \propto |I(\theta)|^{1/2}$$

where $|\ |$ denotes the determinant and $I(\theta)$ is the Fisher information matrix based on the likelihood function $p(\mathbf{y}|\theta)$:

$$I(\theta) = -E\left[\frac{\partial^2 \log p(\mathbf{y}|\theta)}{\partial \theta^2}\right]$$

Jeffreys' prior is locally uniform and hence noninformative. It provides an automated scheme for finding a noninformative prior for any parametric model $p(\mathbf{y}|\theta)$. Another appealing property of Jeffreys' prior is that it is invariant with respect to one-to-one transformations. The invariance property means that if you have a locally uniform prior on θ and $\phi(\theta)$ is a one-to-one function of θ, then $p(\phi(\theta)) = \pi(\theta) \cdot |\phi'(\theta)|^{-1}$ is a locally uniform prior for $\phi(\theta)$. This invariance principle carries through to multidimensional parameters as well. While Jeffreys' prior provides a general recipe for obtaining noninformative priors, it has some shortcomings: the prior is improper for many models, and it can lead to improper posterior in some cases; and the prior can be cumbersome to use in high dimensions. PROC GENMOD calculates Jeffreys' prior

automatically for any generalized linear model. You can set it as your prior density for the coefficient parameters, and it does not lead to improper posteriors. You can construct Jeffreys' prior for a variety of statistical models in the MCMC procedure. See the section "Example 74.4: Logistic Regression Model with Jeffreys' Prior" on page 5799 in Chapter 74, "The MCMC Procedure," for an example. PROC MCMC does not guarantee that the corresponding posterior distribution is proper, and you need to exercise extra caution in this case.

Bayesian Inference

Bayesian inference about θ is primarily based on the posterior distribution of θ. There are various ways in which you can summarize this distribution. For example, you can report your findings through point estimates. You can also use the posterior distribution to construct hypothesis tests or probability statements.

Point Estimation and Estimation Error

Classical methods often report the maximum likelihood estimator (MLE) or the method of moments estimator (MOME) of a parameter. In contrast, Bayesian approaches often use the posterior mean. The definition of the posterior mean is given by

$$E(\theta|\mathbf{y}) = \int \theta \; p(\theta|\mathbf{y}) \; d\theta$$

Other commonly used posterior estimators include the posterior median, defined as

$$\theta\colon P(\theta \geq \text{median}|\mathbf{y}) = P(\theta \leq \text{median}|\mathbf{y}) = \frac{1}{2}$$

and the posterior mode, defined as the value of θ that maximizes $p(\theta|\mathbf{y})$.

The variance of the posterior density (simply referred to as the *posterior variance*) describes the uncertainty in the parameter, which is a random variable in the Bayesian paradigm. A Bayesian analysis typically uses the posterior variance, or the posterior standard deviation, to characterize the dispersion of the parameter. In multidimensional models, covariance or correlation matrices are used.

If you know the distributional form of the posterior density of interest, you can report the exact posterior point estimates. When models become too difficult to analyze analytically, you have to use simulation algorithms, such as the Markov chain Monte Carlo (MCMC) method to obtain posterior estimates (see the section "Markov Chain Monte Carlo Method" on page 129). All of the Bayesian procedures rely on MCMC to obtain all posterior estimates. Using only a finite number of samples, simulations introduce an additional level of uncertainty to the accuracy of the estimates. *Monte Carlo standard error (MCSE)*, which is the standard error of the posterior mean estimate, measures the simulation accuracy. See the section "Standard Error of the Mean Estimate" on page 150 for more information.

The posterior standard deviation and the MCSE are two completely different concepts: the posterior standard deviation describes the uncertainty in the parameter, while the MCSE describes only the uncertainty in the parameter estimate as a result of MCMC simulation. The posterior standard deviation is a function of the sample size in the data set, and the MCSE is a function of the number of iterations in the simulation.

Hypothesis Testing

Suppose you have the following null and alternative hypotheses: H_0 is $\theta \in \Theta_0$ and H_1 is $\theta \in \Theta_0^c$, where Θ_0 is a subset of the parameter space and Θ_0^c is its complement. Using the posterior distribution $\pi(\theta|\mathbf{y})$, you can compute the posterior probabilities $P(\theta \in \Theta_0|\mathbf{y})$ and $P(\theta \in \Theta_0^c|\mathbf{y})$, or the probabilities that H_0 and H_1 are true, respectively. One way to perform a Bayesian hypothesis test is to accept the null hypothesis if $P(\theta \in \Theta_0|\mathbf{y}) \geq P(\theta \in \Theta_0^c|\mathbf{y})$ and vice versa, or to accept the null hypothesis if $P(\theta \in \Theta_0|\mathbf{y})$ is greater than a predefined threshold, such as 0.75, to guard against falsely accepted null distribution.

It is more difficult to carry out a point null hypothesis test in a Bayesian analysis. A point null hypothesis is a test of $H_0: \theta = \theta_0$ versus $H_1: \theta \neq \theta_0$. If the prior distribution $\pi(\theta)$ is a continuous density, then the posterior probability of the null hypothesis being true is 0, and there is no point in carrying out the test. One alternative is to restate the null to be a small interval hypothesis: $\theta \in \Theta_0 = (\theta_0 - a, \theta_0 + a)$, where a is a very small constant. The Bayesian paradigm can deal with an interval hypothesis more easily. Another approach is to give a mixture prior distribution to θ with a positive probability of p_0 on θ_0 and the density $(1 - p_0)\pi(\theta)$ on $\theta \neq \theta_0$. This prior ensures a nonzero posterior probability on θ_0, and you can then make realistic probabilistic comparisons. For more detailed treatment of Bayesian hypothesis testing, see Berger (1985).

Interval Estimation

The Bayesian set estimates are called *credible sets*, which are also known as *credible intervals*. This is analogous to the concept of confidence intervals used in classical statistics. Given a posterior distribution $p(\theta|\mathbf{y})$, A is a credible set for θ if

$$P(\theta \in A|\mathbf{y}) = \int_A p(\theta|\mathbf{y})d\theta$$

For example, you can construct a 95% credible set for θ by finding an interval, A, over which $\int_A p(\theta|\mathbf{y}) = 0.95$.

You can construct credible sets that have equal tails. A $100(1-\alpha)\%$ equal-tail interval corresponds to the $100(\alpha/2)$th and $100(1-\alpha/2)$th percentiles of the posterior distribution. Some statisticians prefer this interval because it is invariant under transformations. Another frequently used Bayesian credible set is called the *highest posterior density* (HPD) interval.

A $100(1-\alpha)\%$ HPD interval is a region that satisfies the following two conditions:

1. The posterior probability of that region is $100(1-\alpha)\%$.
2. The minimum density of any point within that region is equal to or larger than the density of any point outside that region.

The HPD is an interval in which most of the distribution lies. Some statisticians prefer this interval because it is the smallest interval.

One major distinction between Bayesian and classical sets is their interpretation. The Bayesian probability reflects a person's subjective beliefs. Following this approach, a statistician can make the claim that θ is inside a credible interval with measurable probability. This property is appealing because it enables you to

make a direct probability statement about parameters. Many people find this concept to be a more natural way of understanding a probability interval, which is also easier to explain to nonstatisticians. A confidence interval, on the other hand, enables you to make a claim that the interval covers the true parameter. The interpretation reflects the uncertainty in the sampling procedure; a confidence interval of $100(1-\alpha)\%$ asserts that, in the long run, $100(1-\alpha)\%$ of the realized confidence intervals cover the true parameter.

Bayesian Analysis: Advantages and Disadvantages

Bayesian methods and classical methods both have advantages and disadvantages, and there are some similarities. When the sample size is large, Bayesian inference often provides results for parametric models that are very similar to the results produced by frequentist methods. Some advantages to using Bayesian analysis include the following:

- It provides a natural and principled way of combining prior information with data, within a solid decision theoretical framework. You can incorporate past information about a parameter and form a prior distribution for future analysis. When new observations become available, the previous posterior distribution can be used as a prior. All inferences logically follow from Bayes' theorem.
- It provides inferences that are conditional on the data and are exact, without reliance on asymptotic approximation. Small sample inference proceeds in the same manner as if one had a large sample. Bayesian analysis also can estimate any functions of parameters directly, without using the "plug-in" method (a way to estimate functionals by plugging the estimated parameters in the functionals).
- It obeys the likelihood principle. If two distinct sampling designs yield proportional likelihood functions for θ, then all inferences about θ should be identical from these two designs. Classical inference does not in general obey the likelihood principle.
- It provides interpretable answers, such as "the true parameter θ has a probability of 0.95 of falling in a 95% credible interval."
- It provides a convenient setting for a wide range of models, such as hierarchical models and missing data problems. MCMC, along with other numerical methods, makes computations tractable for virtually all parametric models.

There are also disadvantages to using Bayesian analysis:

- It does not tell you how to select a prior. There is no correct way to choose a prior. Bayesian inferences require skills to translate subjective prior beliefs into a mathematically formulated prior. If you do not proceed with caution, you can generate misleading results.
- It can produce posterior distributions that are heavily influenced by the priors. From a practical point of view, it might sometimes be difficult to convince subject matter experts who do not agree with the validity of the chosen prior.
- It often comes with a high computational cost, especially in models with a large number of parameters. In addition, simulations provide slightly different answers unless the same random seed is used. Note that slight variations in simulation results do not contradict the early claim that Bayesian inferences are

exact. The posterior distribution of a parameter is exact, given the likelihood function and the priors, while simulation-based estimates of posterior quantities can vary due to the random number generator used in the procedures.

For more in-depth treatments of the pros and cons of Bayesian analysis, see Berger (1985, Sections 4.1 and 4.12), Berger and Wolpert (1988), Bernardo and Smith (1994, with a new edition coming out), Carlin and Louis (2000, Section 1.4), Robert (2001, Chapter 11), and Wasserman (2004, Section 11.9).

The following sections provide detailed information about the Bayesian methods provided in SAS.

Markov Chain Monte Carlo Method

The Markov chain Monte Carlo (MCMC) method is a general simulation method for sampling from posterior distributions and computing posterior quantities of interest. MCMC methods sample successively from a target distribution. Each sample depends on the previous one, hence the notion of the Markov chain. A Markov chain is a sequence of random variables, $\theta^1, \theta^2, \ldots$, for which the random variable θ^t depends on all previous θs only through its immediate predecessor θ^{t-1}. You can think of a Markov chain applied to sampling as a mechanism that traverses randomly through a target distribution without having any memory of where it has been. Where it moves next is entirely dependent on where it is now.

Monte Carlo, as in Monte Carlo integration, is mainly used to approximate an expectation by using the Markov chain samples. In the simplest version

$$\int_S g(\theta)p(\theta)d\theta \cong \frac{1}{n}\sum_{t=1}^{n} g(\theta^t)$$

where $g(\cdot)$ is a function of interest and θ^t are samples from $p(\theta)$ on its support S. This approximates the expected value of $g(\theta)$. The earliest reference to MCMC simulation occurs in the physics literature. Metropolis and Ulam (1949) and Metropolis et al. (1953) describe what is known as the Metropolis algorithm (see the section "Metropolis and Metropolis-Hastings Algorithms" on page 130). The algorithm can be used to generate sequences of samples from the joint distribution of multiple variables, and it is the foundation of MCMC. Hastings (1970) generalized their work, resulting in the Metropolis-Hastings algorithm. Geman and Geman (1984) analyzed image data by using what is now called Gibbs sampling (see the section "Gibbs Sampler" on page 131). These MCMC methods first appeared in the mainstream statistical literature in Tanner and Wong (1987).

The Markov chain method has been quite successful in modern Bayesian computing. Only in the simplest Bayesian models can you recognize the analytical forms of the posterior distributions and summarize inferences directly. In moderately complex models, posterior densities are too difficult to work with directly. With the MCMC method, it is possible to generate samples from an arbitrary posterior density $p(\theta|\mathbf{y})$ and to use these samples to approximate expectations of quantities of interest. Several other aspects of the Markov chain method also contributed to its success. Most importantly, if the simulation algorithm is implemented correctly, the Markov chain is guaranteed to converge to the target distribution $p(\theta|\mathbf{y})$ under rather broad conditions, regardless of where the chain was initialized. In other words, a Markov chain is able to improve its approximation to the true distribution at each step in the simulation. Furthermore, if the chain is run for a very long time (often required), you can recover $p(\theta|\mathbf{y})$ to any precision. Also, the simulation algorithm is

easily extensible to models with a large number of parameters or high complexity, although the "curse of dimensionality" often causes problems in practice.

Properties of Markov chains are discussed in Feller (1968), Breiman (1968), and Meyn and Tweedie (1993). Ross (1997) and Karlin and Taylor (1975) give a non-measure-theoretic treatment of stochastic processes, including Markov chains. For conditions that govern Markov chain convergence and rates of convergence, see Amit (1991), Applegate, Kannan, and Polson (1990), Chan (1993), Geman and Geman (1984), Liu, Wong, and Kong (1991a, b), Rosenthal (1991a, b), Tierney (1994), and Schervish and Carlin (1992). Besag (1974) describes conditions under which a set of conditional distributions gives a unique joint distribution. Tanner (1993), Gilks, Richardson, and Spiegelhalter (1996), Chen, Shao, and Ibrahim (2000), Liu (2001), Gelman et al. (2004), Robert and Casella (2004), and Congdon (2001, 2003, 2005) provide both theoretical and applied treatments of MCMC methods. You can also see the section "A Bayesian Reading List" on page 153 for a list of books with varying levels of difficulty of treatment of the subject and its application to Bayesian statistics.

Metropolis and Metropolis-Hastings Algorithms

The Metropolis algorithm is named after its inventor, the American physicist and computer scientist Nicholas C. Metropolis. The algorithm is simple but practical, and it can be used to obtain random samples from any arbitrarily complicated target distribution of any dimension that is known up to a normalizing constant.

Suppose you want to obtain T samples from a univariate distribution with probability density function $f(\theta|\mathbf{y})$. Suppose θ^t is the tth sample from f. To use the Metropolis algorithm, you need to have an initial value θ^0 and a symmetric *proposal* density $q(\theta^{t+1}|\theta^t)$. For the $(t+1)$ iteration, the algorithm generates a sample from $q(\cdot|\cdot)$ based on the current sample θ^t, and it makes a decision to either accept or reject the new sample. If the new sample is accepted, the algorithm repeats itself by starting at the new sample. If the new sample is rejected, the algorithm starts at the current point and repeats. The algorithm is self-repeating, so it can be carried out as long as required. In practice, you have to decide the total number of samples needed in advance and stop the sampler after that many iterations have been completed.

Suppose $q(\theta_{\text{new}}|\theta^t)$ is a symmetric distribution. The proposal distribution should be an easy distribution from which to sample, and it must be such that $q(\theta_{\text{new}}|\theta^t) = q(\theta^t|\theta_{\text{new}})$, meaning that the likelihood of jumping to θ_{new} from θ^t is the same as the likelihood of jumping back to θ^t from θ_{new}. The most common choice of the proposal distribution is the normal distribution $N(\theta^t, \sigma)$ with a fixed σ. The Metropolis algorithm can be summarized as follows:

1. Set $t = 0$. Choose a starting point θ^0. This can be an arbitrary point as long as $f(\theta^0|\mathbf{y}) > 0$.
2. Generate a new sample, θ_{new}, by using the proposal distribution $q(\cdot|\theta^t)$.
3. Calculate the following quantity:

$$r = \min\left\{\frac{f(\theta_{\text{new}}|\mathbf{y})}{f(\theta^t|\mathbf{y})}, 1\right\}$$

4. Sample u from the uniform distribution $U(0,1)$.
5. Set $\theta^{t+1} = \theta_{\text{new}}$ if $u < r$; otherwise set $\theta^{t+1} = \theta^t$.
6. Set $t = t + 1$. If $t < T$, the number of desired samples, return to step 2. Otherwise, stop.

Note that the number of iteration keeps increasing regardless of whether a proposed sample is accepted.

This algorithm defines a chain of random variates whose distribution will converge to the desired distribution $f(\theta|\mathbf{y})$, and so from some point forward, the chain of samples is a sample from the distribution of interest. In Markov chain terminology, this distribution is called the *stationary distribution* of the chain, and in Bayesian statistics, it is the posterior distribution of the model parameters. The reason that the Metropolis algorithm works is beyond the scope of this documentation, but you can find more detailed descriptions and proofs in many standard textbooks, including Roberts (1996) and Liu (2001). The random-walk Metropolis algorithm is used in the MCMC procedure.

You are not limited to a symmetric random-walk proposal distribution in establishing a valid sampling algorithm. A more general form, the Metropolis-Hastings (MH) algorithm, was proposed by Hastings (1970). The MH algorithm uses an asymmetric proposal distribution: $q(\theta_{\text{new}}|\theta^t) \neq q(\theta^t|\theta_{\text{new}})$. The difference in its implementation comes in calculating the ratio of densities:

$$r = \min\left\{\frac{f(\theta_{\text{new}}|\mathbf{y})q(\theta^t|\theta_{\text{new}})}{f(\theta^t|\mathbf{y})q(\theta_{\text{new}}|\theta^t)}, 1\right\}$$

Other steps remain the same.

The extension of the Metropolis algorithm to a higher-dimensional $\boldsymbol{\theta}$ is straightforward. Suppose $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)'$ is the parameter vector. To start the Metropolis algorithm, select an initial value for each θ_k and use a multivariate version of proposal distribution $q(\cdot|\cdot)$, such as a multivariate normal distribution, to select a k-dimensional new parameter. Other steps remain the same as those previously described, and this Markov chain eventually converges to the target distribution of $f(\boldsymbol{\theta}|\mathbf{y})$. Chib and Greenberg (1995) provide a useful tutorial on the algorithm.

Gibbs Sampler

The Gibbs sampler, named by Geman and Geman (1984) after the American physicist Josiah W. Gibbs, is a special case of the "Metropolis and Metropolis-Hastings Algorithms" on page 130 in which the proposal distributions exactly match the posterior conditional distributions and proposals are accepted 100% of the time. Gibbs sampling requires you to decompose the joint posterior distribution into full conditional distributions for each parameter in the model and then sample from them. The sampler can be efficient when the parameters are not highly dependent on each other and the full conditional distributions are easy to sample from. Some researchers favor this algorithm because it does not require an instrumental proposal distribution as Metropolis methods do. However, while deriving the conditional distributions can be relatively easy, it is not always possible to find an efficient way to sample from these conditional distributions.

Suppose $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)'$ is the parameter vector, $p(\mathbf{y}|\boldsymbol{\theta})$ is the likelihood, and $\pi(\boldsymbol{\theta})$ is the prior distribution. The full posterior conditional distribution of $\pi(\theta_i|\theta_j, i \neq j, \mathbf{y})$ is proportional to the joint posterior density; that is,

$$\pi(\theta_i|\theta_j, i \neq j, \mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$$

For instance, the one-dimensional conditional distribution of θ_1 given $\theta_j = \theta_j^*, 2 \leq j \leq k$, is computed as the following:

$$\pi(\theta_1|\theta_j = \theta_j^*, 2 \le j \le k, \mathbf{y}) = p(\mathbf{y}|(\boldsymbol{\theta} = (\theta_1, \theta_2^*, \ldots, \theta_k^*)')\pi(\boldsymbol{\theta} = (\theta_1, \theta_2^*, \ldots, \theta_k^*)')$$

The Gibbs sampler works as follows:

1. Set $t = 0$, and choose an arbitrary initial value of $\boldsymbol{\theta}^{(0)} = \{\theta_1^{(0)}, \ldots, \theta_k^{(0)}\}$.
2. Generate each component of $\boldsymbol{\theta}$ as follows:
   - draw $\theta_1^{(t+1)}$ from $\pi(\theta_1|\theta_2^{(t)}, \ldots, \theta_k^{(t)}, \mathbf{y})$
   - draw $\theta_2^{(t+1)}$ from $\pi(\theta_2|\theta_1^{(t+1)}, \theta_3^{(t)}, \ldots, \theta_k^{(t)}, \mathbf{y})$
   - ...
   - draw $\theta_k^{(t+1)}$ from $\pi(\theta_k|\theta_1^{(t+1)}, \ldots, \theta_{k-1}^{(t+1)}, \mathbf{y})$
3. Set $t = t + 1$. If $t < T$, the number of desired samples, return to step 2. Otherwise, stop.

The name "Gibbs" was introduced by Geman and Geman (1984). Gelfand et al. (1990) first used Gibbs sampling to solve problems in Bayesian inference. See Casella and George (1992) for a tutorial on the sampler. The GENMOD, LIFEREG, and PHREG procedures update parameters using the Gibbs sampler.

Adaptive Rejection Sampling Algorithm

The GENMOD, LIFEREG, and PHREG procedures use the adaptive rejection sampling (ARS) algorithm to sample parameters sequentially from their univariate full conditional distributions. The ARS algorithm is a rejection algorithm that was originally proposed by Gilks and Wild (1992). Given a log-concave density (the log of the density is concave), you can construct an envelope for the density by using linear segments. You then use the linear segment envelope as a proposal density (it becomes a piecewise exponential density on the original scale and is easy to generate samplers from) in the rejection sampling.

The log-concavity condition is met in some of the models that are fit by the procedures. For example, the posterior densities for the regression parameters in the generalized linear models are log-concave under flat priors. When this condition fails, the ARS algorithm calls for an additional Metropolis-Hastings step (Gilks, Best, and Tan 1995), and the modified algorithm becomes the adaptive rejection Metropolis sampling (ARMS) algorithm. The GENMOD, LIFEREG, and PHREG procedures can recognize whether a model is log-concave and select the appropriate sampler for the problem at hand.

Although samples obtained from the ARMS algorithm often exhibit less dependence with lower autocorrelations, the algorithm could have a high computational cost because it requires repeated evaluations of the objective function (usually five to seven repetitions) at each iteration for each univariate parameter.[1]

Implementation the ARMS algorithm in the GENMOD, LIFEREG, and PHREG procedures is based on code that is provided by Walter R. Gilks, University of Leeds (Gilks 2003). For a detailed description and explanation of the algorithm, see Gilks and Wild (1992); Gilks, Best, and Tan (1995).

[1] The extension to the multivariate ARMS algorithm is possible in theory but problematic in practice because the computational cost associated with constructing a multidimensional hyperbola envelop is often prohibitive.

Slice Sampler

The slice sampler (Neal 2003), like the ARMS algorithm, is a general algorithm that can be used to sample parameters from their target distribution. As with the ARMS algorithm, the only requirement of the slice sampler is the ability to evaluate the objective function (the unnormalized conditional distribution in a Gibbs step, for example) at a given parameter value. In theory, you can draw a random number from any given distribution as long as you can first obtain a random number uniformly under the curve of that distribution. Treat the area under the curve of $p(\theta)$ as a two-dimensional space that is defined by the θ-axis and the Y-axis, the latter being the axis for the density function. You draw uniformly in that area, obtain a two-dimensional vector of (θ_i, y_i), ignore the y_i, and keep the θ_i. The θ_i's are distributed according to the right density.

To solve the problem of sampling uniformly under the curve, Neal (2003) proposed the idea of slices (hence the name of the sampler), which can be explained as follows:

1. Start the algorithm at θ_0.
2. Calculate the objective function $p(\theta_0)$ and draw a line between $y = 0$ and $y = p(\theta_0)$, which defines a vertical slice. You draw a uniform number, y_1, on this slice, between $(0, p(\theta_0))$.
3. Draw a horizontal line at y_1 and find the two points where the line intercepts with the curve, (L_1, R_1). These two points define a horizontal slice. Draw a uniform number, x_1, on this slice, between (L_1, R_1).
4. Repeat steps 2 and 3 many times.

The challenging part of the algorithm is finding the horizontal slice (L_i, R_i) at each iteration. The closed form expressions of $p_L^{-1}(y_i)$ and $p_R^{-1}(y_i)$ are virtually impossible to obtain analytically in most problems. Neal (2003) proved that although exact solutions would be nice, devising a search algorithm that finds portions of this horizontal slice is sufficient for the sampler to work. The search algorithm is based on the rejection method to expand and contract, when needed.

The sampler is implemented as an optional algorithm in the MCMC procedure, where you can use it to draw either model parameters or random-effects parameters. As with the ARMS algorithm, only the univariate version of the slice sampler is implemented. The slice sampler requires repeated evaluations of the objective function; this happens in the search algorithm to identify each horizontal slice at every iteration. Hence, the computational cost could be high if each evaluation of the objective function requires one pass through the entire data set.

Independence Sampler

Another type of Metropolis algorithm is the "independence" sampler. It is called the independence sampler because the proposal distribution in the algorithm does not depend on the current point as it does with the random-walk Metropolis algorithm. For this sampler to work well, you want to have a proposal distribution that mimics the target distribution and have the acceptance rate be as high as possible.

1. Set $t = 0$. Choose a starting point $\theta^{\mathbf{0}}$. This can be an arbitrary point as long as $f(\theta^{\mathbf{0}}|\mathbf{y}) > 0$.
2. Generate a new sample, θ_{new}, by using the proposal distribution $q(\cdot)$. The proposal distribution does not depend on the current value of θ^t.

3. Calculate the following quantity:

$$r = \min\left\{\frac{f(\theta_{\text{new}}|\mathbf{y})/q(\theta_{\text{new}})}{f(\theta^t|\mathbf{y})/q(\theta^t)}, 1\right\}$$

4. Sample u from the uniform distribution $U(0, 1)$.
5. Set $\theta^{t+1} = \theta_{\text{new}}$ if $u < r$; otherwise set $\theta^{t+1} = \theta^t$.
6. Set $t = t + 1$. If $t < T$, the number of desired samples, return to step 2. Otherwise, stop.

A good proposal density should have thicker tails than those of the target distribution. This requirement sometimes can be difficult to satisfy especially in cases where you do not know what the target posterior distributions are like. In addition, this sampler does not produce independent samples as the name seems to imply, and sample chains from independence samplers can get stuck in the tails of the posterior distribution if the proposal distribution is not chosen carefully. The MCMC procedure uses the independence sampler.

Gamerman Algorithm

The Gamerman algorithm, named after the inventor Dani Gamerman, is a special case of the Metropolis algorithm (see the section "Metropolis and Metropolis-Hastings Algorithms" on page 130) in which the proposal distribution is derived from one iteration of the iterative weighted least squares (IWLS) algorithm. As the name suggests, a weighted least squares algorithm is carried out inside an iteration loop. For each iteration, a set of weights for the observations is used in the least squares fit. The weights are constructed by applying a weight function to the current residuals. The proposal distribution uses the current iteration's values of the parameters to form the proposal distribution from which to generate a proposed random value (Gamerman 1997).

The multivariate sampling algorithm is simple but practical, and can be used to obtain random samples from the posterior distribution of the regression parameters in a generalized linear model (GLM). See "Generalized Linear Regression" on page 75 in Chapter 4, "Introduction to Regression Procedures," for further details on generalized linear regression models. See McCullagh and Nelder (1989) for a discussion of transformed observations and diagonal matrix of weights pertaining to IWLS.

The GENMOD procedure uses the Gamerman algorithm to sample parameters from their multivariate posterior conditional distributions. For a detailed description and explanation of the algorithm, see Gamerman (1997).

Hamiltonian Monte Carlo Sampler

The Hamiltonian Monte Carlo (HMC) algorithm, also known as the hybrid Monte Carlo algorithm, is a version of the Metropolis algorithm that uses gradients information and auxiliary momentum variables to draw samples from the posterior distribution (Neal 2011). The algorithm uses Hamiltonian dynamics to enable distant proposals in the Metropolis algorithm, making it efficient in many scenarios. The HMC algorithm is applicable only to continuous parameters.

HMC translates the target density function to a potential energy function and adds an auxiliary momentum variable $\mathbf{r}$ for each model parameter $\boldsymbol{\theta}$. The resulting joint density has the form

$$p(\boldsymbol{\theta}, \mathbf{r}) \propto p(\boldsymbol{\theta}) \exp\left(-\frac{1}{2}\mathbf{r}'\mathbf{r}\right)$$

where $p(\boldsymbol{\theta})$ is the posterior of the parameters $\boldsymbol{\theta}$ (up to a normalizing constant). HMC draws from the joint space of $(\boldsymbol{\theta}, \mathbf{r})$, throws away $\mathbf{r}$, and retains $\boldsymbol{\theta}$ as samples from $p(\boldsymbol{\theta})$. The algorithm uses the idea of Hamiltonian dynamics in preserving the total energy of a physics system, in which $\boldsymbol{\theta}$ is part of the potential energy function and $\mathbf{r}$ is part of the kinetic energy (velocity). As the velocity changes, the potential energy changes accordingly, leading to the movements in the parameter space.

At each iteration, HMC first generates the momentum variables $\mathbf{r}$, usually from standard normal distributions, that are independent of $\boldsymbol{\theta}$. Then the algorithm follows with a Metropolis update that includes many steps along a trajectory while maintaining the total energy of the system. One of the most frequently used methods in moving along this trajectory is the leapfrog method, which involves L steps with a step size ϵ,

$$\begin{aligned}
\mathbf{r}^{t+\epsilon/2} &= \mathbf{r}^{t} + (\epsilon/2) \bigtriangledown_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}^{t}) \\
\boldsymbol{\theta}^{t+\epsilon} &= \boldsymbol{\theta}^{t} + \epsilon \mathbf{r}^{t+\epsilon/2} \\
\mathbf{r}^{t+\epsilon} &= \mathbf{r}^{t+\epsilon/2} + (\epsilon/2) \bigtriangledown_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}^{t+\epsilon})
\end{aligned}$$

where $\bigtriangledown_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta})$ is the gradient of the log-posterior with respect to $\boldsymbol{\theta}$. After L steps, the proposed state $(\boldsymbol{\theta}^*, \mathbf{r}^*)$ is accepted as the next state of the Markov chain with probability $\min\{1, p(\boldsymbol{\theta}^*, \mathbf{r}^*)/p(\boldsymbol{\theta}, \mathbf{r})\}$.

Although HMC can lead to rapid convergence, it also heavily relies on two requirements: the gradient calculation of the logarithm of the posterior density and carefully selected tuning parameters, in step size ϵ and number of steps L. Step sizes that are too large or too small can lead to overly low or overly high acceptance rates, both of which affect the convergence of the Markov chain. Large L leads to large trajectory length ($\epsilon \cdot L$), which can move the parameters back to their original positions. Small L limits the movement of the chain. To tune ϵ and L, you usually want to run a few trials, starting with a small number of L and an ϵ with a value around 0.1. See Neal (2011) for advice on tuning these parameters.

An example of adaptive HMC with automatic tuning of ϵ and L is the No-U-Turn Sampler (NUTS; Hoffman and Gelman 2014). The NUTS algorithm uses a doubling process to build a binary tree whose leaf nodes correspond to the states of the parameters and momentum variables. The initial tree has a single node with no heights ($j = 0$). The doubling process expands the tree either left or right in a binary fashion, and in each direction, the algorithm takes 2^j leapfrog steps of size ϵ. Obviously, as the height of the tree (j) increases, the computational cost increases dramatically. The tree expands until one sampling trajectory makes a U-turn and starts to revisit parameter space that has been already explored. The NUTS algorithm tunes ϵ so that the actual acceptance rate during the doubling process is close to a predetermined target acceptance probability δ (usually set to 0.6 or higher). When the tuning stage ends, the NUTS algorithm proceeds to the main sampling stage and starts to draw posterior samples that have a fixed ϵ value. Increasing the targeted acceptance probability δ can often improve mixing, but it can also slow down the process significantly. For more information about the NUTS algorithm and its efficiency, see Hoffman and Gelman (2014).

Burn-In, Thinning, and Markov Chain Samples

Burn-in refers to the practice of discarding an initial portion of a Markov chain sample so that the effect of initial values on the posterior inference is minimized. For example, suppose the target distribution is $N(0, 1)$ and the Markov chain was started at the value 10^6. The chain might quickly travel to regions around 0 in a few iterations. However, including samples around the value 10^6 in the posterior mean calculation can produce substantial bias in the mean estimate. In theory, if the Markov chain is run for an infinite amount of time, the effect of the initial values decreases to zero. In practice, you do not have the luxury of infinite samples. In practice, you assume that after t iterations, the chain has reached its target distribution and you

can throw away the early portion and use the good samples for posterior inference. The value of t is the burn-in number.

With some models you might experience poor mixing (or slow convergence) of the Markov chain. This can happen, for example, when parameters are highly correlated with each other. Poor mixing means that the Markov chain slowly traverses the parameter space (see the section "Visual Analysis via Trace Plots" on page 137 for examples of poorly mixed chains) and the chain has high dependence. High sample autocorrelation can result in biased Monte Carlo standard errors. A common strategy is to *thin* the Markov chain in order to reduce sample autocorrelations. You thin a chain by keeping every kth simulated draw from each sequence. You can safely use a thinned Markov chain for posterior inference as long as the chain converges. It is important to note that thinning a Markov chain can be wasteful because you are throwing away a $\frac{k-1}{k}$ fraction of all the posterior samples generated. MacEachern and Berliner (1994) show that you always get more precise posterior estimates if the entire Markov chain is used. However, other factors, such as computer storage or plotting time, might prevent you from keeping all samples.

To use the BCHOICE, FMM, GENMOD, LIFEREG, MCMC, and PHREG procedures, you need to determine the total number of samples to keep ahead of time. This number is not obvious and often depends on the type of inference you want to make. Mean estimates do not require nearly as many samples as small-tail percentile estimates. In most applications, you might find that keeping a few thousand iterations is sufficient for reasonably accurate posterior inference. In all four procedures, the relationship between the number of iterations requested, the number of iterations kept, and the amount of thinning is as follows:

$$\text{kept} = \left[\frac{\text{requested}}{\text{thinning}}\right]$$

where [ ] is the rounding operator.

Assessing Markov Chain Convergence

Simulation-based Bayesian inference requires using simulated draws to summarize the posterior distribution or calculate any relevant quantities of interest. You need to treat the simulation draws with care. There are usually two issues. First, you have to decide whether the Markov chain has reached its stationary, or the desired posterior, distribution. Second, you have to determine the number of iterations to keep after the Markov chain has reached stationarity. Convergence diagnostics help to resolve these issues. Note that many diagnostic tools are designed to verify a necessary but not sufficient condition for convergence. There are no conclusive tests that can tell you when the Markov chain has converged to its stationary distribution. You should proceed with caution. Also, note that you should check the convergence of *all* parameters, and not just those of interest, before proceeding to make any inference. With some models, certain parameters can appear to have very good convergence behavior, but that could be misleading due to the slow convergence of other parameters. If some of the parameters have bad mixing, you cannot get accurate posterior inference for parameters that appear to have good mixing. See Cowles and Carlin (1996) and Brooks and Roberts (1998) for discussions about convergence diagnostics.

Visual Analysis via Trace Plots

Trace plots of samples versus the simulation index can be very useful in assessing convergence. The trace tells you if the chain has not yet converged to its stationary distribution—that is, if it needs a longer burn-in period. A trace can also tell you whether the chain is mixing well. A chain might have reached stationarity if the distribution of points is not changing as the chain progresses. The aspects of stationarity that are most recognizable from a trace plot are a relatively constant mean and variance. A chain that mixes well traverses its posterior space rapidly, and it can jump from one remote region of the posterior to another in relatively few steps. Figure 7.1 through Figure 7.4 display some typical features that you might see in trace plots. The traces are for a parameter called γ.

Figure 7.1 Essentially Perfect Trace for γ

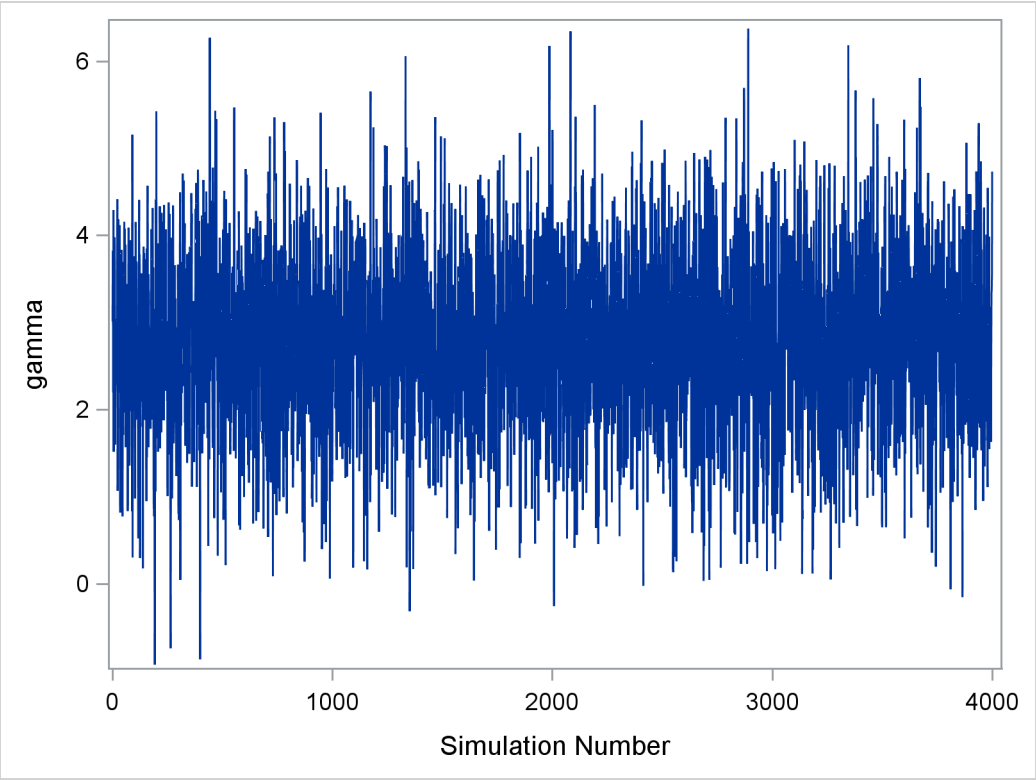


Figure 7.1 displays a "perfect" trace plot. Note that the center of the chain appears to be around the value 3, with very small fluctuations. This indicates that the chain could have reached the right distribution. The chain is mixing well; it is exploring the distribution by traversing to areas where its density is very low. You can conclude that the mixing is quite good here.

Figure 7.2 Initial Samples of γ Need to be Discarded

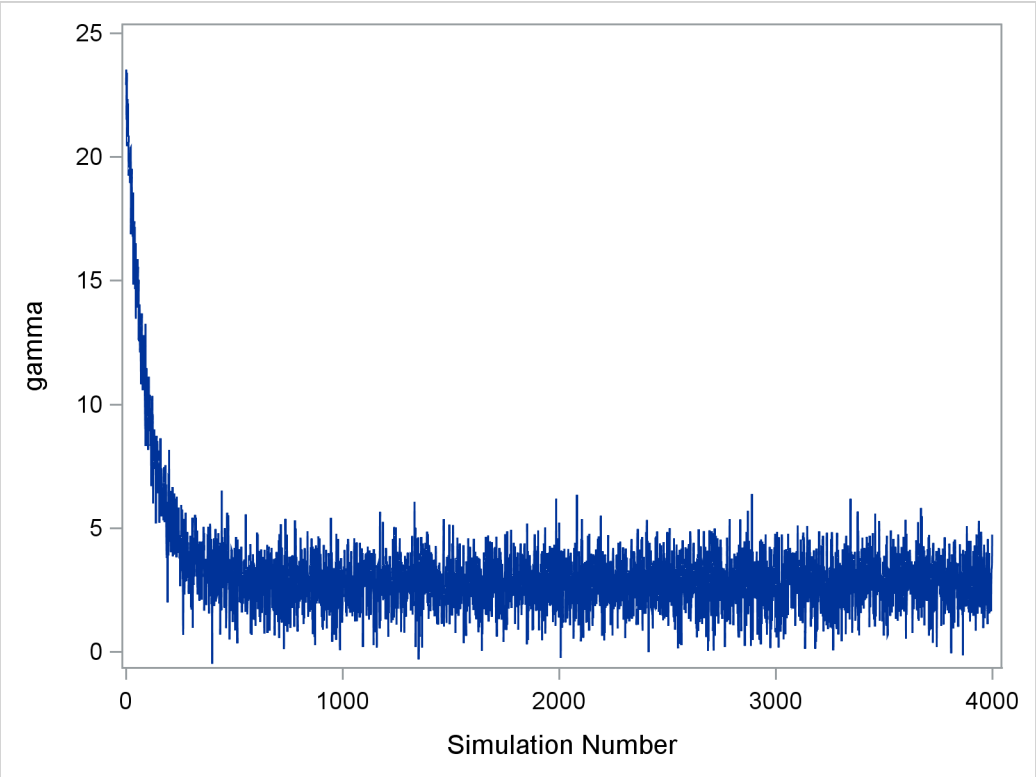


Figure 7.2 displays a trace plot for a chain that starts at a very remote initial value and makes its way to the targeting distribution. The first few hundred observations should be discarded. This chain appears to be mixing very well locally. It travels relatively quickly to the target distribution, reaching it in a few hundred iterations. If you have a chain that looks like this, you would want to increase the burn-in sample size. If you need to use this sample to make inferences, you would want to use only the samples toward the end of the chain.

Figure 7.3 Marginal Mixing for γ

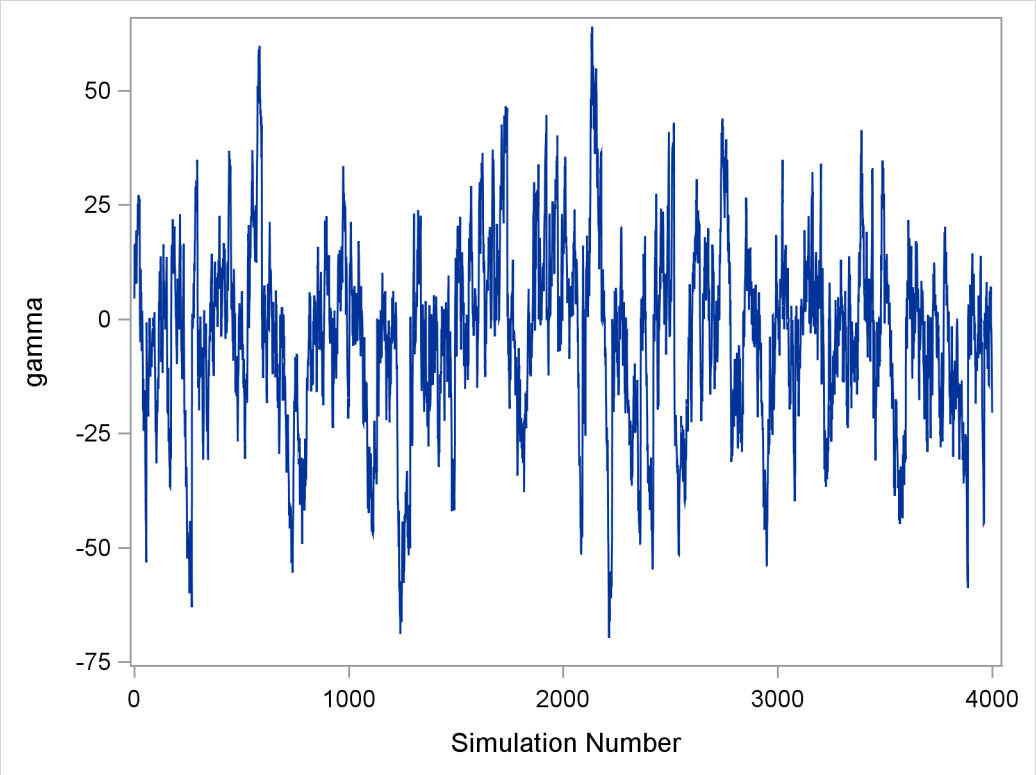
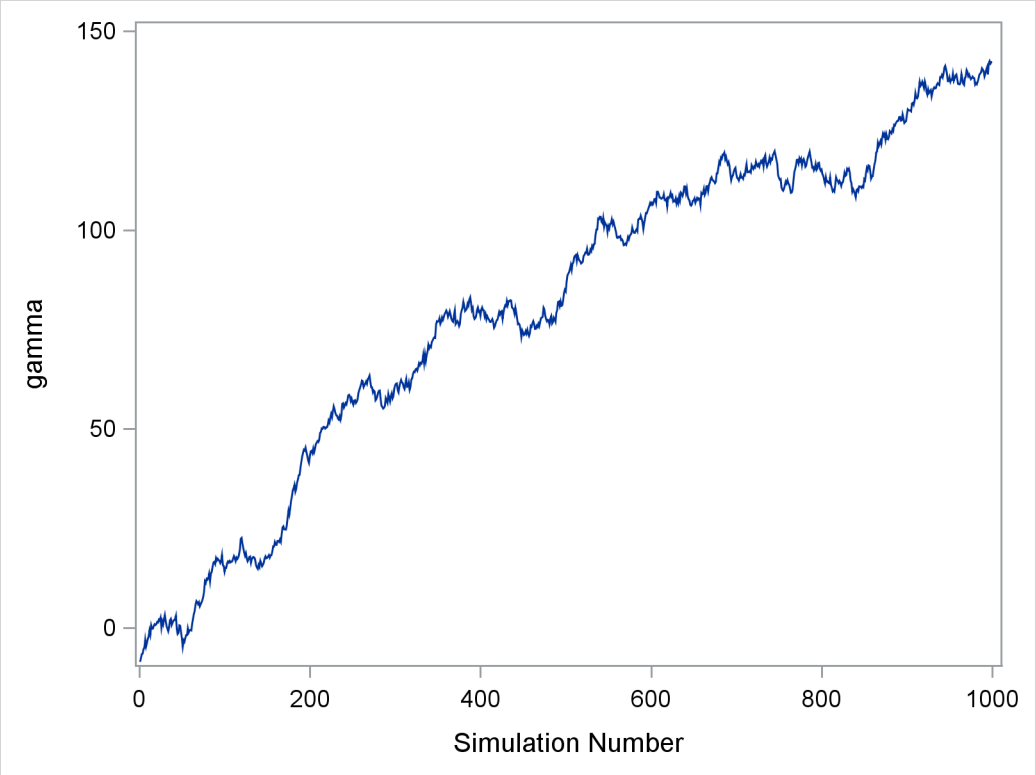


Figure 7.3 demonstrates marginal mixing. The chain is taking only small steps and does not traverse its distribution quickly. This type of trace plot is typically associated with high autocorrelation among the samples. To obtain a few thousand independent samples, you need to run the chain for much longer.

Figure 7.4 Bad Mixing, Nonconvergence of γ



The trace plot shown in Figure 7.4 depicts a chain with serious problems. It is mixing very slowly, and it offers no evidence of convergence. You would want to try to improve the mixing of this chain. For example, you might consider reparameterizing your model on the log scale. Run the Markov chain for a long time to see where it goes. This type of chain is entirely unsuitable for making parameter inferences.

Statistical Diagnostic Tests

The Bayesian procedures include several statistical diagnostic tests that can help you assess Markov chain convergence. For a detailed description of each of the diagnostic tests, see the following subsections. Table 7.1 provides a summary of the diagnostic tests and their interpretations.

Table 7.1 Convergence Diagnostic Tests Available in the Bayesian Procedures

Name	Description	Interpretation of the Test
Gelman-Rubin	Uses parallel chains with dispersed initial values to test whether they all converge to the same target distribution. Failure could indicate the presence of a multi-mode posterior distribution (different chains converge to different local modes) or the need to run a longer chain (burn-in is yet to be completed).	One-sided test based on a variance ratio test statistic. Large $\widehat{\mathrm{R}}_c$ values indicate rejection.
Geweke	Tests whether the mean estimates have converged by comparing means from the early and latter part of the Markov chain.	Two-sided test based on a z-score statistic. Large absolute z values indicate rejection.
Heidelberger-Welch (stationarity test)	Tests whether the Markov chain is a covariance (or weakly) stationary process. Failure could indicate that a longer Markov chain is needed.	One-sided test based on a Cramer–von Mises statistic. Small p-values indicate rejection.
Heidelberger-Welch (half-width test)	Reports whether the sample size is adequate to meet the required accuracy for the mean estimate. Failure could indicate that a longer Markov chain is needed.	If a relative half-width statistic is greater than a predetermined accuracy measure, this indicates rejection.
Raftery-Lewis	Evaluates the accuracy of the estimated (desired) percentiles by reporting the number of samples needed to reach the desired accuracy of the percentiles. Failure could indicate that a longer Markov chain is needed.	If the total samples needed are fewer than the Markov chain sample, this indicates rejection.
autocorrelation	Measures dependency among Markov chain samples.	High correlations between long lags indicate poor mixing.
effective sample size	Relates to autocorrelation; measures mixing of the Markov chain.	Large discrepancy between the effective sample size and the simulation sample size indicates poor mixing.

Gelman and Rubin Diagnostics

Gelman and Rubin diagnostics (Gelman and Rubin 1992; Brooks and Gelman 1997) are based on analyzing multiple simulated MCMC chains by comparing the variances within each chain and the variance between chains. Large deviation between these two variances indicates nonconvergence.

Define $\{\theta^t\}$, where $t = 1, \ldots, n$, to be the collection of a single Markov chain output. The parameter θ^t is the tth sample of the Markov chain. For notational simplicity, θ is assumed to be single dimensional in this section.

Suppose you have M parallel MCMC chains that were initialized from various parts of the target distribution. Each chain is of length n (after discarding the burn-in). For each θ^t, the simulations are labeled as θ^t_m, where $t = 1, \ldots, n$ and $m = 1, \ldots, M$. The between-chain variance B and the within-chain variance W are calculated as

$$
\begin{aligned}
B &= \frac{n}{M-1}\sum_{m=1}^{M}(\bar{\theta}^{\cdot}_m - \bar{\theta}^{\cdot}_{\cdot})^2, \quad \text{where } \bar{\theta}^{\cdot}_m = \frac{1}{n}\sum_{t=1}^{n}\theta^t_m, \quad \bar{\theta}^{\cdot}_{\cdot} = \frac{1}{M}\sum_{m=1}^{M}\bar{\theta}^{\cdot}_m \\
W &= \frac{1}{M}\sum_{m=1}^{M}s^2_m, \quad \text{where } s^2_m = \frac{1}{n-1}\sum_{t=1}^{n}(\theta^t_m - \bar{\theta}^{\cdot}_m)^2
\end{aligned}
$$

The posterior marginal variance, $\mathrm{var}(\theta|\mathbf{y})$, is a weighted average of W and B. The estimate of the variance is

$$
\widehat{V} = \frac{n-1}{n}W + \frac{M+1}{nM}B
$$

If all M chains have reached the target distribution, this posterior variance estimate should be very close to the within-chain variance W. Therefore, you would expect to see the ratio $\widehat{V}/W$ be close to 1. The square root of this ratio is referred to as the *potential scale reduction factor* (PSRF). A large PSRF indicates that the between-chain variance is substantially greater than the within-chain variance, so that longer simulation is needed. If the PSRF is close to 1, you can conclude that each of the M chains has stabilized, and they are likely to have reached the target distribution.

A refined version of PSRF is calculated, as suggested by Brooks and Gelman (1997), as

$$
\widehat{\mathrm{R}}_c = \sqrt{\frac{\hat{d}+3}{\hat{d}+1}\cdot\frac{\widehat{V}}{W}} = \sqrt{\frac{\hat{d}+3}{\hat{d}+1}\left(\frac{n-1}{n} + \frac{M+1}{nM}\frac{B}{W}\right)}
$$

where

$$
\hat{d} = \frac{2\widehat{V}^2}{\widehat{\mathrm{Var}}(\widehat{V})}
$$

and

$$\begin{aligned}\widehat{\mathrm{Var}}(\widehat{V}) &= \left(\frac{n-1}{n}\right)^2 \frac{1}{M}\widehat{\mathrm{Var}}(s_m^2) + \left(\frac{M+1}{nM}\right)^2 \frac{2}{M-1}B^2 \\ &\quad + 2\frac{(M+1)(n-1)}{n^2 M}\frac{n}{M}\left(\widehat{\mathrm{cov}}(s_m^2, (\bar{\theta}_m^{\cdot})^2) - 2\bar{\theta}^{\cdot}\widehat{\mathrm{cov}}(s_m^2, \bar{\theta}_m^{\cdot})\right)\end{aligned}$$

All the Bayesian procedures also produce an upper $100(1-\alpha/2)\%$ confidence limit of $\widehat{\mathrm{R}}_c$. Gelman and Rubin (1992) showed that the ratio B/W in $\widehat{\mathrm{R}}_c$ has an F distribution with degrees of freedom $M-1$ and $2W^2 M/\widehat{\mathrm{Var}}(s_m^2)$. Because you are concerned only if the scale is large, not small, only the upper $100(1-\alpha/2)\%$ confidence limit is reported. This is written as

$$\sqrt{\left(\frac{n-1}{n} + \frac{M+1}{nM}\cdot F_{1-\alpha/2}\left(M-1, \frac{2W^2}{\widehat{\mathrm{Var}}(s_m^2)/M}\right)\right)\cdot\frac{\hat{d}+3}{\hat{d}+1}}$$

In the Bayesian procedures, you can specify the number of chains that you want to run. Typically three chains are sufficient. The first chain is used for posterior inference, such as mean and standard deviation; the other $M-1$ chains are used for computing the diagnostics and are discarded afterward. This test can be computationally costly, because it prolongs the simulation M-fold.

It is best to choose different initial values for all M chains. The initial values should be as dispersed from each other as possible so that the Markov chains can fully explore different parts of the distribution before they converge to the target. Similar initial values can be risky because all of the chains can get stuck in a local maximum; that is something this convergence test cannot detect. If you do not supply initial values for all the different chains, the procedures generate them for you.

Geweke Diagnostics

The Geweke test (Geweke 1992) compares values in the early part of the Markov chain to those in the latter part of the chain in order to detect failure of convergence. The statistic is constructed as follows. Two subsequences of the Markov chain $\{\theta^t\}$ are taken out, with $\{\theta_1^t : t = 1, \ldots, n_1\}$ and $\{\theta_2^t : t = n_a, \ldots, n\}$, where $1 < n_1 < n_a < n$. Let $n_2 = n - n_a + 1$, and define

$$\bar{\theta}_1 = \frac{1}{n_1}\sum_{t=1}^{n_1}\theta^t \quad \text{and} \quad \bar{\theta}_2 = \frac{1}{n_2}\sum_{t=n_a}^{n}\theta^t$$

Let $\hat{s}_1(0)$ and $\hat{s}_2(0)$ denote consistent spectral density estimates at zero frequency (see the subsection "Spectral Density Estimate at Zero Frequency" on page 144 for estimation details) for the two MCMC chains, respectively. If the ratios n_1/n and n_2/n are fixed, $(n_1+n_2)/n < 1$, and the chain is stationary, then the following statistic converges to a standard normal distribution as $n \to \infty$:

$$Z_n = \frac{\bar{\theta}_1 - \bar{\theta}_2}{\sqrt{\frac{\hat{s}_1(0)}{n_1} + \frac{\hat{s}_2(0)}{n_2}}}$$

This is a two-sided test, and large absolute z-scores indicate rejection.

Spectral Density Estimate at Zero Frequency

For one sequence of the Markov chain $\{\theta_t\}$, the relationship between the h-lag covariance sequence of a time series and the spectral density, f, is

$$s_h = \frac{1}{2\pi}\int_{-\pi}^{\pi} \exp(\mathrm{i}\omega h) f(\omega) d\omega$$

where i indicates that ωh is the complex argument. Inverting this Fourier integral,

$$f(\omega) = \sum_{h=-\infty}^{\infty} s_h \exp(-\mathrm{i}\omega h) = s_0 \left(1 + 2\sum_{h=1}^{\infty} \rho_h \cos(\omega h)\right)$$

It follows that

$$f(0) = \sigma^2 \left(1 + 2\sum_{h=1}^{\infty} \rho_h\right)$$

which gives an autocorrelation adjusted estimate of the variance. In this equation, σ^2 is the naive variance estimate of the sequence $\{\theta_t\}$ and ρ_h is the lag h autocorrelation. Due to obvious computational difficulties, such as calculation of autocorrelation at infinity, you cannot effectively estimate $f(0)$ by using the preceding formula. The usual route is to first obtain the *periodogram* $p(\omega)$ of the sequence, and then estimate $f(0)$ by smoothing the estimated periodogram. The periodogram is defined to be

$$p(\omega) = \frac{1}{n}\left[\left(\sum_{t=1}^{n} \theta_t \sin(\omega t)\right)^2 + \left(\sum_{t=1}^{n} \theta_t \cos(\omega t)\right)^2\right]$$

The procedures use the following way to estimate $\hat{f}(0)$ from p (Heidelberger and Welch 1981). In $p(\omega)$, let $\omega = \omega_k = 2\pi k/n$ and $k = 1, \ldots, [\frac{n}{2}]$.[2] A smooth spectral density in the domain of $(0, \pi]$ is obtained by fitting a gamma model with the log link function, using $p(\omega_k)$ as response and $x_1(\omega_k) = \sqrt{3}(4\omega_k/(2\pi) - 1)$ as the only regressor. The predicted value $\hat{f}(0)$ is given by

$$\hat{f}(0) = \exp(\hat{\beta}_0 - \sqrt{3}\hat{\beta}_1)$$

where $\hat{\beta}_0$ and $\hat{\beta}_1$ are the estimates of the intercept and slope parameters, respectively.

[2] This is equivalent to the fast Fourier transformation of the original time series θ_t.

Heidelberger and Welch Diagnostics

The Heidelberger and Welch test (Heidelberger and Welch 1981, 1983) consists of two parts: a stationary portion test and a half-width test. The stationarity test assesses the stationarity of a Markov chain by testing the hypothesis that the chain comes from a covariance stationary process. The half-width test checks whether the Markov chain sample size is adequate to estimate the mean values accurately.

Given $\{\theta^t\}$, set $S_0 = 0$, $S_n = \sum_{t=1}^{n} \theta^t$, and $\bar{\theta} = (1/n) \sum_{t=1}^{n} \theta^t$. You can construct the following sequence with s coordinates on values from $\frac{1}{n}, \frac{2}{n}, \ldots, 1$:

$$B_n(s) = (S_{[ns]} - [ns]\bar{\theta})/(n\hat{p}(0))^{1/2}$$

where [ ] is the rounding operator, and $\hat{p}(0)$ is an estimate of the spectral density at zero frequency that uses the second half of the sequence (see the section "Spectral Density Estimate at Zero Frequency" on page 144 for estimation details). For large n, B_n converges in distribution to a Brownian bridge (Billingsley 1986). So you can construct a test statistic by using B_n. The statistic used in these procedures is the Cramer–von Mises statistic[3]; that is $\int_0^1 B_n(s)^2 ds = \mathrm{CVM}(B_n)$. As $n \to \infty$, the statistic converges in distribution to a standard Cramer–von Mises distribution. The integral $\int_0^1 B_n(s)^2 ds$ is numerically approximated using Simpson's rule.

Let $y_i = B_n(s)^2$, where $s = 0, \frac{1}{n}, \ldots, \frac{n-1}{n}, 1$, and $i = ns = 0, 1, \ldots, n$. If n is even, let $m = n/2$; otherwise, let $m = (n-1)/2$. The Simpson's approximation to the integral is

$$\int_0^1 B_n(s)^2 ds \approx \frac{1}{3n} [y_0 + 4(y_1 + \cdots + y_{2m-1}) + 2(y_2 + \cdots + y_{2m-2}) + y_{2m}]$$

Note that Simpson's rule requires an even number of intervals. When n is odd, y_n is set to be 0 and the value does not contribute to the approximation.

This test can be performed repeatedly on the same chain, and it helps you identify a time t when the chain has reached stationarity. The whole chain, $\{\theta^t\}$, is first used to construct the Cramer–von Mises statistic. If it passes the test, you can conclude that the entire chain is stationary. If it fails the test, you drop the initial 10% of the chain and redo the test by using the remaining 90%. This process is repeated until either a time t is selected or it reaches a point where there are not enough data remaining to construct a confidence interval (the cutoff proportion is set to be 50%).

The part of the chain that is deemed stationary is put through a half-width test, which reports whether the sample size is adequate to meet certain accuracy requirements for the mean estimates. Running the simulation less than this length of time would not meet the requirement, while running it longer would not provide any additional information that is needed. The statistic calculated here is the *relative half-width* (RHW) of the confidence interval. The RHW for a confidence interval of level $1 - \alpha$ is

$$\mathrm{RHW} = \frac{z_{(1-\alpha/2)} \cdot (\hat{s}_n/n)^{1/2}}{\hat{\theta}}$$

[3] The von Mises distribution was first introduced by Von Mises (1918). The density function is $p(\theta|\mu\kappa) \sim M(\mu, \kappa) = [2\pi I_0(\kappa)]^{-1} \exp(\kappa \cos(\theta - \mu))$ $(0 \le \theta \le 2\pi)$, where the function $I_0(\kappa)$ is the modified Bessel function of the first kind and order zero, defined by $I_0(\kappa) = (2\pi)^{-1} \int_0^{2\pi} \exp(\kappa \cos(\theta - \mu)) d\theta$.

where $z_{(1-\alpha/2)}$ is the z-score of the $100(1-\alpha/2)$th percentile (for example, $z_{(1-\alpha/2)} = 1.96$ if $\alpha = 0.05$), $\hat{s}_n$ is the variance of the chain estimated using the spectral density method (see explanation in the section "Spectral Density Estimate at Zero Frequency" on page 144), n is the length, and $\hat{\theta}$ is the estimated mean. The RHW quantifies accuracy of the $1-\alpha$ level confidence interval of the mean estimate by measuring the ratio between the sample standard error of the mean and the mean itself. In other words, you can stop the Markov chain if the variability of the mean stabilizes with respect to the mean. An implicit assumption is that large means are often accompanied by large variances. If this assumption is not met, then this test can produce false rejections (such as a small mean around 0 and large standard deviation) or false acceptance (such as a very large mean with relative small variance). As with any other convergence diagnostics, you might want to exercise caution in interpreting the results.

The stationarity test is one-sided; rejection occurs when the p-value is greater than $1-\alpha$. To perform the half-width test, you need to select an α level (the default of which is 0.05) and a predetermined tolerance value ϵ (the default of which is 0.1). If the calculated RHW is greater than ϵ, you conclude that there are not enough data to accurately estimate the mean with $1-\alpha$ confidence under tolerance of ϵ.

Raftery and Lewis Diagnostics

If your interest lies in posterior percentiles, you want a diagnostic test that evaluates the accuracy of the estimated percentiles. The Raftery-Lewis test (Raftery and Lewis 1992, 1995) is designed for this purpose. Notation and deductions here closely resemble those in Raftery and Lewis (1995).

Suppose you are interested in a quantity θ_q such that $P(\theta \leq \theta_q|\mathbf{y}) = q$, where q can be an arbitrary cumulative probability, such as 0.025. This θ_q can be empirically estimated by finding the $100nq$th number of the sorted $\{\theta^t\}$. Let $\hat{\theta}_q$ denote the estimand, which corresponds to an estimated probability $P(\theta \leq \hat{\theta}_q) = \hat{P}_q$. Because the simulated posterior distribution converges to the true distribution as the simulation sample size grows, $\hat{\theta}_q$ can achieve any degree of accuracy if the simulator is run for a very long time. However, running too long a simulation can be wasteful. Alternatively, you can use coverage probability to measure accuracy and stop the chain when a certain accuracy is reached.

A stopping criterion is reached when the estimated probability is within $\pm r$ of the true cumulative probability q, with probability s, such as $P(\hat{P}_q \in (q-r, q+r)) = s$. For example, suppose you want the coverage probability s to be 0.95 and the amount of tolerance r to be 0.005. This corresponds to requiring that the estimate of the cumulative distribution function of the 2.5th percentile be estimated to within ± 0.5 percentage points with probability 0.95.

The Raftery-Lewis diagnostics test finds the number of iterations, M, that need to be discarded (burn-ins) and the number of iterations needed, N, to achieve a desired precision. Given a predefined cumulative probability q, these procedures first find $\hat{\theta}_q$, and then they construct a binary $0-1$ process $\{Z_t\}$ by setting $Z_t = 1$ if $\theta^t \leq \hat{\theta}_q$ and 0 otherwise for all t. The sequence $\{Z_t\}$ is itself not a Markov chain, but you can construct a subsequence of $\{Z_t\}$ that is approximately Markovian if it is sufficiently k-thinned. When k becomes reasonably large, $\{Z_t^{(k)}\}$ starts to behave like a Markov chain.

Next, the procedures find this thinning parameter k. The number k is estimated by comparing the Bayesian information criterion (BIC) between two Markov models: a first-order and a second-order Markov model. A jth-order Markov model is one in which the current value of $\{Z_t^{(k)}\}$ depends on the previous j values. For example, in a second-order Markov model,

$$p\left(Z_t^{(k)} = z_t | Z_{t-1}^{(k)} = z_{t-1}, Z_{t-2}^{(k)} = z_{t-2}, \ldots, Z_0^{(k)} = z_0\right)$$
$$= \quad p\left(Z_t^{(k)} = z_t | Z_{t-1}^{(k)} = z_{t-1}, Z_{t-2}^{(k)} = z_{t-2}\right)$$

where $z_i = \{0, 1\}, i = 0, \ldots, t$. Given $\{Z_t^{(k)}\}$, you can construct two transition count matrices for a second-order Markov model:

	$z_t = 0$		$z_t = 1$	
	$z_{t-1} = 0$	$z_{t-1} = 1$	$z_{t-1} = 0$	$z_{t-1} = 1$
$z_{t-2} = 0$	w_{000}	w_{010}	w_{001}	w_{011}
$z_{t-2} = 1$	w_{100}	w_{110}	w_{101}	w_{111}

For each k, the procedures calculate the BIC that compares the two Markov models. The BIC is based on a likelihood ratio test statistic that is defined as

$$G_k^2 = 2 \sum_{i=0}^{1} \sum_{j=0}^{1} \sum_{l=0}^{1} w_{ijl} \log \frac{w_{ijl}}{\hat{w}_{ijl}}$$

where $\hat{w}_{ijl}$ is the expected cell count of w_{ijl} under the null model, the first-order Markov model, where the assumption $(Z_t^{(k)} \perp Z_{t-2}^{(k)}) | Z_{t-1}^{(k)}$ holds. The formula for the expected cell count is

$$\hat{w}_{ijl} = \frac{\sum_i w_{ijl} \cdot \sum_l w_{ijl}}{\sum_i \sum_l w_{ijl}}$$

The BIC is $G_k^2 - 2\log(n_k - 2)$, where n_k is the k-thinned sample size (every kth sample starting with the first), with the last two data points discarded due to the construction of the second-order Markov model. The thinning parameter k is the smallest k for which the BIC is negative. When k is found, you can estimate a transition probability matrix between state 0 and state 1 for $\{Z_t^{(k)}\}$:

$$Q = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}$$

Because $\{Z_t^{(k)}\}$ is a Markov chain, its equilibrium distribution exists and is estimated by

$$\pi = (\pi_0, \pi_1) = \frac{(\beta, \alpha)}{\alpha + \beta}$$

where $\pi_0 = P(\theta \leq \theta_q|\mathbf{y})$ and $\pi_1 = 1 - \pi_0$. The goal is to find an iteration number m such that after m steps, the estimated transition probability $P(Z_m^{(k)} = i|Z_0^{(k)} = j)$ is within ϵ of equilibrium π_i for $i, j = 0, 1$. Let $e_0 = (1, 0)$ and $e_1 = 1 - e_0$. The estimated transition probability after step m is

$$P(Z_m^{(k)} = i|Z_0^{(k)} = j) = e_j \left[\begin{pmatrix} \pi_0 & \pi_1 \\ \pi_0 & \pi_1 \end{pmatrix} + \frac{(1-\alpha-\beta)^m}{\alpha+\beta} \begin{pmatrix} \alpha & -\alpha \\ -\beta & \beta \end{pmatrix} \right] e_j'$$

which holds when

$$m = \frac{\log\left(\frac{(\alpha+\beta)\epsilon}{\max(\alpha,\beta)}\right)}{\log(1-\alpha-\beta)}$$

assuming $1 - \alpha - \beta > 0$.

Therefore, by time m, $\{Z_t^{(k)}\}$ is sufficiently close to its equilibrium distribution, and you know that a total size of $M = mk$ should be discarded as the burn-in.

Next, the procedures estimate N, the number of simulations needed to achieve desired accuracy on percentile estimation. The estimate of $P(\theta \leq \theta_q|\mathbf{y})$ is $\bar{Z}_n^{(k)} = \frac{1}{n}\sum_{t=1}^{n} Z_t^{(k)}$. For large n, $\bar{Z}_n^{(k)}$ is normally distributed with mean q, the true cumulative probability, and variance

$$\frac{1}{n}\frac{(2-\alpha-\beta)\alpha\beta}{(\alpha+\beta)^3}$$

$P(q - r \leq \bar{Z}_n^{(k)} \leq q + r) = s$ is satisfied if

$$n = \frac{(2-\alpha-\beta)\alpha\beta}{(\alpha+\beta)^3} \left\{ \frac{\Phi^{-1}\left(\frac{s+1}{2}\right)}{r} \right\}^2$$

Therefore, $N = nk$.

By using similar reasoning, the procedures first calculate the minimal number of iterations needed to achieve the desired accuracy, assuming the samples are independent:

$$N_{min} = \left\{ \Phi^{-1}\left(\frac{s+1}{2}\right) \right\}^2 \frac{q(1-q)}{r^2}$$

If $\{\theta^t\}$ does not have that required sample size, the Raftery-Lewis test is not carried out. If you still want to carry out the test, increase the number of Markov chain iterations.

The ratio N/N_{min} is sometimes referred to as the *dependence factor*. It measures deviation from posterior sample independence: the closer it is to 1, the less correlated are the samples. There are a few things to keep

in mind when you use this test. This diagnostic tool is specifically designed for the percentile of interest and does not provide information about convergence of the chain as a whole (Brooks and Roberts 1999). In addition, the test can be very sensitive to small changes. Both N and N_{min} are inversely proportional to r^2, so you can expect to see large variations in these numbers with small changes to input variables, such as the desired coverage probability or the cumulative probability of interest. Last, the time until convergence for a parameter can differ substantially for different cumulative probabilities.

Autocorrelations

The sample autocorrelation of lag h for a parameter θ is defined in terms of the sample autocovariance function:

$$\hat{\rho}_h(\theta) = \frac{\hat{\gamma}_h(\theta)}{\hat{\gamma}_0(\theta)}, \quad |h| < n$$

The sample autocovariance function of lag h of θ is defined by

$$\hat{\gamma}_h(\theta) = \frac{1}{n-h}\sum_{t=1}^{n-h}\left(\theta^{t+h} - \bar{\theta}\right)\left(\theta^{t} - \bar{\theta}\right), \quad 0 \le h < n$$

Effective Sample Size

You can use autocorrelation and trace plots to examine the mixing of a Markov chain. A closely related measure of mixing is the effective sample size (ESS) (Kass et al. 1998).

ESS is defined as follows:

$$\text{ESS} = \frac{n}{\tau} = \frac{n}{1 + 2\sum_{k=1}^{\infty}\rho_k(\theta)}$$

where n is the total sample size and $\rho_k(\theta)$ is the autocorrelation of lag k for θ. The quantity τ is referred to as the autocorrelation time. To estimate τ, the Bayesian procedures first find a cutoff point k after which the autocorrelations are very close to zero, and then sum all the ρ_k up to that point. The cutoff point k is such that $|\rho_k| < \min\{0.01, 2s_k\}$, where s_k is the estimated standard deviation:

$$s_k = \sqrt{\left(\frac{1}{n}\left(1 + 2\sum_{j=1}^{k-1}\hat{\rho}_j^2(\theta)\right)\right)}$$

ESS and τ are inversely proportional to each other, and low ESS or high τ indicates bad mixing of the Markov chain.

Summary Statistics

Let $\boldsymbol{\theta}$ be a p-dimensional parameter vector of interest: $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_p\}$. For each $i \in \{1, \ldots, p\}$, there are n observations: $\theta_i = \{\theta_i^t,\ t = 1, \ldots, n\}$.

Mean

The posterior mean is calculated by using the following formula:

$$E(\theta_i|\mathbf{y}) \approx \bar{\theta}_i = \frac{1}{n}\sum_{t=1}^{n}\theta_i^t, \text{ for } i = 1, \ldots, n$$

Standard Deviation

Sample standard deviation (expressed in variance term) is calculated by using the following formula:

$$\mathrm{Var}(\theta_i|\mathbf{y}) \approx s_i^2 = \frac{1}{n-1}\sum_{t=1}^{n}\left(\theta_i^t - \bar{\theta}_i\right)^2$$

Standard Error of the Mean Estimate

Suppose you have n iid samples, the mean estimate is $\bar{\theta}_i$, and the sample standard deviation is s_i. The standard error of the estimate is $\hat{\sigma}_i/\sqrt{n}$. However, positive autocorrelation (see the section "Autocorrelations" on page 149 for a definition) in the MCMC samples makes this an underestimate. To take account of the autocorrelation, the Bayesian procedures correct the standard error by using effective sample size (see the section "Effective Sample Size" on page 149).

Given an effective sample size of m, the standard error for $\bar{\theta}_i$ is $\hat{\sigma}_i/\sqrt{m}$. The procedures use the following formula (expressed in variance term):

$$\widehat{\mathrm{Var}}(\bar{\theta}_i) = \frac{1 + 2\sum_{k=1}^{\infty}\rho_k(\theta_i)}{n} \cdot \frac{\sum_{t=1}^{n}\left(\theta_i^t - \bar{\theta}_i\right)^2}{(n-1)}$$

The standard error of the mean is also known as the Monte Carlo standard error (MCSE). The MCSE provides a measurement of the accuracy of the posterior estimates, and small values do not necessarily indicate that you have recovered the true posterior mean.

Percentiles

Sample percentiles are calculated using Definition 5 (see Chapter 4, "The UNIVARIATE Procedure" (*Base SAS Procedures Guide: Statistical Procedures*)).

Correlation

Correlation between θ_i and θ_j is calculated as

$$r_{ij} = \frac{\sum_{t=1}^{n} \left(\theta_i^t - \bar{\theta}_i\right)\left(\theta_j^t - \bar{\theta}_j\right)}{\sqrt{\sum_t \left(\theta_i^t - \bar{\theta}_i\right)^2 \sum_t \left(\theta_j^t - \bar{\theta}_j\right)^2}}$$

Covariance

Covariance θ_i and θ_j is calculated as

$$s_{ij} = \sum_{t=1}^{n} \left(\theta_i^t - \bar{\theta}_i\right)\left(\theta_j^t - \bar{\theta}_j\right)/(n-1)$$

Equal-Tail Credible Interval

Let $\pi\left(\theta_i|\mathbf{y}\right)$ denote the marginal posterior cumulative distribution function of θ_i. A $100\left(1-\alpha\right)\%$ Bayesian equal-tail credible interval for θ_i is $\left(\theta_i^{\alpha/2}, \theta_i^{1-\alpha/2}\right)$, where $\pi\left(\theta_i^{\alpha/2}|\mathbf{y}\right) = \frac{\alpha}{2}$, and $\pi\left(\theta_i^{1-\alpha/2}|\mathbf{y}\right) = 1 - \frac{\alpha}{2}$. The interval is obtained using the empirical $\frac{\alpha}{2}$th and $(1-\frac{\alpha}{2})$th percentiles of $\left\{\theta_i^t\right\}$.

Highest Posterior Density (HPD) Interval

For a definition of an HPD interval, see the section "Interval Estimation" on page 127. The procedures use the Chen-Shao algorithm (Chen and Shao 1999; Chen, Shao, and Ibrahim 2000) to estimate an empirical HPD interval of θ_i:

1. Sort $\left\{\theta_i^t\right\}$ to obtain the ordered values:

   $$\theta_{i(1)} \leq \theta_{i(2)} \leq \cdots \leq \theta_{i(n)}$$

2. Compute the $100\left(1-\alpha\right)\%$ credible intervals:

   $$R_j\left(n\right) = \left(\theta_{i(j)}, \theta_{i(j+[(1-\alpha)n])}\right)$$

   for $j = 1, 2, \ldots, n - [(1-\alpha)\,n]$.

3. The $100\left(1-\alpha\right)\%$ HPD interval, denoted by $R_{j^*}\left(n\right)$, is the one with the smallest interval width among all credible intervals.

Deviance Information Criterion (DIC)

The deviance information criterion (DIC) (Spiegelhalter et al. 2002) is a model assessment tool, and it is a Bayesian alternative to Akaike's information criterion (AIC) and the Bayesian information criterion (BIC, also known as the Schwarz criterion). The DIC uses the posterior densities, which means that it takes the prior information into account. The criterion can be applied to nonnested models and models that have non-iid data. Calculation of the DIC in MCMC is trivial—it does not require maximization over the parameter space, like the AIC and BIC. A smaller DIC indicates a better fit to the data set.

Letting $\boldsymbol{\theta}$ be the parameters of the model, the deviance information formula is

$$\mathrm{DIC} = \overline{D(\boldsymbol{\theta})} + p_D = D(\overline{\boldsymbol{\theta}}) + 2p_D$$

where

$D(\boldsymbol{\theta}) = 2\left(\log(f(\mathbf{y})) - \log(p(\mathbf{y}|\boldsymbol{\theta}))\right)$: deviance

where

$p(\mathbf{y}|\boldsymbol{\theta})$: likelihood function with the normalizing constants.

$f(\mathbf{y})$: a standardizing term that is a function of the data alone. This term is constant with respect to the parameter and is irrelevant when you compare different models that have the same likelihood function. Since the term cancels out in DIC comparisons, its calculation is often omitted.

NOTE: You can think of the deviance as the difference in twice the log likelihood between the saturated, $f(\mathbf{y})$, and fitted, $p(\mathbf{y}|\boldsymbol{\theta})$, models.

$\overline{\boldsymbol{\theta}}$: posterior mean, approximated by $\frac{1}{n}\sum_{t=1}^{n}\boldsymbol{\theta}^t$

$\overline{D(\boldsymbol{\theta})}$: posterior mean of the deviance, approximated by $\frac{1}{n}\sum_{t=1}^{n}D(\boldsymbol{\theta}^t)$. The expected deviation measures how well the model fits the data.

$D(\overline{\boldsymbol{\theta}})$: deviance evaluated at $\bar{\boldsymbol{\theta}}$, equal to $-2\log(p(\mathbf{y}|\bar{\boldsymbol{\theta}}))$. It is the deviance evaluated at your "best" posterior estimate.

p_D: effective number of parameters. It is the difference between the measure of fit and the deviance at the estimates: $\overline{D(\boldsymbol{\theta})} - D(\overline{\boldsymbol{\theta}})$. This term describes the complexity of the model, and it serves as a penalization term that corrects deviance's propensity toward models with more parameters.

A Bayesian Reading List

This section lists a number of Bayesian textbooks of varying difficulty degrees and a few tutorial/review papers.

Textbooks

Introductory Books

Berry, D. A. (1996). *Statistics: A Bayesian Perspective*. Belmont, CA: Duxbury Press.

Bolstad, W. M. (2007). *Introduction to Bayesian Statistics*. 2nd ed. Hoboken, NJ: John Wiley & Sons.

DeGroot, M. H., and Schervish, M. J. (2002). *Probability and Statistics*. 3rd ed. Reading, MA: Addison-Wesley.

Gamerman, D., and Lopes, H. F. (2006). *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*. 2nd ed. Boca Raton, FL: Chapman & Hall/CRC.

Ghosh, J. K., Delampady, M., and Samanta, T. (2006). *An Introduction to Bayesian Analysis: Theory and Methods*. New York: Springer-Verlag.

Lee, P. M. (2004). *Bayesian Statistics: An Introduction*. 3rd ed. London: Edward Arnold.

Sivia, D. S. (1996). *Data Analysis: A Bayesian Tutorial*. Oxford: Oxford University Press.

Intermediate-Level Books

Box, G. E. P., and Tiao, G. C. (1992). *Bayesian Inference in Statistical Analysis*. New York: John Wiley & Sons.

Chen, M.-H., Shao, Q.-M., and Ibrahim, J. G. (2000). *Monte Carlo Methods in Bayesian Computation*. New York: Springer-Verlag.

Gelman, A., and Hill, J. (2007). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge: Cambridge University Press.

Goldstein, M., and Wooff, D. A. (2007). *Bayes Linear Statistics: Theory and Methods*. Chichester, UK: John Wiley & Sons.

Harney, H. L. (2003). *Bayesian Inference: Parameter Estimation and Decisions*. Berlin: Springer-Verlag.

Leonard, T., and Hsu, J. S. J. (1999). *Bayesian Methods: An Analysis for Statisticians and Interdisciplinary Researchers*. Cambridge: Cambridge University Press.

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

Marin, J.-M., and Robert, C. P. (2007). *Bayesian Core: A Practical Approach to Computational Bayesian Statistics*. New York: Springer-Verlag.

Press, S. J. (2002). *Subjective and Objective Bayesian Statistics: Principles, Models, and Applications*. 2nd ed. New York: Wiley-Interscience.

Robert, C. P. (2001). *The Bayesian Choice*. 2nd ed. New York: Springer-Verlag.

Robert, C. P., and Casella, G. (2004). *Monte Carlo Statistical Methods*. 2nd ed. New York: Springer-Verlag.

Tanner, M. A. (1993). *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*. New York: Springer-Verlag.

Advanced Titles

Berger, J. O. (1985). *Statistical Decision Theory and Bayesian Analysis*. 2nd ed. New York: Springer-Verlag.

Bernardo, J. M., and Smith, A. F. M. (2007). *Bayesian Theory*. 2nd ed. Chichester, UK: John Wiley & Sons.

De Finetti, B. (1992). *Theory of Probability: A Critical Introductory Treatment*. Chichester, UK: John Wiley & Sons.

Jeffreys, H. (1998). *Theory of Probability*. 3rd ed. Oxford: Oxford University Press.

O'Hagan, A. (1994). *Bayesian Inference*. Volume 2B of Kendall's Advanced Theory of Statistics. London: Edward Arnold.

Savage, L. J. (1954). *The Foundations of Statistics*. New York: John Wiley & Sons.

Books Motivated by Statistical Applications and Data Analysis

Carlin, B. P., and Louis, T. A. (2000). *Bayes and Empirical Bayes Methods for Data Analysis*. 2nd ed. London: Chapman & Hall.

Congdon, P. (2003). *Applied Bayesian Modeling*. New York: John Wiley & Sons.

Congdon, P. (2005). *Bayesian Models for Categorical Data*. Chichester, UK: John Wiley & Sons.

Congdon, P. (2006). *Bayesian Statistical Modeling*. 2nd ed. New York: John Wiley & Sons.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004). *Bayesian Data Analysis*. 2nd ed. London: Chapman & Hall.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1996). *Markov Chain Monte Carlo in Practice*. London: Chapman & Hall.

Tutorial and Review Papers on MCMC

Besag, J., Green, P., Higdon, D., and Mengersen, K. (1995). "Bayesian Computation and Stochastic Systems." *Statistical Science* 10:3–66. With discussion.

Casella, G., and George, E. I. (1992). "Explaining the Gibbs Sampler." *American Statistician* 46:167–174.

Chib, S., and Greenberg, E. (1995). "Understanding the Metropolis-Hastings Algorithm." *American Statistician* 49:327–335.

Chib, S., and Greenberg, E. (1996). "Markov Chain Monte Carlo Simulation Methods in Econometrics." *Econometric Theory* 12:409–431.

Kass, R. E., Carlin, B. P., Gelman, A., and Neal, R. M. (1998). "Markov Chain Monte Carlo in Practice: A Roundtable Discussion." *Statistical Science* 52:93–100.

References

Amit, Y. (1991). "On Rates of Convergence of Stochastic Relaxation for Gaussian and Non-Gaussian Distributions." *Journal of Multivariate Analysis* 38:82–99.

Applegate, D. L., Kannan, R., and Polson, N. (1990). *Random Polynomial Time Algorithms for Sampling from Joint Distributions*. Technical report, School of Computer Science, Carnegie Mellon University.

Berger, J. O. (1985). *Statistical Decision Theory and Bayesian Analysis*. 2nd ed. New York: Springer-Verlag.

Berger, J. O. (2006). "The Case for Objective Bayesian Analysis." *Bayesian Analysis* 3:385–402. http://www.stat.cmu.edu/bayesworkshop/2005/berger.pdf.

Berger, J. O., and Wolpert, R. (1988). *The Likelihood Principle*. 2nd ed. Hayward, CA: Institute of Mathematical Statistics.

Bernardo, J. M., and Smith, A. F. M. (1994). *Bayesian Theory*. New York: John Wiley & Sons.

Besag, J. (1974). "Spatial Interaction and the Statistical Analysis of Lattice Systems." *Journal of the Royal Statistical Society, Series B* 36:192–236.

Billingsley, P. (1986). *Probability and Measure*. 2nd ed. New York: John Wiley & Sons.

Box, G. E. P., and Tiao, G. C. (1973). *Bayesian Inference in Statistical Analysis*. New York: John Wiley & Sons.

Breiman, L. (1968). *Probability*. Reading, MA: Addison-Wesley.

Brooks, S. P., and Gelman, A. (1997). "General Methods for Monitoring Convergence of Iterative Simulations." *Journal of Computational and Graphical Statistics* 7:434–455.

Brooks, S. P., and Roberts, G. O. (1998). "Assessing Convergence of Markov Chain Monte Carlo Algorithms." *Statistics and Computing* 8:319–335.

Brooks, S. P., and Roberts, G. O. (1999). "On Quantile Estimation and Markov Chain Monte Carlo Convergence." *Biometrika* 86:710–717.

Carlin, B. P., and Louis, T. A. (2000). *Bayes and Empirical Bayes Methods for Data Analysis*. 2nd ed. London: Chapman & Hall.

Casella, G., and George, E. I. (1992). "Explaining the Gibbs Sampler." *American Statistician* 46:167–174.

Chan, K. S. (1993). "Asymptotic Behavior of the Gibbs Sampler." *Journal of the American Statistical Association* 88:320–326.

Chen, M.-H., and Shao, Q.-M. (1999). "Monte Carlo Estimation of Bayesian Credible and HPD Intervals." *Journal of Computational and Graphical Statistics* 8:69–92.

Chen, M.-H., Shao, Q.-M., and Ibrahim, J. G. (2000). *Monte Carlo Methods in Bayesian Computation*. New York: Springer-Verlag.

Chib, S., and Greenberg, E. (1995). "Understanding the Metropolis-Hastings Algorithm." *American Statistician* 49:327–335.

Congdon, P. (2001). *Bayesian Statistical Modeling*. Chichester, UK: John Wiley & Sons.

Congdon, P. (2003). *Applied Bayesian Modeling*. New York: John Wiley & Sons.

Congdon, P. (2005). *Bayesian Models for Categorical Data*. Chichester, UK: John Wiley & Sons.

Cowles, M. K., and Carlin, B. P. (1996). "Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review." *Journal of the American Statistical Association* 91:883–904.

DeGroot, M. H., and Schervish, M. J. (2002). *Probability and Statistics*. 3rd ed. Reading, MA: Addison-Wesley.

Feller, W. (1968). *An Introduction to Probability Theory and Its Applications*. 3rd ed. New York: John Wiley & Sons.

Gamerman, D. (1997). "Sampling from the Posterior Distribution in Generalized Linear Models." *Statistics and Computing* 7:57–68.

Gelfand, A. E., Hills, S. E., Racine-Poon, A., and Smith, A. F. M. (1990). "Illustration of Bayesian Inference in Normal Data Models Using Gibbs Sampling." *Journal of the American Statistical Association* 85:972–985.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004). *Bayesian Data Analysis*. 2nd ed. London: Chapman & Hall.

Gelman, A., and Rubin, D. B. (1992). "Inference from Iterative Simulation Using Multiple Sequences." *Statistical Science* 7:457–472.

Geman, S., and Geman, D. (1984). "Stochastic Relaxation, Gibbs Distribution, and the Bayesian Restoration of Images." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 6:721–741.

Geweke, J. (1992). "Evaluating the Accuracy of Sampling-Based Approaches to Calculating Posterior Moments." In *Bayesian Statistics*, vol. 4, edited by J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, 169–193. Oxford: Clarendon Press.

Gilks, W. R. (2003). "Adaptive Metropolis Rejection Sampling (ARMS)." Software from MRC Biostatistics Unit, Cambridge, UK. http://www.maths.leeds.ac.uk/~wally.gilks/adaptive.rejection/web_page/Welcome.html.

Gilks, W. R., Best, N. G., and Tan, K. K. C. (1995). "Adaptive Rejection Metropolis Sampling within Gibbs Sampling." *Journal of the Royal Statistical Society, Series C* 44:455–472.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1996). *Markov Chain Monte Carlo in Practice*. London: Chapman & Hall.

Gilks, W. R., and Wild, P. (1992). "Adaptive Rejection Sampling for Gibbs Sampling." *Journal of the Royal Statistical Society, Series C* 41:337–348.

Goldstein, M. (2006). "Subjective Bayesian Analysis: Principles and Practice." *Bayesian Analysis* 3:403–420. http://www.stat.cmu.edu/bayesworkshop/2005/goldstein.pdf.

Hastings, W. K. (1970). "Monte Carlo Sampling Methods Using Markov Chains and Their Applications." *Biometrika* 57:97–109.

Heidelberger, P., and Welch, P. D. (1981). "A Spectral Method for Confidence Interval Generation and Run Length Control in Simulations." *Communications of the ACM* 24:233–245.

Heidelberger, P., and Welch, P. D. (1983). "Simulation Run Length Control in the Presence of an Initial Transient." *Operations Research* 31:1109–1144.

Hoffman, M. D., and Gelman, A. (2014). "The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo." *Journal of Machine Learning Research* 15:1351–1381.

Jeffreys, H. (1961). *Theory of Probability*. 3rd ed. Oxford: Oxford University Press.

Karlin, S., and Taylor, H. (1975). *A First Course in Stochastic Processes*. 2nd ed. Orlando, FL: Academic Press.

Kass, R. E., Carlin, B. P., Gelman, A., and Neal, R. M. (1998). "Markov Chain Monte Carlo in Practice: A Roundtable Discussion." *American Statistician* 52:93–100.

Kass, R. E., and Wasserman, L. (1996). "Formal Rules of Selecting Prior Distributions: A Review and Annotated Bibliography." *Journal of the American Statistical Association* 91:343–370.

Liu, C., Wong, W. H., and Kong, A. (1991a). *Correlation Structure and Convergence Rate of the Gibbs Sampler (I): Application to the Comparison of Estimators and Augmentation Scheme*. Technical report, Department of Statistics, University of Chicago.

Liu, C., Wong, W. H., and Kong, A. (1991b). *Correlation Structure and Convergence Rate of the Gibbs Sampler (II): Applications to Various Scans*. Technical report, Department of Statistics, University of Chicago.

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

MacEachern, S. N., and Berliner, L. M. (1994). "Subsampling the Gibbs Sampler." *American Statistician* 48:188–190.

McCullagh, P., and Nelder, J. A. (1989). *Generalized Linear Models*. 2nd ed. London: Chapman & Hall.

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

Metropolis, N., and Ulam, S. (1949). "The Monte Carlo Method." *Journal of the American Statistical Association* 44:335–341.

Meyn, S. P., and Tweedie, R. L. (1993). *Markov Chains and Stochastic Stability*. Berlin: Springer-Verlag.

Neal, R. M. (2003). "Slice Sampling." *Annals of Statistics* 31:705–757.

Neal, R. M. (2011). "MCMC Using Hamiltonian Dynamics." In *Handbook of Markov Chain Monte Carlo*, edited by S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng, 113–161. Boca Raton, FL: CRC Press.

Press, S. J. (2003). *Subjective and Objective Bayesian Statistics*. New York: John Wiley & Sons.

Raftery, A. E., and Lewis, S. M. (1992). "One Long Run with Diagnostics: Implementation Strategies for Markov Chain Monte Carlo." *Statistical Science* 7:493–497.

Raftery, A. E., and Lewis, S. M. (1995). "The Number of Iterations, Convergence Diagnostics, and Generic Metropolis Algorithms." In *Markov Chain Monte Carlo in Practice*, edited by W. R. Gilks, D. J. Spiegelhalter, and S. Richardson, 115–130. London: Chapman & Hall.

Robert, C. P. (2001). *The Bayesian Choice*. 2nd ed. New York: Springer-Verlag.

Robert, C. P., and Casella, G. (2004). *Monte Carlo Statistical Methods*. 2nd ed. New York: Springer-Verlag.

Roberts, G. O. (1996). "Markov Chain Concepts Related to Sampling Algorithms." In *Markov Chain Monte Carlo in Practice*, edited by W. R. Gilks, D. J. Spiegelhalter, and S. Richardson, 45–58. London: Chapman & Hall.

Rosenthal, J. S. (1991a). *Rates of Convergence for Data Augmentation on Finite Sample Spaces*. Technical report, Department of Mathematics, Harvard University.

Rosenthal, J. S. (1991b). *Rates of Convergence for Gibbs Sampling for Variance Component Models*. Technical report, Department of Mathematics, Harvard University.

Ross, S. M. (1997). *Simulation*. 2nd ed. Orlando, FL: Academic Press.

Schervish, M. J., and Carlin, B. P. (1992). "On the Convergence of Successive Substitution Sampling." *Journal of Computational and Graphical Statistics* 1:111–127.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van der Linde, A. (2002). "Bayesian Measures of Model Complexity and Fit." *Journal of the Royal Statistical Society, Series B* 64:583–616. With discussion.

Tanner, M. A. (1993). *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*. New York: Springer-Verlag.

Tanner, M. A., and Wong, W. H. (1987). "The Calculation of Posterior Distributions by Data Augmentation." *Journal of the American Statistical Association* 82:528–540.

Tierney, L. (1994). "Markov Chains for Exploring Posterior Distributions." *Annals of Statistics* 22:1701–1762.

Von Mises, R. (1918). "Über die 'Ganzzahligkeit' der Atomgewicht und verwandte Fragen." *Physikalische Zeitschrift* 19:490–500.

Wasserman, L. (2004). *All of Statistics: A Concise Course in Statistical Inference*. New York: Springer-Verlag.

Index