12
Markov chain Monte Carlo

12.1 Overview

In the last chapter, we discussed a variety of approaches to estimate the most probable set of parameters for nonlinear models. The primary rationale for these approaches is that they circumvent the need to carry out the multi-dimensional integrals required in a full Bayesian computation of the desired marginal posteriors. This chapter provides an introduction to a very efficient mathematical tool to estimate the desired posterior distributions for high-dimensional models that has been receiving a lot of attention recently. The method is known as *Markov Chain Monte Carlo* (MCMC). MCMC was first introduced in the early 1950s by statistical physicists (N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller) as a method for the simulation of simple fluids. Monte Carlo methods are now widely employed in all areas of science and economics to simulate complex systems and to evaluate integrals in many dimensions. Among all Monte Carlo methods, MCMC provides an enormous scope for dealing with very complicated systems. In this chapter we will focus on its use in evaluating the multi-dimensional integrals required in a Bayesian analysis of models with many parameters.

The chapter starts with an introduction to Monte Carlo integration and examines how a Markov chain, implemented by the Metropolis–Hastings algorithm, can be employed to concentrate samples to regions with significant probability. Next, *tempering* improvements are investigated that prevent the MCMC from getting stuck in the region of a local peak in the probability distribution. One such method called *parallel tempering* is used to re-analyze the spectral line problem of Section 3.6. We also demonstrate how to use the results of parallel tempering MCMC for model comparison. Although MCMC methods are relatively simple to implement, in practice, a great deal of time is expended in optimizing some of the MCMC parameters. Section 12.8 describes one attempt at automating the selection of these parameters. The capabilities of this automated MCMC algorithm are demonstrated in a re-analysis of an astronomical data set used to discover an extrasolar planet.

12.2 Metropolis–Hastings algorithm

Suppose we can write down the joint posterior density,[1] $p(X|D,I)$, of a set of model parameters represented by X. We now want to calculate the expectation value of some function $f(X)$ of the parameters. The expectation value is obtained by integrating the function weighted by $p(X|D,I)$.

$$\langle f(X)\rangle = \int f(X)p(X|D,I)dX = \int g(X)dX. \tag{12.1}$$

For example, if there is only one parameter and we want to compute its mean value, then $f(X) = X$. Also, we frequently want to compute the marginal probability of a subset X_A of the parameters and need to integrate over the remaining parameters designated X_B. Unfortunately, in many cases, we are unable to perform the integrals required in a reasonable length of time. In this section, we develop an efficient method to approximate the desired integrals, starting with a discussion of Monte Carlo integration. Given a value of X, the discussion below assumes we can compute the value of $g(X)$.

In straight Monte Carlo integration, the procedure is to pick n points, uniformly randomly distributed in a multi-dimensional volume (V) of our parameter space X. The volume must be large enough to contain all regions where $g(X)$ contributes significantly to the integral. Then the basic theorem of Monte Carlo integration estimates the integral of $g(X)$ over the volume V by

$$\langle f(X)\rangle = \int_V g(X)dX \approx V \times \langle g(X)\rangle \pm V \times \sqrt{\frac{\langle g^2(X)\rangle - \langle g(X)\rangle^2}{n}}, \tag{12.2}$$

where

$$\langle g(X)\rangle = \frac{1}{n}\sum_{i=1}^{n} g(X_i);\ \ \langle g^2(X)\rangle = \frac{1}{n}\sum_{i=1}^{n} g^2(X_i). \tag{12.3}$$

There is no guarantee that the error is distributed as a Gaussian, so the error term is only a rough indicator of the probable error. When the random samples X_i are independent, the law of large numbers ensures that the approximation can be made as accurate as desired by increasing n. Note: n is the number of random samples of $g(X)$, not the size of the fixed data sample. The problem with Monte Carlo integration is that too much time is wasted sampling regions where $p(X|D,I)$ is very small. Suppose in a one-parameter problem the fraction of the time spent sampling regions of high probability is 10^{-1}. Then in an M-parameter problem, this fraction could easily fall to 10^{-M}. A variation of the simple Monte Carlo described above, which involves reweighting the integrand and adjusting the sample rules (known as "importance sampling"), helps considerably but it is difficult to design the reweighting for large numbers of parameters.

[1] In the literature dealing with MCMC, it is common practice to write $\pi(X)$ instead of $p(X|D,I)$.

In general, drawing samples independently from $p(X|D,I)$ is not currently computationally feasible for problems where there are large numbers of parameters. However, the samples need not necessarily be independent. They can be generated by any process that generates samples from the *target distribution*, $p(X|D,I)$, in the correct proportions. All MCMC algorithms generate the desired samples by constructing a kind of random walk in the model parameter space such that the probability for being in a region of this space is proportional to the posterior density for that region. The random walk is accomplished using a Markov chain, whereby the new sample, X_{t+1}, depends on the previous sample X_t according to an entity called the *transition probability* or *transition kernel*, $p(X_{t+1}|X_t)$. The transition kernel is assumed to be time independent. The remarkable property of $p(X_{t+1}|X_t)$ is that after an initial burn-in period (which is discarded) it generates samples of X with a probability density equal to the desired posterior $p(X|D,I)$.

How does it work? There are two steps. In the first step, we pick a proposed value for X_{t+1} which we call Y, from a *proposal distribution*, $q(Y|X_t)$, which is easy to evaluate. As we show below, $q(Y|X_t)$ can have almost any form. To help in developing your intuition, it is perhaps convenient to contemplate a multivariate normal (Gaussian) distribution for $q(Y|X_t)$, with a mean equal to the current sample X_t. With such a proposal distribution, the probability density decreases with distance away from the current sample.

The second step is to decide on whether to accept the candidate Y for X_{t+1} on the basis of the value of a ratio r given by

$$r = \frac{p(Y|D,I)}{p(X_t|D,I)} \frac{q(X_t|Y)}{q(Y|X_t)}, \tag{12.4}$$

where r is called the *Metropolis ratio*. If the proposal distribution is symmetric, then the second factor is $= 1$. If $r \geq 1$, then we set $X_{t+1} = Y$. If $r < 1$, then we accept it with a probability $= r$. This is done by sampling a random variable U from Uniform(0, 1), a uniform distribution in the interval 0 to 1. If $U \leq r$ we set $X_{t+1} = Y$, otherwise we set $X_{t+1} = X_t$. This second step can be summarized by a term called the *acceptance probability* $\alpha(X_t, Y)$ given by

$$\alpha(X_t, Y) = \min(1, r) = \min\left(1, \frac{p(Y|D,I)}{p(X_t|D,I)} \frac{q(X_t|Y)}{q(Y|X_t)}\right). \tag{12.5}$$

The MCMC method as initially proposed by Metropolis *et al*. in 1953, considered only symmetric proposal distributions, having the form $q(Y|X_t) = q(X_t|Y)$. Hastings (1970) generalized the algorithm to include asymmetric proposal distributions and the generalization is commonly referred to as the Metropolis–Hastings algorithm. There are now many different versions of the algorithm.

The Metropolis–Hastings algorithm is extremely simple:

```
1. Initialize $X_0$; set $t = 0$.
2. Repeat {Obtain a new sample $Y$ from $q(Y|X_t)$
   Sample a Uniform(0,1) random variable $U$
       If $U \leq r$ set $X_{t+1} = Y$ otherwise set $X_{t+1} = X_t$ Increment $t$}
```

Example 1:
Suppose the posterior is a Poisson distribution, $p(X|D,I) = \lambda^X e^{-\lambda}/X!$. For our proposal distribution $q(Y|X_t)$, we will use a simple random walk such that:

1. Given X_t, pick a random number $U_1 \sim U(0,1)$
2. If $U_1 > 0.5$, propose $Y = X_t + 1$ otherwise $Y = X_t - 1$
3. Compute the Metropolis ratio $r = p(Y|D,I)/p(X_t|D,I) = \lambda^{Y-X_t} X_t!/Y!$
4. Acceptance/rejection: $U_2 \sim \mathbf{U(0,1)}$
   Accept $X_{t+1} = Y$ if $U_2 \leq r$ otherwise set $X_{t+1} = X_t$

Figure 12.1 illustrates the results for the above simple MCMC simulation using a value of $\lambda = 3$ and starting from an initial $X_0 = 10$ which is far out in the tail of the posterior. Panel (a) shows a sequence of 1000 samples from the MCMC. It is clear that the samples quickly move from our starting point far out in the tail to the vicinity of the posterior mean. Panel (b) compares a histogram of the last 900 samples from the MCMC with the true Poisson posterior which is indicated by the solid line. The

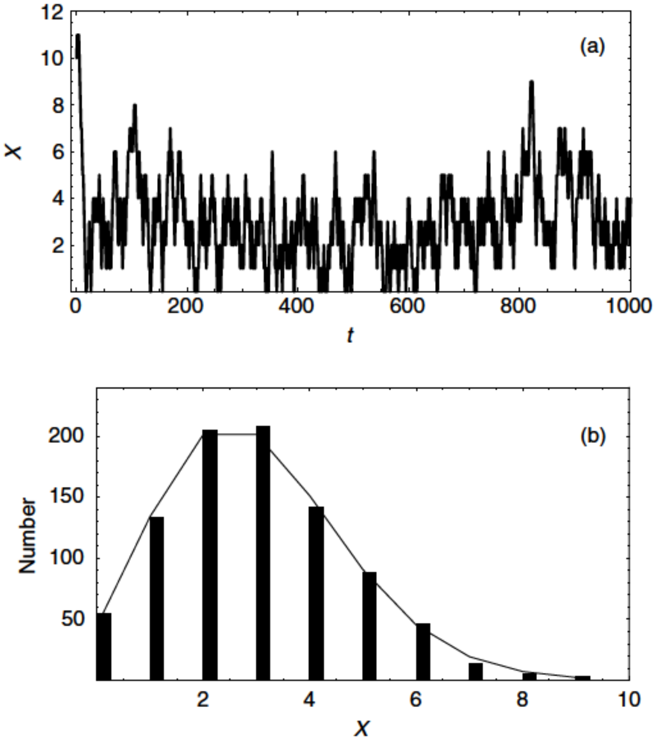

Figure 12.1 The results from a simple one-dimensional Markov chain Monte Carlo simulation for a Poisson posterior for X. Panel (a) shows a sequence of 1000 samples from the MCMC. Panel (b) shows a comparison of the last 900 MCMC samples with the true posterior indicated by the solid curve.

agreement is very good. We treated the first 100 samples as an estimate of the burn-in period and did not use them.

Example 2:

Now consider a MCMC simulation of samples from a joint posterior $p(X_1, X_2|D, I)$ in two parameters X_1 and X_2, which has a double peak structure. Note: if we want to refer to the tth time sample of the ith parameter from a Markov chain, we will do so with the designation $X_{t,i}$. We define the posterior in *Mathematica* with the following commands.

Needs["Statistics 'MultinormalDistribution' "]

dist1 = MultinormalDistribution [{0, 0}, {{1, 0}, {0, 1}}]
The first argument $\{0, 0\}$ indicates the multinormal distribution is centered at 0,0.
The second argument $\{\{1, 0\}, \{0, 1\}\}$ gives the covariance of the distribution.
dist2 = MultinormalDistribution[{{4, 0}, {{2, 0.8}, {0.8, 2}}]
Posterior = 0.5 (PDF[dist1, $\{\mathbf{X_1}, \mathbf{X_2}\}$]+ PDF[dist2, $\{\mathbf{X_1}, \mathbf{X_2}\}$])
The factor of 0.5 ensures the posterior is normalized to an area of one.

In this example, we used a proposal density function $q(Y_1, Y_2|X_1, X_2)$ which is a two-dimensional Gaussian (normal) distribution.

$$[\mathbf{MultinormalDistribution}[\{X_1,X_2\}, \{\{\sigma_1^2,0\},\{0,\sigma_2^2\}\}]]$$

The results for 8000 samples of the posterior generated with this MCMC are shown in Figure 12.2. Note that the first 50 samples were treated as the burn-in period and are not included in this plot. Panel (a) shows a sequence of 7950 samples from the MCMC with $\sigma_1 = \sigma_2 = 1$. The two model parameters represented by X_1 and X_2 could be very different physical quantities each characterized by a different scale. In that case, σ_1 and σ_2 could be very different. Panel (b) shows the same points with contours of the posterior overlaid. The distribution of sample points matches the contours of the true posterior very well. Panel (c) shows a comparison of the true marginal posterior (solid curve) for X_1 and the MCMC marginal (dots). The MCMC marginal is simply a normalized histogram of the X_1 sample values. Panel (d) shows a comparison of the true marginal posterior (solid curve) for X_2 and the MCMC marginal (dots). In both cases, the agreement is very good.

We also investigated the evolution of the MCMC samples for proposal distributions with different values of σ. Panel (a) in Figure 12.3 shows the case for a $\sigma \sim 1/10$ the scale of the smallest features in the true posterior. The starting point for each simulation was at $X_1 = -4.5$, $X_2 = 4.5$. In this case, the burn-in period is considerably longer and it appears that a larger number of samples would be needed to do justice to the posterior which is indicated by the contours. Panel (b) illustrates the case for $\sigma = 1$, the value used for Figure 12.2. Panel (c) uses a $\sigma \sim 10$ times the scale of the smallest features in the posterior. From the density of the points it appears that we have used a

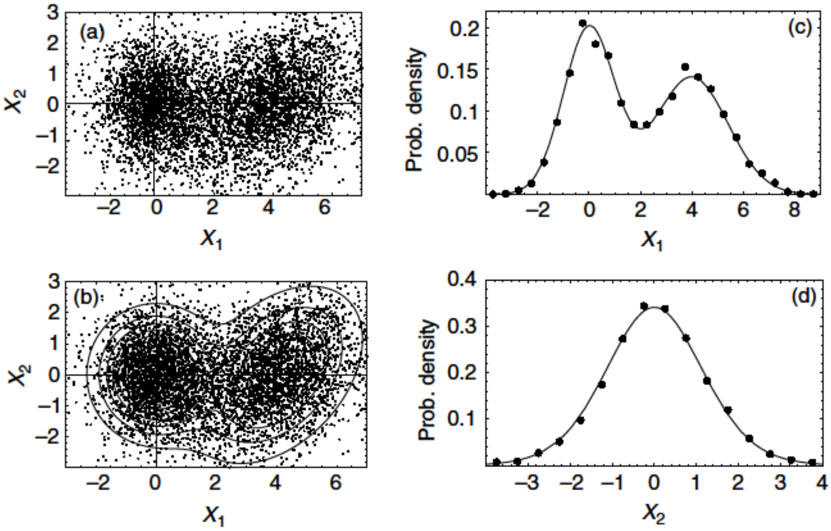

Figure 12.2 The results from a two-dimensional Markov chain Monte Carlo simulation of a double peaked posterior. Panel (a) shows a sequence of 7950 samples from the MCMC. Panel (b) shows the same points with contours of the posterior overlaid. Panel (c) shows a comparison of the marginal posterior (solid curve) for X_1 and the MCMC marginal (dots). Panel (d) shows a comparison of the marginal posterior (solid curve) for X_2 and the MCMC marginal (dots).

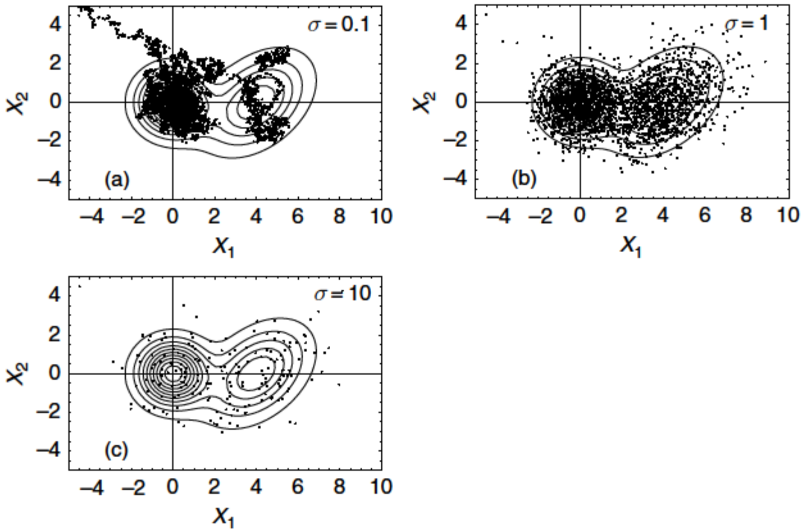

Figure 12.3 A comparison of the samples from three Markov chain Monte Carlo runs using Gaussian proposal distributions with differing values of the standard deviation: (a) $\sigma = 0.1$, (b) $\sigma = 1$, (c) $\sigma = 10$. The starting point for each run was at $X_1 = -4.5$ and $X_2 = 4.5$.

much smaller number of MCMC samples. In fact we used the same number of samples. Recall that in MCMC we carry out a test to decide whether to accept the new proposal (see discussion following Equation (12.4)). If we fail to accept the proposal, then we set $X_{t+1} = X_t$. Thus, many of the points in panel (c) are repeats of the same sample as the proposed sample was rejected on many occasions.

It is commonly agreed that finding an ideal proposal distribution is an art. If we restrict the conversation to Gaussian proposal distributions then the question becomes what is the optimum choice of σ? As mentioned earlier, the samples from a MCMC are not independent, but exhibit correlations. In Figure 12.4, we illustrate the correlations of samples corresponding to the three choices of σ used in Figure 12.3 by plotting the *autocorrelation functions* (ACFs) for X_2. The ACF, $\rho(h)$, which was introduced in Section 5.13.2, is given by

$$\rho(h) = \frac{\sum_{\text{overlap}}[(X_t - \overline{X})(X_{t+h} - \overline{X})]}{\sqrt{\sum_{\text{overlap}}(X_t - \overline{X})^2} \times \sqrt{\sum_{\text{overlap}}(X_{t+h} - \overline{X})^2}}, \tag{12.6}$$

where X_{t+h} is a shifted version of X_t and the summation is carried out over the subset of samples that overlap. The shift h is referred to as the *lag*. It is often observed that $\rho(h)$ is roughly exponential in shape so we can model the ACF

$$\rho(h) \sim \exp\{-\frac{h}{\tau_{\text{exp}}}\}. \tag{12.7}$$

The autocorrelation time constant, τ_{exp}, reflects the convergence speed of the MCMC sampler and is approximately equal to the interval between independent samples. In general, the smaller the value of τ_{exp} the better, i.e., the more efficient, the MCMC

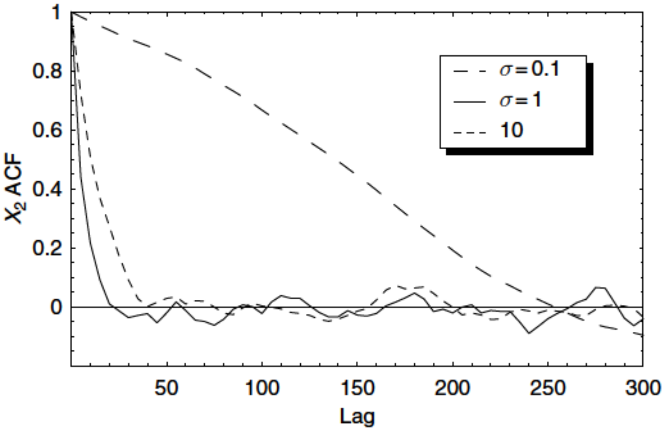

Figure 12.4 A comparison of the autocorrelation functions for three Markov chain Monte Carlo runs using Gaussian proposal distributions with differing values of the standard deviation: $\sigma = 0.1, \sigma = 1, \sigma = 10$.

sampler is. Examination of Figure 12.4 indicates that of the three choices of σ chosen above, $\sigma = 1.0$ leads to the smallest values of τ_{exp} for X_2. Of course, in this example just considered, we have set $\sigma_{X_1} = \sigma_{X_2}$. In general, they will not be equal. Related to the optimum choice of σ is the average rate at which proposed state changes are accepted, called the *acceptance rate*. Based on empirical studies, Roberts, Gelman, and Gilks (1997) recommend calibrating the acceptance rate to about 25% for a high-dimensional model and to about 50% for models of one or two dimensions. The acceptance rates corresponding to our three choices of σ in Figure 12.3 are 95%, 63%, and 5%, respectively.

A number of issues arise from a consideration of these two simple examples. How do we decide: (a) the length of the burn-in period, (b) when to stop the Markov chain, and (c) what is a suitable proposal distribution? For a discussion of these points, the reader is referred to a collection of review and application papers (Gilks, Richardson, and Spiegelhalter 1996). For an unpublished 1996 roundtable discussion of informal advice for novice practitioners, moderated by R. E. Kass, see *www.amstat.org/publications/tas/kass.pdf*. The treatment of MCMC given in this text is intended only as an introduction to this topic.

Loredo (1999) gives an interesting perspective on the relationship between the development of MCMC in statistics and certain computational physics techniques. Define a function $\Lambda(X) = -\ln[p(X|I)\ p(D|X, I)]$. Then the posterior distribution can be written as $p(X|D, I) = e^{-\Lambda(X)}/Z$, where $Z = \int dX\ e^{-\Lambda(X)}$. Evaluation of the posterior resembles two classes of problems familiar to physicists: evaluating Boltzmann factors and partition functions in statistical mechanics, and evaluating Feynman path weights and path integrals in Euclidean quantum field theory. For a discussion of some useful modern extensions of the Metropolis algorithm that are particularly accessible to physical scientists, see Liu (2001) and the first section of Toussaint (1989). A readable tutorial for statistics students is available in Chib and Greenberg (1995).

12.3 Why does Metropolis–Hastings work?

Remarkably, for a wide range of proposal distributions $q(Y|X)$, the Metropolis–Hastings algorithm generates samples of X with a probability density which converges on the desired target posterior $p(X|D, I)$, called the *stationary distribution* of the Markov chain. For the distribution of X_t to converge to a stationary distribution, the Markov chain must have three properties (Roberts, 1996). First, it must be *irreducible*. That is, from all starting points, the Markov chain must be able (eventually) to jump to all states in the target distribution with positive probability. Second it must be *aperiodic*. This stops the chain from oscillating between different states in a regular periodic movement. Finally the chain must be *positive recurrent*. This can be expressed in terms of the existence of a stationary distribution $\pi(X)$, say, such that if an initial value X_0 is sampled from $\pi(X)$, then all subsequent iterates will also be distributed according to $\pi(X)$.

To see that the target distribution is the stationary distribution of the Markov chain generated by the Metropolis–Hastings algorithm, consider the following: suppose we start with a sample X_t from the target distribution. The probability of drawing X_t from the posterior is $p(X_t|D,I)$. The probability that we will draw and accept a sample X_{t+1} is given by the transition kernel, $p(X_{t+1}|X_t) = q(X_{t+1}|X_t)\ \alpha(X_t, X_{t+1})$, where $\alpha(X_t, X_{t+1})$ is given by Equation (12.5). The joint probability of X_t and X_{t+1} is then given by

$$\begin{aligned}
&\text{Joint probability}(X_t, X_{t+1}) = p(X_t|D,I)\ p(X_{t+1}|X_t)\\
&= p(X_t|D,I)\ q(X_{t+1}|X_t)\alpha(X_t, X_{t+1})\\
&= p(X_t|D,I)\ q(X_{t+1}|X_t)\min\left(1, \frac{p(X_{t+1}|D,I)\ q(X_t|X_{t+1})}{p(X_t|D,I)\ q(X_{t+1}|X_t)}\right)\\
&= \min(p(X_t|D,I)\ q(X_{t+1}|X_t), p(X_{t+1}|D,I)q(X_t|X_{t+1}))\\
&= p(X_{t+1}|D,I)\ q(X_t|X_{t+1})\alpha(X_{t+1}, X_t)\\
&= p(X_{t+1}|D,I)\ p(X_t|X_{t+1}).
\end{aligned} \tag{12.8}$$

Thus, we have shown

$$p(X_t|D,I)\ p(X_{t+1}|X_t) = p(X_{t+1}|D,I)\ p(X_t|X_{t+1}), \tag{12.9}$$

which is called the *detailed balance equation.*

In statistical mechanics, detailed balance occurs for systems in thermodynamic equilibrium.[2] In the present case, the condition of detailed balance means that the Markov chain generated by the Metropolis–Hastings algorithm converges to a stationary distribution.

Recall from Equation (12.8) that $p(X_t|D,I)p(X_{t+1}|X_t)$ is the joint probability of X_t and X_{t+1}. We will now integrate this joint probability with respect to X_t, making use of Equation (12.9), and demonstrate that the result is simply the marginal probability distribution of X_{t+1}.

$$\begin{aligned}
\int p(X_t|D,I)\ p(X_{t+1}|X_t)dX_t &= \int p(X_{t+1}|D,I)p(X_t|X_{t+1})\ dX_t\\
&= p(X_{t+1}|D,I)\ \int p(X_t|X_{t+1})\ dX_t\\
&= p(X_{t+1}|D,I).
\end{aligned} \tag{12.10}$$

Thus, we have shown that once a sample from the stationary target distribution has been obtained, all subsequent samples will be from that distribution.

[2] It may help to consider the following analogy: suppose we have a collection of hydrogen atoms. The number of atoms making a transition from excited state t to state $t+1$ in 1 s is given by $N \times p(t) \times p(t+1|t)$, where N equals the total number of atoms, $p(t)$ is the probability of an atom being in state t, and $p(t+1|t)$ is the probability that an atom in state t will make a transition to state $t+1$ in 1 s. Similarly the number making transitions from $t+1$ to t in 1 s is given by $N \times p(t+1) \times p(t|t+1)$. In thermodynamic equilibrium, the rate of transition from t to $t+1$ is equal to the rate from $t+1$ to t, so

$$p(t) \times p(t+1|t) = p(t+1) \times p(t|t+1).$$

12.4 Simulated tempering

The simple Metropolis–Hastings algorithm outlined in Section 12.2 can run into difficulties if the target probability distribution is multi-modal. The MCMC can become stuck in a local mode and fail to fully explore other modes which contain significant probability. This problem is very similar to the one encountered in finding a global minimum in a nonlinear model fitting problem. One solution to that problem was to use simulated annealing (see Section 11.4.1) by introducing a temperature parameter $\mathcal{T}$. The analogous process applied to drawing samples from a target probability distribution (e.g., Geyer and Thompson, 1995) is often referred to as *simulated tempering* (ST). In annealing, the temperature parameter is gradually decreased. In ST, we create a discrete set of progressively flatter versions of the target distribution using a temperature parameter. For $\mathcal{T} = 1$, the distribution is the desired target distribution which is referred to as the cold sampler. For $\mathcal{T} \gg 1$, the distribution is much flatter. The basic idea is that by repeatedly heating up the distribution (making it flatter), the new sampler can escape from local modes and increase its chance of reaching all regions of the target distribution that contain significant probability. Typical inference is based on samples drawn from the cold sampler and the remaining observations discarded. Actually, in Section 12.7 we will see how to use the samples from the hotter distributions to evaluate Bayes factors in model selection problems.

Again, let $p(X|D,I)$ be the target posterior distribution we want to sample. Applying Bayes' theorem, we can write this as

$$p(X|D,I) = C\, p(X|I) \times p(D|X,I),$$

where $C = 1/p(D|I)$ is the usual normalization constant which is not important at this stage and will be dropped. We can construct other flatter distributions as follows:

$$\begin{aligned}\pi(X|D,\beta,I) &= p(X|I)p(D|X,I)^{\beta}\\ &= p(X|I)\ \exp(\beta\ \ln[p(D|X,I)]), \quad \text{for } 0 < \beta < 1.\end{aligned} \tag{12.11}$$

Rather than use a temperature which varies from 1 to infinity, we prefer to use its reciprocal which we label β and refer to as the tempering parameter. Thus β varies from 1 to zero. We will use a discrete set of β values labeled $\{1, \beta_2, \cdots, \beta_m\}$, where $\beta = 1$ corresponds to the cold sampler (target distribution) and β_m corresponds to our hottest sampler which is generally much flatter. This particular formulation is also convenient for our later discussion on determining the Bayes factor in model selection problems. Rather than describe ST in detail, we will describe a more efficient related algorithm called *parallel tempering* in the next section.

12.5 Parallel tempering

Parallel tempering (PT) is an attractive alternative to simulated tempering (Liu, 2001). Again, multiple copies of the simulation are run in parallel, each at a different temperature (i.e., a different $\beta = 1/\mathcal{T}$). One of the simulations, corresponding to $\beta = 1/\mathcal{T} = 1$, is the desired target probability distribution. The other simulations correspond to a ladder of higher temperature distributions indexed by i. Let $n\beta$ equal the number of parallel MCMC simulations. At intervals, a pair of adjacent simulations on this ladder are chosen at random and a proposal made to swap their parameter states. Suppose simulations β_i and β_{i+1} are chosen. At time t, simulation β_i is in state $X_{t,i}$ and simulation β_{i+1} is in state $X_{t,i+1}$. If the swap is accepted by the test given below then these states are interchanged. In the example discussed in Section 12.6, we specify that on average, a swap is proposed after every n_s iterations ($n_s = 30$ was used) of the parallel simulations in the ladder. This is done by choosing a random number, $U_1 \sim$ Uniform[0,1], at each time iteration and proposing a swap only if $U_1 \leq 1/n_s$. If a swap is to be proposed, we use a second random number to pick one of the ladder simulations i in the range $i = 1$ to $(n\beta - 1)$, and propose swapping the parameter states of i and $i+1$. A Monte Carlo acceptance rule determines the probability for the proposed swap to occur. Accept the swap with probability

$$r = \min\left\{1, \frac{\pi(X_{t,i+1}|D,\beta_i,I)\ \pi(X_{t,i}|D,\beta_{i+1},I)}{\pi(X_{t,i}|D,\beta_i,I)\ \pi(X_{t,i+1}|D,\beta_{i+1},I)}\right\}, \qquad (12.12)$$

where $\pi(X|D,\beta,I)$ is given by Equation (12.11). We accept the swap if $U_2 \sim$ Uniform[0,1] $\leq r$.

This swap allows for an exchange of information across the population of parallel simulations. In the higher temperature simulations, radically different configurations can arise, whereas in lower temperature states, a configuration is given the chance to refine itself. By making exchanges, we can capture and improve the higher probability configurations generated by the population by putting them into lower temperature simulations. Some experimentation is needed to refine suitable choices of β_i values. Adjacent simulations need to have some overlap to achieve a sufficient acceptance probability for an exchange operation.

12.6 Example

Although MCMC really comes into its own when the number of model parameters is very large, we will apply it to the toy spectral line problem we analyzed in Section 3.6, because we can compare with our earlier results. The objective of that problem was to test two competing models, represented by M_1 and M_2, on the basis of some spectral line data. Only M_1 predicts the existence of a particular spectral line. In the simplest version of the problem, the line frequency and shape is exactly predicted by M_1; the only quantity which is uncertain is the line strength T expressed in temperature units. The odds ratio in favor of M_1 was found to be 11:1 assuming a Jeffreys prior for the line strength. We also computed the most probable line strength. In Section 3.9, we investigated how our conclusions would be altered if the line frequency were uncertain, i.e., it could occur anywhere between channels 1 to 44. In that case, the odds ratio favoring M_1 dropped from 11:1 to $\approx$ 1:1, assuming a uniform prior for the line center frequency. Below, we apply both the Metropolis–Hastings and parallel tempering versions of MCMC to the problem of estimating the marginal posteriors of the line strength and center frequency to compare with our previous results. In Section 12.7, we will employ parallel tempering to compute the Bayes factor needed for model comparison.

Metropolis–Hastings results

In this section, we will draw samples from $p(X|D, M_1, I)$, where X is a vector representing the two parameters of model M_1, namely the line strength T and the line center frequency ν expressed as channel number. We use a Jeffreys prior for T in the range $T_{\min} = 0.1\,\text{mK}$ to $T_{\max} = 100\,\text{mK}$. We assume a uniform prior for ν in the range channel 1 to 44. The steps in the calculation are as follows:

```
1. Initialize $X_0$; set $t = 0$.
      In this example we set $X_0 = \{T_0 = 5, \nu_0 = 30\}$
2. Repeat {
   a) Obtain a new sample $Y$ from $q(Y|X_t)$
      $Y = \{T', \nu'\}$
      we set $q(T'|T_t) = $ Random[NormalDistribution[$T_t, \sigma_T = 1.0$]]
      and $q(\nu'|\nu_t) = $ Random[NormalDistribution[$\nu_t, \sigma_f = 1.0$]]
   b) Compute the Metropolis ratio

      $r = \frac{p(Y|D, M_1, I)}{p(X_t|D, M_1, I)} = \frac{p(T', \nu'|M_1, I)\ p(D|M_1, T', \nu', I)}{p(T_t, \nu_t|M_1, I)\ p(D|M_1, T_t, \nu_t, I)}$

      where $p(D|M_1, T, \nu, I)$ is given by Equations (3.44) and (3.41).
      The priors $p(T, \nu|M_1, I) = p(T|M_1, I)\ p(\nu|M_1, I)$ are given by Equations (3.38)
      and (3.33).
            Note: if $T'$ or $\nu'$ lie outside the prior boundaries set $r = 0$.
   c) Acceptance/rejection: $U \sim \mathbf{U(0, 1)}$
   d) Accept $X_{t+1} = Y$ if $U \leq r$, otherwise set $X_{t+1} = X_t$
   e) Increment $t$}
```

Figure 12.5 shows results for 10^5 iterations of a Metropolis–Hastings Markov chain Monte Carlo. Panel (a) shows every 50th value of parameter ν, expressed as a channel number, and panel (c) the same for parameter T. It is clear that the ν values move quickly to a region centered on channel 37 with occasional jumps to a region centered on channel 24 and only one jump to small channel numbers. The T parameter can be seen to fluctuate between 0.1 and $\sim$3.5 mK. Panels (b) and (d) show a blow-up of the first 500 iterations. It is apparent from these panels that the burn-in period is very short, $<$ 50 iterations for a starting state of $T = 5$ and $\nu = 30$.

Figure 12.6 shows distributions of the two parameters. In panel (a), the joint distribution of T and ν is apparent from the scatter plot of every 20th iteration obtained

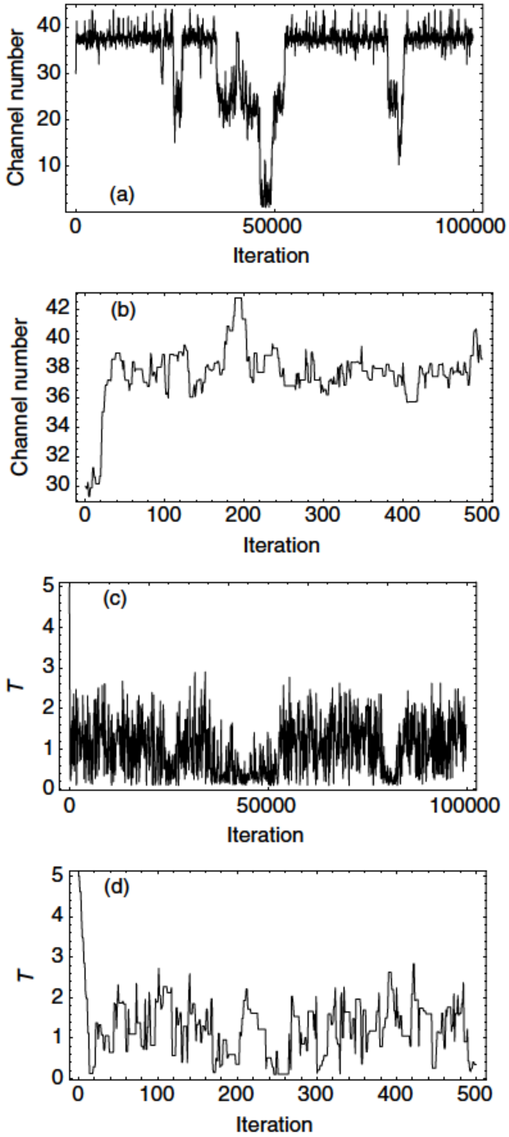

Figure 12.5 Results for 10^5 iterations of a Metropolis–Hastings Markov chain Monte Carlo. Panel (a) shows every 50th value of parameter ν and panel (c) the same for parameter T. Panels (b) and (d) show a blow-up of the first 500 iterations.

after dropping the burn-in period consisting of the first 50 iterations. To obtain the marginal posterior density for the ν parameter, we simply plot a histogram of all the ν values (post burn-in) normalized by dividing by the sum of the ν values multiplied by the width of each bin. This is shown plotted in panel (b) together with our earlier marginal distribution (solid curve) computed by numerical integration. It is clear that

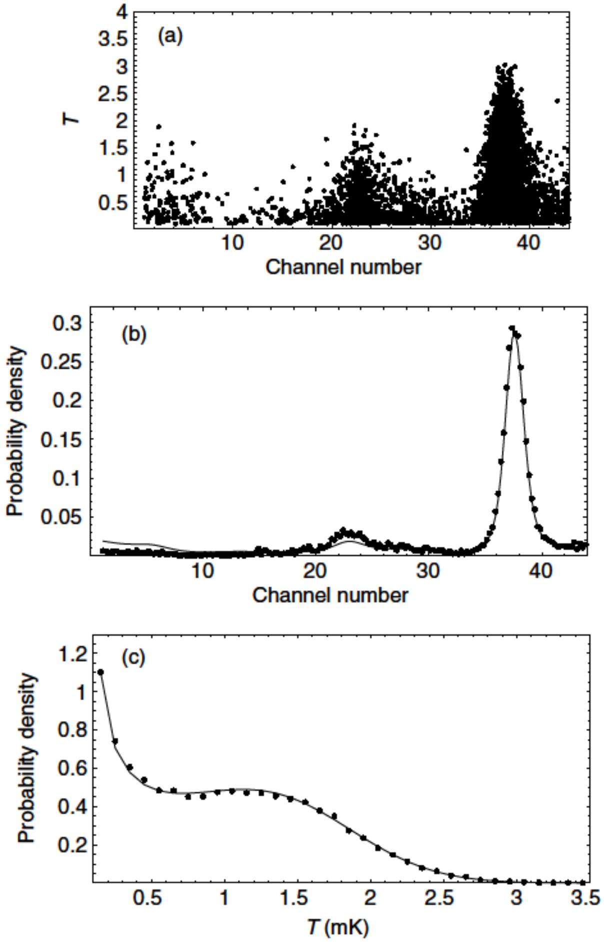

Figure 12.6 Results for the spectral line problem using a Metropolis–Hastings Markov chain Monte Carlo analysis. Panel (a) is a scatter plot of the result for every 20th iteration in the two model parameters, channel number and line strength T. Panel (b) shows the marginal probability density for channel number (points) compared to our earlier numerical integration result indicated by the solid curve. Panel (c) shows the marginal probability density for line strength T (points) compared to our earlier numerical integration result indicated by the solid curve.

10^5 iterations of Metropolis–Hastings does a good job of defining the dominant peak of the probability distribution for ν but does a poor job of capturing two other widely separated islands containing significant probability. On the other hand, it is clear from panel (c) that it has done a great job of defining the distribution of T.

Parallel tempering results

We also analyzed the spectral line data with a parallel tempering (PT) version of MCMC described in Section 12.5. We used five values for the tempering parameter, β, uniformly spaced between 0.01 and 1.0, and ran all five chains in parallel. At intervals (on average every 50 iterations) a pair of adjacent simulations on this ladder are chosen at random and a proposal made to swap their parameter states. We used the same starting state of $T = 5, \nu = 30$ and executed 10^5 iterations. The final results for the $\beta = 1$, corresponding to the target distribution, are shown in Figures 12.7 and 12.8. The acceptance rate for this simulation was 37%.

Comparing panel (a) of Figures 12.7 and 12.5, we see that the PT version visits the two low-lying regions of ν probability much more frequently than the Metropolis–Hastings version. Comparing the marginal densities of Figures 12.8 and 12.6 we see that the PT marginal density for ν is in better agreement with the expected results indicated by the solid curves. For both versions, the marginal densities for T are in excellent agreement with the expected result. In more complicated problems, we often cannot conveniently compute the marginal densities by another method. In this case, it is useful to compare the results from a number of PT simulations with different starting parameter states.

12.7 Model comparison

So far we have demonstrated how to use MCMC to compute the marginal posteriors for model parameters. In this section, we will show how to use the results of parallel tempering to compute the Bayes factor used in model comparison (Skilling, 1998; Goggans and Chi, 2004). In the toy spectral line problem of Section 3.6, we were interested in computing the odds ratio of two models M_1 and M_2 which from Equation (3.30) is equal to the prior odds times the Bayes factor given by

$$B_{12} = \frac{p(D|M_1, I)}{p(D|M_2, I)}, \tag{12.13}$$

where $p(D|M_1, I)$ and $p(D|M_2, I)$ are the global likelihoods for the two models. In the version of this problem analyzed in Section 12.6, M_1 has two parameters ν and T. For independent priors,

$$p(D|M_1, I) = \int d\nu\, p(\nu|M_1, I) \int dT\, p(T|M_1, I) p(D|M_1, \nu, T, I). \tag{12.14}$$

In what follows, we will generalize the model parameter set to an arbitrary number of parameters which we represent by the vector X.

To evaluate $p(D|M_1, I)$, using parallel tempering MCMC, we first define a partition function

$$\begin{aligned} Z(\beta) &= \int dX\; p(X|M_1, I)\; p(D|M_1, X, I)^{\beta} \\ &= \int dX \exp\{\ln[p(X|M_1, I)] + \beta \ln[p(D|M_1, X, I)]\}, \end{aligned} \tag{12.15}$$

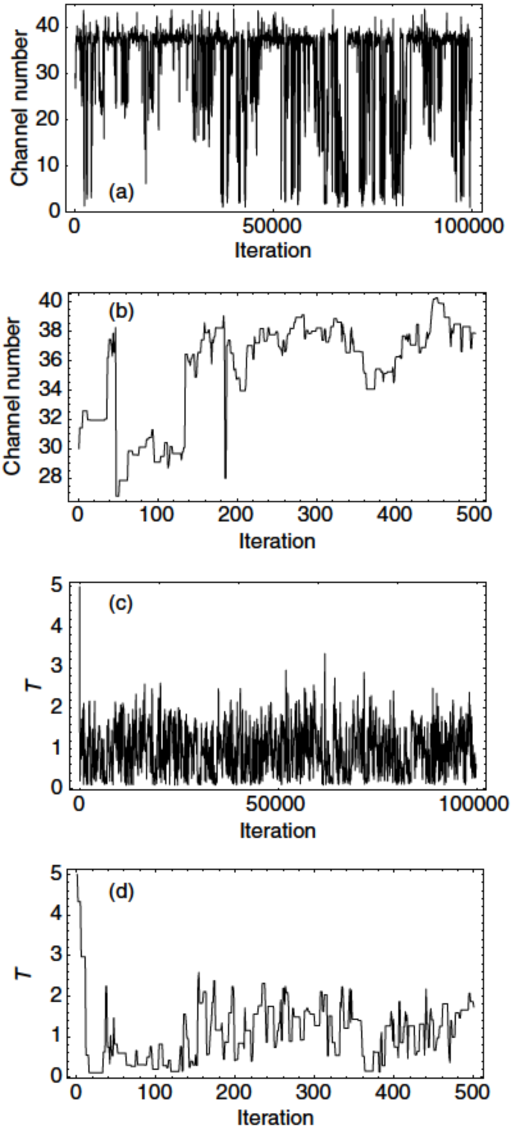

Figure 12.7 Results for 10^5 iterations of a parallel tempering Markov chain Monte Carlo. Panel (a) shows every 50th value of parameter ν and panel (c) the same for parameter T. Panels (b) and (d) show a blow-up of the first 500 iterations.

where β is the tempering parameter introduced in Section 12.4. Now take the derivative of $\ln[Z(\beta)]$.

$$\frac{d}{d\beta}\ln[Z(\beta)] = \frac{1}{Z(\beta)}\frac{d}{d\beta}Z(\beta) \tag{12.16}$$

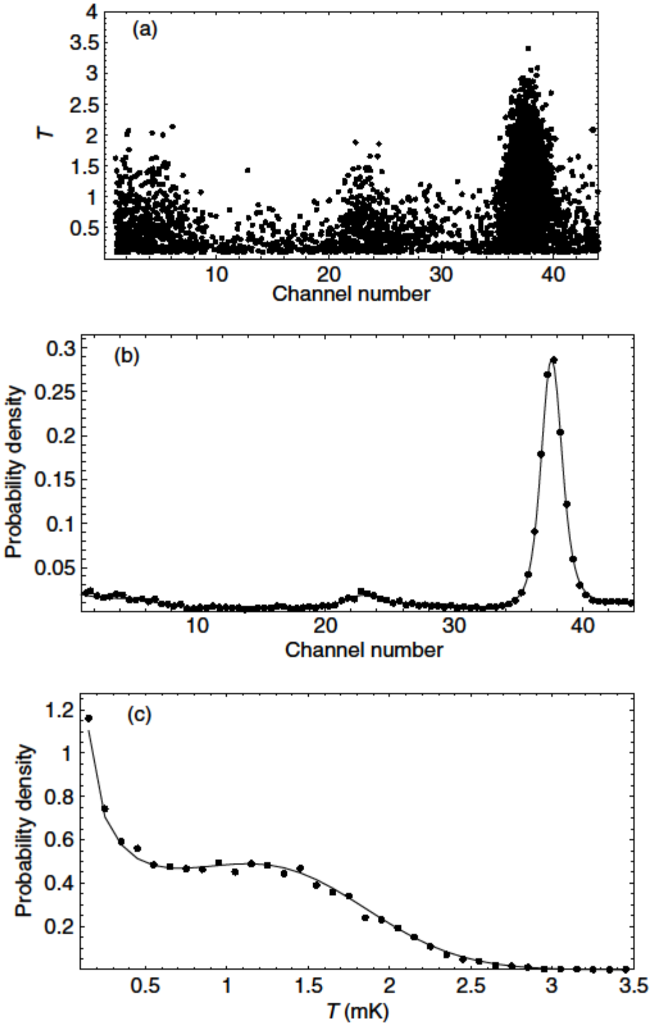

Figure 12.8 Results for the spectral line problem using a Markov chain Monte Carlo analysis with parallel tempering. Panel (a) is a scatter plot of the result for every 20th iteration in the two model parameters, channel number and line strength T. Panel (b) shows the marginal probability density for channel number (points) compared to our earlier numerical integration result indicated by the solid line. Panel (c) shows the marginal probability density for line strength T (points) compared to our earlier numerical integration result indicated by the solid line.

$$\begin{aligned}\frac{d}{d\beta}Z(\beta) &= \int dX \ln[p(D|M_1,X,I)] \\ &\quad \times \exp\{\ln[p(X|M_1,I)] + \beta\ln[p(D|M_1,X,I)]\} \\ &= \int dX \ln[p(D|M_1,X,I)]\; p(X|M_1,I)\; p(D|M_1,X,I)^{\beta}.\end{aligned} \tag{12.17}$$

Substituting Equation (12.17) into Equation (12.16), we obtain

$$\begin{aligned}\frac{d}{d\beta}\ln[Z(\beta)] &= \frac{\int dX \ln[p(D|M_1,X,I)]\; p(X|M_1,I)\; p(D|M_1,X,I)^{\beta}}{\int dX\; p(X|M_1,I)\; p(D|M_1,X,I)^{\beta}} \\ &= \langle \ln[p(D|M_1,X,I)]\rangle_{\beta},\end{aligned} \tag{12.18}$$

where $\langle \ln[p(D|M_1,X,I)]\rangle_{\beta}$ is the expectation value of the $\ln[p(D|M_1,X,I)]$. This quantity is easily evaluated from the MCMC results which consist of sets of X_t samples, one set for each value of the tempering parameter β. Let $\{X_{t,\beta}\}$ represent the samples for tempering parameter β.

$$\langle \ln[p(D|M_1,X,I)]\rangle_{\beta} = \frac{1}{n}\sum_t \ln[p(D|M_1,X_{t,\beta},I)], \tag{12.19}$$

where n is the number of samples in each set after the burn-in period. From Equation (12.18) we can write

$$\begin{aligned}\int_0^1 d\ln[Z(\beta)] &= \ln[Z(1)] - \ln[Z(0)] \\ &= \int d\beta\, \langle \ln[p(D|M_1,X,I)]\rangle_{\beta}.\end{aligned} \tag{12.20}$$

Now from Equation (12.15)

$$Z(1) = \int dX\; p(X|M_1,I)\; p(D|M_1,X,I) = p(D|M_1,I), \tag{12.21}$$

and

$$Z(0) = \int dX\; p(X|M_1,I). \tag{12.22}$$

From Equations (12.20) and (12.21) we can write

$$\ln[p(D|M_1,I)] = \ln[Z(0)] + \int d\beta \langle \ln[p(D|M_1,X,I)]\rangle_{\beta}. \tag{12.23}$$

For a normalized prior, $Z(0) = 1$ and Equation (12.23) becomes

$$\ln[p(D|M_1,I)] = \int d\beta \langle \ln[p(D|M_1,X,I)]\rangle_{\beta}. \tag{12.24}$$

Armed with Equation (12.24) we are now in a position to evaluate the Bayes factor given by Equation (12.13), which is at the heart of model comparison.

Returning to the spectral line problem,

$$\langle \ln[p(D|M_1,\nu,T,I)]\rangle_{\beta} = \frac{1}{n}\sum_t \ln[p(D|M_1,\nu_{t,\beta},T_{t,\beta},I)]. \tag{12.25}$$

We evaluated Equation (12.25) for the five values of $\beta = 0.01$, 0.2575, 0.505, 0.7525, 1.0 used in the PT MCMC analysis of Section 12.6. The results were $-97.51, -87.1937, -86.4973, -85.9128, -85.1565$, respectively. We then evaluated the integral in Equation (12.24) by generating an interpolating function and integrating the interpolating function in the interval 0 to 1. This yielded $\ln[p(D|M_1, I)] = -87.4462$. A more sophisticated interpolation of the results yielded $\ln[p(D|M_1, I)] = -87.3369$. Model M_2 had no free parameters and $p(D|M_2, I) = 1.133 \times 10^{-38}$ from Equation (3.49). The resulting Bayes factors for the two interpolations were $B_{1,2} = 0.93$ and 1.04, respectively. This should be compared to $B_{1,2} = 1.06$ obtained from our earlier solution to this problem.

12.8 Towards an automated MCMC

As the number of model parameters increases, so does the time required to choose a suitable σ value for each of the parameter proposal distributions. Suitable means that MCMC solutions, starting from different locations in the prior parameter space, yield equilibrium distributions of model parameter values that are not significantly different, in an acceptable number of iterations. Generally this involves running a series of chains, each time varying σ for one or more of the parameter proposal distributions, until the chain appears to converge on an equilibrium distribution with a proposal acceptance rate, λ, that is reasonable for the number of parameters involved, e.g., approximately 25% for a large number of parameters (Roberts, Gelman, and Gilks, 1997). This is especially time consuming if each parameter corresponds to a different physical quantity, so that the σ values can be very different. In this section, we describe one attempt at automating this process, which we apply to the detection of an extrasolar planet using some real astronomical data.

Suppose we are dealing with M parameters that are represented collectively by $\{X_\alpha\}$. Let σ_α represent the characteristic width of a symmetric proposal distribution for X_α. We will assume Gaussian proposal distributions but the general approach should also be applicable to other forms of proposal distributions. To automate the MCMC, we need to incorporate a control system that makes use of some form of error signal to steer the selection of the $\{\sigma_\alpha\}$.

For a manually controlled MCMC, a useful approach is to start with a large value of σ_α, approximately one tenth of the prior uncertainty of that parameter. In a PT MCMC, this will normally be sufficient to provide access to all areas with significant probability within the prior range, but may result in a very small acceptance rate for the $\beta = 1$ member of the PT MCMC chain. By running a number of smaller iteration chains, each time perturbing one or more of the $\{\sigma_\alpha\}$, it soon becomes clear which parameters are restraining the acceptance rate from a more desirable level. Larger $\{\sigma_\alpha\}$ values yield larger jumps in parameter proposal values. The general approach of refining the $\{\sigma_\alpha\}$ towards smaller values is analogous to an annealing operation. The refinement is terminated when the target proposal acceptance rate is reached.

In the automated version of this process described below, the error signal used for the control system is the difference between the current acceptance rate and a target acceptance rate. The control system steers the proposal σ's to desirable values during the burn-in stage of a single parallel tempering MCMC run. Although inclusion of the control system may result in a somewhat longer burn-in period, there is a huge saving in time because it eliminates many trial runs to manually establish a suitable set of $\{\sigma_\alpha\}$. In addition the control system error monitor provides another indication of the length of the burn-in period. In practice, it is important to repeat the operation for a few different choices of initial parameter values, to ensure that the MCMC results converge.

The automatic parallel tempering MCMC (APT MCMC) algorithm contains major and minor cycles. During the major cycles the current set of $\{\sigma_\alpha\}$ are used for n_1 iterations. The acceptance rate achieved during this major cycle is compared to the target acceptance rate. If the difference (control system error signal), ϵ, is greater than a chosen threshold, tol_1, then a set of minor cycles, one cycle of n_2 iterations for each σ_α, are employed to explore the sensitivity of the acceptance rate to each σ_α. The $\{\sigma_\alpha\}$ are updated and another major cycle run. If tol_1 is set $= 0$, then the minor cycles are always performed after each major cycle. At this point, the reader might find it useful to examine the evolution of the error signal, and the $\{\sigma_\alpha\}$, for the examples shown in Figures 12.12 and 12.13. One can clearly see the expected Poisson fluctuations in the error signal after the $\{\sigma_\alpha\}$ stabilize. For these examples we set $\text{tol}_1 = 1.5\sqrt{\lambda n_1}$ to reduce the number of minor cycles. Normally the control system is turned off after ϵ is less than some threshold, tol_2. Typically $\text{tol}_2 = \sqrt{\lambda n_1}$.

Full details of the control system are not included here as it is considered experimental and in a process of evolution. The latest version is included in the *Mathematica* tutorial in the section entitled "Automatic parallel tempering MCMC," along with useful default values for the algorithm parameters and input data format. Figure 12.9 provides a summary of the inputs and outputs for the APT MCMC algorithm. In the following section we demonstrate the behavior of the algorithm with a set of astronomical data used to detect an extrasolar planet.

12.9 Extrasolar planet example

In this section, we will apply the automated parallel tempering MCMC described in Section 12.8 to some real astronomical data, which were used to discover (Tinney *et al.*, 2003) an extrasolar planet orbiting a star with a catalog number HD 2039. Although light from the planet is too faint to be detected, the gravitational tug of the planet on the star is sufficient to produce a measurable Doppler shift in the velocity of absorption lines in the star's spectrum. By fitting a Keplerian orbit to the measured radial velocity data, v_i, it is possible to obtain information about the orbit and a lower limit on the mass of the unseen planet. The predicted model radial velocity, f_i, for a particular orbit is given below, and involves six unknowns. The geometry of a stellar orbit with respect to the observer is shown in Figure 12.10. The points labeled *F*, *P*,

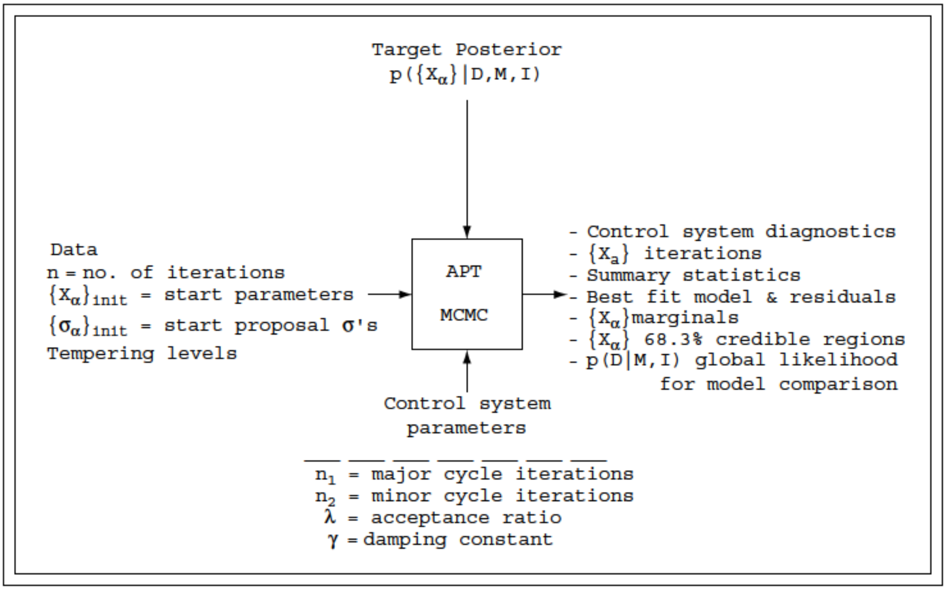
Target Posterior
p({Xα}|D,M,I)
Data
n = no. of iterations
{Xα}init = start parameters
{σα}init = start proposal σ's
Tempering levels
APT
MCMC
- Control system diagnostics
- {Xa} iterations
- Summary statistics
- Best fit model & residuals
- {Xα}marginals
- {Xα} 68.3% credible regions
- p(D|M,I) global likelihood
for model comparison
Control system
parameters
n1 = major cycle iterations
n2 = minor cycle iterations
λ = acceptance ratio
γ = damping constant

Figure 12.9 An overview schematic of the inputs and outputs for the automated parallel tempering MCMC.

and S, are the location of the focus of the elliptical orbit, periastron, and the star's position at time t_i, respectively.

$$f_i = V + K[\cos\{\theta(t_i + t_0) + \omega\} + e\cos\omega], \tag{12.26}$$

where

$V =$ the systematic velocity of the system.
$K =$ velocity amplitude $= 2\pi P^{-1}(1 - e^2)^{-1/2} a \sin i$.
$P =$ the orbital period.
$a =$ the semi-major axis of the orbit.
$e =$ the eccentricity of the elliptical orbit.
$i =$ the inclination of the orbit as defined in Figure 12.10.
$\omega =$ the longitude of periastron, angle LFA in Figure 12.10.
$\chi =$ the fraction of an orbit prior to the start of data-taking that periastron occurred at. Thus, $t_0 = \chi P =$ the number of days prior to $t_i = 0$ that the star was at periastron, for an orbital period of P days. At $t_i = 0$, the star is at an angle AFB from periastron.
$\theta(t_i + t_0) =$ the angle (AFS) of the star in its orbit relative to periastron at time t_i.

The dependence of θ on t_i, which follows from the conservation of angular momentum, is given by the solution of

$$\frac{d\theta}{dt} - \frac{2\pi[1 + e\cos\theta(t_i + t_0)]^2}{P(1 - e^2)^{3/2}} = 0. \tag{12.27}$$

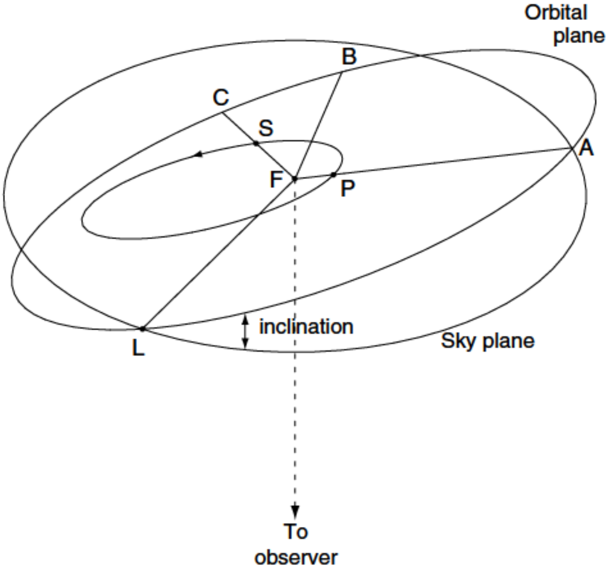

Figure 12.10 The geometry of a stellar orbit with respect to the observer. The sky plane is perpendicular to the dashed line connecting the star and the observer.

To fit Equation (12.26) to the data, we need to specify the six model parameters, P, K, V, e, ω, χ.

The measured radial velocities and their errors are shown Figure 12.11. As we have discussed before, it is good idea not to assume that the quoted measurement errors are the only error component in the data.

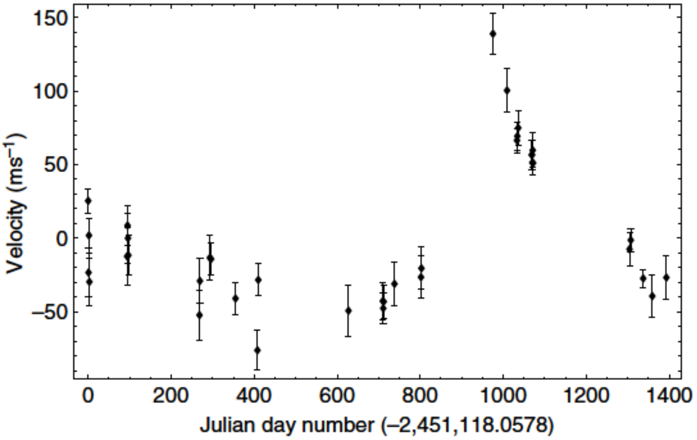

Figure 12.11 HD 2039 radial velocity measurements plotted from the data given in Tinney *et al.*, (2003).

We can represent the measured velocities by the equation

$$v_i = f_i + e_i, \tag{12.28}$$

where e_i is the component of v_i which arises from measurement errors plus any real signal in the data that cannot be explained by the model prediction f_i. For example, suppose that the star actually has two planets, and the model assumes only one is present. In regard to the single planet model, the velocity variations induced by the second planet act like an additional unknown noise term. In the absence of detailed knowledge of the effective noise distribution, other than that it has a finite variance, the maximum entropy principle tells us that a Gaussian distribution would be the most conservative choice (i.e., maximally non-committal about the information we don't have). We will assume the noise variance is finite and adopt a Gaussian distribution for e_i with a variance σ_i^2.

In a Bayesian analysis where the variance of e_i is unknown, but assumed to be the same for all data points, we can treat σ as an unknown nuisance parameter. Marginalizing over σ has the desirable effect of treating anything in the data that can't be explained by the model as noise and this leads to the most conservative estimates of model parameters.

In the current problem, the quoted measurement errors are not all the same. We let $s_i =$ the experimenter's estimate of σ_i, prior to fitting the model and examining the model residuals. The σ_i values are not known, but the s_i values are our best initial estimates. They also contain information on the relative weight we want to associate with each point. Since we do not know the absolute values of the σ_i, we introduce a parameter called the noise scale parameter, b, to allow for this.[3] It could also be called a noise weight parameter. Several different definitions of b are possible including $\sigma_i^2 = bs_i^2$ and $\sigma_i = bs_i$. The definition we use here is given by

$$\frac{1}{\sigma_i^2} = \frac{b}{s_i^2}. \tag{12.29}$$

Again marginalizing over b has the desirable effect of treating anything in the data that can't be explained by the model as noise, leading to the most conservative estimates of orbital parameters. Since b is a scale parameter, we assume a Jeffreys prior (see Section 3.10).

[3] Note added in proof:
A better choice for parameterizing any additional unknown noise term is to rewrite equation (12.28) as

$$v_i = f_i + e_i + e_0$$

where e_i is the noise component arising from known but unequal measurement errors, and e_0 is the additional unknown noise term. From the arguments given above, we can characterize the combination of $e_i + e_0$ by a Gaussian distribution with variance $= \sigma_i^2 + \sigma_0^2$. With this form of parameterization we would marginalize over σ_0 instead of b. The one advantage of using b is that it can allow for the possibility that the measurement errors have been overestimated.

$$p(b|I) = \frac{1}{b \ln \frac{b_{\max}}{b_{\min}}}, \tag{12.30}$$

with $b_{\max} = 2$ and $b_{\min} = 0.1$. We also compute $p(b|D, \text{Model}, I)$. If the most probable estimate of $b \approx 1$, then the one-planet model is doing a good job accounting for everything that is not noise based on the s_i estimates. If $b < 1$, then either the model is not accounting for significant real features in the data or the initial noise estimates, s_i, were low.

12.9.1 Model probabilities

In this section, we set up the equations needed to (a) specify the joint posterior probability of the model parameters (parameter estimation problem) for use in the MCMC analysis, and (b) decide if a planet has been detected (model selection problem). To decide if a planet has been detected, we will compare the probability of $M_1 \equiv$ "the star's radial velocity variations are caused by one planet" to the probability of $M_0 \equiv$ "the radial velocity variations are consistent with noise." From Bayes' theorem, we can write

$$p(M_1|D,I) = \frac{p(M_1|I)\; p(D|M_1,I)}{p(D|I)} = C\; p(M_1|I)\; p(D|M_1,I), \tag{12.31}$$

where

$$\begin{aligned} p(D|M_1,I) = &\int dP \int dK \int dV \int de \int d\chi \int d\omega \int db\, p(P,K,V,e,\chi,\omega,b|M_1,I) \\ &\times p(D|M_1,P,K,V,e,\chi,\omega,b,I). \end{aligned} \tag{12.32}$$

The joint prior for the model parameters, assuming independence, is given by

$$\begin{aligned} p(P,K,V,e,\chi,\omega,b|M_1,I) = &\frac{1}{P \ln\left(\frac{P_{\max}}{P_{\min}}\right)} \frac{1}{K \ln\left(\frac{K_{\max}}{K_{\min}}\right)} \frac{1}{(V_{\max} - V_{\min})} \\ &\times \frac{1}{(e_{\max} - e_{\min})} \frac{1}{2\pi} \frac{1}{b \ln\left(\frac{b_{\max}}{b_{\min}}\right)}. \end{aligned} \tag{12.33}$$

Note: we have assumed a uniform prior for χ in the range 0 to 1, so $p(\chi|M_1,I) = 1$.

$$p(D|M_1,P,K,V,e,\chi,\omega,b,I) = Ab^{N/2} \times \exp\left[-\frac{b}{2}\sum_{i=1}^{N}\frac{(v_i - f_i)^2}{s_i^2}\right], \tag{12.34}$$

where

$$A = (2\pi)^{-N/2}\left[\prod_{i=1}^{N} s_i^{-1}\right]. \tag{12.35}$$

For the purposes of estimating the model parameters, we will assume a prior uncertainty in b in the range $b_{\min} = 0.1$ and $b_{\max} = 2$.

When it comes to comparing the probability of M_1 to M_0, or to a model which assumes there are two planets present, we will set $b = 1$ and perform the model comparison based on the errors quoted in Tinney *et al.*, (2003). The probability of M_0 is given by

$$p(M_0|D, I) = C\, p(M_0|I) p(D|M_0, I), \tag{12.36}$$

where

$$\begin{aligned} p(D|M_0, I) &= \int db \int dV\, p(V, b|D, M_0, I) \\ &= \int db \int dV\, p(V, b|M_0, I)\; p(D|M_0, V, b, I), \end{aligned} \tag{12.37}$$

$$p(V|M_0, I) = \frac{1}{(V_{\max} - V_{\min})} \frac{1}{b \ln\left(\frac{b_{\max}}{b_{\min}}\right)}, \tag{12.38}$$

and

$$p(D|M_0, V, b, I) = (2\pi)^{-N/2} \left[\prod_{i=1}^{N} s_i^{-1}\right] b^{\frac{N}{2}} \exp\left[-\frac{b}{2} \sum_{i=1}^{N} \frac{(v_i - V)^2}{s_i^2}\right]. \tag{12.39}$$

The integral over V in Equation (12.37) can be performed analytically yielding

$$\begin{aligned} p(D|M_0, I) = A \sqrt{\frac{\pi}{2}} W^{-1/2} \int db\, b^{\frac{N-3}{2}} \exp\left[-\frac{bW}{2} \sum_{i=1}^{N} (\overline{v_w^2} - (\overline{v_w})^2)^2\right] \\ \times [\operatorname{erf}(u_{\max}) - \operatorname{erf}(u_{\min})], \end{aligned} \tag{12.40}$$

where

$$\overline{v_w} = \sum_{i=1}^{N} w_i\; v_i, \tag{12.41}$$

$$\overline{v_w^2} = \sum_{i=1}^{N} w_i\; v_i^2, \tag{12.42}$$

$$w_i = 1/s_i^2, \tag{12.43}$$

$$W = \sum_{i=1}^{N} w_i, \tag{12.44}$$

$$u_{\max} = \left(\frac{bW}{2}\right)^{-1/2} (V_{\max} - \overline{v_w}), \tag{12.45}$$

$$u_{\min} = \left(\frac{bW}{2}\right)^{-1/2} (V_{\min} - \overline{v_w}). \tag{12.46}$$

In conclusion, Equations (12.31) and (12.34) are required for the parameter estimation part of the problem, and Equations (12.32) and (12.40) answer the model selection part of the problem. Equation (12.32) is evaluated from the results of the parallel tempering chains according to the method discussed in Section 12.7.

12.9.2 Results

The APT MCMC algorithm described in Section 12.8 was used to re-analyze the measurements of Tinney *et al.* (2003). Figures 12.12 and 12.13 show the diagnostic information output by the MCMC control system for two runs of the APT MCMC algorithm that use different starting values for the parameters and different starting values for the proposal σ's. The top left panel shows the evolution of the control system error for 100 000 iterations. Even for the best set of $\{\sigma_\alpha\}$, the control system error will exhibit statistical fluctuations of order $\sqrt{\lambda n_1}$ which will result in fluctuations of $\{\sigma_\alpha\}$ throughout the run. Recall, $\lambda =$ the target acceptance fraction and $n_1 =$ the number of iterations in major cycles (see Section 12.8). These fluctuations are of no consequence since the equilibrium distribution of parameter values is insensitive to small fluctuations in $\{\sigma_\alpha\}$. To reduce the time spent in perturbing $\{\sigma_\alpha\}$ values, we set a threshold on the control system error of $1.5\sqrt{\lambda n_1}$. When the error is less than this value no minor cycles are executed. Normally, the control system is disabled the first time the error is $<1.5\sqrt{\lambda n_1}$. This was not done in the two examples shown in order to illustrate the behavior of the control system and evolution of the $\{\sigma_\alpha\}$. For the two runs, the error drops to a level consistent with the minimum threshold set for initiating a change in $\{\sigma_\alpha\}$ in 8000 and 9600 iterations, respectively. The other six panels exhibit the evolution of the $\{\sigma_\alpha\}$ to relatively stable values. Table 12.1 compares the starting and final values for two APT MCMC runs with a set of $\{\sigma_\alpha\}$ values arrived at manually. The starting parameter values for the two APT MCMC runs are shown in Table 12.2. Control system parameters were: $\mathrm{sc}_{\max} = 0.1$, $n_1 = 1000$, $n_2 = 100$, $\lambda = 0.25$, and a damping factor, $\gamma = 1.6$. $\mathrm{sc}_{\max}$ specifies the maximum scaling of $\{\sigma_\alpha\}$ to be used in a minor cycle. Tempering β values used were $\{0.01, 0.2575, 0.505, 0.7525, 1\}$. β values are chosen to give $\simeq$50% swap acceptance between adjacent levels.

Figure 12.14 shows the iterations of the six model parameters, P, K, V, e, χ, ω, for the 100 000 iterations of APT MCMC 1. Only every 100th value is plotted. The plot for K shows clear evidence that parallel tempering is doing its job, enabling regions of significant probability to be explored apart from the biggest peak region. A conservative burn-in period of 8000 samples was arrived at from an examination of the

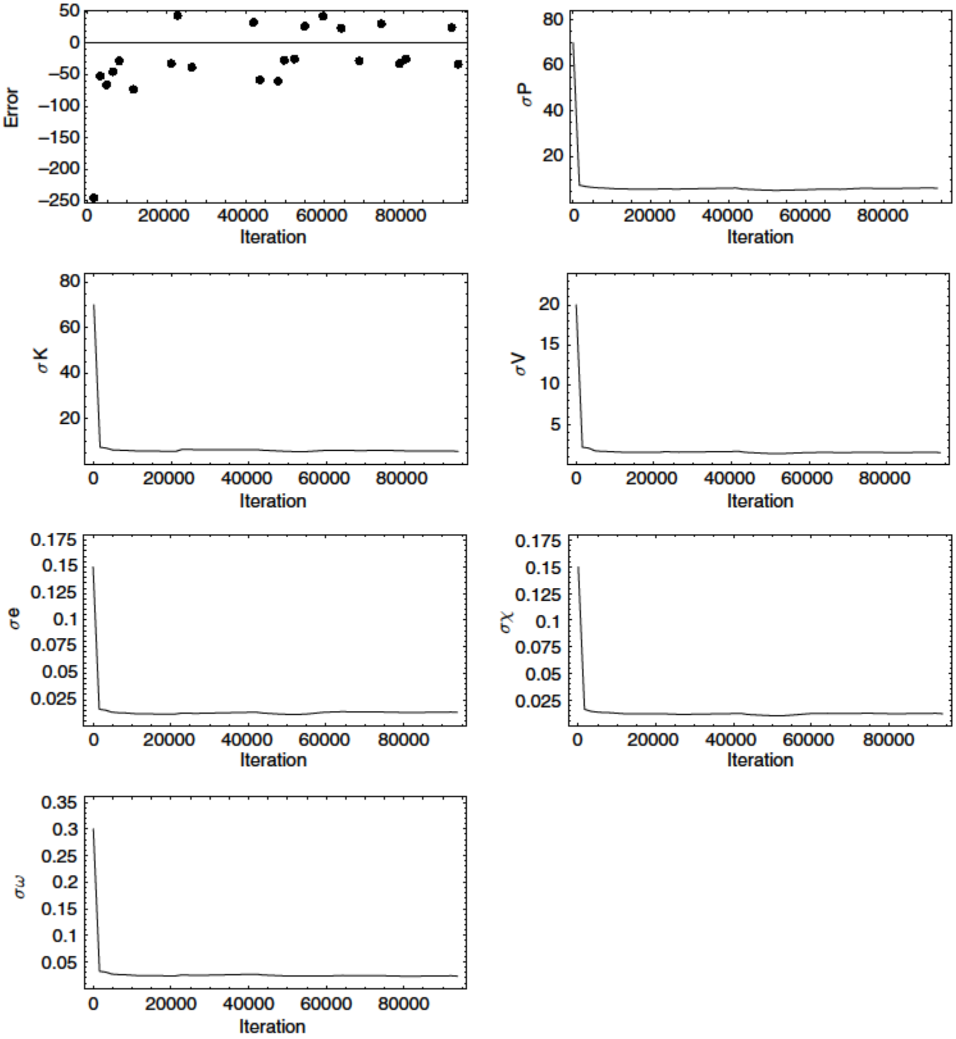

Figure 12.12 The upper left panel shows the evolution of the APT MCMC control system error versus iteration number for the first run. The other six panels exhibit the evolution of the Gaussian parameter proposal distribution σ's.

control system error, shown in the upper left panel of Figure 12.12, and the parameter iterations of Figure 12.14.

The level of agreement between two different MCMC runs can be judged from a comparison of the marginal distributions of the parameters. Figures 12.15 and 12.16 show the posterior marginals for the six model parameters, P, K, V, e, χ, ω, and the noise scale parameter b for APT MCMC 1 and APT MCMC 2, respectively. The final model

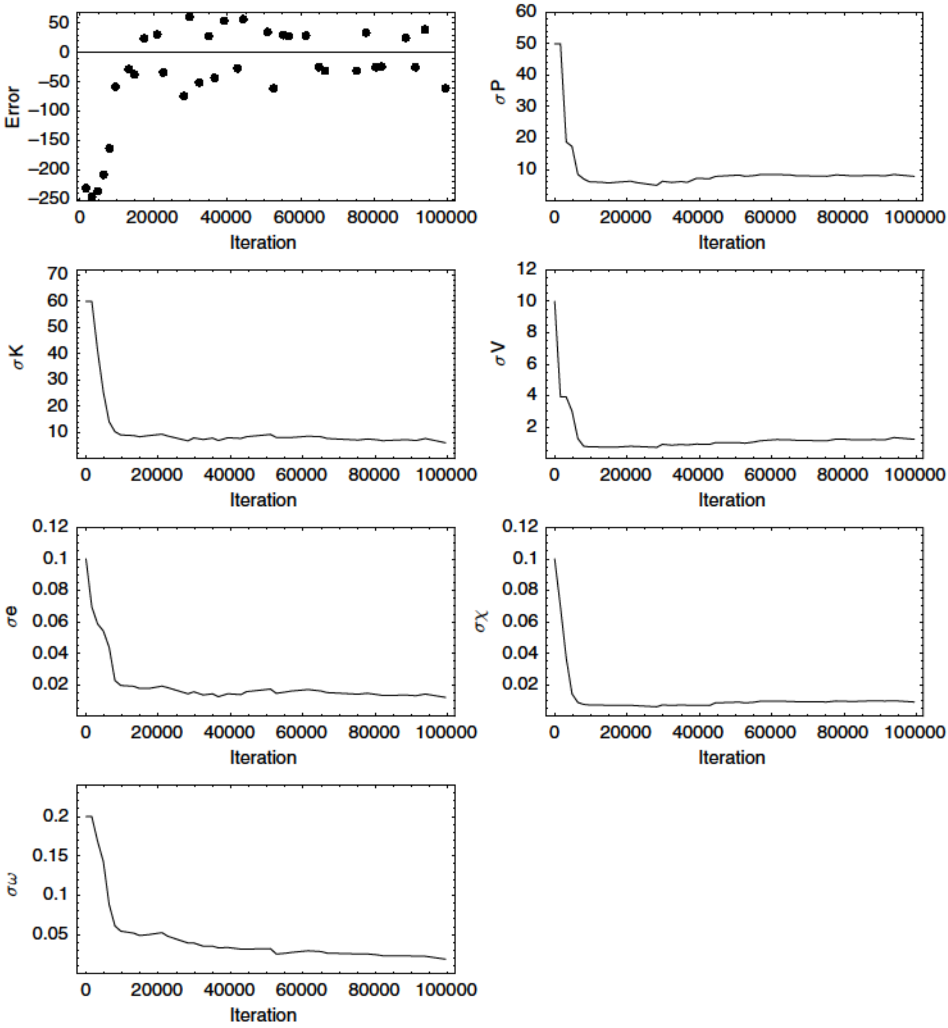

Figure 12.13 The upper left panel shows the evolution of the APT MCMC control system error versus iteration number for the second run. The other six panels exhibit the evolution of the Gaussian parameter proposal distribution σ's.

parameter values are given in Table 12.3, along with values of $a \sin i$, $M \sin i$, and the Julian date of periastron passage, that were derived from the parameter values.

$$a \sin i(\text{km}) = 1.38 \times 10^5 KP\sqrt{1 - e^2}, \tag{12.47}$$

where K is in units of m s^{-1} and P is in days.

$$M \sin i = 4.91 \times 10^{-3} (M_*)^{2/3} KP^{1/3}\sqrt{1 - e^2}, \tag{12.48}$$

Table 12.1 *Comparison of the starting and final values of proposal distribution σ's for two automatic parallel tempering MCMC runs, to manually derived values.*

Proposal σ	APT MCMC 1		APT MCMC 2		Manual
	Start	Final	Start	Final	Final
σP (days)	70	6.2	50	7.8	10
σK(m s^{-1})	70	5.7	60	6.0	5
σV(m s^{-1})	20	1.5	10	1.2	2
σe	0.15	0.012	0.1	0.012	0.005
$\sigma\chi$	0.15	0.013	0.1	0.009	0.007
$\sigma\omega$	0.3	0.023	0.2	0.019	0.05

Table 12.2 *Starting parameter values for the two automatic parallel tempering MCMC runs.*

Trial	P	K	V	e	χ	ω	b
1	950	80	−2	0.4	0.0	0.0	1.0
2	1300	250	5	0.2	0.0	0.0	1.0

where M is the mass of the planet measured in Jupiter masses, and M_* is the mass of the star in units of solar masses.

One important issue concerns what summary statistic to use to represent the best estimate of the parameter values. We explore the question of a suitable robust summary statistic further in Section 12.10. In Table 12.3, the final quoted parameter values correspond to the MAP values. The median values are shown in brackets below. The error bars correspond to the boundaries of the 68.3% credible region of the marginal distribution. The MAP parameter values for APT MCMC 1 were used to construct the model plotted in panel (a) of Figure 12.17. The residuals are shown in panel (b).

Figure 12.18 shows the posterior probability distributions for $a \sin i$, $M \sin i$, and the Julian date of periastron passage, that are derived from the MCMC samples of the orbital parameters.

The Bayes factors, $p(D|M_1, I)/p(D|M_0, I)$, determined from the two APT MCMC runs were 1.4×10^{14} and 1.6×10^{14}. Clearly, both trials overwhelmingly favor M_1 over M_0.

The upper panel of Figure 12.19 shows a comparison of the marginal and projected probability density functions for the velocity amplitude, K, derived from the APT MCMC parameter samples. To understand the difference, it is useful to examine the strong correlation that is evident between K and orbital eccentricity in the lower panel.

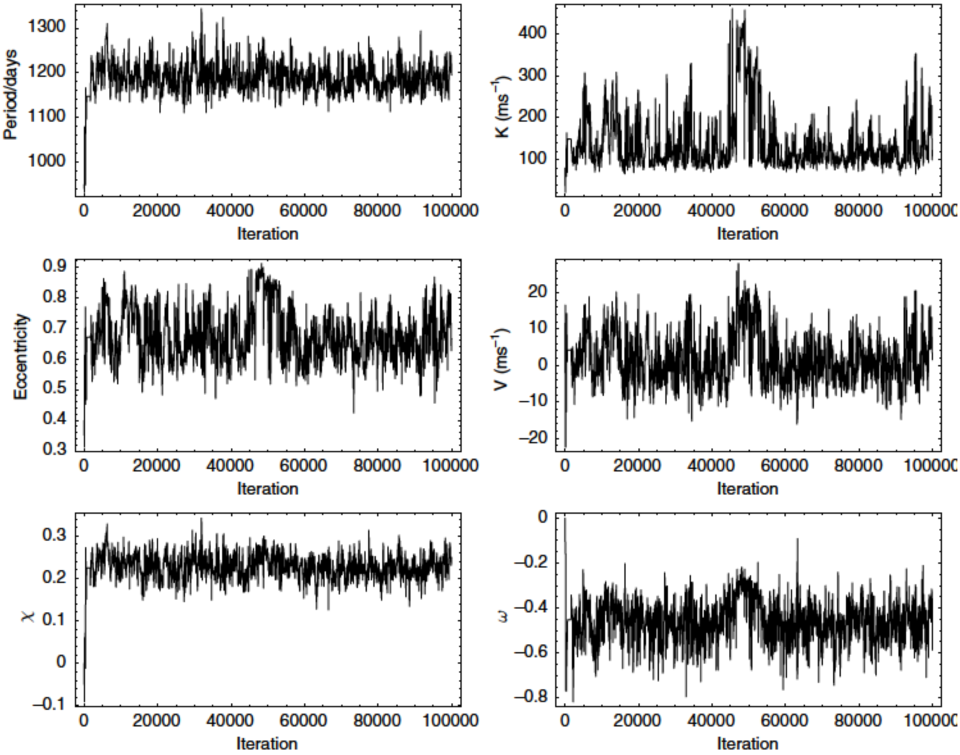

Figure 12.14 The figure shows every 100th APT MCMC iteration of the six model parameters, P, K, V, e, χ, ω.

Not only is the density of samples much higher at low K values, but the characteristic width of the K sample distribution is also much broader, giving rise to an enhancement in the marginal beyond that seen in the projected.

Finally, even though the 68.3% credible region contains $b = 1$, we decided to analyze the best-fit residuals, shown in the lower panel of Figure 12.17, to see what probability theory had to say about the evidence for another planet.[4] The APT MCMC program was re-run on the residuals to look for evidence of a second planet in the period range 2 to 500 days, $K = 1$ to $40\,\mathrm{m\ s^{-1}}$, $V = -10$ to $10\,\mathrm{m\ s^{-1}}$, $e = 0$ to 0.95, $\chi = 0$ to 1, and $\omega = -\pi$ to π. The most probable orbital solution had a period of 11.90 ± 0.02 days, $K = 18^{+9}_{-15}\,\mathrm{m\ s^{-1}}$, $V = -2.7^{+2.4}_{-1.6}\,\mathrm{m\ s^{-1}}$, eccentricity $= 0.626^{+0.16}_{-0.18}$, $\omega = 156^{+2}_{-4}$ deg, periastron passage $= 1121 \pm 1$ days (JD $-2{,}450{,}000$), and an $M \sin i = 0.14^{+0.07}_{-0.04}$. Figure 12.20 shows this orbital solution overlaid on the residuals for two cycles of phase. Note: the second cycle is just a repeat of the first. The computed Bayes factor $p(D|M_2, I)/p(D|M_1, I) = 0.7$. Assuming *a priori* that

[4] Note: a better approach would be to fit a two-planet model to the original radial velocity data.

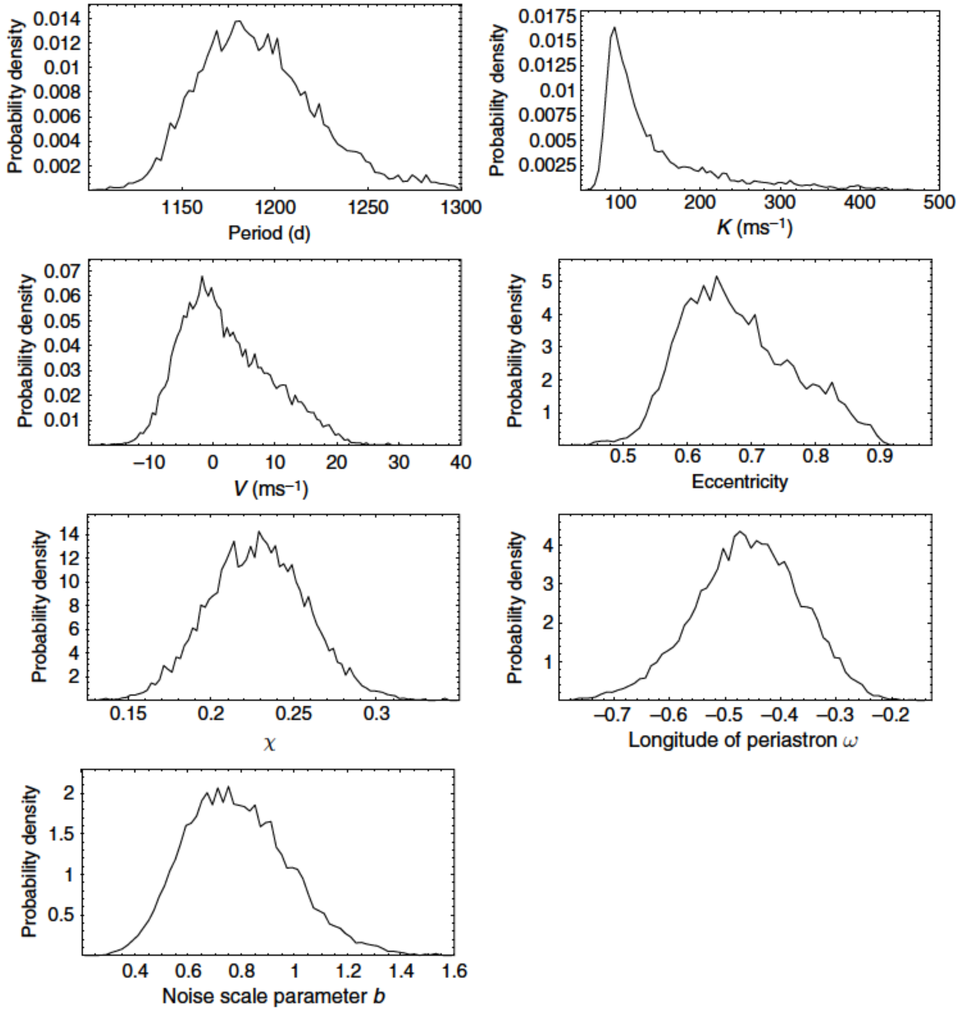

Figure 12.15 The marginal probabilities for the six model parameters, P, K, V, e, χ, ω, and the noise scale parameter b for the run APT MCMC 1.

$p(M_2|I) = p(M_1|I)$, this result indicates that it is more probable that the orbital solution for the residuals arises from fitting a noise feature than from the existence of a second planet. Thus, there is insufficient evidence at this time to claim the presence of a second planet.

12.10 MCMC robust summary statistic

In the previous section, the best estimate of each model parameter is based on the maximum *a posteriori* (MAP) value. It has been argued, e.g., Fox and Nicholls (2001),

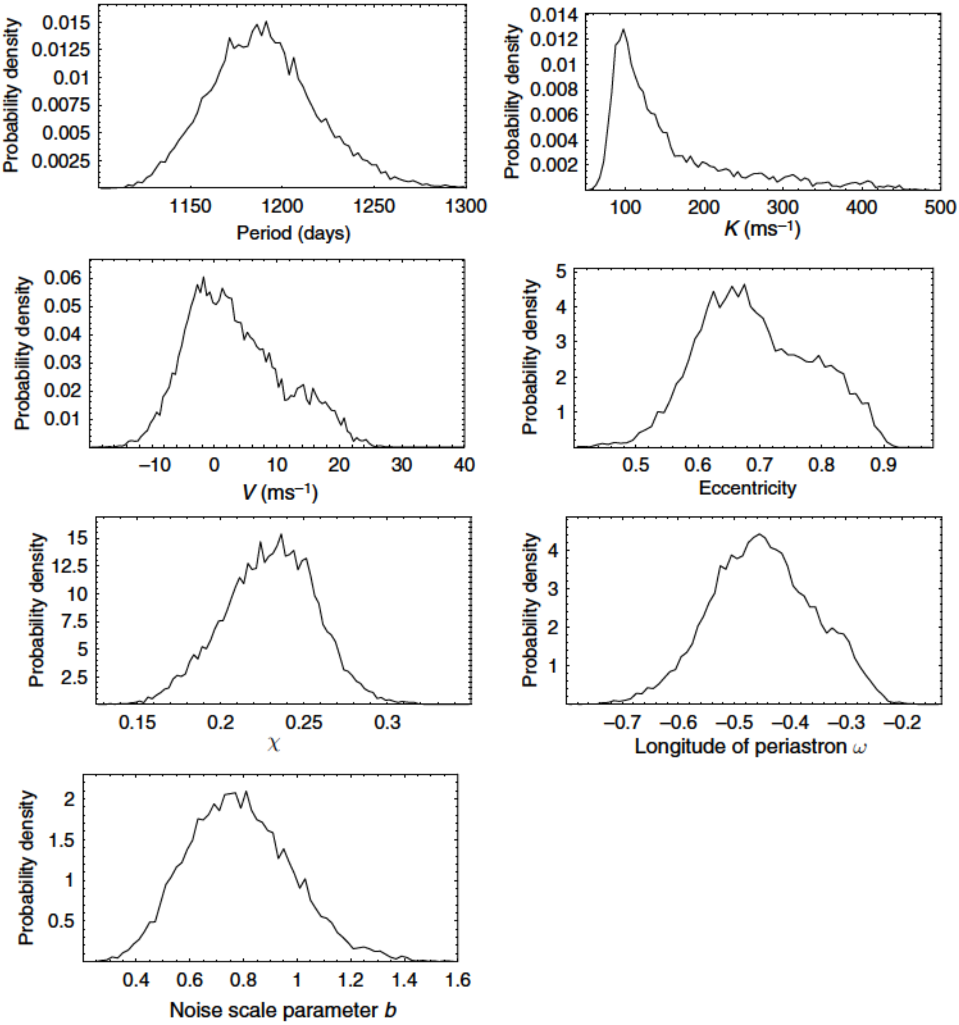

Figure 12.16 The marginal probabilities for the six model parameters, P, K, V, e, χ, ω, and the noise scale parameter b for the run APT MCMC 2.

that MAP values are sometimes unrepresentative of the bulk of the posterior probability. Fox and Nicholls were considering the reconstruction of degraded binary images. The current problem is very different but the issue remains the same: what choice of summary statistic to use? Two desirable properties are: a) that it be representative of the marginal probability distribution, and b) the set of summary parameter values provides a good fit to the data. Here, we consider three other possible choices of summary statistic. They are the mean, the median, and the marginal posterior mode (MPM), all of which satisfy point (a). In repeated APT MCMC runs, it was found that

Table 12.3 *Comparison of the results from two parallel tempering MCMC Bayesian runs with the analysis of Tinney et al. (2003). The values quoted for the two APT MCMC runs are MAP (maximum a posterior) values. The error bars correspond to the boundaries of the 68.3% credible region of the marginal distribution. The median values are given in brackets on the line below. Note: the periastron time and error quoted by Tinney et al. is identical with their P value and is assumed to be a typographical error.*

Parameter	Tinney *et al.*	APT MCMC	APT MCMC
	(2003)	1	2
Orbital period P (days)	1183 ± 150	1188^{+28}_{-35} (1188)	1177^{+36}_{-21} (1188)
Velocity amplitude K (m s^{-1})	130 ± 20	106^{+46}_{-29} (115)	116^{+56}_{-39} (125)
Eccentricity e	0.67 ± 0.1	$0.63^{+0.12}_{-0.06}$ (0.67)	$0.65^{+0.15}_{-0.06}$ (0.68)
Longitude of periastron ω (deg)	333 ± 15	333^{+6}_{-5} (334)	332^{+8}_{-3} (334)
$a \sin i$ (units of 10^6 km)	1.56 ± 0.3	$1.35^{+0.4}_{-0.3}$ (1.42)	$1.4^{+0.4}_{-0.4}$ (1.66)
Periastron time (JD −2,450,000)	1183 ± 150	864^{+18}_{-58} (845)	856^{+52}_{-28} (844)
Systematic velocity V (m s^{-1})		-0.7^{+8}_{-6} (0.8)	1.4^{+7}_{-7} (2.1)
$M \sin i$ (M_J)	4.9 ± 1.0	$4.2^{+1.2}_{-1.0}$ (4.5)	$4.5^{+1.3}_{-1.3}$ (4.7)
RMS about fit	15	13.8 (14.1)	14.0 (14.0)

the MPM solution provided a relatively poor fit to the data, while the mean was somewhat better, and in all cases, the median provided a good fit – almost as good as the MAP fits. One example of the fits is shown in Figure 12.21. The residuals were as follows: (a) 14.0 m s^{-1} (MAP), (b) 16.1 (mean), (c) 18.7 (MPM), and (d) 14.0 (median).

In the previous example the Bayes factor favored the one-planet model, M_1, compared to the no-planet model, M_0, by a factor of approximately 10^{14}. It is also

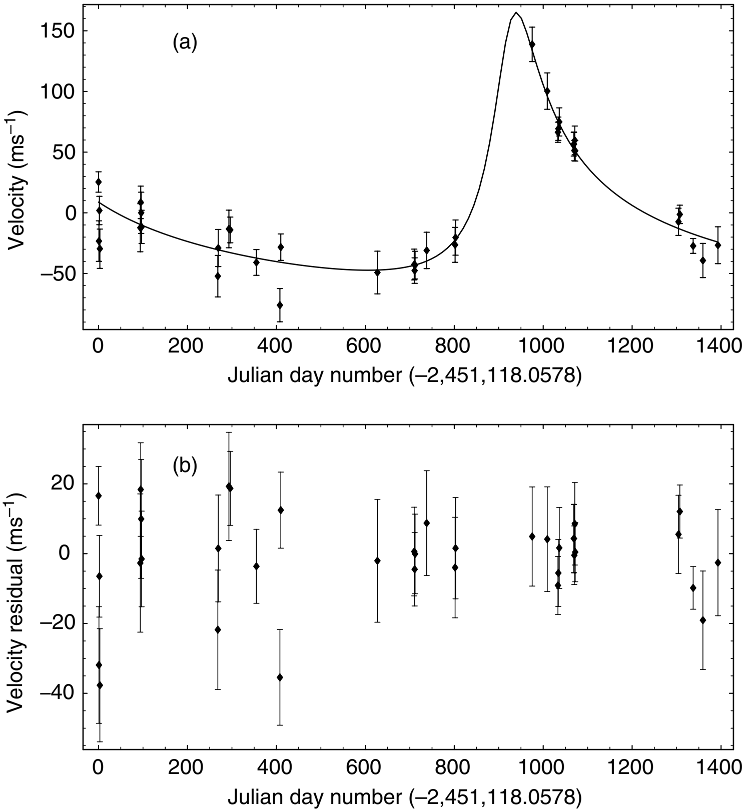

Figure 12.17 Panel (a) shows the raw data with error bars plotted together with the model radial velocity curve using the MAP (maximum *a posteriori*) summary statistic. Panel (b) shows the radial velocity residuals.

interesting to compare the four different summary statistics in the case where the Bayes factor is close to 1, as we found for the toy spectral line problem in Section 12.7, i.e., neither model is preferred. Figure 12.22 shows a comparison of the fits obtained using (a) the MAP, (b) the mean, (c) the MPM, and (d) the median. Both the MAP and median summary statistic placed the model line at the actual location of the simulated spectral line (channel 37). The MAP achieved a slightly lower RMS residual (RMS = 0.87) compared to the median (RMS = 0.89). The mean statistic performed rather poorly and the MPM not much better.

The conclusion, based on the current studies, is that the median statistic provides a robust alternative to the common MAP statistic for summarizing the posterior distribution. Unfortunately, the median was not one of the statistics considered in Fox and Nicholls (2001).

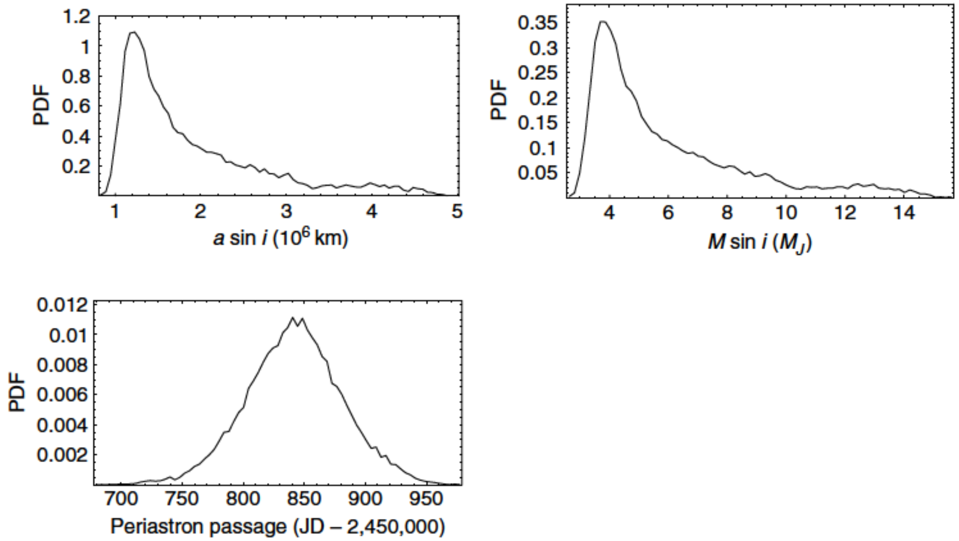

Figure 12.18 The figure shows the distribution of three useful astronomical quantities; $a \sin i$, $M \sin i$ and epoch of periastron passage, that are derived from the MCMC samples of the orbital parameters.

12.11 Summary

This chapter provides a brief introduction to the powerful role MCMC methods can play in a full Bayesian analysis of a complex inference problem involving models with large numbers of parameters. We have only demonstrated their use for models with a small to a moderate number of parameters, where they can easily be compared with results from other methods. These comparisons will provide a useful introduction and calibration of these methods for readers wishing to handle more complex problems. For the examples considered, the median statistic proved to be a robust alternative to the common MAP statistic for summarizing the MCMC posterior distribution.

The most ambitious topic treated in this chapter dealt with an experimental new algorithm for automatically annealing the σ values for the parameter proposal distributions in a parallel tempering Markov chain Monte Carlo (APT MCMC) calculation. This was applied to the analysis of a set of astronomical data used in the detection of an extrasolar planet. Existing analyses are based on the use of nonlinear least-squares methods which typically require a good initial guess of the parameter values (see Section 11.5). Frequently, the first indication of a periodic signal comes from a periodogram analysis of the data. As we show in the next chapter, a Bayesian analysis based on prior information of the shape of the periodic signal can frequently do a better job of detection than the ordinary Fourier power spectrum, otherwise known as the Schuster periodogram. In the extrasolar planet Kepler problem, the mathematical form of the signal is well known and is built into the Bayesian analysis. The APT

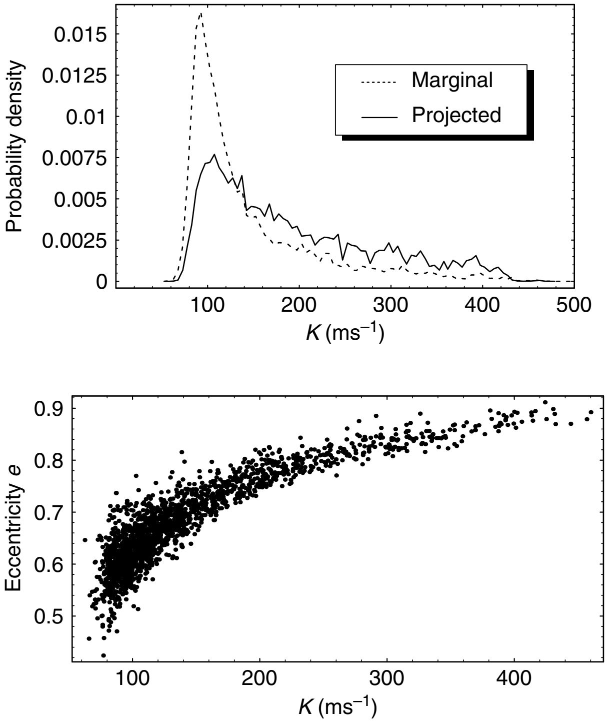

Figure 12.19 The upper panel shows a comparison of the marginal and projected probability density functions for the velocity amplitude, K. The lower panel illustrates the strong correlation between K and orbital eccentricity.

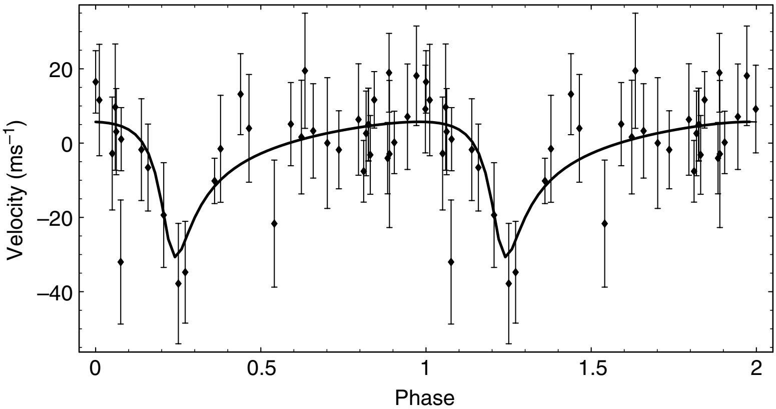

Figure 12.20 The figure shows the most probable orbital solution to the data residuals (for two cycles of phase), after removing the best fitting model of the first planet.

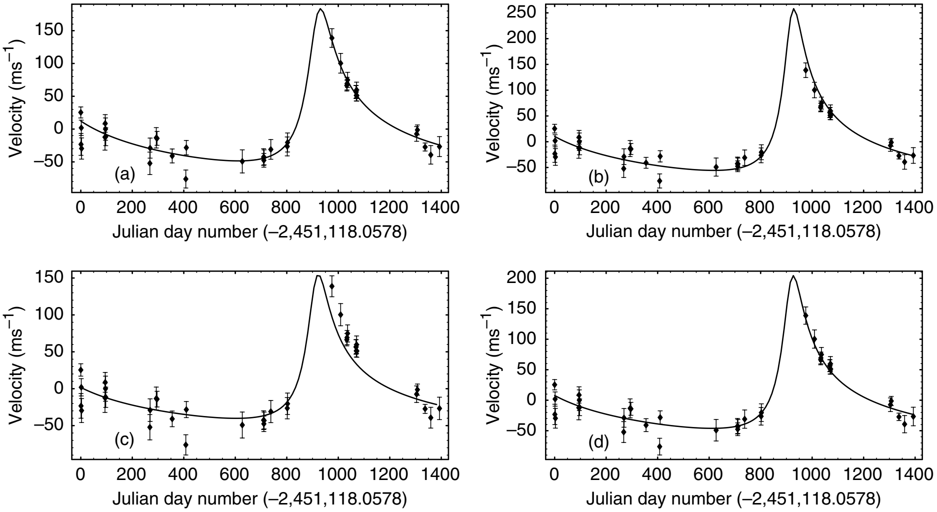

Figure 12.21 The four panels illustrate typical fits obtained in the extrasolar planet problem using different choices of summary statistic to represent the MCMC parameter distributions. They correspond to: (a) the MAP (maximum *a posteriori*), (b) the mean, (c) the MPM (marginal posterior mode), and (d) the median.

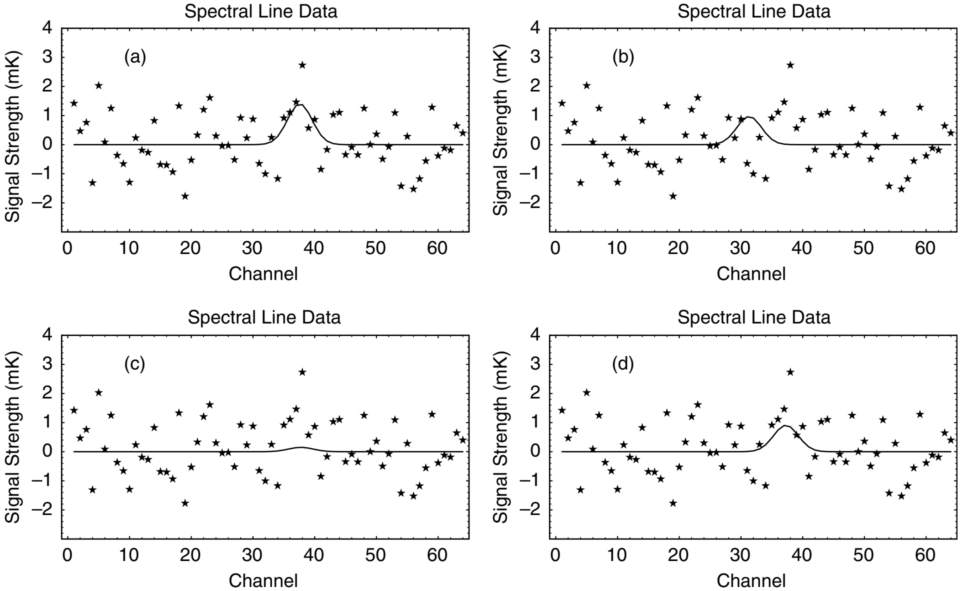

Figure 12.22 The four panels illustrate the fits obtained in the toy spectral line problem using different choices of summary statistic to represent the MCMC parameter distributions. They correspond to: (a) the MAP (maximum *a posteriori*), (b) the mean, (c) the MPM (marginal posterior mode), and (d) the median.

MCMC algorithm implemented in Section 12.9 is thus effective for both detecting and characterizing the orbits of extrasolar planets. Another advantage is that a good initial guess of the orbital parameter values is not required, which allows for the earliest possible detection of a new planet. Moreover, the built-in Occam's razor in the Bayesian analysis can save a great deal of time in deciding whether a detection is believable.

Finally, it is important to remember that the MCMC techniques described in this chapter are basically tools to allow us to evaluate the integrals needed for a full Bayesian analysis of some problem of interest. The APT MCMC algorithm discussed in the context of the extrasolar planet problem can readily be modified to tackle other very different problems.

12.12 Problems

1. In Section 12.6, we used both the Metropolis–Hastings and parallel tempering (PT) versions of MCMC to re-analyze the toy spectral line problem of Section 3.6. A program to perform the PT calculations is given in the Markov chain Monte Carlo section of the *Mathematica* tutorial. Use this program to analyze the spectrum given in Table 12.4, for $n = 10\,000$ to $50\,000$ iterations, depending on the speed of your computer. As part of your solution, recompute Figures 12.7, 12.8, and the Bayes factor used to compare the two competing models. Explain how you arrived at your choice for the number of burn-in samples.

   The prior information is the same as that assumed in Section 12.6. Theory predicts the spectral line has a Gaussian shape with a line width $\sigma_L = 2$ frequency channels. The noise in each channel is known to be Gaussian with a $\sigma = 1.0\,\text{mK}$ and the spectrometer output is in units of mK.
2. Repeat the analysis of problem 1 with the following changes. In addition to the unknown line strength and center frequency, the line width is also uncertain. Assume a uniform prior for the line width, with upper and lower bounds of 0.5 and 4 frequency channels, respectively. You will need to modify the parallel tempering MCMC program to allow for the addition of the line width parameter. Experiment with your choice of σ, for the line width in the Gaussian proposal distribution, to obtain a reasonable value for the acceptance rate somewhere in the range 0.25 to 0.5. Your solution should include a plot of the marginal probability distribution for each of the three parameters and a calculation of the Bayes factor for comparing the two models. Justify your choice for the number of burn-in samples.
3. Carry out the analysis described in problem 2 by modifying the experimental APT MCMC software provided in the *Mathematica* tutorial, and discussed in Section 12.8.
4. In Section 11.6, we illustrated the solution of a simple nonlinear model fitting problem using *Mathematica*'s **NonlinearRegress**, which implements the Levenberg–Marquardt method. In this problem we want to analyze the same spectral line data (Table 12.5) using the experimental APT MCMC software given in the *Mathematica* tutorial and discussed in Section 12.8. It will yield a fully Bayesian solution to the problem without the need to assume the asymptotic normal approximation, or to assume the Laplacian approximations for computing the Bayes factor and marginals. In general, MCMC solutions come into their own for

Table 12.4 *Spectral line data consisting of 64 frequency channels obtained with a radio astronomy spectrometer. The output voltage from each channel has been calibrated in units of effective black body temperature expressed in mK. The existence of negative values arises from receiver channel noise which gives rise to both positive and negative fluctuations.*

ch. #	mK	ch. #	mK	ch. #	mK	ch. #	mK
1	0.82	17	−0.90	33	−0.03	49	−0.72
2	−2.07	18	0.33	34	1.47	50	0.38
3	0.38	19	0.80	35	1.70	51	0.02
4	0.99	20	−1.42	36	1.89	52	−1.26
5	−0.12	21	0.28	37	4.55	53	1.35
6	−1.35	22	−0.42	38	3.59	54	−0.04
7	−0.20	23	0.12	39	2.02	55	−1.45
8	0.36	24	0.14	40	0.21	56	1.48
9	0.78	25	−0.63	41	0.05	57	−1.16
10	1.01	26	−1.77	42	0.54	58	−0.40
11	0.44	27	−0.67	43	−0.09	59	0.01
12	0.34	28	0.55	44	−0.61	60	0.29
13	1.58	29	1.98	45	2.49	61	−1.35
14	0.08	30	−0.08	46	0.07	62	−0.21
15	0.38	31	1.16	47	−1.45	63	−1.67
16	−0.71	32	0.48	48	0.56	64	0.70

Table 12.5 *Spectral line data consisting of 51 pairs of frequency and signal strength (mK) measurements.*

f	mK	*f*	mK	*f*	mK	*f*	mK
0.00	0.86	1.56	0.97	3.12	1.95	4.68	1.39
0.12	1.08	1.68	0.97	3.24	1.75	4.80	0.64
0.24	0.70	1.80	1.06	3.36	2.03	4.92	0.79
0.36	1.16	1.92	0.85	3.48	1.42	5.04	1.27
0.48	0.98	2.04	1.94	3.60	1.06	5.16	1.17
0.60	1.32	2.16	2.34	3.72	0.79	5.28	1.23
0.72	1.05	2.28	3.55	3.84	1.11	5.40	1.23
0.84	1.17	2.40	3.53	3.96	0.88	5.52	0.71
0.96	0.96	2.52	4.11	4.08	0.88	5.64	0.71
1.08	0.86	2.64	3.72	4.20	0.68	5.76	0.80
1.20	1.12	2.76	3.52	4.32	1.39	5.88	1.16
1.32	0.79	2.88	2.78	4.44	0.62	6.00	1.12
1.44	0.86	3.00	3.03	4.56	0.80		

higher dimensional problems but it is desirable to gain experience working with simpler problems.

Modify the APT MCMC software to analyze these data for the two models described in Section 11.6.

In *Mathematica*, model 1 has the form:

model[a0_, a1_,f1_]: = a0 + a1 line[f1]

where

$$\textbf{line}[f1\_] := \frac{\sin[2\pi(f-f1)/\Delta f]}{2\pi(f-f1)/\Delta f} \text{ and } \Delta f = 1.5.$$

Model 2 has the form:

model[a0_, a1_, a2_, f1_, f2_] := a0 + a1 line[f1] + a2 line[f2],

where $f2$ is assumed to be the higher frequency line.

Adopt uniform priors for all parameters and assume a lower bound of 0 and an upper bound of 10 for $a0$, $a1$ and $a2$. For the two spectral line model, we need to carefully consider the prior boundaries for $f1$ and $f2$ to prevent the occurrence of two degenerate peaks in the joint posterior. Adopt a range for $f2$ = 1.0 to 5.0. Since by definition, $f1$ is the lower frequency line, at any iteration the current value of $f1$ must be less than current value of $f2$. Thus

$$p(f1|f2, M_2, I) = \frac{1}{f_2 - 1.0}.$$