Western Swiss Doctoral School in
Statistics and Probability

Statistical computing for systems biology

CUSO, Villars-sur-Ollon, Switzerland
September 2019

Darren Wilkinson
Newcastle University, UK
and
The Alan Turing Institute, UK

darrenjw.github.io

Lecture 2: Bayesian inference and MCMC

Likelihood and Bayesian inference

This lecture provides a brief introduction to the essential ideas of Bayesian inference, together with some basic MCMC algorithms

Later we will examine the problem of inference for Markov processes. Here we will begin by examining the essential concepts of Bayesian inference and the reasons why analytic approaches to Bayesian inference in complex models are generally intractable, before moving on to MCMC and its application to Bayesian inference.

Bayesian inference

Often we are able to understand the probability of some **outcome**, X conditional on various possible **hypotheses**, $H_i,\ i = 1, 2, \ldots n$, where the H_i form a partition. We can then compute probabilities of the form $\mathsf{P}(X = x'|H_i), i = 1, \ldots, n,\ x' \in S_X$. However, when we actually **observe** some outcome $X = x$, we are interested in the probabilities of the hypotheses **conditional** on the outcome, $\mathsf{P}(H_i|X = x)$. Bayes Theorem tells us how to compute these, but the answer also depends on the prior probabilities for the hypotheses, $\mathsf{P}(H_i)$, and hence to use Bayes Theorem, these too must be specified. Thus Bayes Theorem provides us with a coherent way of updating our prior beliefs about the hypotheses $\mathsf{P}(H_i)$ to $\mathsf{P}(H_i|X = x)$, our posterior beliefs based on the occurrence of $X = x$, as

$$\mathsf{P}(H_i|X = x) = \frac{\mathsf{P}(X = x|H_i)\,\mathsf{P}(H_i)}{\sum_{j=1}^{n} \mathsf{P}\left(X = x|H_j\right)\mathsf{P}\left(H_j\right)}, \quad i = 1, \ldots n.$$

Note that the probabilities $\mathsf{P}(X = x|H_i)$ are known as **likelihoods**, and are often written $L(H_i; x)$, as they tend to be regarded as a function of the H_i for given fixed outcome x. Note, however, that the **likelihood function** does not represent a PMF for the H_i; in particular, there is no reason to suppose that it will sum to 1.

This is how it all works for purely discrete problems, but some adaptation is required before it can be used with continuous or mixed problems. Let us first stay with discrete outcome X and consider a continuum of hypotheses represented by a continuous parameter Θ. Our prior beliefs must now be represented by a density function, traditionally written, $\pi(\theta)$. Taking the continuous limit in the usual way, Bayes Theorem becomes

$$\pi(\theta|X = x) = \frac{\pi(\theta)\,\mathsf{P}(X = x|\theta)}{\int_\Theta \mathsf{P}\left(X = x|\theta'\right)\pi(\theta')d\theta'}.$$

In this case, the likelihood function is $L(\theta; x) = \mathsf{P}(X = x|\theta)$, regarded as a function of θ for given fixed x. Again, note that the likelihood function is not a density for θ, as it does not integrate to 1.

Using this notation we can rewrite Bayes Theorem as

$$\pi(\theta|X = x) = \frac{\pi(\theta)L(\theta; x)}{\int_{\Theta} \pi(\theta')L(\theta'; x)d\theta'},$$

and this is the way it is usually written in the context of Bayesian statistics, though the likelihood function $L(\theta; x)$ means slightly different things depending on the context. Note that the integral on the bottom line of Bayes Theorem is not a function of θ, and so simply represents a constant of proportionality. Thus, we can rewrite Bayes Theorem in the simpler form

$$\pi(\theta|X = x) \propto \pi(\theta)L(\theta; x), \qquad (1)$$

giving rise to the Bayesian mantra **"the posterior is proportional to the prior times the likelihood".**

Example

Suppose that for a particular gene in a particular cell, transcription events occur according to a Poisson process with rate θ per minute. Prior to carrying out an experiment, a biological expert specifies his opinion regarding θ in the form of a $Ga(a, b)$ distribution. Suppose that for our expert, $a = 2,\ b = 1$. Counts of the number of transcript events are gathered from n separate one-minute intervals to get data $x = (x_1, x_2, \ldots, x_n)^\mathsf{T}$. In this case the likelihood for θ is

$$\begin{aligned} L(\theta; x) &= \mathsf{P}(x|\theta) \\ &= \prod_{i=1}^{n} \mathsf{P}(x_i|\theta) \\ &= \prod_{i=1}^{n} \frac{\theta^{x_i} e^{-\theta}}{x_i!} \\ &\propto \prod_{i=1}^{n} \theta^{x_i} e^{-\theta} \\ &= \theta^{\sum_{i=1}^{n} x_i} e^{-n\theta}. \end{aligned}$$

The second line follows from the first because the data are independent (given θ). The likelihood depends on the data only through n and $\bar{x}$, so n and $\bar{x}$ are said to be **sufficient statistics** for the likelihood function. Then since θ is gamma, we have

$$\pi(\theta) \propto \theta^{a-1} e^{-b\theta}$$

giving

$$\begin{aligned} \pi(\theta|x) &\propto \pi(\theta) L(\theta; x) \\ &\propto \theta^{a+\sum_{i=1}^{n} x_i - 1} e^{-(b+n)\theta}. \end{aligned}$$

In other words,

$$\theta|x \sim Ga\left(a + \sum_{i=1}^{n} x_i, b+n\right).$$

So in this case, starting with a gamma prior results in a gamma posterior. Problems of this nature are said to be **conjugate**, and so in this case the gamma prior is said to be conjugate for the Poisson likelihood. In the context of our example, observing the data $x = (4, 2, 3)$ leads to a $Ga(11, 4)$ posterior distribution. This distribution (which has an expectation of $11/4$ and a variance of $11/16$) represents our belief about the value of θ having observed the data, and is shown in Figure 1.

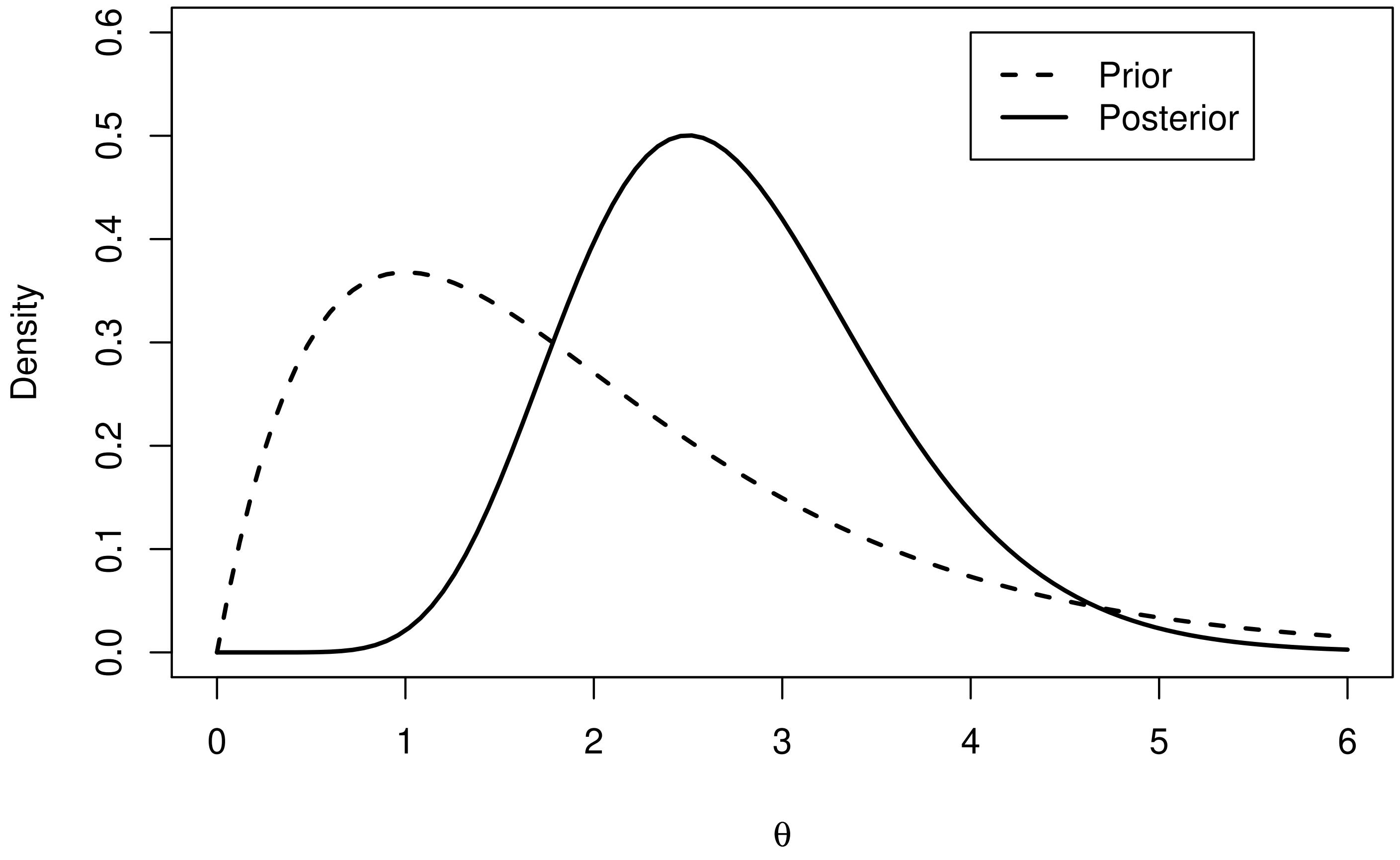

Figure 1: Plot showing the prior and posterior for the Poisson rate example. Note how the prior is modified to give a posterior more consistent with the data (which has a sample mean of 3).

The analysis is essentially the same when X is continuous rather than discrete, except that here the likelihood is evaluated using the PDF rather than the PMF.

Bayesian computation

In principle, the previous section covers everything we need to know about Bayesian inference — the posterior is nothing more (or less) than a conditional distribution for the parameters given the data. In practice, however, this may not be entirely trivial to work with.

The first problem one encounters is choosing the constant of proportionality so that the density integrates to 1. If the density is non-standard (as is usually the case for non-trivial problems), then the problem reduces to integrating the product of the likelihood and the prior (known as the **kernel** of the posterior) over the support of Θ. If the support is infinite in extent, and/or multi-dimensional, then this is a highly non-trivial numerical problem.

Even if we have the constant of integration, if the parameter space is multi-dimensional, we will want to know what the marginal distribution of each component looks like. For each component, we have a very difficult numerical integration problem.

Example

Consider the case where we have a collection of observations, X_i, which we believe to be independent identically distributed (iid) normal with unknown mean and precision (the reciprocal of variance). We write

$$X_i|\mu, \tau \sim N(\mu, 1/\tau).$$

The likelihood for a single observation is

$$L(\mu, \tau; x_i) = f(x_i|\mu, \tau) = \sqrt{\frac{\tau}{2\pi}} \exp\left\{-\frac{\tau}{2}(x_i - \mu)^2\right\}$$

and so for n independent observations, $x = (x_1, \ldots, x_n)^\mathsf{T}$ is

$$\begin{aligned} L(\mu, \tau; x) = f(x|\mu, \tau) &= \prod_{i=1}^{n} \sqrt{\frac{\tau}{2\pi}} \exp\left\{-\frac{\tau}{2}(x_i - \mu)^2\right\} \\ &= \left(\frac{\tau}{2\pi}\right)^{\frac{n}{2}} \exp\left\{-\frac{\tau}{2}\left[(n-1)s^2 + n(\bar{x} - \mu)^2\right]\right\} \\ &\propto \tau^{\frac{n}{2}} \exp\left\{-\frac{\tau}{2}\left[(n-1)s^2 + n(\bar{x} - \mu)^2\right]\right\} \end{aligned}$$

where

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad \text{and} \quad s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2.$$

For a Bayesian analysis, we need also to specify prior distributions for the parameters, μ and τ. There is a conjugate analysis for this problem based on the specifications

$$\tau \sim Ga(a,b), \qquad \mu|\tau \sim N\left(c, \frac{1}{d\tau}\right).$$

However, this specification is rather unsatisfactory — μ and τ are not independent, and in many cases our prior beliefs for μ and τ will separate into independent specifications. For example, we may prefer to specify independent priors for the parameters:

$$\tau \sim Ga(a,b), \qquad \mu \sim N\left(c, \frac{1}{d}\right).$$

However, this specification is no longer conjugate, making analytic analysis intractable. Let us see why. We have

$$\pi(\mu) = \sqrt{\frac{d}{2\pi}} \exp\left\{-\frac{d}{2}(\mu-c)^2\right\} \propto \exp\left\{-\frac{d}{2}(\mu-c)^2\right\}$$

and

$$\pi(\tau) = \frac{b^a}{\Gamma(a)}\tau^{a-1}\exp\{-b\tau\} \propto \tau^{a-1}\exp\{-b\tau\},$$

so

$$\pi(\mu,\tau) \propto \tau^{a-1} \exp\left\{-\frac{d}{2}(\mu-c)^2 - b\tau\right\},$$

giving

$$\begin{aligned}\pi(\mu,\tau|x) &\propto \tau^{a-1} \exp\left\{-\frac{d}{2}(\mu-c)^2 - b\tau\right\} \times \tau^{\frac{n}{2}} \exp\left\{-\frac{\tau}{2}\left[(n-1)s^2\right.\right.\\ &\qquad\qquad\qquad \left.\left.+n(\bar{x}-\mu)^2\right]\right\}\\ &= \tau^{a+\frac{n}{2}-1} \exp\left\{-\frac{\tau}{2}\left[(n-1)s^2 + n(\bar{x}-\mu)^2\right] - \frac{d}{2}(\mu-c)^2 - b\tau\right\}.\end{aligned}$$

The posterior density for μ and τ certainly will not factorise (μ and τ are not independent **a posteriori**), and will not even separate into the form of the conditional normal-gamma conjugate form mentioned earlier.

So we have the kernel of the posterior for μ and τ, but it is not in a standard form. We can gain some idea of the likely values of (μ, τ) by plotting the bivariate surface (the integration constant is not necessary for that), but we cannot work out the posterior mean or variance, or the forms of the marginal posterior distributions for μ or τ, since we cannot integrate out the other variable. We need a way of understanding posterior densities which does not rely on being able to analytically integrate the posterior density.

In fact, there is nothing particularly special about the fact that the density represents a Bayesian posterior. Given any complex non-standard multivariate probability distribution, we need ways to understand it, to calculate its moments, and to compute its conditional and marginal distributions and their moments. Markov chain Monte Carlo (MCMC) algorithms such as the Gibbs sampler and the Metropolis–Hastings method provide a possible solution.

The Gibbs sampler

Introduction

The Gibbs sampler is a way of simulating from multivariate distributions based only on the ability to simulate from conditional distributions. In particular, it is appropriate when sampling from marginal distributions is not convenient or possible.

Example

Reconsider the problem of Bayesian inference for the mean and variance of a normally distributed random sample. In particular, consider the non-conjugate approach based on independent prior distributions for the mean and variance. The posterior took the form

$$\pi(\mu, \tau | x) \propto \tau^{a+\frac{n}{2}-1} \exp\left\{-\frac{\tau}{2}\left[(n-1)s^2 + n(\bar{x}-\mu)^2\right] - \frac{d}{2}(\mu - c)^2 - b\tau\right\}.$$

As explained previously, this distribution is not in a standard form.

However, while clearly not conjugate, this problem is often referred to as **semi-conjugate**, because the two **full conditional** distributions $\pi(\mu|\tau,x)$ and $\pi(\tau|\mu,x)$ **are** of standard form, and further, are of the same form as the independent prior specifications. That is, $\tau|\mu,x$ is gamma distributed and $\mu|\tau,x$ is normally distributed. In fact, by picking out terms in the variable of interest and regarding everything else as a constant of proportionality, we get

$$\begin{aligned}\tau|\mu,x &\sim Ga\left(a+\frac{n}{2}, b+\frac{1}{2}\left[(n-1)s^2+n(\bar{x}-\mu)^2\right]\right),\\ \mu|\tau,x &\sim N\left(\frac{cd+n\tau\bar{x}}{n\tau+d}, \frac{1}{n\tau+d}\right).\end{aligned}$$

So providing that we can simulate normal and gamma quantities, we can simulate from the full conditionals. We therefore need a way to simulate from the joint density (and hence the marginals) based only on the ability to sample from the full-conditionals.

Sampling from bivariate densities

Consider a bivariate density $\pi(x, y)$. We have

$$\pi(x, y) = \pi(x)\pi(y|x),$$

and so we can simulate from $\pi(x, y)$ by first simulating $X = x$ from $\pi(x)$, and then simulating $Y = y$ from $\pi(y|x)$. On the other hand, if we can simulate from the marginal for y, we can write

$$\pi(x, y) = \pi(y)\pi(x|y)$$

and simulate $Y = y$ from $\pi(y)$ and then $X = x$ from $\pi(x|y)$. Either way we need to be able to simulate from one of the marginals. So let us just suppose that we can. That is, we have an $X = x$ from $\pi(x)$. Given this, we can now simulate a $Y = y$ from $\pi(y|x)$ to give a pair of points (x, y) from the bivariate density. However, in that case the y value must be from the marginal $\pi(y)$, and so we can simulate an $X' = x'$ from $\pi(x'|y)$ to give a new pair of points (x', y) also from the joint density. But now x' is from the marginal $\pi(x)$, and so we can keep going.

This alternate sampling from conditional distributions defines a bivariate Markov chain, and we have just given an intuitive explanation for why $\pi(x, y)$ is its stationary distribution. The transition kernel for this bivariate Markov chain is

$$p((x, y), (x', y')) = \pi(x', y'|x, y) = \pi(x'|x, y)\pi(y'|x', x, y) = \pi(x'|y)\pi(y'|x').$$

The Gibbs sampling algorithm

Suppose the density of interest is $\pi(\theta)$, where $\theta = (\theta_1, \ldots, \theta_d)^{\mathsf{T}}$, and that the full conditionals

$$\pi(\theta_i|\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_d) = \pi(\theta_i|\theta_{-i}) = \pi_i(\theta_i), \qquad i = 1, \ldots, d$$

are available for sampling. The Gibbs sampler can be summarised in the following algorithm:

1. Initialise the iteration counter to $j = 1$. Initialise the state of the chain to $\theta^{(0)} = (\theta_1^{(0)}, \ldots, \theta_d^{(0)})^{\mathsf{T}}$.

2. Obtain a new value $\theta^{(j)}$ from $\theta^{(j-1)}$ by successive generation of values

$$\begin{array}{lll}
\theta_1^{(j)} & \sim & \pi(\theta_1 | \theta_2^{(j-1)}, \ldots, \theta_d^{(j-1)}) \\
\theta_2^{(j)} & \sim & \pi(\theta_2 | \theta_1^{(j)}, \theta_3^{(j-1)}, \ldots, \theta_d^{(j-1)}) \\
\vdots & \vdots & \vdots \\
\theta_d^{(j)} & \sim & \pi(\theta_d | \theta_1^{(j)}, \ldots, \theta_{d-1}^{(j)}).
\end{array}$$

3. Change counter j to $j + 1$, and return to step 2.

This clearly defines a homogeneous Markov chain, as each simulated value depends only on the previous simulated value, and not on any other previous values or the iteration counter j. However, we need to show that $\pi(\theta)$ is a stationary distribution of this chain. The transition kernel of the chain is

$$p(\theta, \phi) = \prod_{i=1}^{d} \pi(\phi_i | \phi_1, \ldots, \phi_{i-1}, \theta_{i+1}, \ldots, \theta_d).$$

Therefore, we just need to check that $\pi(\theta)$ is the stationary distribution of the chain with this transition kernel. Unfortunately, the traditional **fixed-sweep** Gibbs sampler just described is **not** reversible, and so we cannot check stationarity by checking for detailed balance (as detailed balance fails). We need to do a direct check of the stationarity of $\pi(\theta)$, that is, we need to check that

$$\pi(\phi) = \int_S p(\theta, \phi) \pi(\theta) \, d\theta.$$

For the bivariate case, we have

$$\begin{aligned}
\int_S p(\theta,\phi)\pi(\theta)\,d\theta &= \int_S \pi(\phi_1|\theta_2)\pi(\phi_2|\phi_1)\pi(\theta_1,\theta_2)\,d\theta_1 d\theta_2 \\
&= \pi(\phi_2|\phi_1)\int_{S_1}\int_{S_2} \pi(\phi_1|\theta_2)\pi(\theta_1,\theta_2)\,d\theta_1 d\theta_2 \\
&= \pi(\phi_2|\phi_1)\int_{S_2} \pi(\phi_1|\theta_2)\,d\theta_2 \int_{S_1} \pi(\theta_1,\theta_2)\,d\theta_1 \\
&= \pi(\phi_2|\phi_1)\int_{S_2} \pi(\phi_1|\theta_2)\pi(\theta_2)\,d\theta_2 \\
&= \pi(\phi_2|\phi_1)\pi(\phi_1) \\
&= \pi(\phi_1,\phi_2) \\
&= \pi(\phi).
\end{aligned}$$

The general case is similar. So, $\pi(\theta)$ is a stationary distribution of this chain. Discussions of uniqueness and convergence are beyond the scope of this course. In particular, these issues are complicated somewhat by the fact that the sampler described is not reversible.

Simulation and analysis

Suppose that we are interested in a multivariate distribution $\pi(\theta)$ (which may be a Bayesian posterior distribution), and that we are able to simulate from the full conditional distributions of $\pi(\theta)$. Simulation from $\pi(\theta)$ is possible by first initialising the sampler somewhere in the support of θ, and then running the Gibbs sampler. The resulting chain should be monitored for convergence, and the “burn-in” period should be discarded for analysis. After convergence, the simulated values are all from $\pi(\theta)$. In particular, the values for a particular component will be simulated values from the marginal distribution of that component. A histogram of these values will give an idea of the “shape” of the marginal distribution, and summary statistics such as the mean and variance will be approximations to the mean and variance of the marginal distribution.

Example

Returning to the case of the posterior distribution for the normal model with unknown mean and precision, we have already established that the full conditional distributions are

$$\begin{aligned}\tau|\mu, x &\sim Ga\left(a+\frac{n}{2}, b+\frac{1}{2}\left[(n-1)s^2+n(\bar{x}-\mu)^2\right]\right),\\ \mu|\tau, x &\sim N\left(\frac{cd+n\tau\bar{x}}{n\tau+d}, \frac{1}{n\tau+d}\right).\end{aligned}$$

We can initialise the sampler anywhere in the half-plane where the posterior (and prior) has support, but convergence will be quicker if the chain is not started in the tails of the distribution. One possibility is to start the sampler near the posterior mode, though this can make convergence more difficult to diagnose. A simple strategy which is often easy to implement for problems in Bayesian inference is to start off the sampler at a point simulated from the prior distribution, or even at the mean of the prior distribution. Here, the prior mean for (τ, μ) is $(a/b, c)$. Once initialised, the sampler proceeds with alternate simulations from the full conditional distributions. The first few (hundred?) values should be discarded, and the rest can give information about the joint posterior distribution and marginals.

An R function to implement a simple Gibbs sampler is given in Figure 2, some example code that uses it is shown in Figure 3, and the results of running the example code are shown in Figure 4; see the figure legends for further details.

```r
normgibbs <- function(N, n, a, b, cc, d, xbar, ssquared)
{
        mat = matrix(ncol = 2, nrow = N)
        mu = cc
        tau = a/b
        mat[1, ] = c(mu, tau)
        for (i in 2:N) {
                muprec = n*tau + d
                mumean = (d*cc + n*tau*xbar)/muprec
                mu = rnorm(1, mumean, sqrt(1/muprec))
                taub = b + 0.5*((n - 1)*ssquared + n*(xbar - mu)^2)
                tau = rgamma(1, a + n/2, taub)
                mat[i, ] = c(mu, tau)
        }
        mat
}
```

Figure 2: An R function to implement a Gibbs sampler for the simple normal random sample model. Example code for using this function is given in Figure 3.

```
postmat=normgibbs(N=11000, n=15, a=3, b=11, cc=10, d=1/100, xbar=25,
   ssquared=20)
postmat=postmat[1001:11000,]
op=par(mfrow=c(3,3))
plot(postmat,col=1:10000)
plot(postmat,type="l")
plot.new()
plot(ts(postmat[,1]))
plot(ts(postmat[,2]))
plot(ts(sqrt(1/postmat[,2])))
hist(postmat[,1],40)
hist(postmat[,2],40)
hist(sqrt(1/postmat[,2]),40)
par(op)
```

Figure 3: Example R code illustrating the use of the function `normgibbs` from Figure 2. The plots generated by running this code are shown in Figure 4. In this example, the prior took the form $\mu \sim N(10, 100)$, $\tau \sim Ga(3, 11)$, and the sufficient statistics for the data were $n = 15$, $\bar{x} = 25$, $s^2 = 20$. The sampler was run for 11,000 iterations with the first 1,000 discarded as burn-in, and the remaining 10,000 iterations used for the main monitoring run.

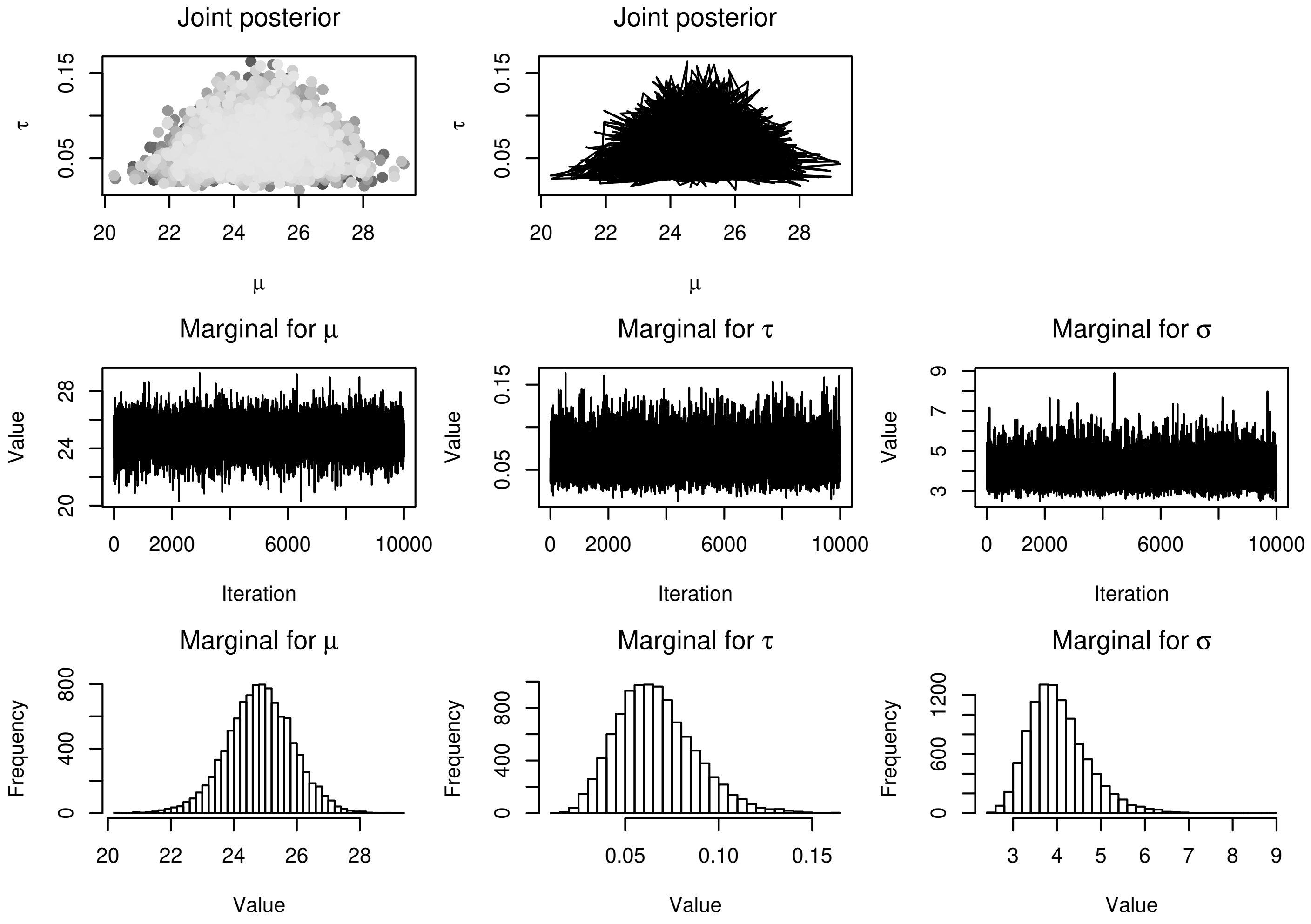

Figure 4: Figure showing the Gibbs sampler output resulting from running the example code in Figure 3. The top two plots give an indication of the bivariate posterior distribution. The second row shows trace plots of the marginal distributions. The final row shows empirical marginal posterior distributions.

It is clear that in principle at least, it ought to be possible to automate the construction of a Gibbs sampler from a specification containing the model, the prior, and the data. There are several freely available software packages that are able to do this for relatively simple models. Examples include, WinBUGS, OpenBugs, and JAGS.

Of course, the Gibbs sampler tacitly assumes that we have some reasonably efficient mechanism for simulating from the full conditional distributions, and yet this is not always the case. Fortunately, the Gibbs sampler can be combined with Metropolis–Hastings algorithms when the full conditionals are difficult to simulate from.

Stochastic simulation

For simple random systems, mathematical analysis alone can provide a complete description of all properties of interest. However, for the kinds of random systems that we are interested in (non-linear Markov processes), mathematical analysis is not possible, and the systems are described as **analytically intractable**. However, this does not mean that it is not possible to understand such systems. With the aid of a computer, it is possible to **simulate** the time-evolution of the system dynamics. Stochastic simulation is concerned with the computer simulation of random (or stochastic) phenomena, and it is therefore essential to know something about this topic before proceeding further.

Monte Carlo integration

The rationale for stochastic simulation can be summarised very easily: to understand a probabilistic model, simulate many realisations from it and study them. To make this more concrete, one way to understand stochastic simulation is to perceive it as a way of numerically solving the difficult integration problems that naturally arise in probability theory.

Suppose we have a (continuous) random variable X, with probability density function (PDF), $f(x)$, and we wish to evaluate $\mathsf{E}(g(X))$ for some function $g(\cdot)$. We know that

$$\mathsf{E}(g(X)) = \int_X g(x) f(x)\, dx,$$

and so the problem is one of integration. For simple $f(\cdot)$ and $g(\cdot)$ this integral might be straightforward to compute directly. On the other hand, in more complex scenarios, it is likely to be analytically intractable.

However, if we can **simulate** realisations $x_1, \ldots, x_n$ of X, then we can form realisations of the random variable $g(X)$ as $g(x_1), \ldots, g(x_n)$. Then, provided that the variance of $g(X)$ is finite, the laws of large numbers assure us that for large n we may approximate the integral by

$$\mathsf{E}(g(X)) \simeq \frac{1}{n}\sum_{i=1}^{n} g(x_i).$$

In fact, even if we cannot simulate realisations of X, but can simulate realisations $y_1, \ldots, y_n$ of Y (a random variable with the same support as X), which has PDF $h(\cdot)$, then

$$\begin{aligned}\mathsf{E}(g(X)) &= \int_X g(x) f(x)\, dx \\ &= \int_X \frac{g(x)f(x)}{h(x)} h(x)\, dx\end{aligned}$$

and so $\mathsf{E}(g(X))$ may be approximated by

$$\mathsf{E}(g(X)) \simeq \frac{1}{n}\sum_{i=1}^{n} \frac{g(y_i)f(y_i)}{h(y_i)}.$$

This procedure is known as **importance sampling**, and it can be very useful when there is reasonable agreement between $f(\cdot)$ and $h(\cdot)$.

Rejection samplers

Proposition 1 (uniform rejection method) *Suppose that we want to simulate from $f(x)$ with (finite) support on $[a, b]$, and that $f(x) \leq m, \ \forall x \in [a, b]$. Then consider simulating*

$$X \sim U(a, b) \quad \textit{and} \quad Y \sim U(0, m).$$

Accept X if $Y < f(X)$, otherwise ***reject*** *and try again. Then the accepted X values have PDF $f(x)$.*

Intuitively we can see that this will work because it has the effect of scattering points uniformly over the region bounded by the PDF and the x-axis.

In summary, we simulate a value x uniformly from the support of X and accept this value with probability $f(x)/m$, otherwise we reject and try again. Obviously the efficiency of this method depends on the overall proportion of candidate points that are accepted. The actual acceptance probability for this method is

$$
\begin{aligned}
\mathsf{P}(\text{Accept}) &= \mathsf{P}((X,Y)\in A) \\
&= \int_a^b \mathsf{P}((X,Y)\in A|X=x)\times\frac{1}{b-a}dx \\
&= \int_a^b \frac{f(x)}{m}\times\frac{1}{b-a}dx \\
&= \frac{1}{m(b-a)}\int_a^b f(x)dx \\
&= \frac{1}{m(b-a)}.
\end{aligned}
$$

If this acceptance probability is very low, the procedure will be very inefficient, and a better procedure should be sought — the **envelope method** is one possibility.

Envelope method

Once we have established that scattering points uniformly over the region bounded by the density and the x-axis generates x-values with the required distribution, we can extend it to distributions with infinite support and make it more efficient by choosing our **enveloping** region more carefully.

Suppose that we wish to simulate X with PDF $f(\cdot)$, but that we can already simulate values of Y (with the same support as X), which has PDF $h(\cdot)$. Suppose further that there exists some constant a such that

$$f(x) \leq a\, h(x), \quad \forall x.$$

That is, a is an upper bound for $f(x)/h(x)$. Note also that $a \geq 1$, as both $f(x)$ and $h(x)$ integrate to 1.

Consider the following algorithm. Draw $Y = y$ from $h(\cdot)$, and then $U = u \sim U(0, a\, h(y))$. Accept y as a simulated value of X if $u < f(y)$, otherwise reject and try again. This works because it distributes points uniformly over a region covering $f(x)$, and then only keeps points in the required region (under $f(x)$).

To summarise, just simulate a proposed value from $h(\cdot)$ and accept this with probability $f(y)/[a\,h(y)]$, otherwise reject and try again. The accepted values will have PDF $f(\cdot)$.

Obviously, this method will work well if the overall acceptance rate is high, but not otherwise. The overall acceptance probability can be computed as

$$\begin{aligned}
\mathsf{P}(U < f(Y)) &= \int_{-\infty}^{\infty} \mathsf{P}(U < f(Y)|Y = y)\, h(y) dy \\
&= \int_{-\infty}^{\infty} \frac{f(y)}{a\,h(y)} h(y) dy \\
&= \int_{-\infty}^{\infty} \frac{f(y)}{a} dy \\
&= \frac{1}{a}.
\end{aligned}$$

Consequently, we want a to be as small as possible (that is, as close as possible to 1). What “small enough” means is context-dependent, but generally speaking, if $a > 10$, the envelope is not adequate — too many points will be rejected, so a better envelope needs to be found. If this is not practical, then an entirely new approach is required.

Importance resampling

Importance resampling is an idea closely related to both importance sampling and the envelope rejection method. One of the problems with using the rejection method is finding a good envelope and computing the envelope bounding constant, a. Importance resampling, like importance sampling, can in principle use any proposal distribution $h(\cdot)$ with the same support as the target distribution $f(\cdot)$, and there is no need to calculate any kind of bounding constant. It is therefore widely applicable, but in practice will work **well** only if $h(\cdot)$ is sufficiently similar to $f(\cdot)$. Further, unlike rejection sampling, importance resampling is not exact, so the generated samples are only approximately from $f(\cdot)$, with the approximation improving as the number of generated samples increases.

Importance sampling was introduced earlier. We demonstrated that an expectation of an arbitrary function $g(\cdot)$ with respect to a target distribution $f(\cdot)$ can be approximated using samples y_i from a proposal distribution $h(\cdot)$ by

$$\mathsf{E}(g(X)) \simeq \frac{1}{n} \sum_{i=1}^{n} \frac{g(y_i) f(y_i)}{h(y_i)}.$$

We can re-write this as

$$\mathsf{E}(g(X)) \simeq \frac{1}{n} \sum_{i=1}^{n} w_i g(y_i),$$

where $w_i = f(y_i)/h(y_i)$. That is, samples from $h(\cdot)$ can be used as if they were samples from $f(\cdot)$, provided that they are re-weighted appropriately by the w_i. This motivates importance resampling: first generate samples from the proposal $h(\cdot)$, then **resample** from the sample, using the weights w_i. Then the new sample is distributed approximately according to $f(\cdot)$.

We can describe the algorithm explicitly as follows.

1. Sample $y_1, y_2, \ldots, y_n \sim h(\cdot)$
2. Compute the weights $w_k = f(y_k)/h(y_k),\ k = 1, 2, \ldots, n$
3. Compute the sum of the weights, $w_0 = \sum_{j=1}^{n} w_j$
4. Compute the **normalised** weights $w'_k = w_k/w_0,\ k = 1, 2, \ldots, n$
5. Sample n times, **with replacement** from the set $\{y_1, y_2, \ldots, y_n\}$ using the probabilities $\{w'_1, w'_2, \ldots, w'_n\}$ (for example, using the lookup method) to generate a new sample $\{x_1, x_2, \ldots, x_n\}$
6. Return the new sample $\{x_1, x_2, \ldots, x_n\}$ as an approximate sample from $f(\cdot)$

As already discussed, this algorithm is very general and is central to several more advanced Monte Carlo techniques, such as **sequential importance resampling** (SIR) and particle filtering that will be discussed later.

A rigorous proof that importance resampling works is beyond the scope of this course (see Doucet et al. (2001) for such details), but an informal justification can be given in the case of a univariate continuous distribution as follows. The method is only approximate, and the approximation improves as the number of particles increases, so consider a **very** large number of particles (samples), N. Let us also consider an arbitrary very small interval $[x, x + dx)$ in the support of $f(\cdot)$ and $h(\cdot)$. The probability that a given sample from the proposal $h(\cdot)$ will be contained in this interval is $h(x)dx$, so the expected number of particles is $Nh(x)dx$. The weight of each of these particles is $w(x) = f(x)/h(x)$. On the other hand, the expected weight of an arbitrary random particle is

$$\mathsf{E}(w(X)) = \int_{\mathbb{R}} w(x)h(x)dx = \int_{\mathbb{R}} \frac{f(x)}{h(x)}h(x)dx = \int_{\mathbb{R}} f(x)dx = 1,$$

and so $w_0 = N$.

The normalised weights are therefore $w'(x) = f(x)/[Nh(x)]$, and so the combined normalised weight of all particles in the interval $[x, x + dx)$ is

$$\frac{f(x)}{Nh(x)} \times Nh(x)dx = f(x)dx.$$

When we resample from our set of particles N times we therefore expect to get $Nf(x)dx$ particles in our interval, corresponding to a proportion $f(x)dx$ and a density $f(x)$, and so our new set of particles has density $f(x)$, asymptotically, as $N \longrightarrow \infty$. Although there are a number of gaps in this argument, it isn't too difficult to tighten up into a reasonably rigorous proof, at least in the case of one-dimensional continuous distributions. The algorithm also works for very general multi-dimensional distributions, but demonstrating the validity of the algorithm in that case requires more work.

The Metropolis–Hastings algorithm

Let us now reconsider MCMC–based methods for sampling from a distribution of interest. Suppose that $\pi(\theta)$ is the density of interest. Suppose further that we have some (arbitrary) transition kernel $q(\theta, \phi)$ (known as the **proposal distribution**), which is easy to simulate from but does not (necessarily) have $\pi(\theta)$ as its stationary density.

Consider the following algorithm:

1. Initialise the iteration counter to $j = 1$, and initialise the chain to $\theta^{(0)}$.

2. Generate a **proposed** value ϕ using the kernel $q(\theta^{(j-1)}, \phi)$.

3. Evaluate the **acceptance probability** $\alpha(\theta^{(j-1)}, \phi)$ of the proposed move, where

$$\alpha(\theta, \phi) = \min\left\{1, \frac{\pi(\phi)q(\phi, \theta)}{\pi(\theta)q(\theta, \phi)}\right\}.$$

4. Put $\theta^{(j)} = \phi$ with probability $\alpha(\theta^{(j-1)}, \phi)$, and put $\theta^{(j)} = \theta^{(j-1)}$ otherwise.

5. Change the counter from j to $j + 1$ and return to step 2.

In other words, at each stage, a new value is generated from the proposal distribution. This is either accepted, in which case the chain moves, or rejected, in which case the chain stays where it is. Whether or not the move is accepted or rejected depends on an acceptance probability which itself depends on the relationship between the density of interest and the proposal distribution. Note that the density of interest, $\pi(\cdot)$, only enters into the acceptance probability as a ratio, and so the method can be used when the density of interest is only known up to a scaling constant. This algorithm is essentially that of Hastings (1970), which is a generalisation of the algorithm introduced by Metropolis et al. 1953.

The Markov chain defined in this way is reversible and has stationary distribution $\pi(\cdot)$ irrespective of the choice of proposal distribution, $q(\cdot,\cdot)$. Let us see why. The transition kernel is clearly given by

$$p(\theta,\phi) = q(\theta,\phi)\alpha(\theta,\phi), \quad \text{if } \theta \neq \phi.$$

But there is also a finite probability that the chain will remain at θ. This is 1 minus the probability that the chain moves, and thus is given by

$$1 - \int q(\theta,\phi)\alpha(\theta,\phi)\, d\phi.$$

So, the transition kernel is part continuous and part discrete. We can easily write down the cumulative distribution form of the transition kernel as

$$P(\theta,\phi) = \int_{-\infty}^{\phi} q(\theta,\phi)\alpha(\theta,\phi)\,d\phi + I(\phi \geq \theta)\left[1 - \int q(\theta,\phi)\alpha(\theta,\phi)\,d\phi\right].$$

We then get the full density form of the kernel by differentiating with respect to ϕ as

$$p(\theta,\phi) = q(\theta,\phi)\alpha(\theta,\phi) + \delta(\theta-\phi)\left[1 - \int q(\theta,\phi)\alpha(\theta,\phi)\,d\phi\right],$$

where $\delta(\cdot)$ is the Dirac δ-function. Now we have the transition kernel we can check whether detailed balance is satisfied:

$$\begin{aligned}\pi(\theta)p(\theta,\phi) &= \pi(\theta)q(\theta,\phi)\min\left\{1, \frac{\pi(\phi)q(\phi,\theta)}{\pi(\theta)q(\theta,\phi)}\right\} \\ &\qquad + \delta(\theta-\phi)\left[\pi(\theta) - \int \pi(\theta)q(\theta,\phi)\min\left\{1, \frac{\pi(\phi)q(\phi,\theta)}{\pi(\theta)q(\theta,\phi)}\right\}\,d\phi\right] \\ &= \min\left\{\pi(\theta)q(\theta,\phi), \pi(\phi)q(\phi,\theta)\right\} \\ &\qquad + \delta(\theta-\phi)\left[\pi(\theta) - \int \min\left\{\pi(\theta)q(\theta,\phi), \pi(\phi)q(\phi,\theta)\right\}\,d\phi\right].\end{aligned}$$

The first term is clearly symmetric in θ and ϕ. Also, the second term must be symmetric in θ and ϕ, because it is only non-zero precisely when $\theta = \phi$. Consequently, detailed balance is satisfied, and the Metropolis–Hastings algorithm defines a reversible Markov chain with stationary distribution $\pi(\cdot)$, irrespective of the form of $q(\cdot, \cdot)$.

Complete freedom in the choice of the proposal distribution $q(\cdot, \cdot)$ leaves us wondering what kinds of choices might be good, or generally quite useful. Some commonly used special cases are discussed below.

Symmetric chains (Metropolis method)

The simplest case is the Metropolis sampler, which is based on the use of a symmetric proposal with $q(\theta, \phi) = q(\phi, \theta), \ \forall \theta, \phi$. We see then that the acceptance probability simplifies to

$$\alpha(\theta, \phi) = \min \left\{ 1, \frac{\pi(\phi)}{\pi(\theta)} \right\},$$

and hence does not involve the proposal density at all. Consequently proposed moves which will take the chain to a region of higher density are always accepted, while moves which take the chain to a region of lower density are accepted with probability proportional to the ratio of the two densities — moves which will take the chain to a region of very low density will be accepted with very low probability. Note that any proposal of the form $q(\theta, \phi) = f(|\theta - \phi|)$ is symmetric, where $f(\cdot)$ is an arbitrary density. In this case, the proposal will represent a symmetric displacement from the current value. This also motivates random walk chains.

Random walk chains

In this case, the proposed value ϕ at stage j is $\phi = \theta^{(j-1)} + w_j$ where the w_j are iid random variables (completely independent of the state of the chain). Suppose that the w_j have density $f(\cdot)$, which is easy to simulate from. We can then simulate an **innovation**, w_j, and set the **candidate** point to $\phi = \theta^{(j-1)} + w_j$. The transition kernel is then $q(\theta, \phi) = f(\phi - \theta)$, and this can be used to compute the acceptance probability. Of course, if $f(\cdot)$ is symmetric about zero, then we have a symmetric chain, and the acceptance probability does not depend on $f(\cdot)$ at all.

So suppose that it is decided to use a symmetric random walk chain with proposed mean zero innovations. There is still the question of how they should be distributed, and what variance they should have. A simple, easy to simulate from distribution is always a good idea, such as uniform or normal (normal is generally better, but is a bit more expensive to simulate). The choice of variance will affect the acceptance probability, and hence the overall proportion of accepted moves.

If the variance of the innovation is too low, then most proposed values will be accepted, but the chain will move very slowly around the space — the chain is said to be too “cold”. On the other hand, if the variance of the innovation is too large, very few proposed values will be accepted, but when they are, they will often correspond to quite large moves — the chain is said to be too “hot”. Experience suggests that an overall acceptance rate of around 30% is desirable, and so it is possible to “tune” the variance of the innovation distribution to get an acceptance rate of around this level. This should be done using a few trial short runs, and then a single fixed value should be adopted for the main monitoring run.

An R function implementing a simple Metropolis random walk sampler is given in Figure 5. In this example the target distribution is a $N(0, 1)$ random quantity, and the innovations are $U(-\alpha, \alpha)$. The results of running the algorithm for different values of α are shown in Figure 6. The auto-correlation function (ACF) plots are a useful diagnostic for assessing the rate of mixing of the chain.

```
metrop <- function(n, alpha)
{
        vec = vector("numeric", n)
        x = 0
        vec[1] = x
        for (i in 2:n) {
                can = x+runif(1,-alpha,alpha)
                aprob = dnorm(can)/dnorm(x)
                u = runif(1)
                if (u < aprob)
                        x=can
                vec[i] = x
        }
        vec
}
```

Figure 5: An R function to implement a Metropolis sampler for a standard normal random quantity based on $U(-\alpha, \alpha)$ innovations. So, `metrop(10000,1)` will execute a run of length 10,000 with an α of 1. See Figure 6 for results of running the algorithm with different values of α.

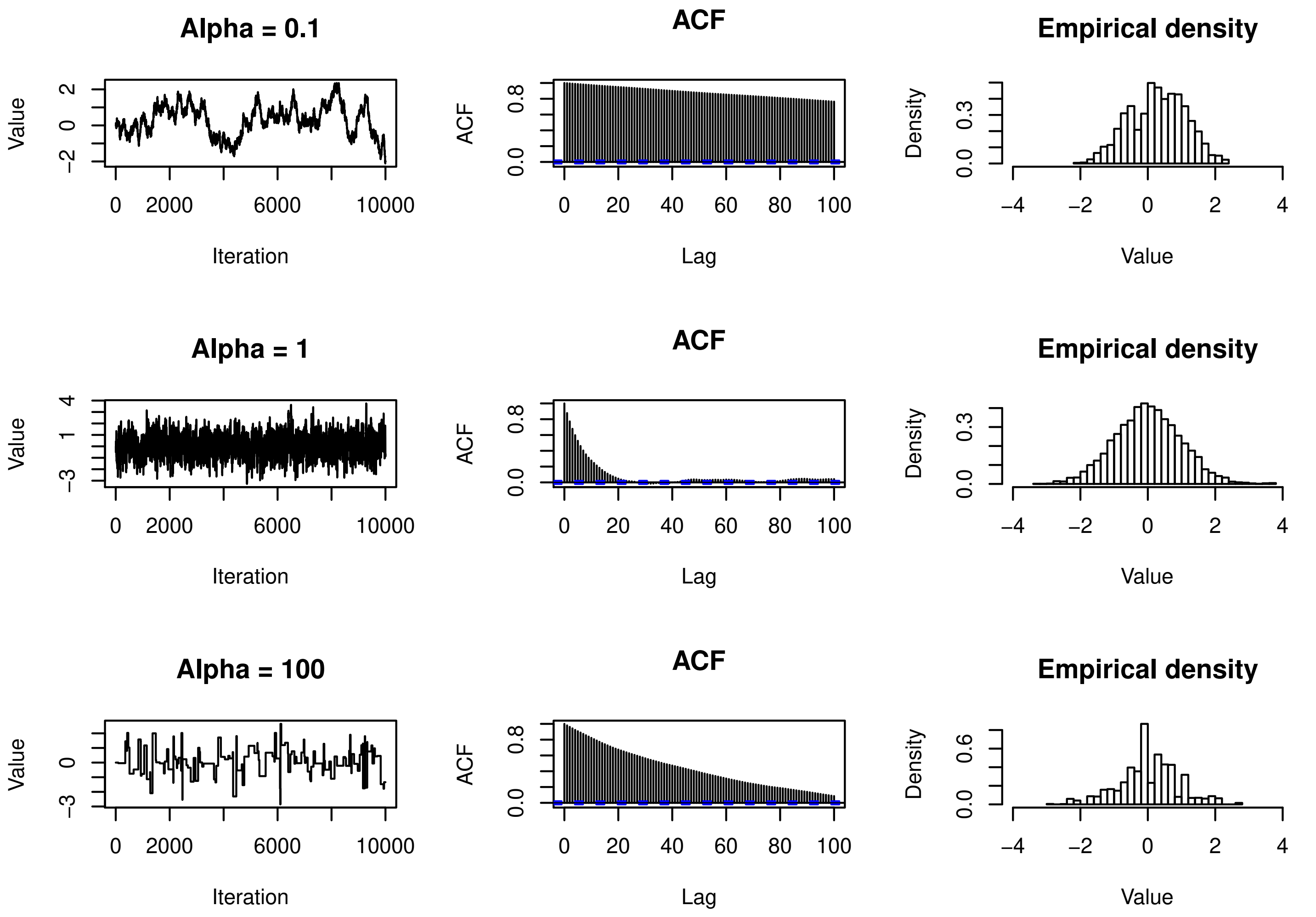

Figure 6: Output from the Metropolis sampler given in Figure 5. The top row shows a chain that is too cold. The middle row shows the results for α close to optimal, and the ACF plot shows auto-correlations in the sampled values decaying away rapidly to zero. The final row shows the results for a chain that is too hot.

Hybrid MCMC schemes

We have seen how we can use the Gibbs sampler to sample from multivariate distributions provided that we can simulate from the full conditionals. We have also seen how we can use Metropolis–Hastings methods to sample from awkward distributions (perhaps full conditionals). If we wish, we can combine these in order to form hybrid Markov chains whose stationary distribution is a distribution of interest.

Componentwise transition: Given a multivariate distribution with full conditionals that are awkward to sample from directly, we can define a Metropolis–Hastings scheme for each full conditional and apply them to each component in turn for each iteration. This is like the Gibbs sampler, but each component update is a Metropolis–Hastings update, rather than a direct simulation from the full conditional. This is in fact the original form of the Metropolis algorithm.

Metropolis within Gibbs: Given a multivariate distribution with full conditionals, some of which may be simulated from directly, and others which have Metropolis–Hastings updating schemes, the Metropolis within Gibbs algorithm goes through each in turn, and simulates directly from the full conditional, or carries out a Metropolis–Hastings update as necessary.

Blocking: The components of a Gibbs sampler, and those of Metropolis–Hastings chains, can be vectors (or matrices) as well as scalars. For many high-dimensional problems, it can be helpful to group related parameters into blocks and use multivariate simulation techniques to update those together if possible. This can greatly improve the mixing of the chain, at the expense of increasing the computational cost of each iteration.

Metropolis–Hastings algorithms for Bayesian inference

Let us now consider the generic problem of using a Metropolis–Hastings sampler in the context of inference for a parameter vector θ given some data x generated from a probability model of the form $\pi(x|\theta)$. We factorise the joint distribution as

$$\pi(\theta, x) = \pi(\theta)\pi(x|\theta)$$

and use this factorisation into “prior” and “likelihood” in order to compute the posterior distribution $\pi(\theta|x)$, which is just the joint density modulo a normalising constant. Consequently we can construct a Metropolis–Hastings scheme which targets $\pi(\theta|x)$ using an essentially arbitrary proposal kernel $q(\theta, \theta^\star)$ for a proposed move from θ to $\theta^\star$ in conjunction with an acceptance probability of the form $\alpha(\theta, \theta^\star) = \min\{1, A\}$, where

$$A = \frac{\pi(\theta^\star)\pi(x|\theta^\star)q(\theta^\star, \theta)}{\pi(\theta)\pi(x|\theta)q(\theta, \theta^\star)}.$$

Bayesian inference for latent variable models

We have introduced Bayesian inference in the context of inference for a parameter vector θ given some data x generated from a probability model of the form $\pi(x|\theta)$. The joint distribution was factorised as

$$\pi(\theta, x) = \pi(\theta)\pi(x|\theta)$$

and this factorisation into “prior” and “likelihood” has been used in order to compute the posterior distribution $\pi(\theta|x)$.

At this point it is useful to consider a commonly encountered extension of the basic inferential framework described above. Suppose now that we cannot observe x, as x is not measured, and represents a “latent”, “missing”, or “hidden” layer in our model. Instead we observe some aspect of x indirectly. We now denote the actual data by y, and it is modelled conditional on the missing data x, and possibly also depends on the parameter vector θ.

So now our joint distribution factorises as

$$\pi(\theta, x, y) = \pi(\theta)\pi(x|\theta)\pi(y|x, \theta).$$

We now consider using this new joint distribution as the basis of our inferential framework, which may be concerned with posterior distributions such as $\pi(\theta|y)$ or $\pi(\theta, x|y)$.

Before proceeding further it may be useful to have a concrete example in mind for motivation. In the next lecture we will consider how to apply the techniques developed in this lecture to the problem of making inference for the parameters of Markov process models given some discrete time measurements of the system state. In that case, the parameter vector, θ, will represent a vector of rate constants, x will represent the complete unobserved sample path of the entire process, and y will represent the actual measurements we obtain.

If we are really just interested in the marginal posterior $\pi(\theta|y)$ we know that in principle we can construct a Metropolis–Hastings scheme targeting this posterior using a proposal $q(\theta, \theta^\star)$ and using the acceptance ratio

$$A = \frac{\pi(\theta^\star)\pi(y|\theta^\star)q(\theta^\star, \theta)}{\pi(\theta)\pi(y|\theta)q(\theta, \theta^\star)},$$

where

$$\pi(y|\theta) = \int_X \pi(y|x, \theta)\pi(x|\theta)\, dx.$$

In practice, marginalising over x is often impossible, but for certain tractable distribution families it can be done, and is often easiest to compute using the basic marginal likelihood identity (BMI) (Chib 1995)

$$\pi(y|\theta) = \frac{\pi(x|\theta)\pi(y|x, \theta)}{\pi(x|y, \theta)}.$$

Note that since the LHS of the above expression is independent of x, the RHS must also be, and hence can be evaluated at any convenient x. However, although there are special cases where this is possible, typically it is not the case, and so another strategy must be adopted. The usual approach is to focus instead on the full posterior distribution $\pi(\theta, x|y)$, since this is often of interest anyway, and if an MCMC scheme can be constructed which targets this, the marginal posterior $\pi(\theta|y)$ can in any case be obtained by considering only the sampled values of θ.

It is clear that the joint posterior $\pi(\theta, x|y)$ can be targeted by a chain with proposal $q((\theta, x), (\theta^\star, x^\star))$ by using the acceptance ratio

$$A = \frac{\pi(\theta^\star)\pi(x^\star|\theta^\star)\pi(y|x^\star, \theta^\star)q((\theta^\star, x^\star), (\theta, x))}{\pi(\theta)\pi(x|\theta)\pi(y|x, \theta)q((\theta, x), (\theta^\star, x^\star))}.$$

There are many possible ways to construct the proposal distribution $q(\cdot, \cdot)$. An interesting special case is where the proposed new $(\theta^\star, x^\star)$ is constructed in two stages: first, a new $\theta^\star$ is proposed from a kernel $f(\theta^\star|\theta)$, and then a new $x^\star$ is proposed from a kernel $g(x^\star|\theta^\star)$, conditional on the newly proposed $\theta^\star$. This allows the proposals to be constructed in such a way as to ensure that the $\theta^\star$ and $x^\star$ are consistent with one another. In this case the proposal is clearly

$$q((\theta, x), (\theta^\star, x^\star)) = f(\theta^\star|\theta)g(x^\star|\theta^\star),$$

and the acceptance ratio becomes

$$A = \frac{\pi(\theta^\star)\pi(x^\star|\theta^\star)\pi(y|x^\star, \theta^\star)f(\theta|\theta^\star)g(x|\theta)}{\pi(\theta)\pi(x|\theta)\pi(y|x, \theta)f(\theta^\star|\theta)g(x^\star|\theta^\star)}.$$

As usual, the proposal for $\theta^\star$, $f(\theta^\star|\theta)$ can be fairly arbitrary, and random walk distributions can often be effective. However, considerable care needs to be taken with the form of the proposal for $x^\star$, $g(x^\star|\theta^\star)$, since $x^\star$ is typically high-dimensional, and a poor choice will lead to a poorly mixing chain. There are two important special cases for $g(\cdot|\cdot)$ which lead to considerable simplification of the acceptance ratio. The first is the so-called **likelihood-free** MCMC (LF-MCMC) proposal, where the proposed new $X^\star$ is generated by simple forward simulation from the model using the newly proposed parameters $\theta^\star$. That is, $g(x^\star|\theta^\star) = \pi(x^\star|\theta^\star)$. This choice obviously leads to cancellation in the acceptance ratio, giving

$$A = \frac{\pi(\theta^\star)\pi(y|x^\star,\theta^\star)f(\theta|\theta^\star)}{\pi(\theta)\pi(y|x,\theta)f(\theta^\star|\theta)}.$$

Again, further simplification can be obtained by choosing $f(\theta^\star|\theta) = \pi(\theta^\star)$, leading to the even simpler acceptance ratio

$$A = \frac{\pi(y|x^\star,\theta^\star)}{\pi(y|x,\theta)},$$

but this will typically not represent an efficient choice. The likelihood-free approach is potentially attractive, as it is usually easier to simulate realisations from the model than to evaluate its likelihood, $\pi(x|\theta)$. However, in the case of high-dimensional y, or “low noise” measurement scenarios, the proposal is very inefficient, leading to a poorly mixing chain. In the context of inference for systems biology models, it is possible to partly alleviate these difficulties by applying the technique sequentially.

The other important special case is the "optimal" choice $g(x^\star|\theta^\star) = \pi(x^\star|\theta^\star, y)$. This clearly gives the acceptance ratio

$$A = \frac{\pi(\theta^\star)\pi(x^\star|\theta^\star)\pi(y|x^\star,\theta^\star)f(\theta|\theta^\star)\pi(x|\theta,y)}{\pi(\theta)\pi(x|\theta)\pi(y|x,\theta)f(\theta^\star|\theta)\pi(x^\star|\theta^\star,y)},$$

but using the BMI this simplifies to

$$A = \frac{\pi(\theta^\star)\pi(y|\theta^\star)f(\theta|\theta^\star)}{\pi(\theta)\pi(y|\theta)f(\theta^\star|\theta)}.$$

It is clear therefore that if it is possible to use this proposal, the acceptance ratio does not depend on the sample path x, and we have a scheme exactly equivalent to that of the marginal updating scheme for $\pi(\theta|y)$. It is rarely possible to implement this scheme, but understanding it and its relationship with the marginal scheme is important for understanding some of the more advanced MCMC schemes that we will use later for inference in Markov process models.

The pseudo-marginal approach to “exact approximate” MCMC

Let us now reconsider the latent variable problem

$$\pi(\theta, x, y) = \pi(\theta)\pi(x|\theta)\pi(y|x, \theta).$$

and the problem of designing an MCMC algorithm for the marginal posterior $\pi(\theta|y)$. We have already seen that using an arbitrary proposal $q(\theta, \theta^\star)$, we can target this posterior distribution using the acceptance ratio

$$A = \frac{\pi(\theta^\star)\pi(y|\theta^\star)q(\theta^\star, \theta)}{\pi(\theta)\pi(y|\theta)q(\theta, \theta^\star)}.$$

As already explained, the difficulty is computation of the marginal likelihood of the data

$$\pi(y|\theta) = \int_X \pi(y|x, \theta)\pi(x|\theta)\,dx.$$

Although this is often analytically intractable, it is often straightforward to construct a Monte Carlo estimate of it, and such estimates can be used as the basis of a (highly computationally intensive) MC within MCMC algorithm. For example, we know that if we simulate realisations $x_1, x_2, \ldots, x_n$ from the model $\pi(x|\theta)$ for a given θ, then the Monte Carlo estimate

$$\widehat{\pi}(y|\theta) = \frac{1}{n}\sum_{i=1}^{n} \pi(y|x_i, \theta)$$

will converge to $\pi(y|\theta)$ as $n \to \infty$ by the law of large numbers. In that sense, $\widehat{\pi}(y|\theta)$ is said to be a **consistent** estimator of $\pi(y|\theta)$. However, it is also clear that for any n,

$$\begin{aligned} \mathsf{E}(\widehat{\pi}(y|\theta)) &= \frac{1}{n}\sum_{i=1}^{n} \mathsf{E}(\pi(y|x_i, \theta)) \\ &= \mathsf{E}(\pi(y|x_1, \theta)) \\ &= \int_X \pi(y|x_1, \theta)\pi(x_1|\theta)dx_1 \\ &= \pi(y|\theta) \end{aligned}$$

and so in that sense, $\widehat{\pi}(y|\theta)$ is said to be an **unbiased** estimator of $\pi(y|\theta)$.

Let us now consider how such estimates can be used within an MCMC algorithm targeting $\pi(\theta|y)$. We know that the “correct” acceptance ratio is

$$A = \frac{\pi(\theta^\star)\pi(y|\theta^\star)q(\theta^\star,\theta)}{\pi(\theta)\pi(y|\theta)q(\theta,\theta^\star)}.$$

If we now just plug in a Monte Carlo estimate of the marginal likelihood, we get the acceptance ratio

$$A = \frac{\pi(\theta^\star)\widehat{\pi}(y|\theta^\star)q(\theta^\star,\theta)}{\pi(\theta)\widehat{\pi}(y|\theta)q(\theta,\theta^\star)}.$$

It is clear that if our Monte Carlo estimate is good, then this ratio will be close to the correct ratio, and we could therefore hope that as a result, the target of the chain will be “close” to the correct target in some appropriate sense.

However, it turns out that if the Monte Carlo estimate that we plug in is **unbiased**, then the target is **exactly** the correct posterior, $\pi(\theta|y)$. To understand why, we must consider the marginal scheme as a joint update on a larger space which includes the Monte Carlo error in the marginal likelihood estimate. Specifically, we define the random variable W as

$$W = \frac{\widehat{\pi}(y|\theta)}{\pi(y|\theta)}.$$

When our Monte Carlo estimate of likelihood is unbiased, we have $\mathsf{E}(W|\theta) = 1,\ \forall\theta$. We can write the distribution of W as $\pi(w|\theta)$, and our unbiasedness property implies that

$$\int_W w\,\pi(w|\theta)\,dw = 1$$

for all θ. This turns out to be key to understanding why the algorithm works.

We can now regard our MCMC scheme as providing a joint update for θ and w, since we sample a new w at each MCMC iteration. In this case our proposal is $q((\theta,w),(\theta^\star,w^\star)) = q(\theta,\theta^\star)\pi(w^\star|\theta^\star)$. We can now re-write our acceptance ratio in the form

$$A = \frac{\pi(\theta^\star)\pi(y|\theta^\star)w^\star\pi(w^\star|\theta^\star)}{\pi(\theta)\pi(y|\theta)w\pi(w|\theta)} \times \frac{q(\theta^\star,\theta)\pi(w|\theta)}{q(\theta,\theta^\star)\pi(w^\star|\theta^\star)}.$$

From this it is clear that the target of the chain must be proportional to $\pi(\theta)\pi(y|\theta)\times\, w\pi(w|\theta)$. The corresponding marginal for θ can be obtained by integrating this target over the range of W. Using our unbiasedness property, this gives a density for θ proportional to $\pi(\theta)\pi(y|\theta)$, which is exactly the required posterior distribution $\pi(\theta|y)$. Our resulting Markov chain is therefore exact, despite the fact that an approximate Monte Carlo estimate of likelihood is used in the calculation of the acceptance ratio. See Beaumont (2003) and Andrieu & Roberts (2009) for further discussion of this technique. We will see how this technique can be used in practice for parameter inference in Markov process models in the next lecture.

*

References

Andrieu, C. & Roberts, G. O. (2009), 'The pseudo-marginal approach for efficient Monte Carlo computations', *Annals of Statistics* **37**(2), 697–725.

Beaumont, M. A. (2003), 'Estimation of population growth or decline in genetically monitored populations', *Genetics* **164**, 1139–1160.

Chib, S. (1995), 'Marginal likelihood from the Gibbs output', *Journal of the American Statistical Association* **90**(432), 1313–1321.

Doucet, A., de Freitas, N. & Gordon, N., eds (2001), *Sequential Monte Carlo Methods in Practice*, Springer, New York.

Hastings, W. K. (1970), ‘Monte Carlo sampling methods using Markov chains and their applications’, *Biometrika* **57**, 97–109.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. (1953), ‘Equations of state calculations by fast computing machines’, *Journal of Chemical Physics* **21**, 1087–1092.