

A primer on the theory of Markov chains as it relates to MCMC may be found in Chapter 3 of *Markov Chain Monte Carlo in Practice*.

Let  $\mathcal{S}$  denote the support of  $g$ .

The following scenario brings to light a condition that  $q$  should satisfy:

Suppose there is a proper subset  $A$  of  $\mathcal{S}$  such that the support of  $q(\cdot | \boldsymbol{x})$  is  $A$  for every  $\boldsymbol{x} \in A$ . Then once the chain has reached a state  $\boldsymbol{x}$  in  $A$ , it will never again leave  $A$ .

To insure that the above scenario doesn't occur, choose  $q$  so that the support of  $q(\cdot | \boldsymbol{x})$  is  $\mathcal{S}$  for every  $\boldsymbol{x}$ .

Choosing  $q$  in this way implies that the chain will be *irreducible*, which means that any set of states can be reached from any other set of states in a finite number of moves.

## *Convergence issues*

The *stationary distribution* (if it exists) of a Markov chain is the distribution  $F$  such that

$$\lim_{t \rightarrow \infty} P(X_t \leq x) = F(x) \quad \forall x.$$

In the Metropolis-Hastings algorithm, it was noted that virtually any proposal distribution will produce a chain that has stationary density  $g$ .

However, the proposal distribution used *does* have an effect on how many values need to be generated. There are two important issues in this regard:

- How quickly does the marginal distribution of the chain converge to  $g$ ?
- Once the chain has converged, how rapidly does it *mix*?

Mixing refers to the tendency of the chain to move around the support of  $g$ . For some proposal distributions, the chain can stay stuck at or near the same state for long periods of time.

---

---

### Example 17

Suppose the stationary distribution ( $g$ ) of the chain is  $N(0, 1)$ . We consider the effect of three different proposal distributions and two different starting values.

The proposals are

(a)  $N(x, (0.5)^2)$

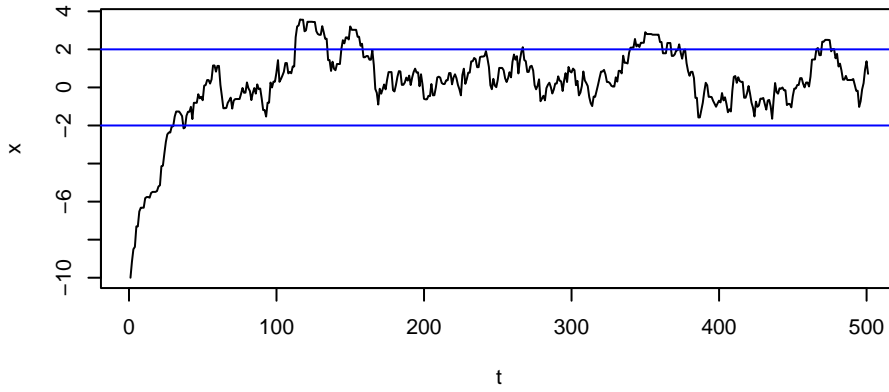
(b)  $N(x, (0.1)^2)$

(c)  $N(x, 10^2)$

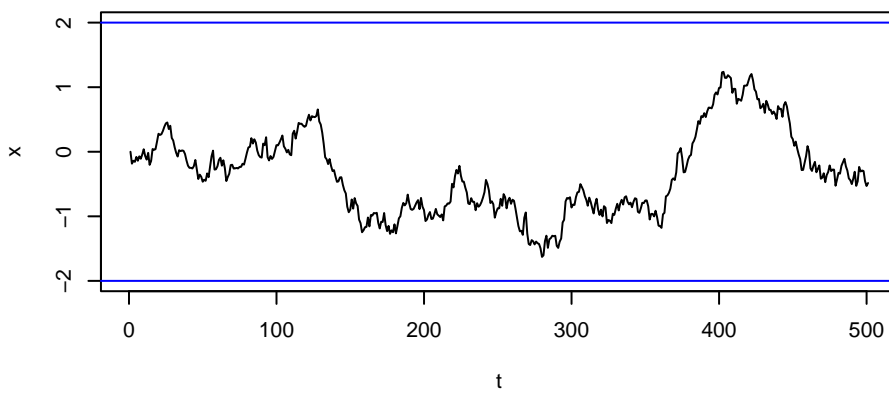
In the case of (a), the starting value is -10, and in the other two it is 0, i.e., the mode of  $g$ .

# Markov chains generated by MCMC

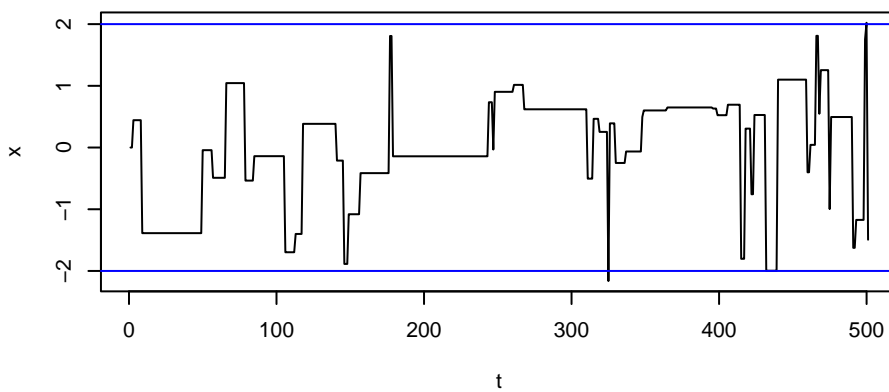
(a)



(b)



(c)



Note that in case (a), the chain converges fairly rapidly and then mixes fairly quickly thereafter.

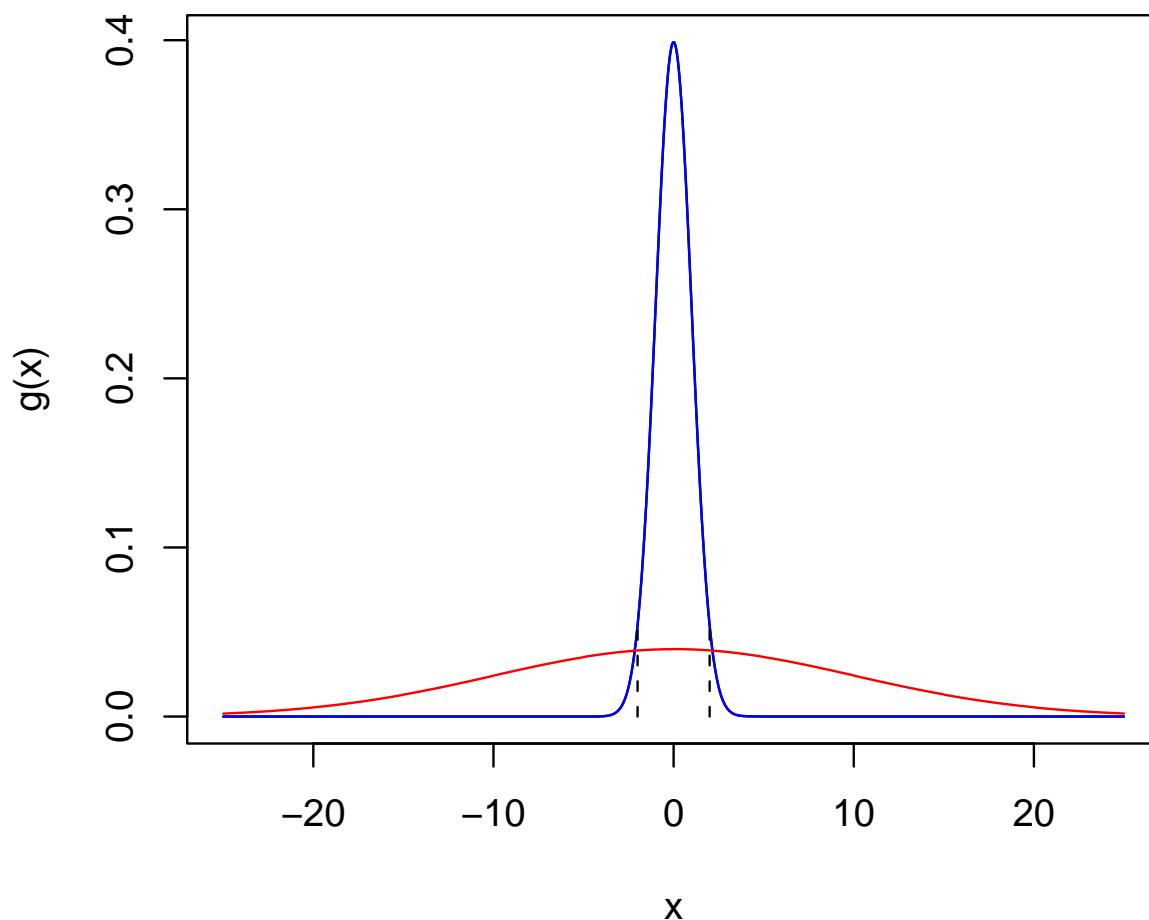
In the other two cases, the chains are mixing slowly. Even though (b) and (c) start at the mode of  $g$ , they will have to run much longer than (a) in order to provide good estimates of quantities associated with  $g$ .

For the proposal in (b), the chain doesn't get stuck at the same value very often, but once near  $x$ , it takes a long time to move away from a neighborhood of  $x$  because of the small variance of the proposal distribution.

Note that the proposal distribution in all three cases has the property that  $q(y|x) = q(x|y)$ , implying that

$$\alpha(x, y) = \min \left( 1, \frac{g(y)}{g(x)} \right).$$

Now in case (c), suppose  $X_t$  takes on a value  $x$  in  $(-2, 2)$ . Then there is a high probability that a value  $y$  from the proposal distribution will be outside  $(-2, 2)$ , as illustrated in the graph on the next page.



Stationary distribution: —

Proposal for  $x = 0$ : —

When the generated  $y$  is outside  $(-2, 2)$ , the ratio  $g(y)/g(0)$  will be very small. This results in a low probability of the chain moving.

## *Canonical forms for proposal distribution*

### Metropolis algorithm

In the Metropolis algorithm, only *symmetric* proposals are considered, i.e., ones satisfying

$$q(\mathbf{y}|\mathbf{x}) = q(\mathbf{x}|\mathbf{y}) \quad \forall \mathbf{x} \text{ and } \mathbf{y}.$$

With such a proposal, the acceptance probability becomes

$$\alpha(\mathbf{x}, \mathbf{y}) = \min \left( 1, \frac{g(\mathbf{y})}{g(\mathbf{x})} \right).$$

An example of a symmetric proposal distribution  $q(\cdot|\mathbf{x})$  is the multivariate normal with mean  $\mathbf{x}$  and covariance  $\Sigma_0$ .

A special case of the Metropolis algorithm is *random-walk Metropolis*, in which

$$q(\mathbf{y}|\mathbf{x}) = q(|\mathbf{x} - \mathbf{y}|).$$

Example 17 was an example of the Metropolis algorithm. It illustrates well that it is important to choose the scale of the proposal distribution wisely. Either too small or too large a scale for the proposal will cause the chain to mix slowly.

### Independence sampler

The independence sampler is a special case of Metropolis-Hastings in which the proposal does not depend on  $x$ , i.e.,

$$q(\mathbf{y}|\mathbf{x}) = q(\mathbf{y}) \quad \forall \mathbf{x} \text{ and } \mathbf{y}.$$

In this case, the acceptance probability takes the form

$$\alpha(\mathbf{x}, \mathbf{y}) = \min \left( 1, \frac{w(\mathbf{y})}{w(\mathbf{x})} \right),$$

where  $w(\mathbf{x}) = g(\mathbf{x})/q(\mathbf{x})$ .

The independence sampler can either work very well or very badly.

To work well,  $q$  should be a reasonably good approximation to  $g$ . It is best to err in the direction of a proposal that is heavier tailed than  $g$ .

### Example 18

We will apply the independence sampler to approximate the joint density of the *unrounded* data in Exercise 5, pp. 96-97 of GCSR, given the rounded data.

Let  $\mathbf{z} = (z_1, \dots, z_5)^T$  be the unrounded data and  $\mathbf{y}$  the rounded data.

Then, one can show that

$$m(\mathbf{z}|\mathbf{y}) \propto \frac{1}{s_z^4} \propto \left( \frac{1}{\sum_{i=1}^5 (z_i - \bar{z})^2} \right)^2,$$

for  $y_i - 0.5 < z_i < y_i + 0.5$ ,  $i = 1, \dots, 5$ . Otherwise  $m(\mathbf{z}|\mathbf{y}) = 0$ .

As a proposal distribution, consider

$$q(\mathbf{z}) = \prod_{i=1}^5 I_{(y_i-0.5, y_i+0.5)}(z_i),$$

which is the density of independent uniform random variables.

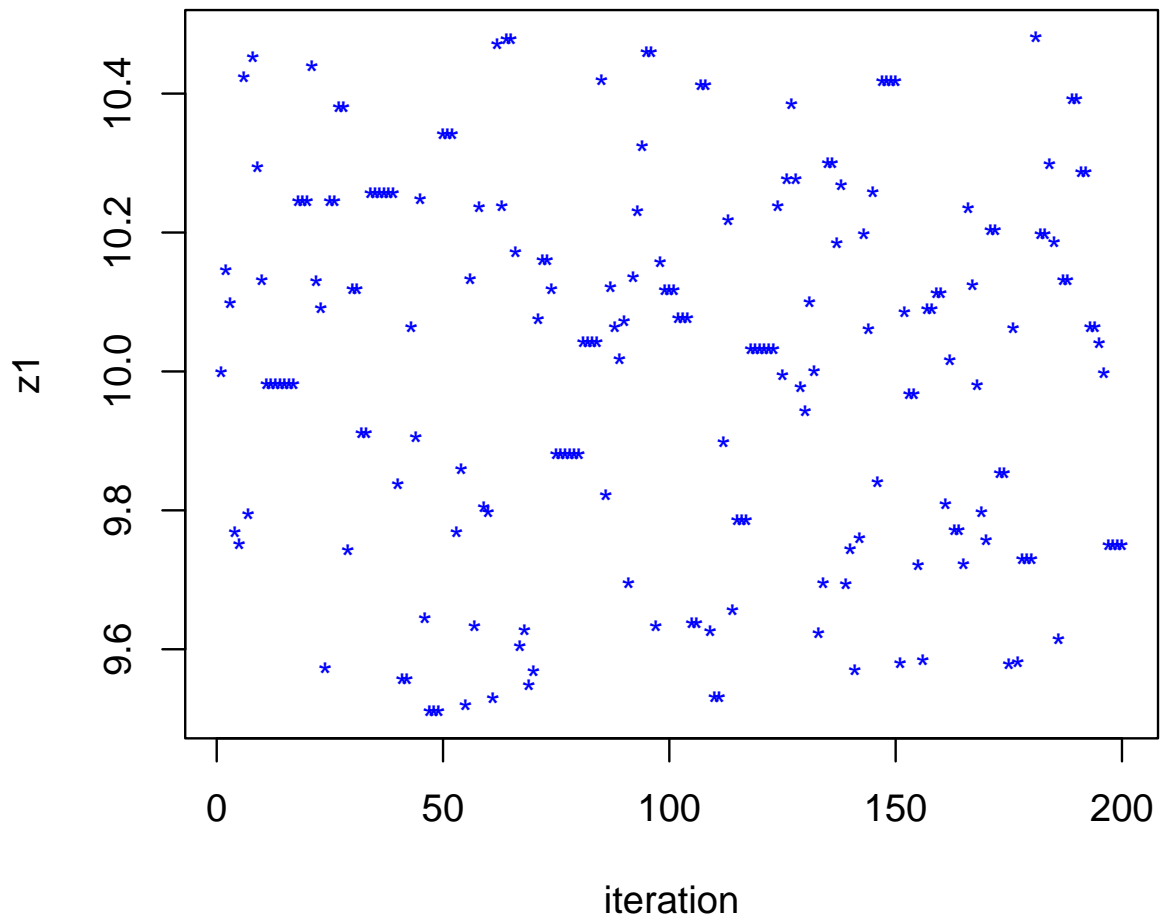
Because  $q$  is constant on the support of  $m(\mathbf{z}|\mathbf{y})$ , this independence sampler is also a Metropolis algorithm, and we have

$$\alpha(\mathbf{u}, \mathbf{v}) = \min\left(1, \frac{s_{\mathbf{u}}^4}{s_{\mathbf{v}}^4}\right)$$

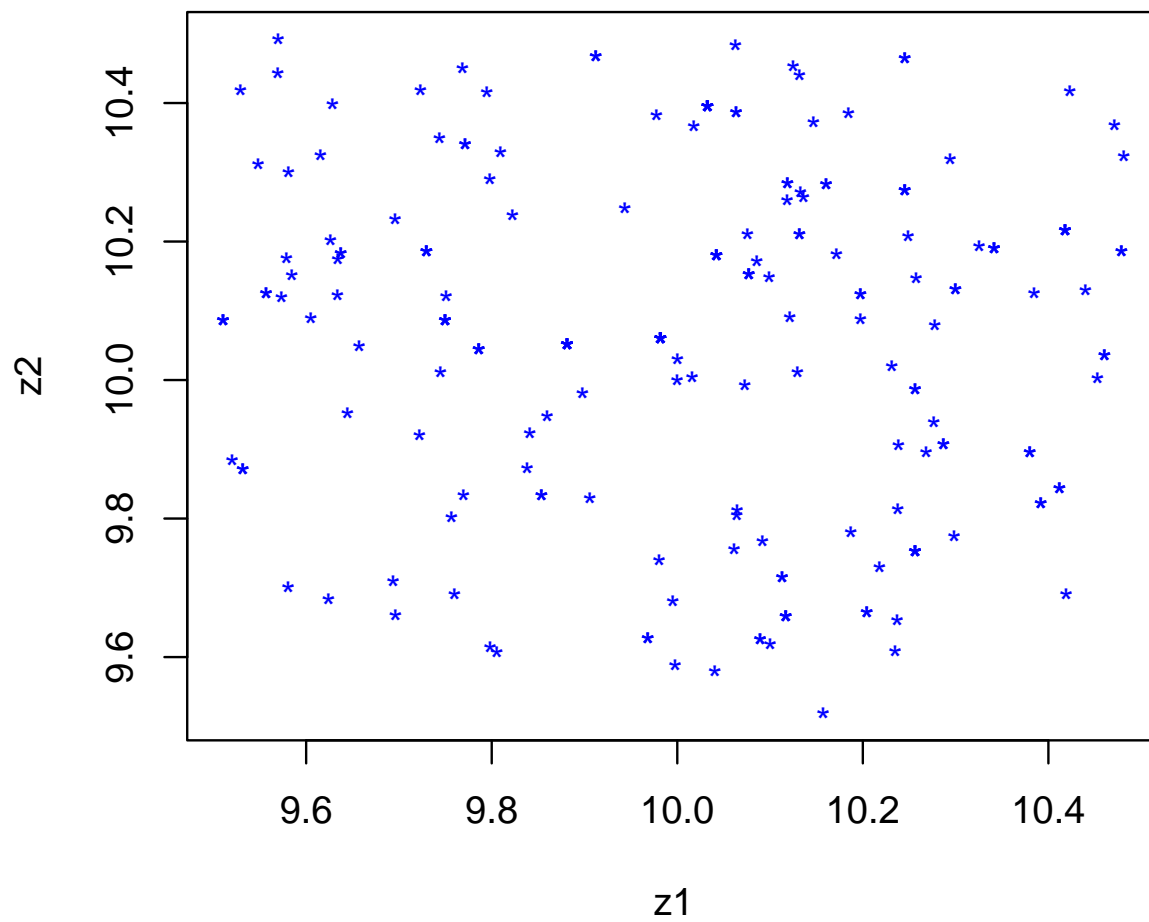
for each  $\mathbf{u}$  and  $\mathbf{v}$  in the support of  $m(\mathbf{z}|\mathbf{y})$ .

The chain was started at  $\mathbf{z} = \mathbf{y}$ , and 10,000 values were generated.

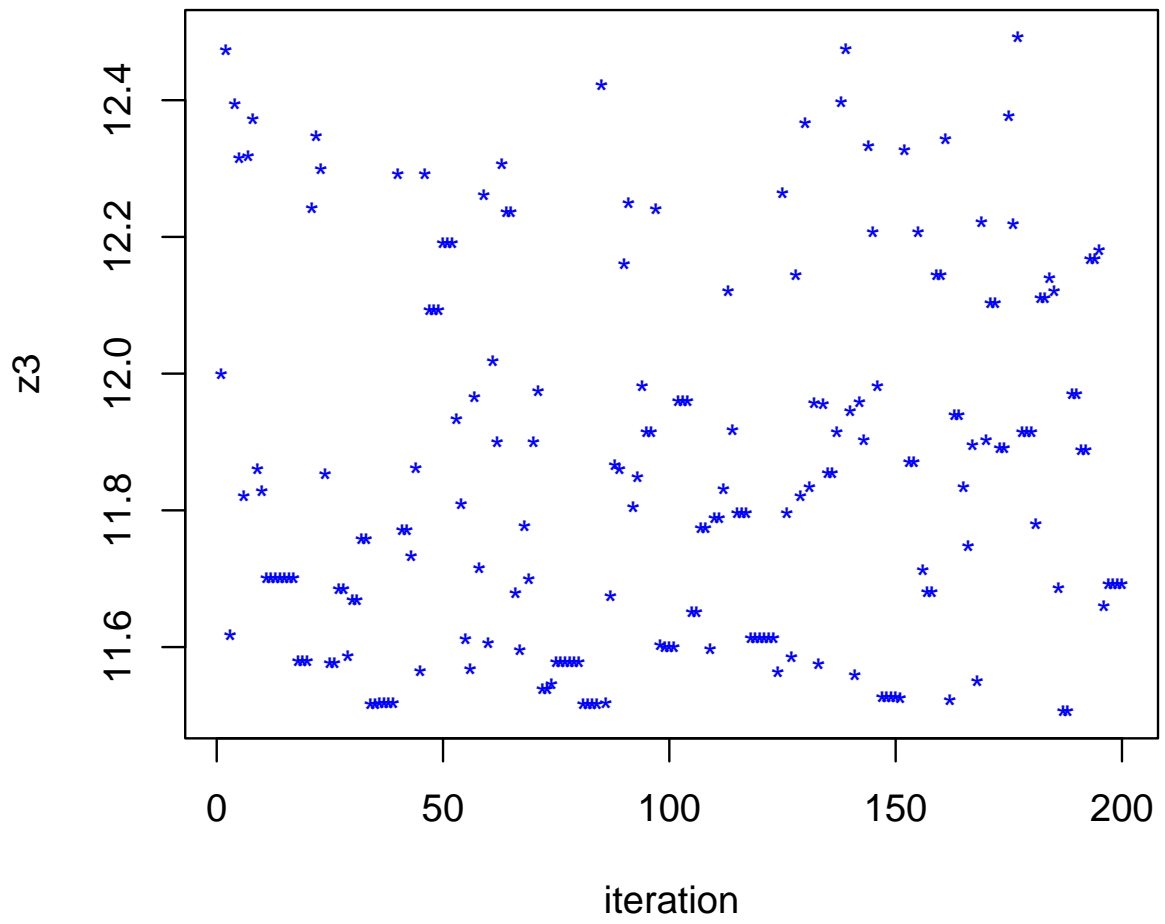
*First 200 draws for  $z_1$*



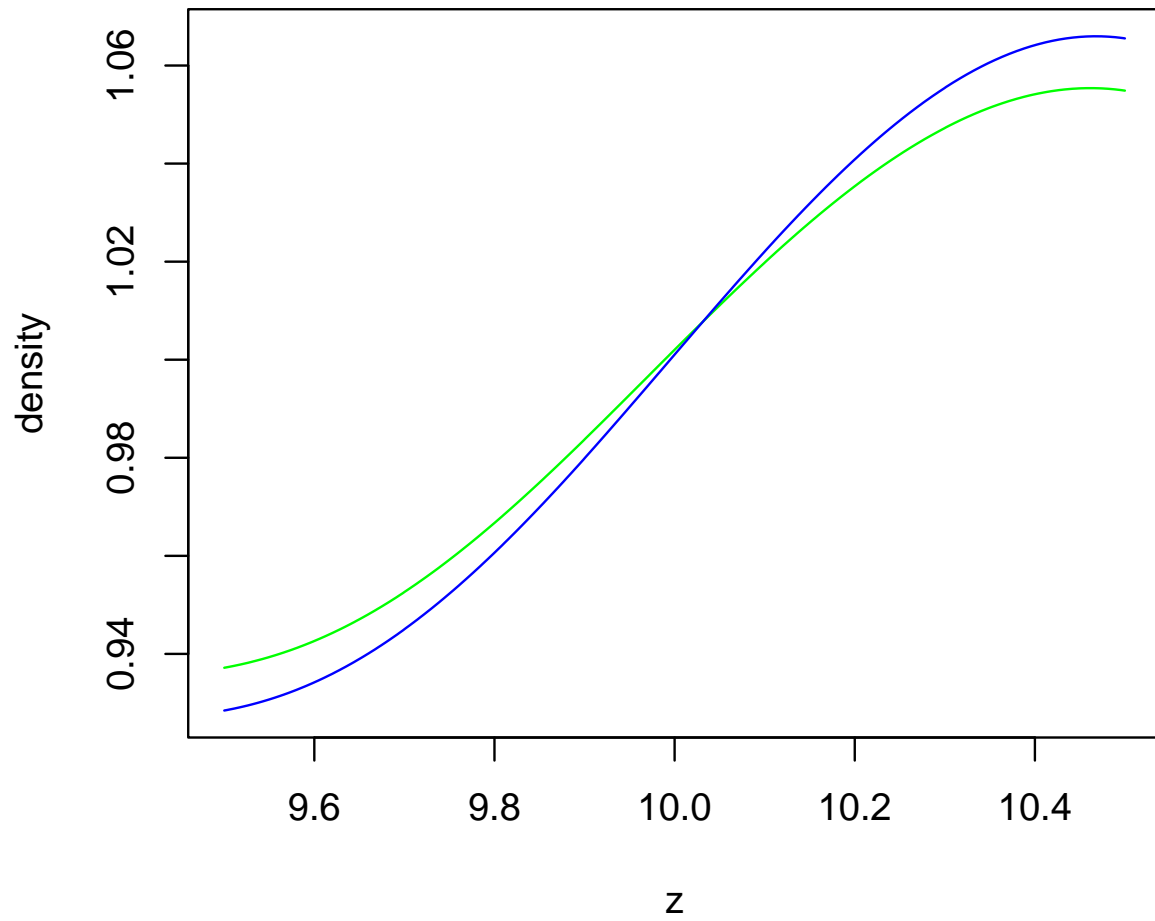
*Scatterplot of  $z_2$  vs.  $z_1$  in first 200 draws*



*First 200 draws for  $z_3$*

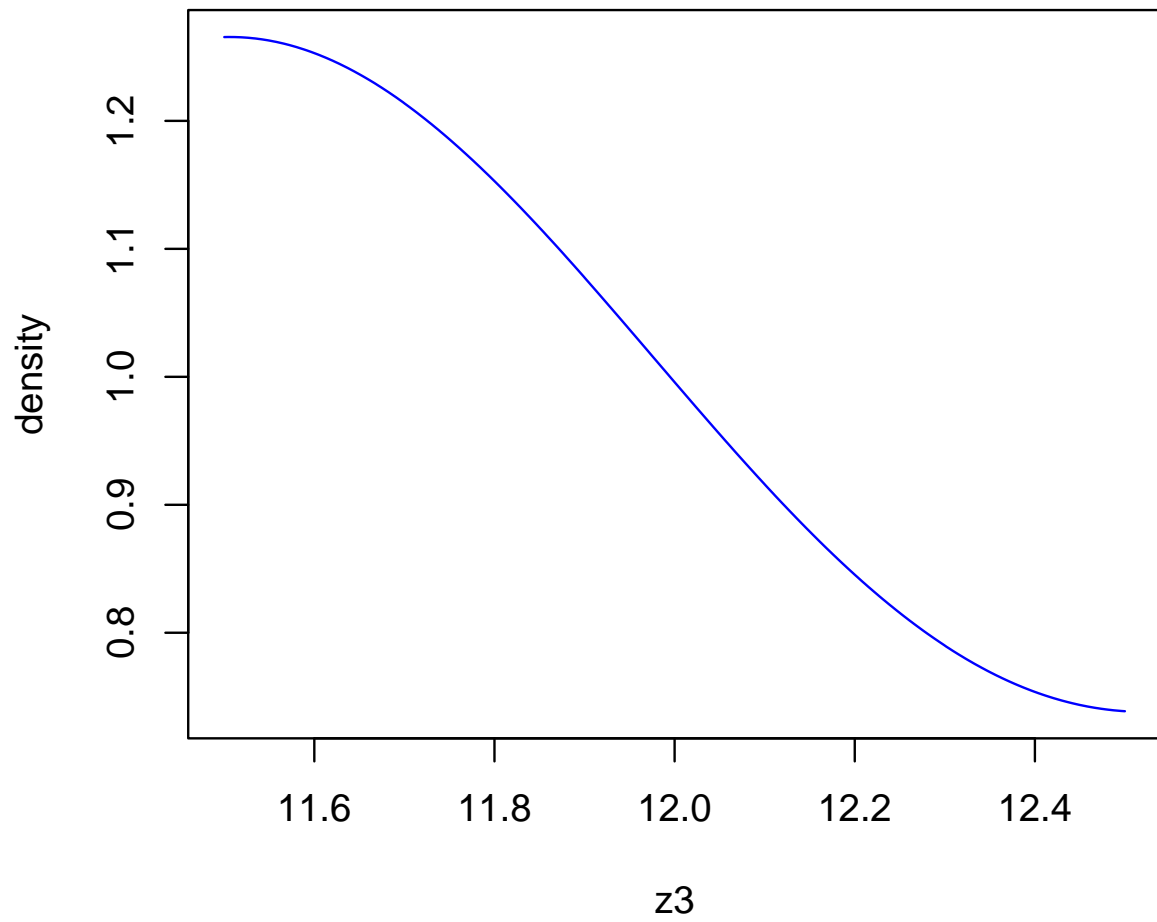


*Density estimates for  $z_1$  and  $z_2$*



$z_1$  estimate: —  
 $z_2$  estimate: —

*Density estimate for  $z_3$*



## Single-component Metropolis-Hastings (scMH)

Suppose values  $\boldsymbol{x}$  in the support of  $g$  are divided into components as follows:

$$\boldsymbol{x} = (\boldsymbol{x}_{\cdot 1}, \dots, \boldsymbol{x}_{\cdot m}),$$

where  $\boldsymbol{x}_{\cdot 1}, \dots, \boldsymbol{x}_{\cdot m}$  are vectors of possibly different dimensions.

In scMH, these components are updated one by one. Let  $\boldsymbol{x}^i$  denote the vector containing all the components of  $\boldsymbol{x}$  except  $\boldsymbol{x}_{\cdot i}$ .

One iteration of scMH consists of  $m$  updating steps. Let  $\boldsymbol{x}_t$  denote the value of the chain at iteration  $t$  and  $\boldsymbol{x}_{t \cdot i}$  be the  $i$ th component of  $\boldsymbol{x}_t$ .

A candidate,  $\boldsymbol{y}_{\cdot i}$ , for  $\boldsymbol{x}_{(t+1) \cdot i}$  is generated from a proposal distribution  $q_i(\cdot | \boldsymbol{x}_{t \cdot i}, \boldsymbol{x}_{t, -i})$ , where

$$\boldsymbol{x}_{t, -i} = (\boldsymbol{x}_{(t+1) \cdot 1}, \dots, \boldsymbol{x}_{(t+1) \cdot (i-1)}, \\ \boldsymbol{x}_{t \cdot (i+1)}, \dots, \boldsymbol{x}_{t \cdot m}).$$

So, the first  $i - 1$  components of  $\mathbf{x}_{t,-i}$  have been updated, but the other  $m - i$  have not. The candidate  $\mathbf{y}_{.i}$  is accepted with probability  $\alpha(\mathbf{x}_{t,-i}, \mathbf{x}_{t.i}, \mathbf{y}_{.i})$ , which is the smaller of 1 and

$$\frac{g(\mathbf{y}_{.i}|\mathbf{x}_{t,-i})q_i(\mathbf{x}_{t.i}|\mathbf{y}_{.i}, \mathbf{x}_{t,-i})}{g(\mathbf{x}_{t.i}|\mathbf{x}_{t,-i})q_i(\mathbf{y}_{.i}|\mathbf{x}_{t.i}, \mathbf{x}_{t,-i})}.$$

Here,  $g(\cdot|\mathbf{x}^i)$  represents the *full conditional density* of  $\mathbf{X}_{.i}$ , i.e., the conditional of  $\mathbf{X}_{.i}$  given all the other components of  $\mathbf{X}$ .

It is important to note that the marginal of each  $\mathbf{X}_{.i}$  *does not* need to be known. This is because

$$\frac{g(\mathbf{y}_{.i}|\mathbf{x}^i)}{g(\mathbf{x}_{.i}|\mathbf{x}^i)} = \frac{g(\mathbf{x}_{.1}, \dots, \mathbf{x}_{.(i-1)}, \mathbf{y}_{.i}, \mathbf{x}_{.(i+1)}, \dots, \mathbf{x}_{.m})}{g(\mathbf{x})}.$$