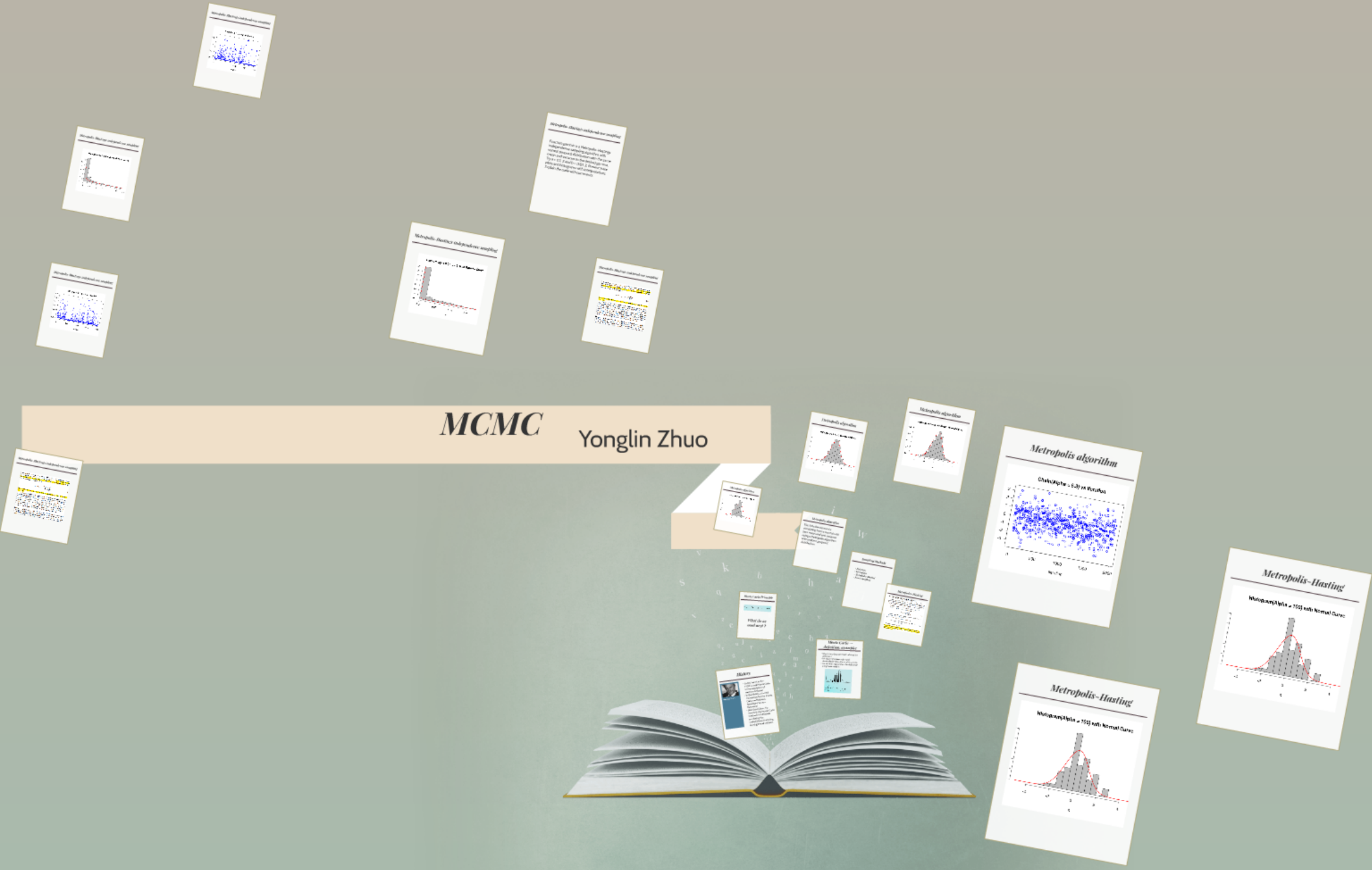
MCMC

Yonglin Zhuo



MCMC

Yonglin Zhuo

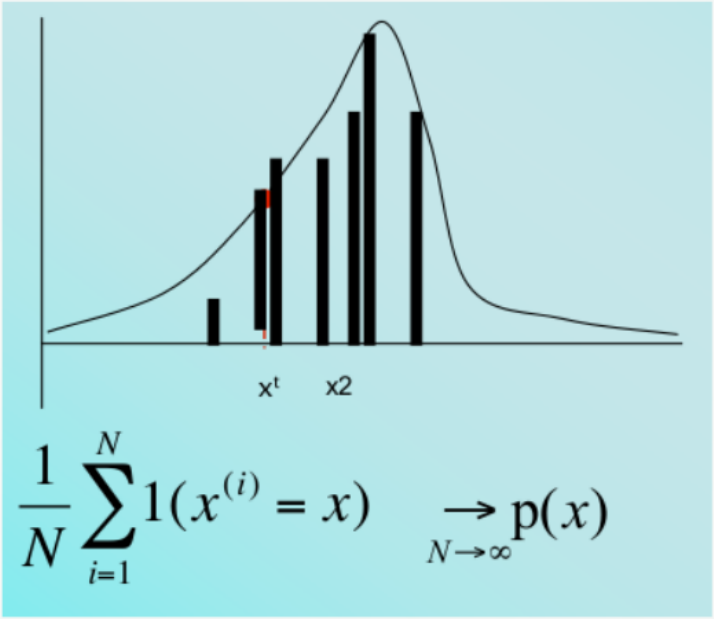
History

Stanislaw Ulam

- Enrico Fermi in the 1930's used Monte Carlo in the calculation of neutron diffusion
- In the 1940's, a formal foundation for the Monte Carlo method was developed by von Neumann
- Stanislaw Ulam, He invented the Monte Carlo method in 1946while pondering the probabilities of winning acard game of solitaire.

Monte Carlo -- definition, examples

- Given a very large set X and a distribution p(x) over it
- We draw i.i.d. (independent and identically distributed) a set of N samples
- We can then approximate the distribution using these samples



Monte Carlo Principle

$$E_N(f) = \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)}) \underset{N\to\infty}{\rightarrow} E(f) = \sum_x f(x)\,\mathrm{p}(x)$$

What do we need next ?

Sampling Methods

- *Rejection*
- *Metropolis*
- *Metropolis-Hasting*
- *Exact Sampling*

a

j

X

Metrop

7.1 METROPOLIS–HASTIN

A very general method for constructi
algorithm [324, 460]. The method be
drawn at random from some starti
$f\left(\mathbf{x}^{(0)}\right) > 0$. Given $\mathbf{X}^{(t)} = \mathbf{x}^{(t)}$, the al

1. Sample a candidate value $\mathbf{X}^*$ fr
2. Compute the *Metropolis–Hasti*

$R(\mathbf{u},$

Note that $R\left(\mathbf{x}^{(t)}, \mathbf{X}^*\right)$ is alway
only occur if $f\left(\mathbf{x}^{(t)}\right) > 0$ and g

Metropolis-Hasting

7.1 METROPOLIS–HASTINGS ALGORITHM

A very general method for constructing a Markov chain is the Metropolis–Hastings algorithm [324, 460]. The method begins at $t = 0$ with the selection of $\mathbf{X}^{(0)} = \mathbf{x}^{(0)}$ drawn at random from some starting distribution g, with the requirement that $f\left(\mathbf{x}^{(0)}\right) > 0$. Given $\mathbf{X}^{(t)} = \mathbf{x}^{(t)}$, the algorithm generates $\mathbf{X}^{(t+1)}$ as follows:

1. Sample a candidate value $\mathbf{X}^*$ from a *proposal distribution* $g\left(\cdot \mid \mathbf{x}^{(t)}\right)$.
2. Compute the *Metropolis–Hastings ratio* $R\left(\mathbf{x}^{(t)}, \mathbf{X}^*\right)$, where

Metropolis-Hasting

7.1 METROPOLIS–HASTINGS ALGORITHM

A very general method for constructing a Markov chain is the Metropolis–Hastings algorithm [324, 460]. The method begins at $t = 0$ with the selection of $\mathbf{X}^{(0)} = \mathbf{x}^{(0)}$ drawn at random from some starting distribution g, with the requirement that $f\left(\mathbf{x}^{(0)}\right) > 0$. Given $\mathbf{X}^{(t)} = \mathbf{x}^{(t)}$, the algorithm generates $\mathbf{X}^{(t+1)}$ as follows:

1. Sample a candidate value $\mathbf{X}^*$ from a *proposal distribution* $g\left(\cdot|\,\mathbf{x}^{(t)}\right)$.
2. Compute the *Metropolis–Hastings ratio* $R\left(\mathbf{x}^{(t)}, \mathbf{X}^*\right)$, where

$$R(\mathbf{u}, \mathbf{v}) = \frac{f(\mathbf{v})\, g(\mathbf{u}|\,\mathbf{v})}{f(\mathbf{u})\, g(\mathbf{v}|\,\mathbf{u})}. \tag{7.1}$$

   Note that $R\left(\mathbf{x}^{(t)}, \mathbf{X}^*\right)$ is always defined, because the proposal $\mathbf{X}^* = \mathbf{x}^*$ can only occur if $f\left(\mathbf{x}^{(t)}\right) > 0$ and $g\left(\mathbf{x}^*|\,\mathbf{x}^{(t)}\right) > 0$.

3. Sample a value for $\mathbf{X}^{(t+1)}$ according to the following:

$$\mathbf{X}^{(t+1)} = \begin{cases} \mathbf{X}^* & \text{with probability } \min\left\{R\left(\mathbf{x}^{(t)}, \mathbf{X}^*\right), 1\right\}, \\ \mathbf{x}^{(t)} & \text{otherwise.} \end{cases} \tag{7.2}$$

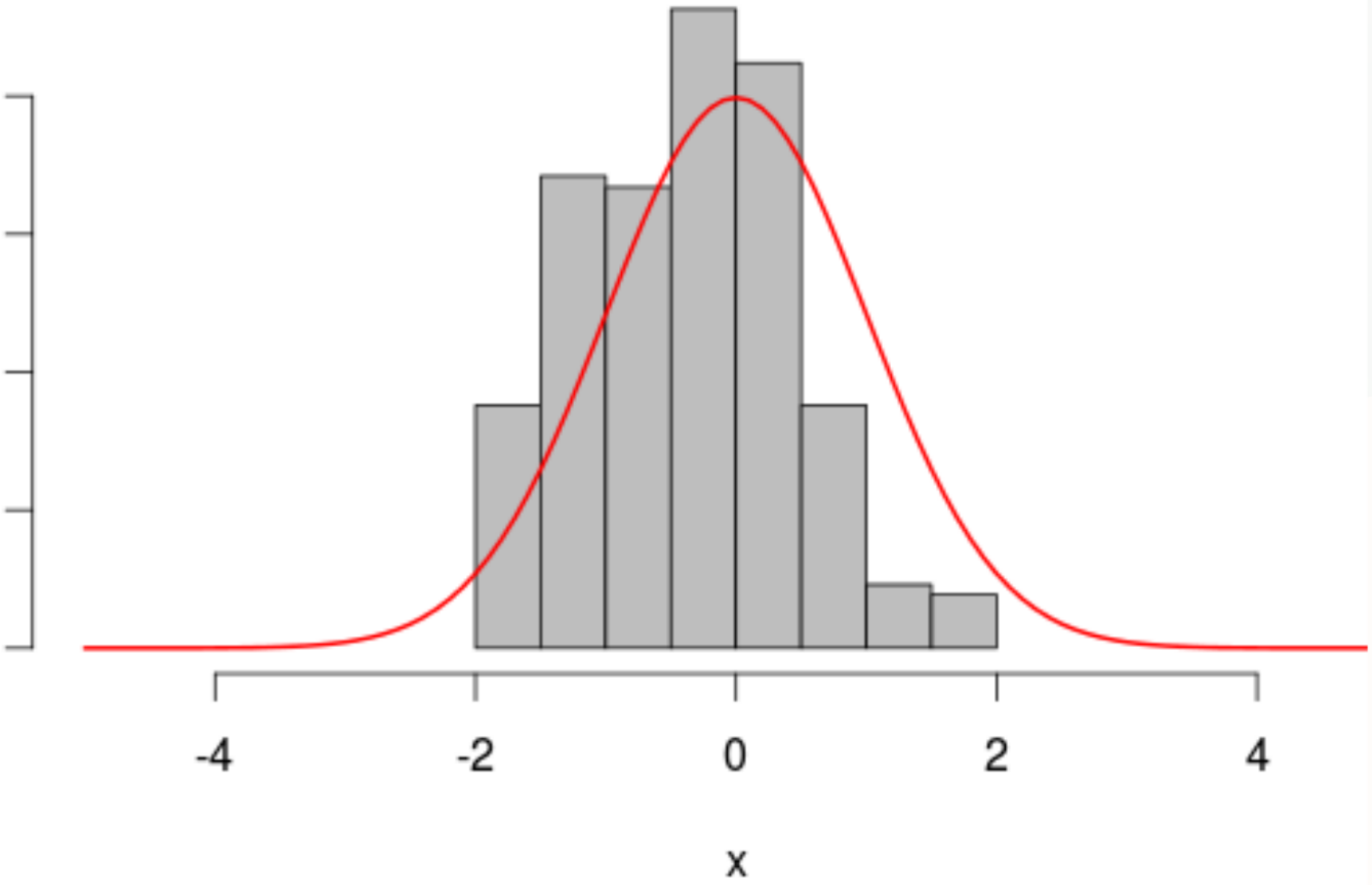
4. Increment t and return to step 1.

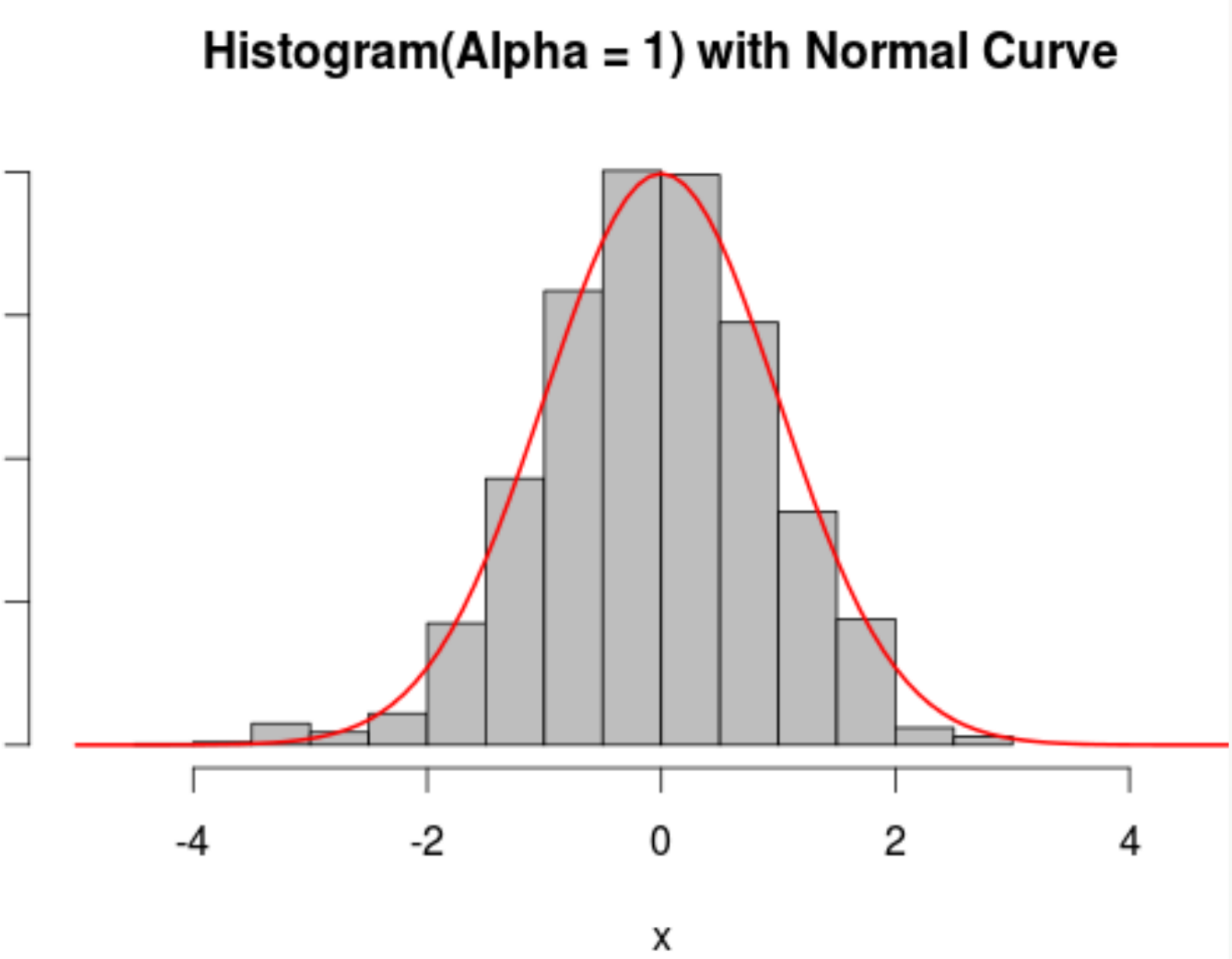
We will call the tth iteration the process that generates $\mathbf{X}^{(t)} = \mathbf{x}^{(t)}$. When the proposal distribution is symmetric so that $g\left(\mathbf{x}^{(t)}|\mathbf{x}^*\right) = g\left(\mathbf{x}^*|\mathbf{x}^{(t)}\right)$, the method is known as the Metropolis algorithm [460].

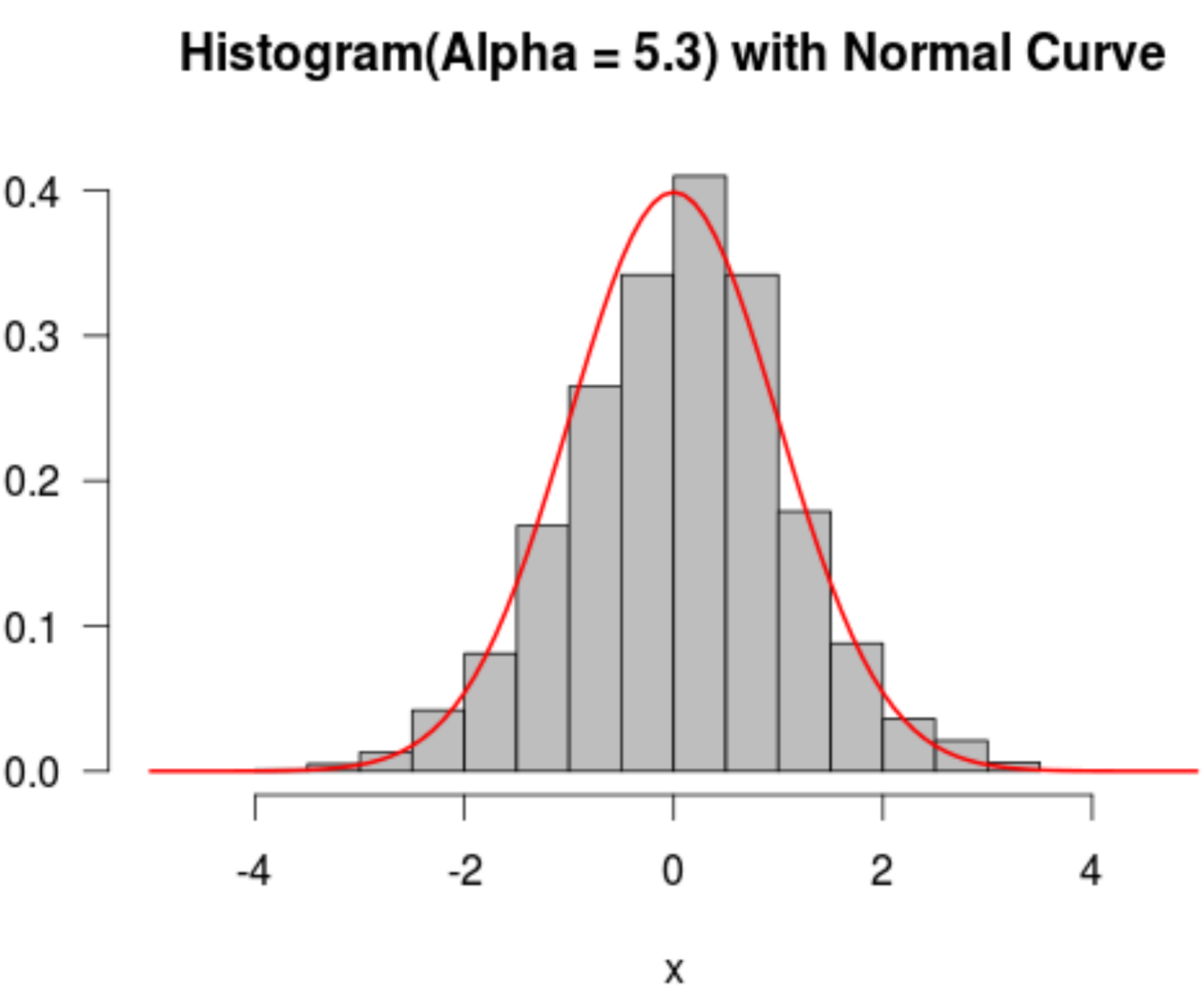
Metropolis algorithm

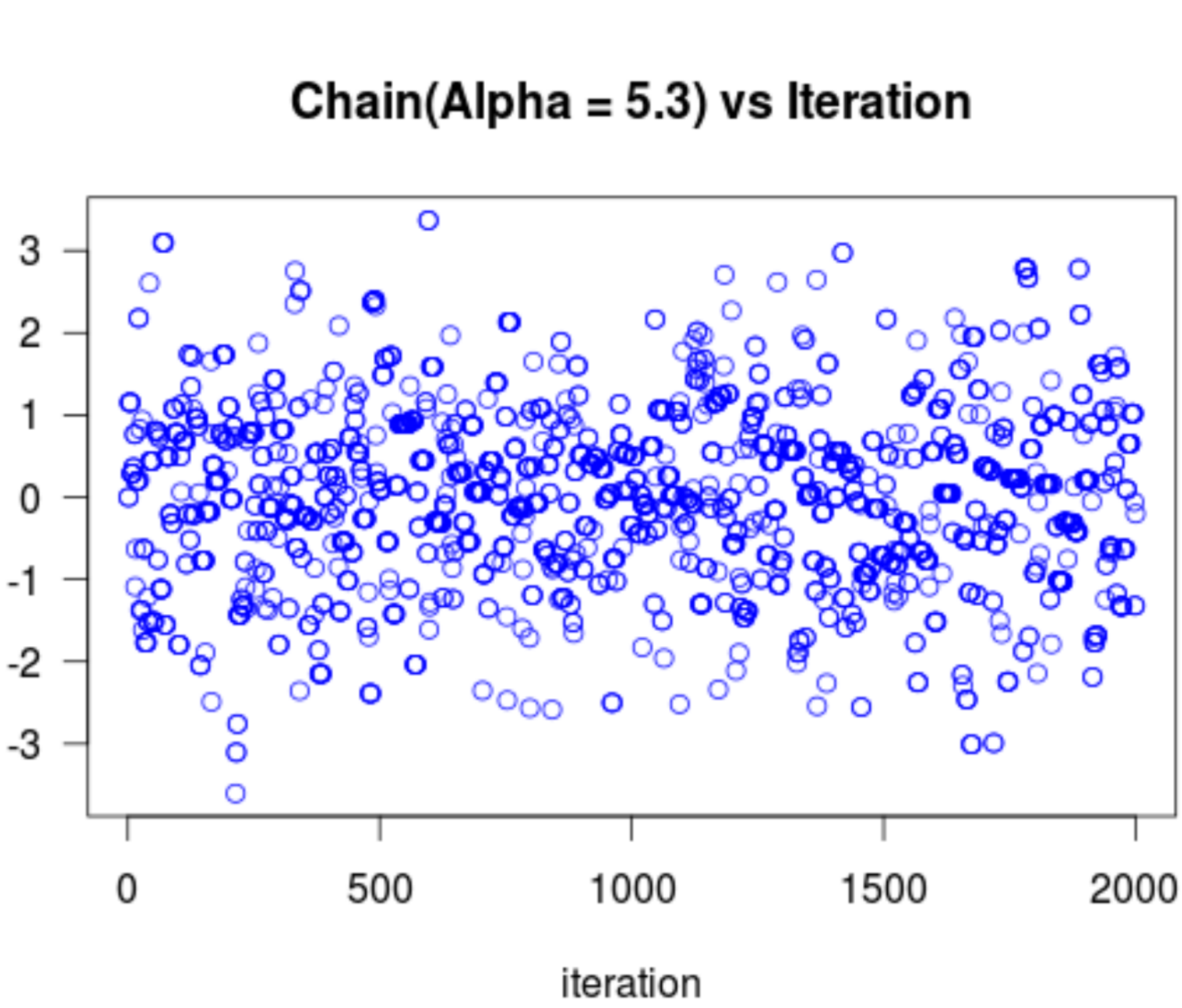
The function normm is simulating from a normal with zero mean and unit variance using a Metropolis algorithm with uniform proposal distribution.

Histogram(Alpha = 0.1) with Normal Curve








Chain(Alpha = 5.3) vs Iteration
3
2
1
0
-1
-2
-3
0
500
1000
1500
2000
iteration

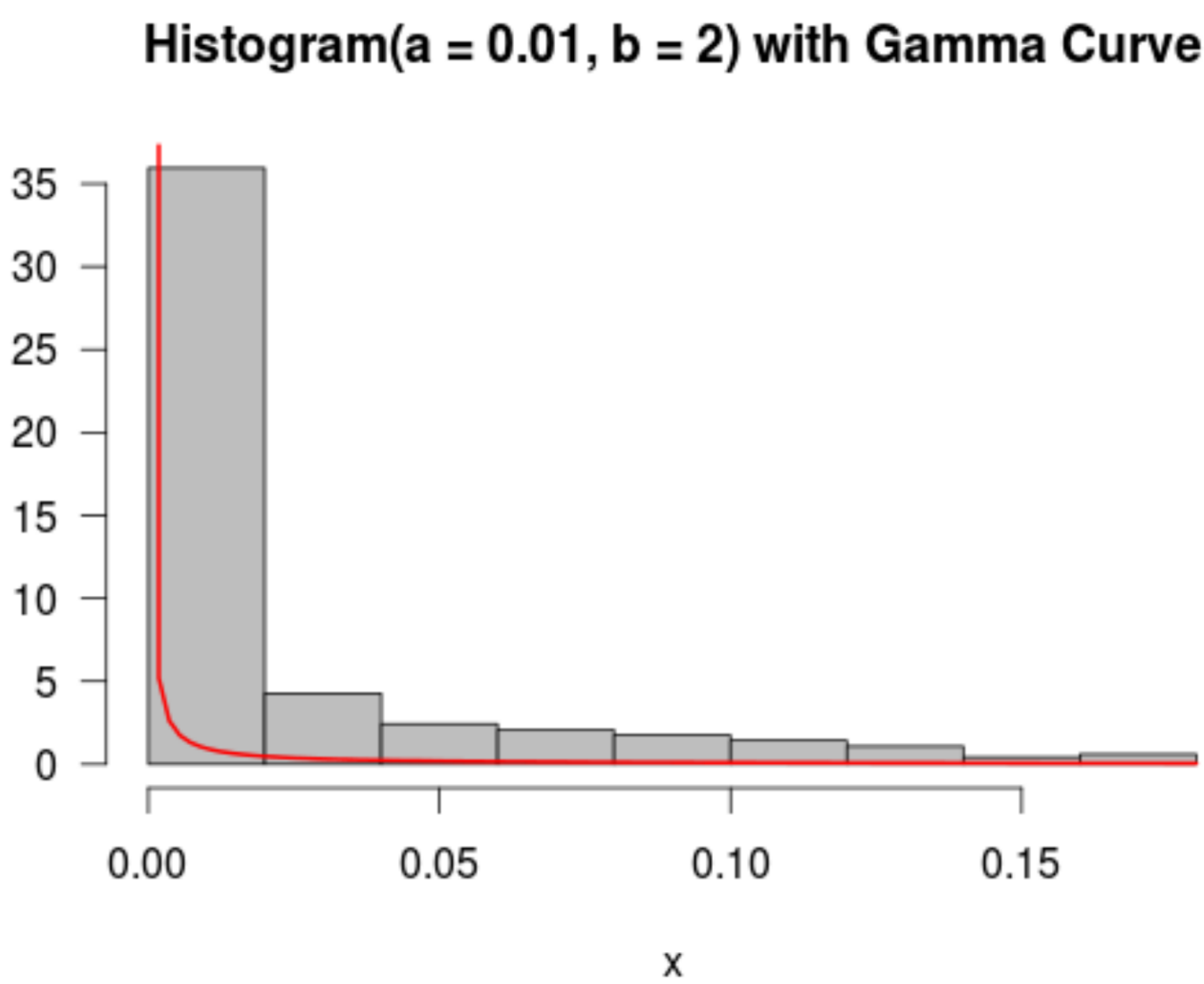
Suppose that the proposal distribution for the Metropolis–Hastings algorithm is chosen such that $g\left(\mathbf{x}^* | \mathbf{x}^{(t)}\right) = g(\mathbf{x}^*)$ for some fixed density g. This yields an independence chain, where each candidate value is drawn independently of the past. In this case, the Metropolis–Hastings ratio is

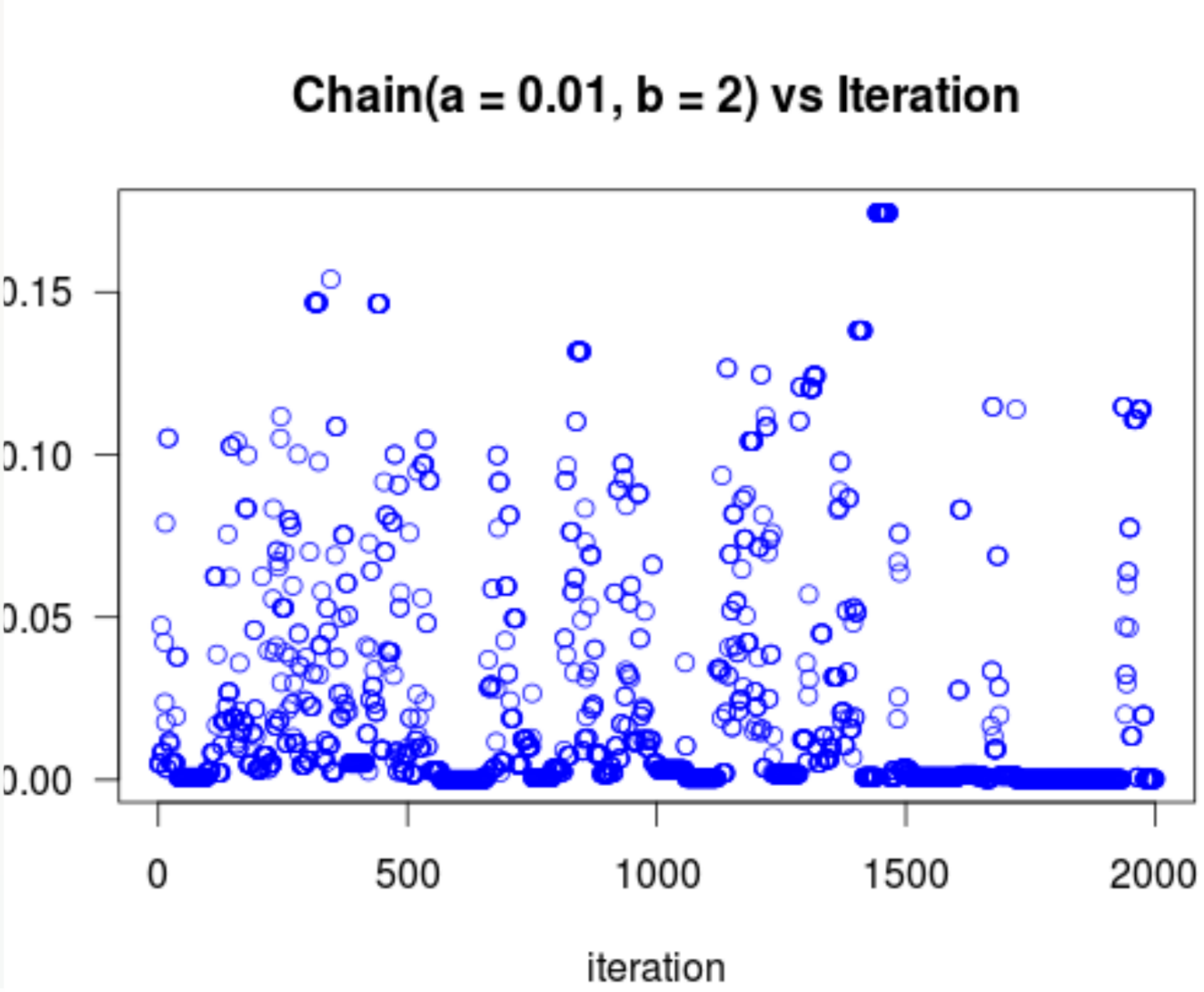
$$R\left(\mathbf{x}^{(t)}, \mathbf{X}^*\right) = \frac{f\left(\mathbf{X}^*\right) g\left(\mathbf{x}^{(t)}\right)}{f\left(\mathbf{x}^{(t)}\right) g\left(\mathbf{X}^*\right)}. \tag{7.4}$$

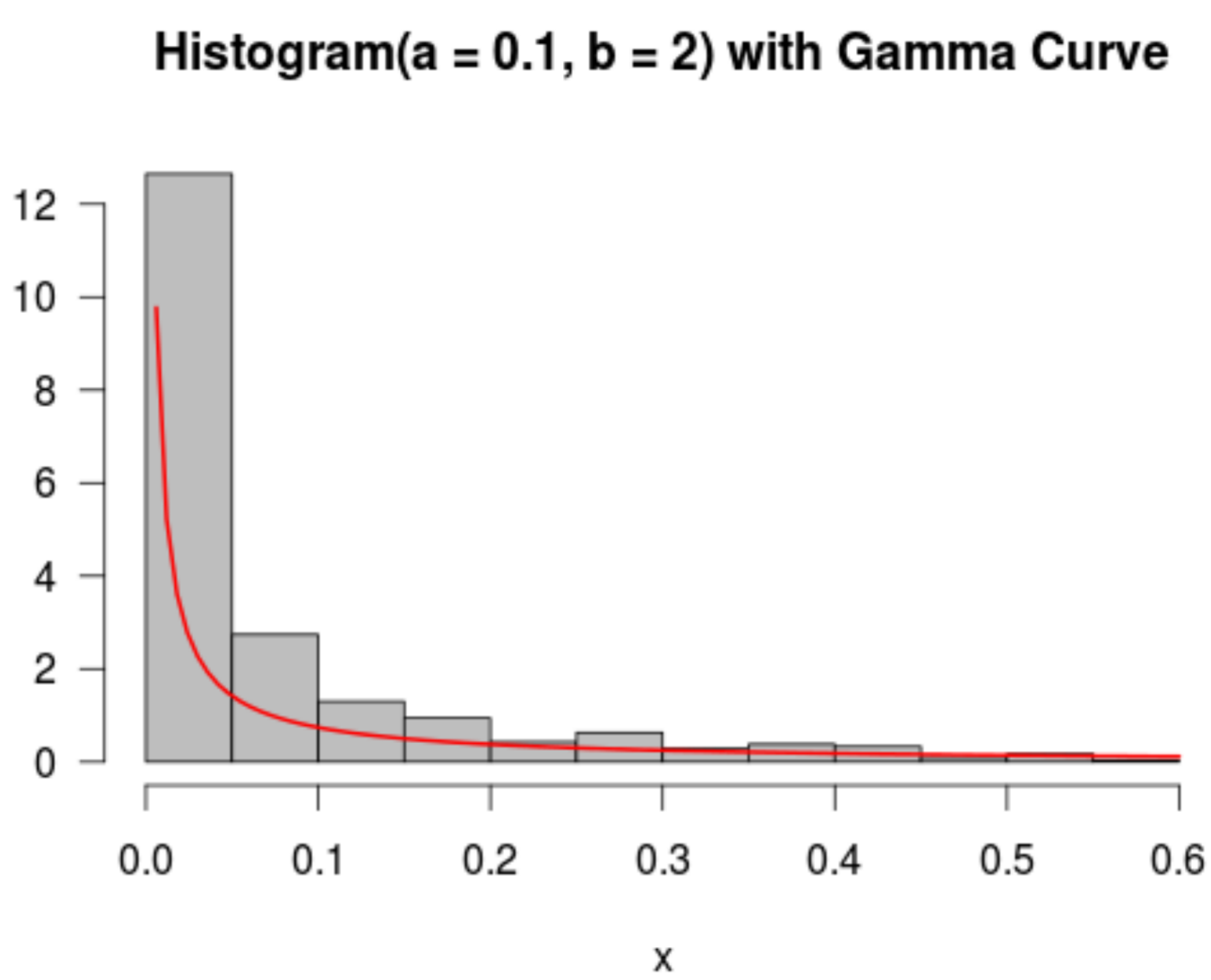
The resulting Markov chain is irreducible and aperiodic if $g(\mathbf{x}) > 0$ whenever $f(\mathbf{x}) > 0$.

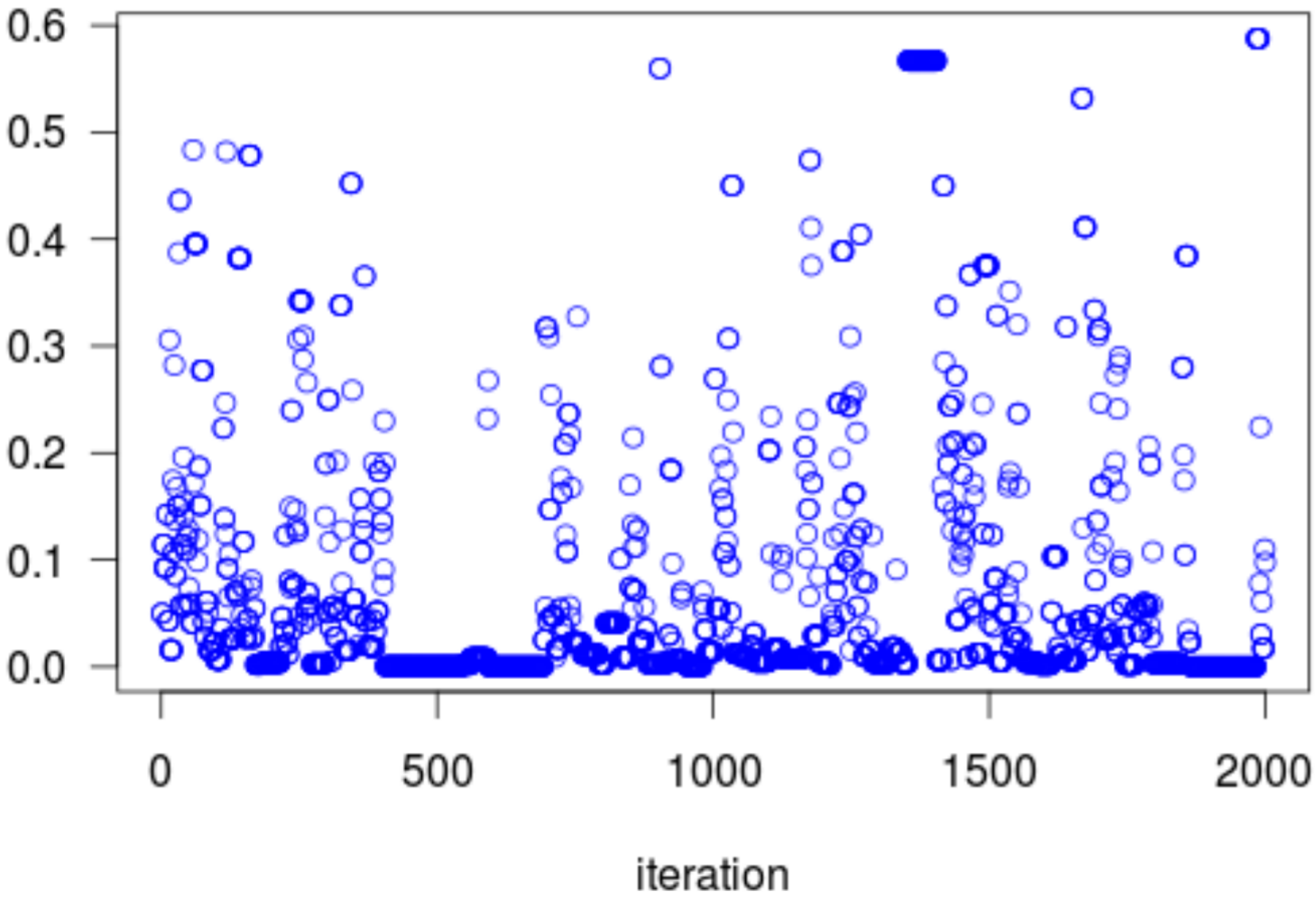
Notice that the Metropolis–Hastings ratio in (7.4) can be reexpressed as the ratio of importance ratios (see Section 6.4.1) where f is the target and g is the envelope: If $w^* = f\left(\mathbf{X}^*\right) / g\left(\mathbf{X}^*\right)$ and $w^{(t)} = f\left(\mathbf{x}^{(t)}\right) / g\left(\mathbf{x}^{(t)}\right)$, then $R\left(\mathbf{x}^{(t)}, \mathbf{X}^*\right) = w^* / w^{(t)}$. This reexpression indicates that when $w^{(t)}$ is much larger than typical w^* values, then the chain will tend to get stuck for long periods at the current value. Therefore, the criteria discussed in Section 6.3.1 for choosing importance sampling envelopes are also relevant here for choosing proposal distributions: The proposal distribution g should resemble the target distribution f, but should cover f in the tails.

Example 7.1 (Bayesian Inference) MCMC methods like the Metropolis–Hastings algorithm are particularly popular tools for Bayesian inference, where some data $\mathbf{y}$ are observed with likelihood function $L(\boldsymbol{\theta}|\mathbf{y})$ for parameters $\boldsymbol{\theta}$ which have prior distribution $p(\boldsymbol{\theta})$. Bayesian inference is based on the posterior distribution $p(\boldsymbol{\theta}|\mathbf{y}) =$


Histogram(a = 0.01, b = 2) with Gamma Curve
35
30
25
20
15
10
5
0
0.00
0.05
0.10
0.15
x


Chain(a = 0.01, b = 2) vs Iteration
0.15
0.10
0.05
0.00
0
500
1000
1500
2000
iteration


Histogram(a = 0.1, b = 2) with Gamma Curve
12
10
8
6
4
2
0
0.0
0.1
0.2
0.3
0.4
0.5
0.6
x


Chain(a = 0.1, b = 2) vs Iteration
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
500
1000
1500
2000
iteration

MCMC

Yonglin Zhuo