Bayesian Statistics

Metropolis-Hastings and the general theory of MCMC

Nan Lin
Department of Mathematics
Washington University in St. Louis

A motivating example for generalized linear models: **Song sparrow reproductive success**

- A sample from a population of 52 female song sparrows was studied over the course of a summer and their reproductive activities were recorded.
  - the age and number of new offspring were recorded for each sparrow (Arcese et al, 1992)
- Goal
  - Understand the relationship between age and reproductive success
  - Make population forecasts for this group of birds

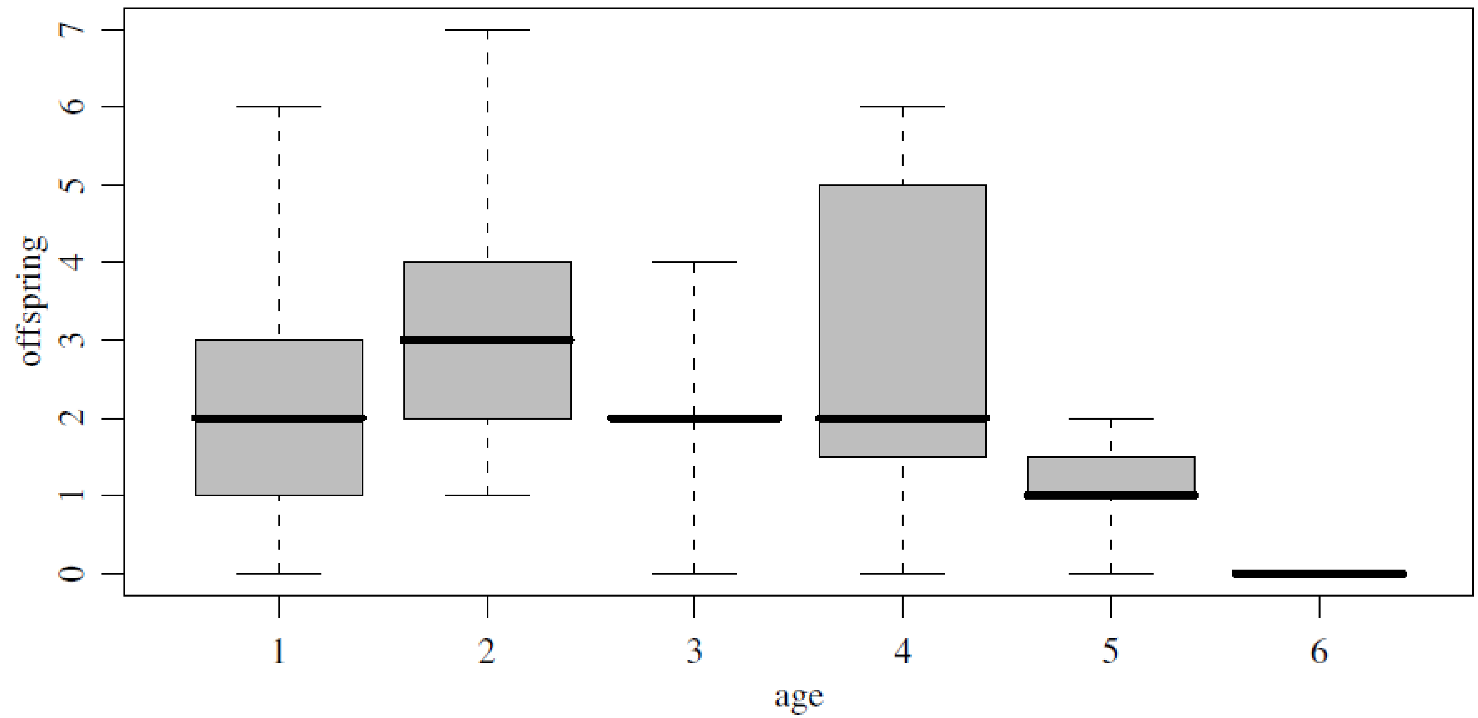

Fig. 10.1. Number of offspring versus age.

Two-year-old birds in this population had the highest median reproductive success, with the number of offspring declining beyond two years of age.

<u>Biological interpretation</u>: One-year-old birds are in their first mating season and are relatively inexperienced compared to two-year-old birds. As birds age beyond two years they experience a general decline in health and activity.

Poisson regression

- Let y=number of offspring, x=age
- $y|x \sim Poisson(\theta_x), \theta_x = E(y|x)$
- Separate estimate of θ_x for each age group may be imprecise when the number of birds of the same age is small
- We may assume $\theta_x = \beta_1 + \beta_2 x + \beta_3 x^2$
  - But this can produce negative values of θ_x
- A better model: $\log\theta_x = \beta_1 + \beta_2 x + \beta_3 x^2$
  - i.e. $\theta_x = \exp(\beta_1 + \beta_2 x + \beta_3 x^2)$
- Poisson regression: $y|x \sim Poisson(\exp(\boldsymbol{\beta}^T \boldsymbol{x}))$
  - Link function: logarithm

Generalized linear model

- Logistic regression
  - $y|x \sim Bernoulli(\theta_x), \theta_x = P(y = 1|x) = E(y|x)$
  - $\text{logit}(\theta_x) = \log \frac{\theta_x}{1-\theta_x} = \boldsymbol{\beta}^T \boldsymbol{x},$
  - i.e. $y|x \sim Bernoulli\left(\frac{\exp(\boldsymbol{\beta}^T \boldsymbol{x})}{1+\exp(\boldsymbol{\beta}^T \boldsymbol{x})}\right)$
  - Link function: logit
- In general, $g\big(E(y|x)\big) = \boldsymbol{\beta}^T \boldsymbol{x}$
  - $g()$: link function
- No conjugate prior except in linear regression, i.e. identity link function → we need a more general MCMC algorithm than Gibbs sampling

The Metropolis algorithm

- When using simulation to describe the posterior distribution $p(\theta|y)$, the general goal is to construct a large collection of θ-values, $\{\theta^{(1)}, \dots, \theta^{(S)}\}$, whose empirical distribution approximates $p(\theta|y)$.
  - That is, we need

$$\frac{\#\{\theta^{(s)}\text{'s in the collection} = \theta_a\}}{\#\{\theta^{(s)}\text{'s in the collection} = \theta_b\}} \approx \frac{p(\theta_a|y)}{p(\theta_b|y)}.$$

- Now, let's think about how to construct $\{\theta^{(1)}, \dots, \theta^{(S)}\}$
  - Suppose we have $\{\theta^{(1)}, \dots, \theta^{(s)}\}$, and we wan to add a new value $\theta^{(s+1)}$
  - We may choose a candidate value θ^* near $\theta^{(s)}$
  - Question: should we include θ^* into the collection?

The Metropolis algorithm

- If $p(\theta^*|y) > p(\theta^{(s)}|y)$, we want more θ^* than $\theta^{(s)}$ in the collection, and since $\theta^{(s)}$ is already in, θ^* should be included as well
- If $p(\theta^*|y) < p(\theta^{(s)}|y)$, we do not necessarily include θ^*
- Decision depends on the ratio $r = \dfrac{p(\theta^*|y)}{p(\theta^{(s)}|y)}$
- And luckily, we can always calculate this ratio easily

$$r = \frac{p(\theta^*|y)}{p(\theta^{(s)}|y)} = \frac{p(y|\theta^*)p(\theta^*)}{p(y)} \frac{p(y)}{p(y|\theta^{(s)})p(\theta^{(s)})} = \frac{p(y|\theta^*)p(\theta^*)}{p(y|\theta^{(s)})p(\theta^{(s)})}$$

The Metropolis algorithm

- How to choose θ^*?
  - sample θ^* from a proposal distribution $J(\theta^*|\theta^{(s)})$
  - The Metropolis algorithm uses a symmetric proposal distribution, i.e. $J(\theta_a|\theta_b) = J(\theta_b|\theta_a)$
  - e.g., $\theta^*|\theta^{(s)} \sim U\left(\theta^{(s)} - \delta, \theta^{(s)} + \delta\right)$ or $\theta^*|\theta^{(s)} \sim N(\theta^{(s)}, \delta^2)$

The Metropolis algorithm

1. Sample $\theta^* \sim J(\theta|\theta^{(s)})$;
2. Compute the acceptance ratio

$$r = \frac{p(\theta^*|y)}{p(\theta^{(s)}|y)} = \frac{p(y|\theta^*)p(\theta^*)}{p(y|\theta^{(s)})p(\theta^{(s)})}.$$

3. Let

$$\theta^{(s+1)} = \begin{cases} \theta^* & \text{with probability } \min(r,1) \\ \theta^{(s)} & \text{with probability } 1-\min(r,1). \end{cases}$$

Step 3 can be accomplished by sampling $u \sim \text{uniform}(0,1)$ and setting $\theta^{(s+1)} = \theta^*$ if $u < r$ and setting $\theta^{(s+1)} = \theta^{(s)}$ otherwise.

<u>Output</u>: a Markov chain $\{\theta^{(1)}, \dots, \theta^{(S)}\}$

Example: normal model with known variance

- Data: $y_1, \ldots, y_n | \theta \sim N(\theta, \sigma^2)$
- Prior: $\theta \sim N(\mu, \tau^2)$
- Suppose that $\sigma^2 = 1, \tau^2 = 10, \mu = 5, n = 5, y = (9.37, 10.18, 9.16, 11.60, 10.33)$.
- From what we learned before, we can show that $\theta | y \sim N(10.03, 0.44)$
- Let's sample from the posterior distribution using the Metropolis algorithm and compare with this exact solution

Example: normal model with known variance

- The acceptance ratio is

$$r = \frac{p(\theta^*|\boldsymbol{y})}{p(\theta^{(s)}|\boldsymbol{y})} = \left(\frac{\prod_{i=1}^n \mathrm{dnorm}(y_i, \theta^*, \sigma)}{\prod_{i=1}^n \mathrm{dnorm}(y_i, \theta^{(s)}, \sigma)}\right) \times \left(\frac{\mathrm{dnorm}(\theta^*, \mu, \tau)}{\mathrm{dnorm}(\theta^{(s)}, \mu, \tau)}\right)$$

  - Often, the above direct calculation is numerically unstable, and one may compute the logarithm instead.

$$\log r = \sum_{i=1}^n [\log \mathrm{dnorm}(y_i, \theta^*, \sigma) - \log \mathrm{dnorm}(y_i, \theta^{(s)}, \sigma)] +$$
$$\log \mathrm{dnorm}(\theta^*, \mu, \tau) - \log \mathrm{dnorm}(\theta^{(s)}, \mu, \tau).$$

  - On the log scale, the proposal is accepted if $\log U(0,1) < \log r$

R code

- Initial value: $\theta^{(0)} = 0$
- Proposal distribution: $\theta^* | \theta^{(s)} \sim N(\theta^{(s)}, \delta^2 = 2)$

```
s2<-1 ; t2<-10 ; mu<-5
y<-c(9.37, 10.18, 9.16, 11.60, 10.33)
theta<-0 ; delta2<-2 ; S<-10000 ; THETA<-NULL ; set.seed(1)

for(s in 1:S)
{

  theta.star<-rnorm(1,theta,sqrt(delta2))

  log.r<-( sum(dnorm(y,theta.star,sqrt(s2),log=TRUE)) +
               dnorm(theta.star,mu,sqrt(t2),log=TRUE) ) -
          ( sum(dnorm(y,theta,sqrt(s2),log=TRUE)) +
               dnorm(theta,mu,sqrt(t2),log=TRUE) )

  if(log(runif(1))<log.r) { theta<-theta.star }

  THETA<-c(THETA,theta)

}
```

Results

- Simulate 10,000 values

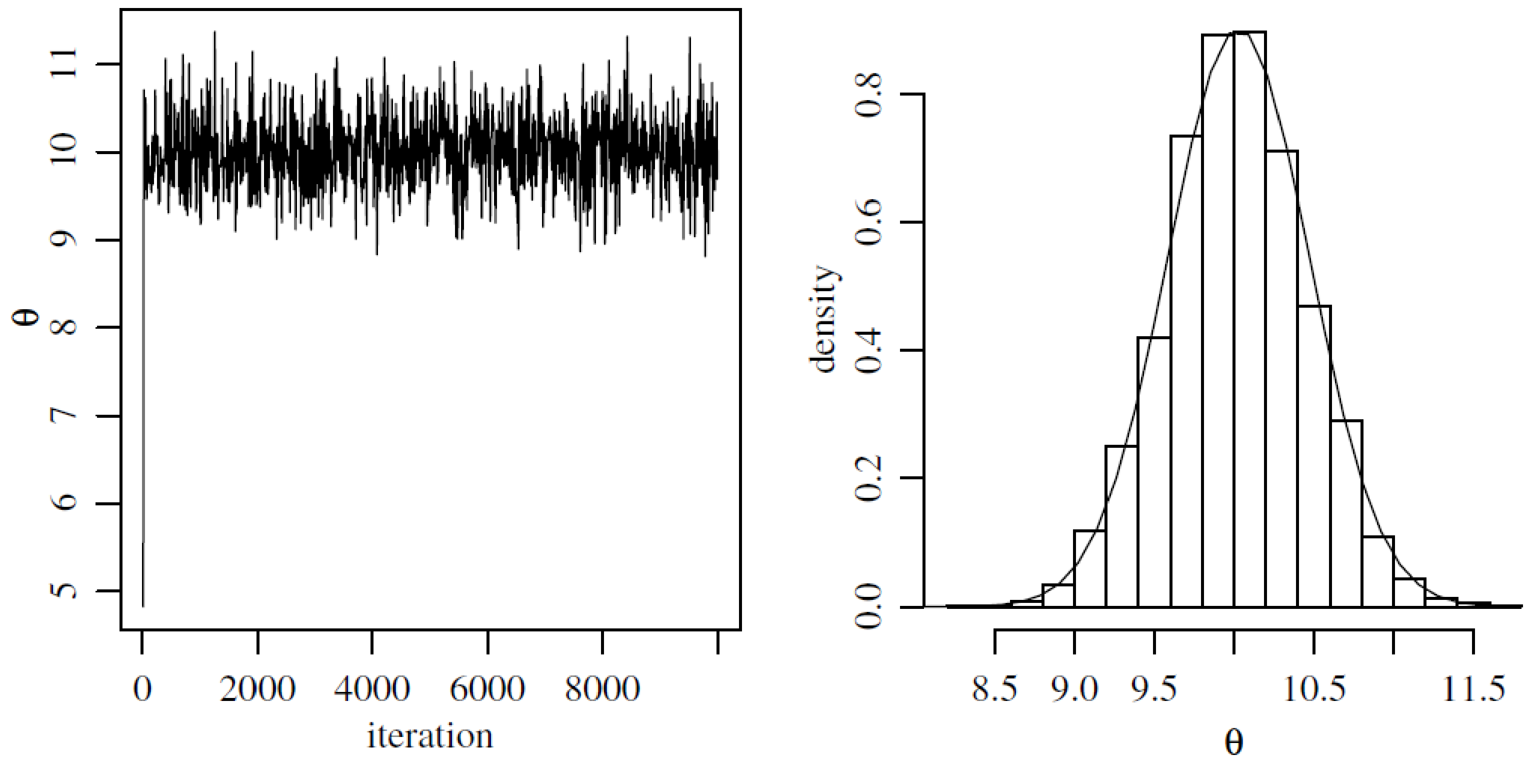

Fig. 10.3. Results from the Metropolis algorithm for the normal model.

General property of the Metropolis algorithm

- Under some mild conditions, the marginal distribution of $\theta^{(s)}$ approximates the posterior distribution $p(\theta|y)$ for large s.
  - For any given value of θ_a,

$$\lim_{S\to\infty} \frac{\#\{\theta\text{'s in the sequence } < \theta_a\}}{S} = p(\theta < \theta_a | y)$$

Tune the proposal distribution

- By choosing a proper proposal variance δ^2, we can decrease the correlation in the Markov chain
  - Faster convergence
  - Better mixing
  - An increase in the effective sample size
- The optimal proposal variance is neither too large nor too small
  - Small δ^2: θ^* is too close to the current value $\theta^{(s)}$. No matter accepted or not, $\theta^{(s+1)}$ is similar to $\theta^{(s)}$. Takes long time to move far away from $\theta^{(s)}$
  - Large δ^2: If $\theta^{(s)}$ is close the posterior mode, θ^* is often very far from $\theta^{(s)}$, and leads to rejection. Then the chain gets 'stuck', because in most iterations, $\theta^{(s+1)} = \theta^{(s)}$.

Different proposal variances

- $\delta^2 = \left\{\frac{1}{32}, \frac{1}{2}, 2, 32, 64\right\}$
- Corresponding autocorrelation: (0.98,0.77,0.69,0.840.86)

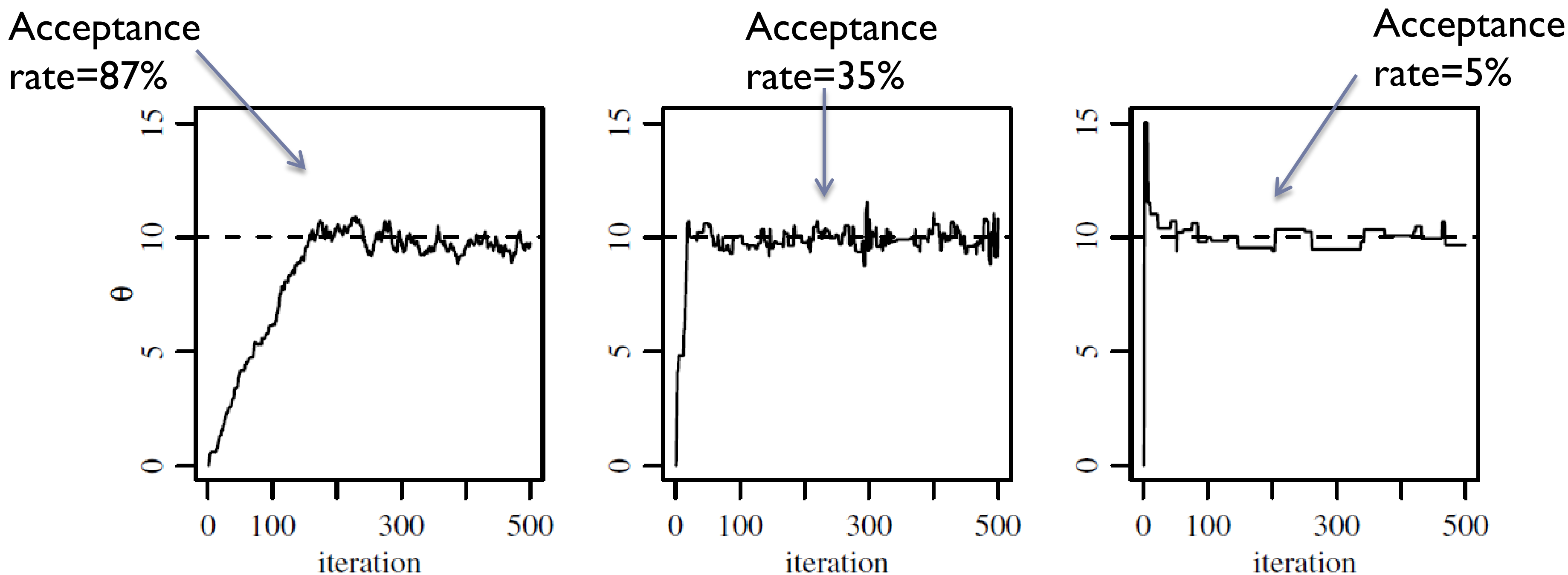

Fig. 10.4. Markov chains under three different proposal distributions. Going from left to right, the values of δ^2 are 1/32, 2 and 64 respectively.

Metropolis algorithm for Poisson regression

- Model: $\log E(y_i|x_i) = \beta_1 + \beta_2 x_i + \beta_3 x_i^2$
- Let $\boldsymbol{x}_i = (1, x_i, x_i^2)$, then $\log E(y_i|x_i) = \boldsymbol{\beta}^T \boldsymbol{x}_i$
- Prior distribution: $\boldsymbol{\beta} \sim N(0{,}100I)$
- Acceptance rate:

$$r = \frac{p(\boldsymbol{\beta}^*|\mathbf{X}, \boldsymbol{y})}{p(\boldsymbol{\beta}^{(s)}|\mathbf{X}, \boldsymbol{y})}$$

$$= \frac{\prod_{i=1}^n \mathrm{dpois}(y_i, \boldsymbol{x}_i^T \boldsymbol{\beta}^*)}{\prod_{i=1}^n \mathrm{dpois}(y_i, \boldsymbol{x}_i^T \boldsymbol{\beta}^{(s)})} \times \frac{\prod_{j=1}^3 \mathrm{dnorm}(\beta_j^*, 0, 10)}{\prod_{j=1}^3 \mathrm{dnorm}(\beta_j^{(s)}, 0, 10)}.$$

- Proposal distribution: multivariate normal
  - Choose the proposal variance similar to posterior variance
  - From linear regression, we know that $\mathrm{var}(\hat{\beta}) = \sigma^2 (X^T X)^{-1}$
  - Construct $\hat{\sigma}^2$ using the sample variance of $\{\log\left(y_1 + \frac{1}{2}\right), \ldots, \log\left(y_n + \frac{1}{2}\right)\}$

R code

```
data(chapter10) ; y<-yX.sparrow[,1] ; X<-yX.sparrow[,-1]
n<-length(y) ; p<-dim(X)[2]

pmn.beta<-rep(0,p)      #prior expectation
psd.beta<-rep(10,p)     #prior var

var.prop<- var(log(y+1/2))*solve( t(X)%*%X )  #proposal var
S<-10000
beta<-rep(0,p) ; acs<-0
BETA<-matrix(0,nrow=S,ncol=p)
set.seed(1)

for(s in 1:S)
{
  beta.p<- t(rmvnorm(1, beta, var.prop ))

  lhr<- sum(dpois(y,exp(X%*%beta.p),log=T)) -
        sum(dpois(y,exp(X%*%beta),log=T)) +
        sum(dnorm(beta.p,pmn.beta,psd.beta,log=T)) -
        sum(dnorm(beta,pmn.beta,psd.beta,log=T))

  if( log(runif(1))< lhr ) { beta<-beta.p ; acs<-acs+1 }

  BETA[s,]<-beta
}
```

Result

- Acceptance rate: 43%

After thinning (every 10^{th} observation)

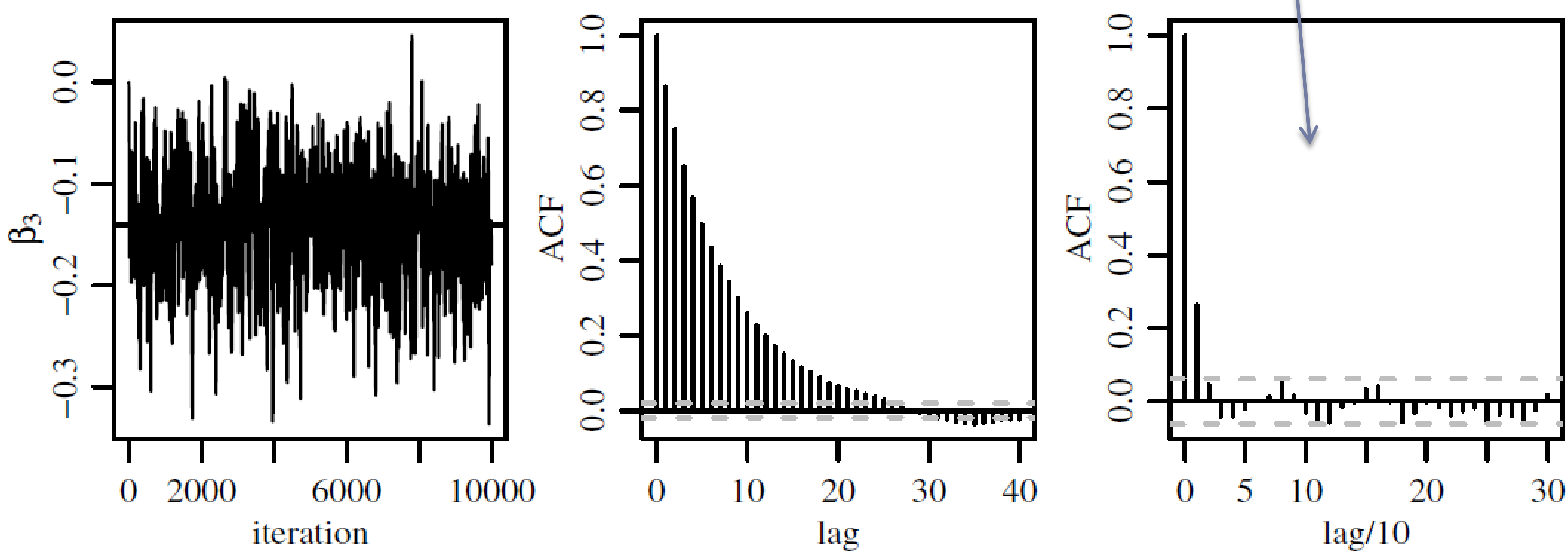

Fig. 10.5. Plot of the Markov chain in β_3 along with autocorrelation functions.

Result

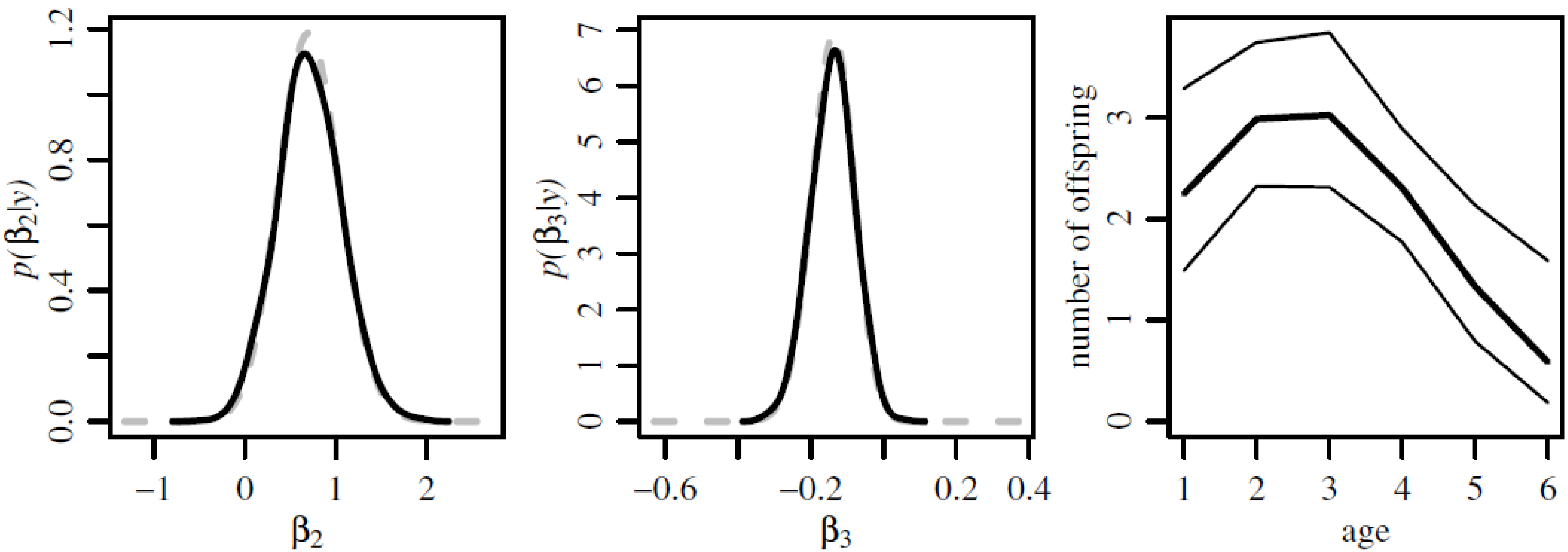

Fig. 10.6. The first two panels give the MCMC approximations to the posterior marginal distributions of β_2 and β_3 in black, with the grid-based approximations in gray. The third panel gives 2.5%, 50% and 97.5% posterior quantiles of $\exp(\boldsymbol{\beta}^T \boldsymbol{x})$.

The Metropolis-Hastings algorithm

- Consider to sample from a bivariate distribution $p_0(u, v)$
- Gibbs sampler

  1. update U: sample $u^{(s+1)} \sim p_0(u|v^{(s)})$;
  2. update V: sample $v^{(s+1)} \sim p_0(v|u^{(s+1)})$.

  1. Proposal: full conditional distribution
  2. Always accept

- The Metropolis algorithm

1. update U:
   a) sample $u^* \sim J_u(u|u^{(s)})$;
   b) compute $r = p_0(u^*, v^{(s)})/p_0(u^{(s)}, v^{(s)})$;
   c) set $u^{(s+1)}$ to u^* or $u^{(s)}$ with probability $\min(1, r)$ and $\max(0, 1-r)$.
2. update V:
   a) sample $v^* \sim J_v(v|v^{(s)})$;
   b) compute $r = p_0(u^{(s+1)}, v^*)/p_0(u^{(s+1)}, v^{(s)})$;
   c) set $v^{(s+1)}$ to v^* or $v^{(s)}$ with probability $\min(1, r)$ and $\max(0, 1-r)$.

Symmetric proposal distribution

The Metropolis-Hastings (M-H) algorithm

- Arbitrary proposal

1. update U:
   a) sample $u^* \sim J_u(u|u^{(s)}, v^{(s)})$;
   b) compute the acceptance ratio

$$r = \frac{p_0(u^*, v^{(s)})}{p_0(u^{(s)}, v^{(s)})} \times \frac{J_u(u^{(s)}|u^*, v^{(s)})}{J_u(u^*|u^{(s)}, v^{(s)})};$$

   c) set $u^{(s+1)}$ to u^* or $u^{(s)}$ with probability $\min(1, r)$ and $\max(0, 1-r)$.
2. update V:
   a) sample $v^* \sim J_v(v|u^{(s+1)}, v^{(s)})$;
   b) compute the acceptance ratio

$$r = \frac{p_0(u^{(s+1)}, v^*)}{p_0(u^{(s+1)}, v^{(s)})} \times \frac{J_v(v^{(s)}|u^{(s+1)}, v^*)}{J_v(v^*|u^{(s+1)}, v^{(s)})};$$

   c) set $v^{(s+1)}$ to v^* or $v^{(s)}$ with probability $\min(1, r)$ and $\max(0, 1-r)$.

Gibbs sampler as the M-H algorithm

- If the proposal distribution is $J_u(u^*|u^{(s)}, v^{(s)}) = p_0(u^*|v^{(s)})$ in the M-H algorithm, the acceptance ratio is

$$
\begin{aligned}
r &= \frac{p_0(u^*, v^{(s)})}{p_0(u^{(s)}, v^{(s)})} \times \frac{J_u(u^{(s)}|u^*, v^{(s)})}{J_u(u^*|u^{(s)}, v^{(s)})} \\
&= \frac{p_0(u^*, v^{(s)})}{p_0(u^{(s)}, v^{(s)})} \frac{p_0(u^{(s)}|v^{(s)})}{p_0(u^*|v^{(s)})} \\
&= \frac{p_0(u^*|v^{(s)})p_0(v^{(s)})}{p_0(u^{(s)}|v^{(s)})p_0(v^{(s)})} \frac{p_0(u^{(s)}|v^{(s)})}{p_0(u^*|v^{(s)})} \\
&= \frac{p_0(v^{(s)})}{p_0(v^{(s)})} = 1,
\end{aligned}
$$

A more general form of the M-H algorithm

1. Generate x^* from $J_s(x^*|x^{(s)})$;
2. Compute the acceptance ratio

$$r = \frac{p_0(x^*)}{p_0(x^{(s)})} \times \frac{J_s(x^{(s)}|x^*)}{J_s(x^*|x^{(s)})};$$

3. Sample $u \sim$uniform$(0,1)$. If $u < r$ set $x^{(s+1)} = x^*$, else set $x^{(s+1)} = x^{(s)}$.

- the proposal distribution may also depend on the iteration number s.
  - For example, in the previous example, J_s can be either J_u or J_v

Requirement for the proposal distribution

- The proposal distribution does not depend on values in the sequence previous to $x^{(s)}$ → Markov property
- Regardless where the chain started, every value x with $p_0(x) > 0$ will eventually be proposed → irreducible
- Aperiodic: A Markov chain lacking any periodic states is called aperiodic
    - A value x is periodic with period $k > 1$ in a Markov chain if it can only be visited every kth iteration.
- Recurrent: A value x is said to be recurrent if, when we continue to run the Markov chain from x, we are guaranteed to eventually return to x. And we want all of the possible values of x to be recurrent in our Markov chain

Ergodic theorem

Theorem 2 *(Ergodic Theorem) If $\{x^{(1)}, x^{(2)}, \ldots\}$ is an irreducible, aperiodic and recurrent Markov chain, then there is a unique probability distribution π such that as $s \to \infty$,*

- $\Pr(x^{(s)} \in A) \to \pi(A)$ *for any set A;*
- $\frac{1}{S} \sum g(x^{(s)}) \to \int g(x)\pi(x)\, dx.$

The distribution π is called the *stationary distribution* of the Markov chain. It is called the stationary distribution because it has the following property:

If $x^{(s)} \sim \pi$,
and $x^{(s+1)}$ is generated from the Markov chain starting at $x^{(s)}$,
then $\Pr(x^{(s+1)} \in A) = \pi(A)$.

$\downarrow$

Once you are sampling from the stationary distribution, you are always sampling from the stationary distribution.

Proof that $p_0(x)$ is the stationary distribution

- Assume X is discrete
- Suppose $x^{(s)}$ is sampled from the target distribution p_0, and then $x^{(s+1)}$ is generated based on $x^{(s)}$ using the Metropolis-Hastings algorithm. To show that p_0 is the stationary distribution we need to show that $\Pr(x^{(s+1)} = x) = p_0(x)$.

Let x_a and x_b be any two values of X such that $p_0(x_a)J_s(x_b|x_a) \geq p_0(x_b)J_s(x_a|x_b)$. Then under the Metropolis-Hastings algorithm the probability that $x^{(s)} = x_a$ and $x^{(s+1)} = x_b$ is equal to the probability of

1. sampling $x^{(s)} = x_a$ from p_0;
2. proposing $x^* = x_b$ from $J_s(x^*|x^{(s)})$;
3. accepting $x^{(s+1)} = x_b$.

The probability of these three things occurring is their product:

$$\Pr(x^{(s)} = x_a, x^{(s+1)} = x_b) = p_0(x_a) \times J_s(x_b|x_a) \times \frac{p_0(x_b)}{p_0(x_a)} \frac{J_s(x_a|x_b)}{J_s(x_b|x_a)}$$
$$= p_0(x_b) J_s(x_a|x_b)\,.$$

On the other hand, the probability that $x^{(s)} = x_b$ and $x^{(s+1)} = x_a$ is the probability that x_b is sampled from p_0, that x_a is proposed from $J_s(x^*|x^{(s)})$ and that x_a is accepted as $x^{(s+1)}$. But in this case the acceptance probability is one because we assumed $p_0(x_a)J_s(x_b|x_a) \geq p_0(x_b)J_s(x_a|x_b)$. This means that $\Pr(x^{(s)} = x_b, x^{(s+1)} = x_a) = p_0(x_b)J_s(x_a|x_b)$.

The above two calculations have shown that the probability of observing $x^{(s)}$ and $x^{(s+1)}$ to be x_a and x_b, respectively, is the same as observing them to be x_b and x_a respectively, for any two values x_a and x_b. The final step of the proof is to use this fact to derive the marginal probability $\Pr(x^{(s+1)} = x)$:

$$
\begin{aligned}
\Pr(x^{(s+1)} = x) &= \sum_{x_a} \Pr(x^{(s+1)} = x, x^{(s)} = x_a) \\
&= \sum_{x_a} \Pr(x^{(s+1)} = x_a, x^{(s)} = x) \\
&= \Pr(x^{(s)} = x)
\end{aligned}
$$

This completes the proof that $\Pr(x^{(s+1)} = x) = p_0(x)$ if $\Pr(x^{(s)} = x) = p_0(x)$.

Combining the Metropolis and Gibbs algorithms

- In complex models it is often the case that conditional distributions are available for some parameters but not for others. In these situations we can combine Gibbs and Metropolis-type proposal distributions to generate a Markov chain to approximate the joint posterior distribution of all of the parameters.

Example: Historical CO2 and temperature data

- Ice cores from East Antarctica allowed scientists to deduce historical atmospheric conditions

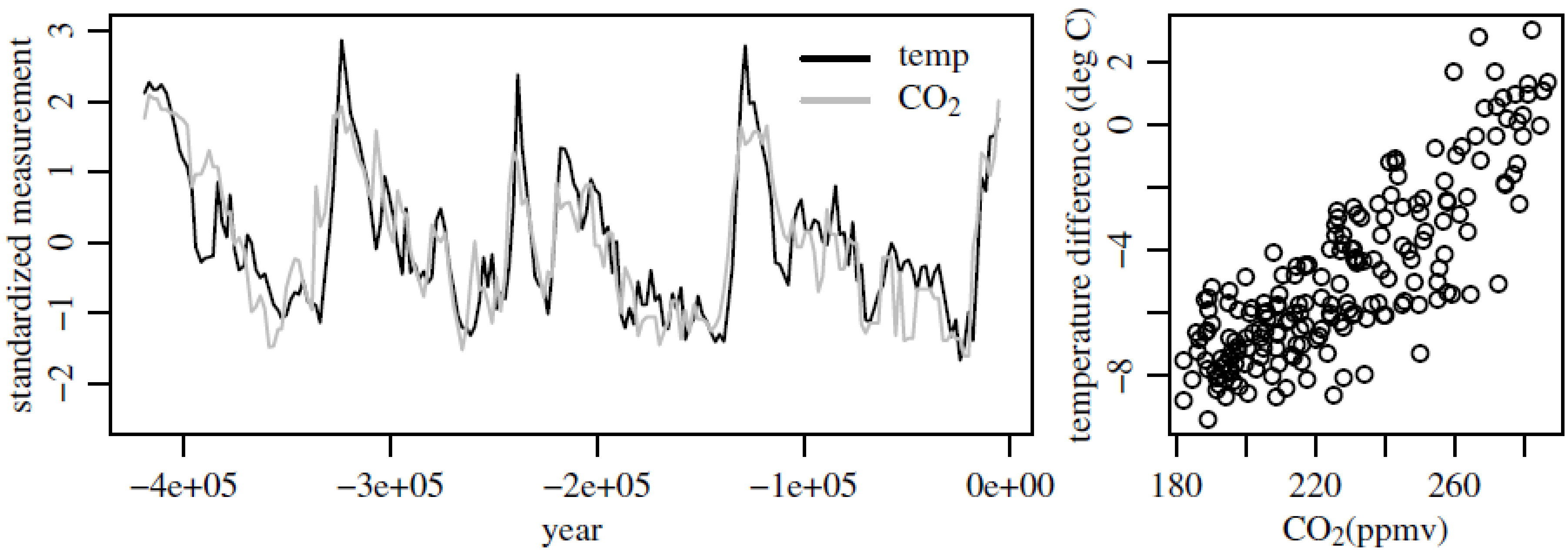

Fig. 10.7. Temperature and carbon dioxide data.

Example: Historical CO2 and temperature data

- Data: 200 values of temperature measured at roughly equal time intervals, with time between consecutive measurements being approximately 2,000 years.
- The plot indicates that the temporal history of temperature and CO2 follow very similar patterns
- Linear regression for temperature (Y) as a function of CO2 (x).

$$\hat{\mathrm{E}}[Y|x] = -23.02 + 0.08x$$

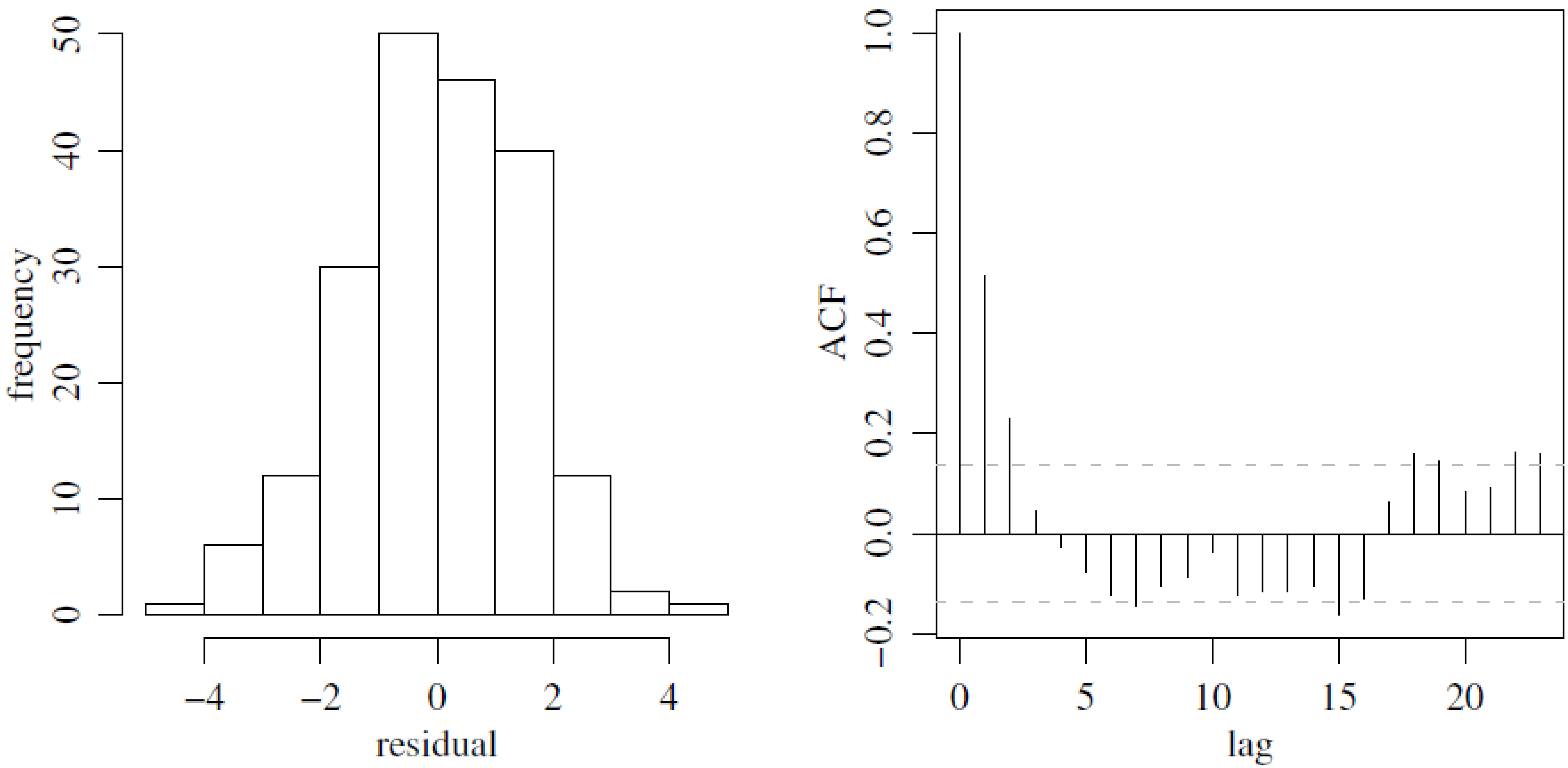

Fig. 10.8. Residual analysis for the least squares estimation.

A regression model with correlated errors

- Ordinary linear regression

$$\boldsymbol{Y} = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix} \sim \text{multivariate normal}(\mathbf{X}\boldsymbol{\beta}, \sigma^2\mathbf{I}).$$

- Introducing AR(1) structure

$$\Sigma = \sigma^2\mathbf{C}_\rho = \sigma^2 \begin{pmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{n-1} \\ \rho & 1 & \rho & \cdots & \rho^{n-2} \\ \rho^2 & \rho & 1 & & \\ \vdots & \vdots & & \ddots & \\ \rho^{n-1} & \rho^{n-2} & & & 1 \end{pmatrix}$$

MCMC

- Prior: $\beta \sim N(\beta_0, \Sigma_0)$ and $\sigma^{-2} \sim gamma(\frac{\nu_0}{2}, \frac{\nu_0 \sigma_0^2}{2})$
- If ρ is known, a Gibbs sampler is available

$$\{\boldsymbol{\beta} | \mathbf{X}, \boldsymbol{y}, \sigma^2, \rho\} \sim \text{multivariate normal}(\boldsymbol{\beta}_n, \Sigma_n) \text{ , where}$$
$$\Sigma_n = (\mathbf{X}^T \mathbf{C}_\rho^{-1} \mathbf{X} / \sigma^2 + \Sigma_0^{-1})^{-1}$$
$$\boldsymbol{\beta}_n = \Sigma_n (\mathbf{X}^T \mathbf{C}_\rho^{-1} \boldsymbol{y} / \sigma^2 + \Sigma_0^{-1} \boldsymbol{\beta}_0) \text{ , and}$$

$$\{\sigma^2 | \mathbf{X}, \boldsymbol{y}, \boldsymbol{\beta}, \rho\} \sim \text{inverse-gamma}([\nu_0 + n]/2, [\nu_0 \sigma_0^2 + \text{SSR}_\rho]/2) \text{ , where}$$
$$\text{SSR}_\rho = (\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})^T \mathbf{C}_\rho^{-1} (\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}).$$

- But ρ is unknown and the full conditional distribution of ρ is not so simple

MCMC

1. Update $\boldsymbol{\beta}$: Sample $\boldsymbol{\beta}^{(s+1)} \sim$ multivariate normal$(\boldsymbol{\beta}_n, \Sigma_n)$, where $\boldsymbol{\beta}_n$ and Σ_n depend on $\sigma^{2(s)}$ and $\rho^{(s)}$.
2. Update σ^2: Sample $\sigma^{2(s+1)} \sim$ inverse-gamma$([\nu_0+n]/2, [\nu_0\sigma_0^2+\mathrm{SSR}_\rho]/2)$, where SSR_ρ depends on $\boldsymbol{\beta}^{(s+1)}$ and $\rho^{(s)}$.
3. Update ρ:
   a) Propose $\rho^* \sim$ uniform$(\rho^{(s)} - \delta, \rho^{(s)} + \delta)$. If $\rho^* < 0$ then reassign it to be $|\rho^*|$. If $\rho^* > 1$ reassign it to be $2 - \rho^*$.
   b) Compute the acceptance ratio

$$r = \frac{p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{\beta}^{(s+1)}, \sigma^{2(s+1)}, \rho^*)p(\rho^*)}{p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{\beta}^{(s+1)}, \sigma^{2(s+1)}, \rho^{(s)})p(\rho^{(s)})}$$

   and sample $u \sim$ uniform(0,1). If $u < r$ set $\rho^{(s+1)} = \rho^*$, otherwise set $\rho^{(s+1)} = \rho^{(s)}$.

This proposal distribution is called reflecting random walk, which ensures $0 < \rho < 1$

- It is symmetric

Example: Historical CO2 and temperature data

- “diffuse” prior:
  - $\beta \sim N(\beta_0, \Sigma_0)$ and $\sigma^{-2} \sim gamma(\frac{\nu_0}{2}, \frac{\nu_0 \sigma_0^2}{2})$ with $\beta_0 = 0, \Sigma_0 = diag(1000), \nu_0 = 1, \sigma_0^2 = 1$
  - $\rho \sim U(0,1)$

High autocorrelation

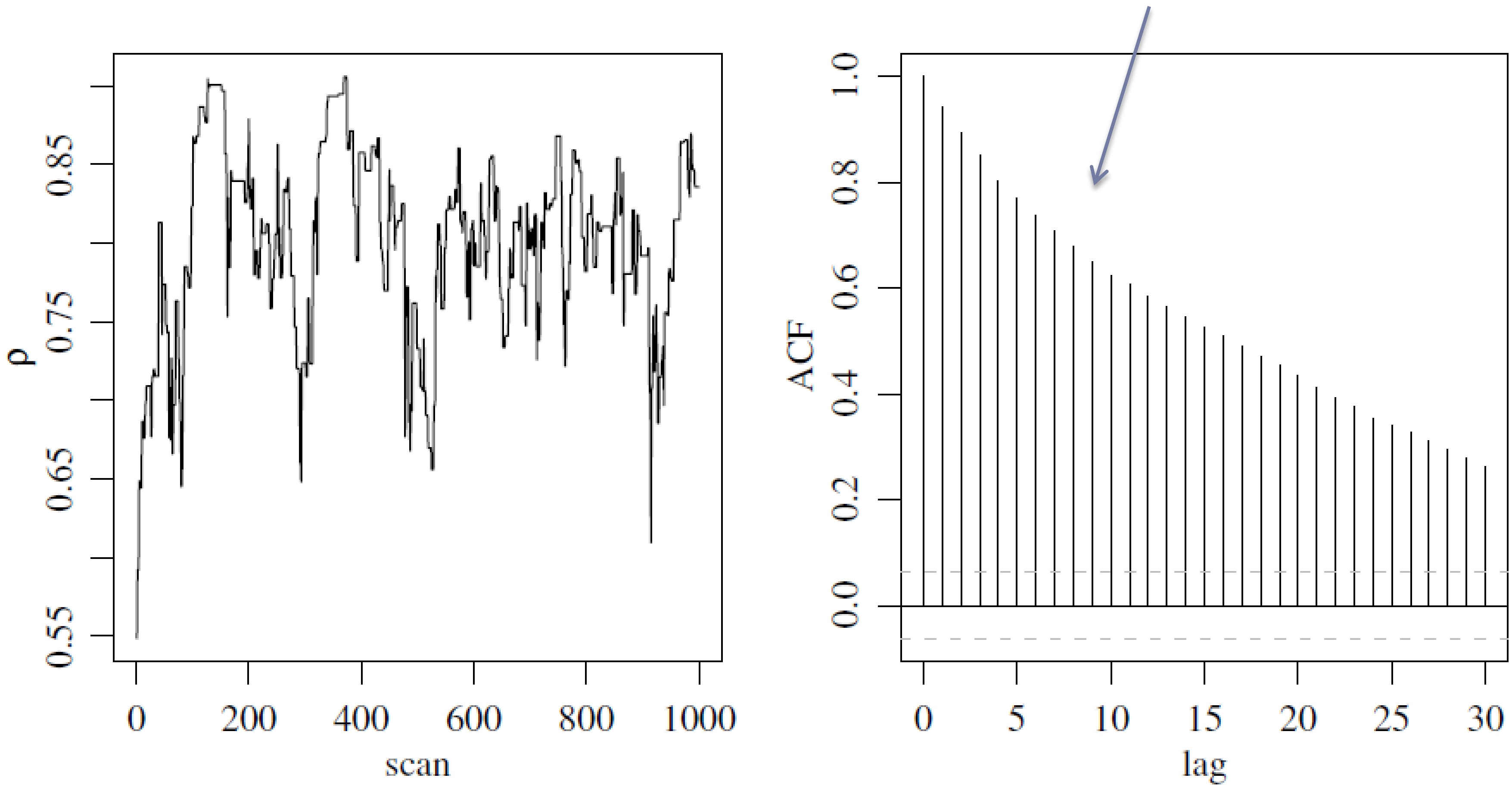

Fig. 10.9. The first 1,000 values of ρ generated from the Markov chain.

After thinning (every 25th value)

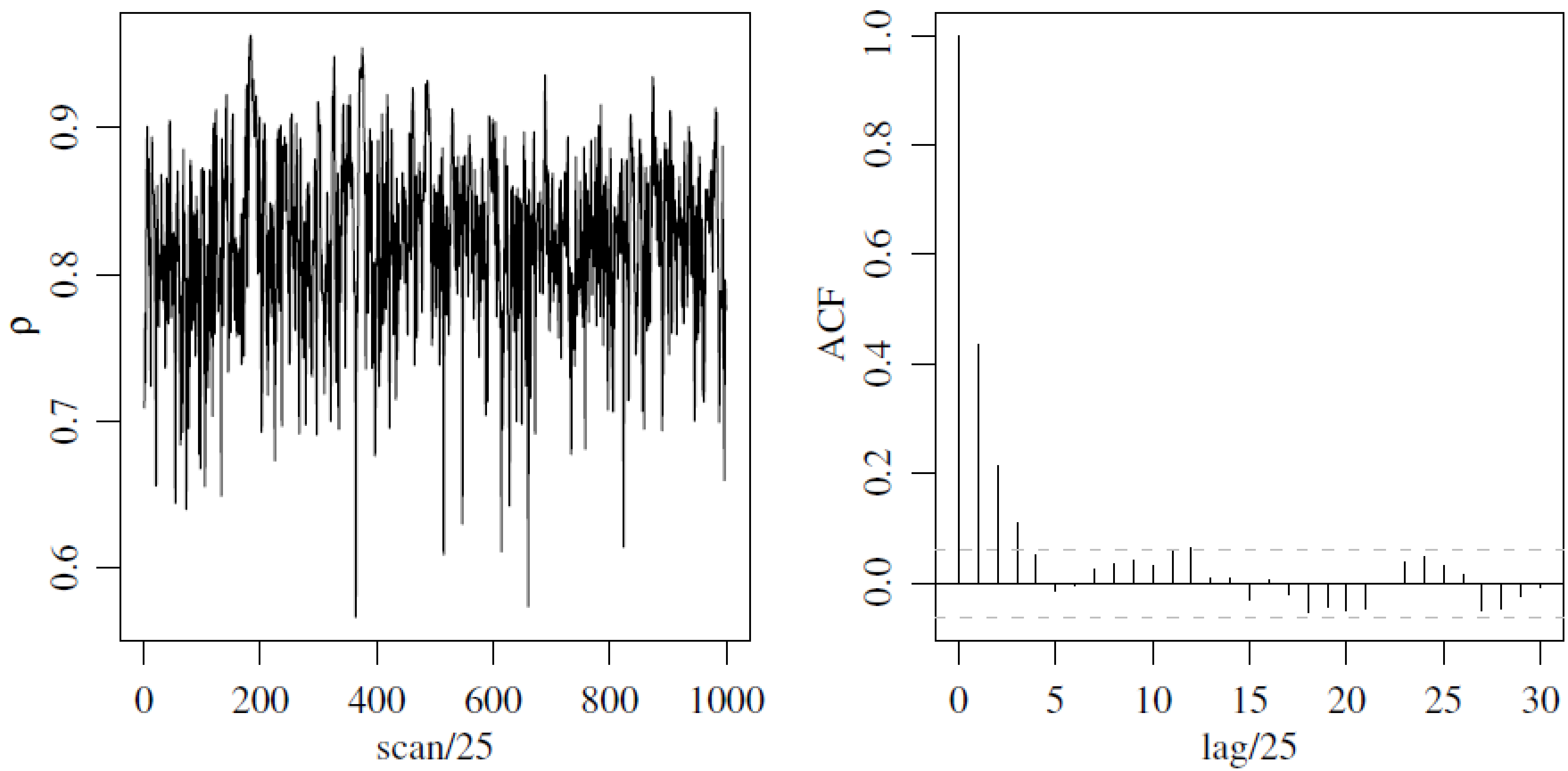

Fig. 10.10. Every 25th value of ρ from a Markov chain of length 25,000.

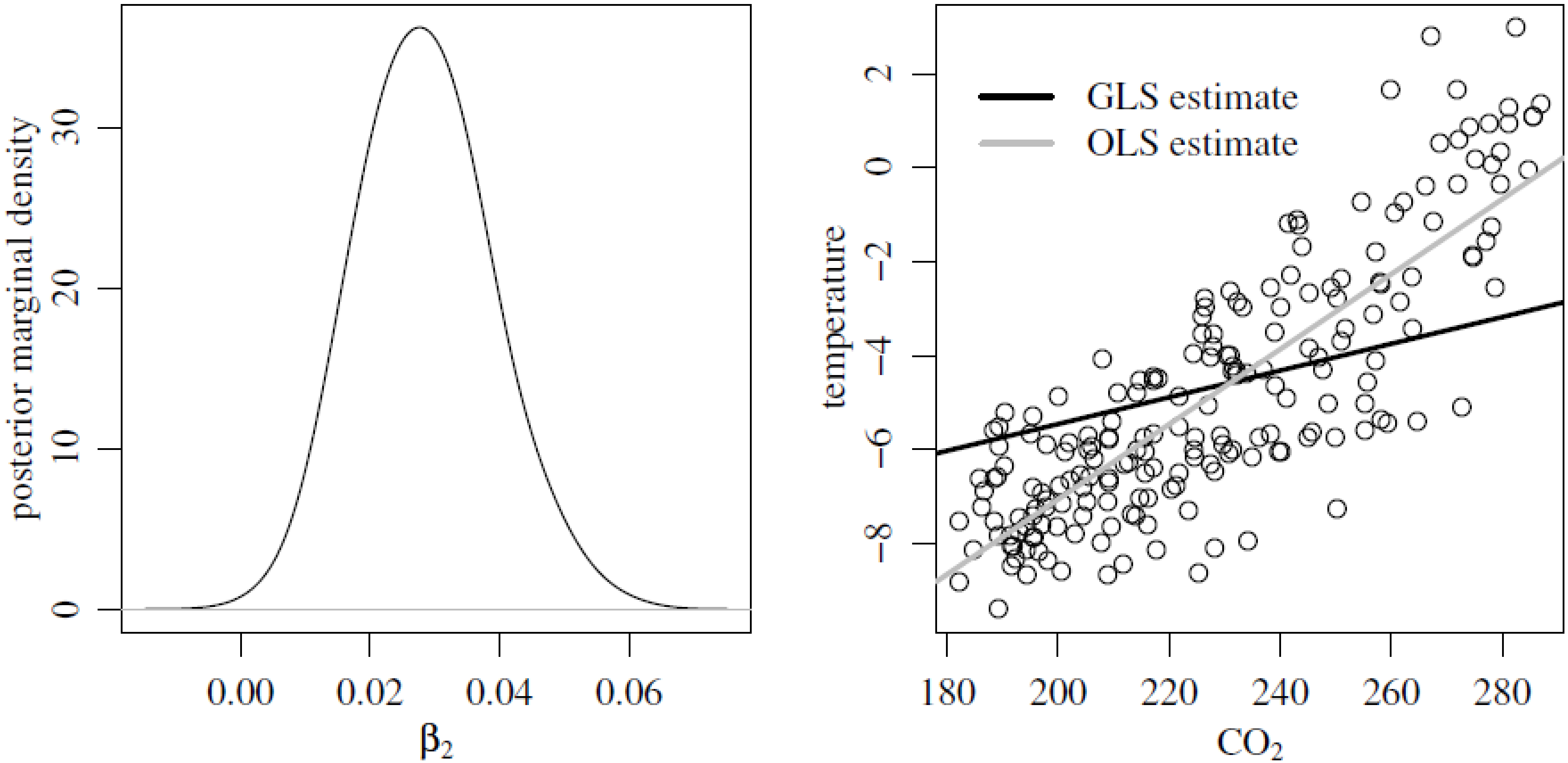

Fig. 10.11. Posterior distribution of the slope parameter β_2, along with the posterior mean regression line.

Posterior mean of the slope β_2 is 0.028 with 95% credible interval (0.01,0.05)

A simple OLS estimate would give $\hat{\beta}_2 = 0.08$