Markov Chain Monte Carlo Methods

Gibbs sampling is but one example of Markov chain Monte Carlo (MCMC) methods.

Why do we need methods besides ordinary Monte Carlo and Gibbs sampling?

- When using a nonconjugate prior, we may get a “weird” posterior from which we don’t know how to sample.

- Even if we use a conjugate prior, we may not know how to sample from all the full conditionals.

- Not knowing $m(\boldsymbol{y})$ makes some approaches to sampling from the posterior infeasible.

We'll discuss the *Metropolis algorithm* and the more general *Metropolis-Hastings algorithm*.

First we define the Metropolis algorithm and then give some intuition for it.

Suppose that $\boldsymbol{\theta}$ is a random variable taking on values in Θ. The density of $\boldsymbol{\theta}$ is f.

We want to be able to draw samples from f but don't know how to.

We use a so-called *proposal distribution, J*.

For every $\boldsymbol{y} \in \Theta$, $J(\,\cdot\,|\boldsymbol{y})$ is a conditional density over Θ. J is assumed to be *symmetric* in the sense that $J(\boldsymbol{x}|\boldsymbol{y}) = J(\boldsymbol{y}|\boldsymbol{x})$ for every $\boldsymbol{x}$ and $\boldsymbol{y}$ in Θ.

Given an initial value $\boldsymbol{\theta}^{(0)}$, and assuming we have arrived at the value $\boldsymbol{\theta}^{(s)}$, the Metropolis algorithm for generating $\boldsymbol{\theta}^{(s+1)}$ from f is as follows:

1. Generate $\boldsymbol{\theta}^*$ from $J(\,\cdot\,|\boldsymbol{\theta}^{(s)})$.

2. Compute the *acceptance ratio*
$$r = \frac{f(\boldsymbol{\theta}^*)}{f(\boldsymbol{\theta}^{(s)})}.$$

3. Generate a value u from the $U(0,1)$ distribution. If $u < r$, set $\boldsymbol{\theta}^{(s+1)} = \boldsymbol{\theta}^*$. Otherwise set $\boldsymbol{\theta}^{(s+1)} = \boldsymbol{\theta}^{(s)}$.

An important aspect of this algorithm is that *we only need to know f up to a constant of proportionality.*

The algorithm only depends on f through the ratio $f(\boldsymbol{\theta}^*)/f(\boldsymbol{\theta}^{(s)})$, which doesn't depend on a constant multiplier.

For example, if we apply the algorithm to generate observations from a posterior, we would have to compute

$$\begin{aligned}\frac{p(\boldsymbol{\theta}^*|\boldsymbol{y})}{p(\boldsymbol{\theta}^{(s)}|\boldsymbol{y})} &= \frac{p(\boldsymbol{y}|\boldsymbol{\theta}^*)p(\boldsymbol{\theta}^*)/m(\boldsymbol{y})}{p(\boldsymbol{y}|\boldsymbol{\theta}^{(s)})p(\boldsymbol{\theta}^{(s)})/m(\boldsymbol{y})} \\ &= \frac{p(\boldsymbol{y}|\boldsymbol{\theta}^*)p(\boldsymbol{\theta}^*)}{p(\boldsymbol{y}|\boldsymbol{\theta}^{(s)})p(\boldsymbol{\theta}^{(s)})},\end{aligned}$$

which doesn't require us to know $m(\boldsymbol{y})$.

An amazing fact about the Metropolis algorithm is that it "works" for virtually any proposal distribution J.

What does "works" mean?

The Metropolis algorithm works in the sense that, *once it has run long enough (i.e., s is big enough), the density of $\boldsymbol{\theta}^{(s+1)}$ is essentially equal to f.*

A sufficient condition for this result is that the support of $J(\,\cdot\,|\boldsymbol{\theta})$ is Θ for each $\boldsymbol{\theta} \in \Theta$.

Still, there are good and bad proposal distributions.

A bad proposal distribution is one for which the generated values display poor mixing. We encountered this phenomenon in Gibbs sampling.

With the Metropolis algorithm, we have a phenomenon that cannot happen in Gibbs sampling: *the generated values can stay stuck at the same value for many consecutive iterations.*

So, *a good proposal distribution is one producing output that doesn't get stuck a lot and mixes well.*

Intuition behind Metropolis algorithm

Suppose we've observed $\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(s)}$. We want to know if some value $\boldsymbol{\theta}^*$ should be included in the sample as well.

Suppose that $f(\boldsymbol{\theta}^*) > f(\boldsymbol{\theta}^{(s)})$. Then since $\boldsymbol{\theta}^*$ is more probable than $\boldsymbol{\theta}^{(s)}$, it makes sense that we should include $\boldsymbol{\theta}^*$ in the sample.

Suppose $f(\boldsymbol{\theta}^*) < f(\boldsymbol{\theta}^{(s)})$. Then the ratio of number of occurrences of $\boldsymbol{\theta}^*$ to number of occurrences of $\boldsymbol{\theta}^{(s)}$ in our sample should be $f(\boldsymbol{\theta}^*)/f(\boldsymbol{\theta}^{(s)})$.

This suggests that we accept $\boldsymbol{\theta}^*$ with probability $f(\boldsymbol{\theta}^*)/f(\boldsymbol{\theta}^{(s)})$.

Example 18 Metropolis algorithm

Suppose we want to generate observations from $f \equiv N(0,1)$. We consider the effect of three different proposal distributions and two different starting values.

The proposals are

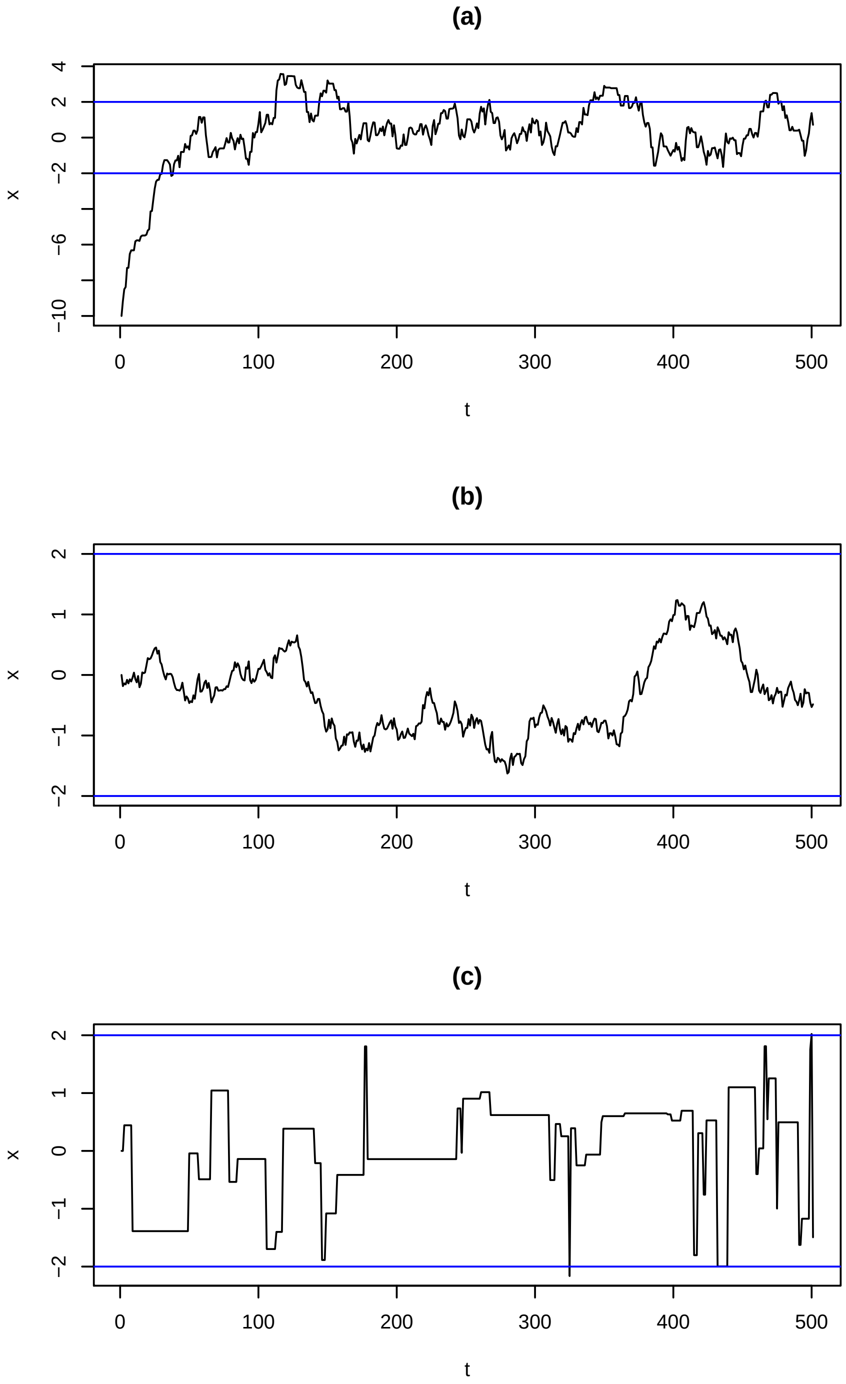
(a) $J(\cdot|\theta) \equiv N(\theta, (0.5)^2)$

(b) $J(\cdot|\theta) \equiv N(\theta, (0.1)^2)$

(c) $J(\cdot|\theta) \equiv N(\theta, 10^2)$

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

Markov chains generated by Metropolis algorithm



Remarks:

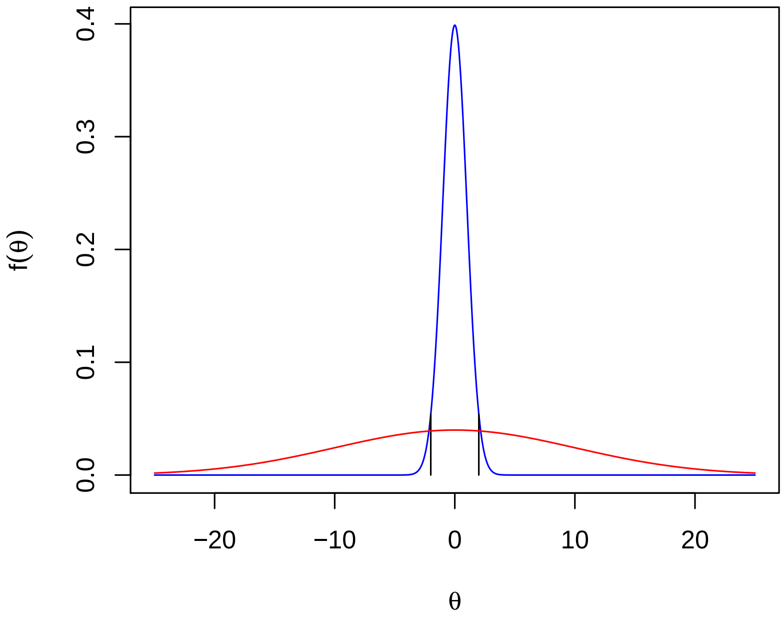
- Note that in case (a), the chain converges fairly rapidly and then mixes fairly quickly thereafter.

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

- 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.

In case (c), suppose $\theta^{(s)}$ is in $(-2, 2)$.

Then there is a high probability that a value θ^* from the proposal distribution will be outside $(-2, 2)$, as illustrated below.



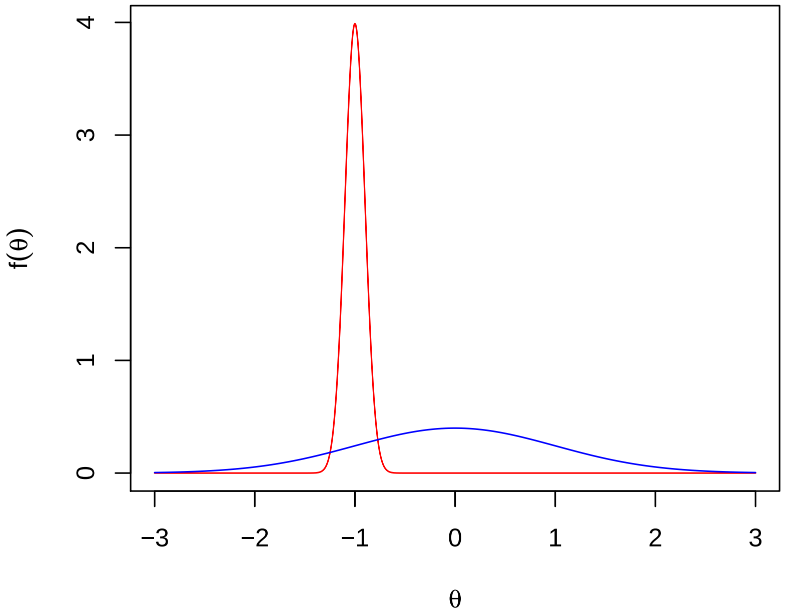
f: —

Proposal when $\theta^{(s)} = 0$: —

When the generated θ^* is outside $(-2, 2)$, the ratio $f(\theta^*)/f(0)$ will be very small. This results in a low probability of the chain moving.

In case (b), regardless of what $\theta^{(s)}$ is, there is a high probability that θ^* will be relatively close to $\theta^{(s)}$.

This means that $f(\theta^*)/f(\theta^{(s)})$ will be close to 1, and hence there is a high probability that the chain moves.



f: —

Proposal when $\theta^{(s)} = -1$: —

However, since $\theta^{(s)}$ and θ^* are always close to each other, the chain does not mix well.

Example 19 Dealing with rounded data

Suppose we have $n = 50$ *rounded* observations from a $N(\mu, \sigma^2)$ distribution.

In other words, the original observations were i.i.d. $N(\mu, \sigma^2)$, but the data we have were rounded to the nearest integer.

Let X_i be the original observation and Y_i its rounded version. *What is the actual likelihood of Y_i?*

$$\begin{aligned} P(Y_i = k|\mu, \sigma) &= P\left(k - \frac{1}{2} < X_i < k + \frac{1}{2}\Big|\mu, \sigma\right) \\ &= \Phi\left(\frac{k + 1/2 - \mu}{\sigma}\right) \\ &\quad -\Phi\left(\frac{k - 1/2 - \mu}{\sigma}\right), \end{aligned}$$

where Φ is the standard normal cdf.

A priori, let's suppose that $\mu|\sigma^2 \sim N(0, \sigma^2)$ and σ^2 is inverse-gamma$(1, 1)$.

This is a conjugate prior for the original experiment, *but not for the rounded data.*

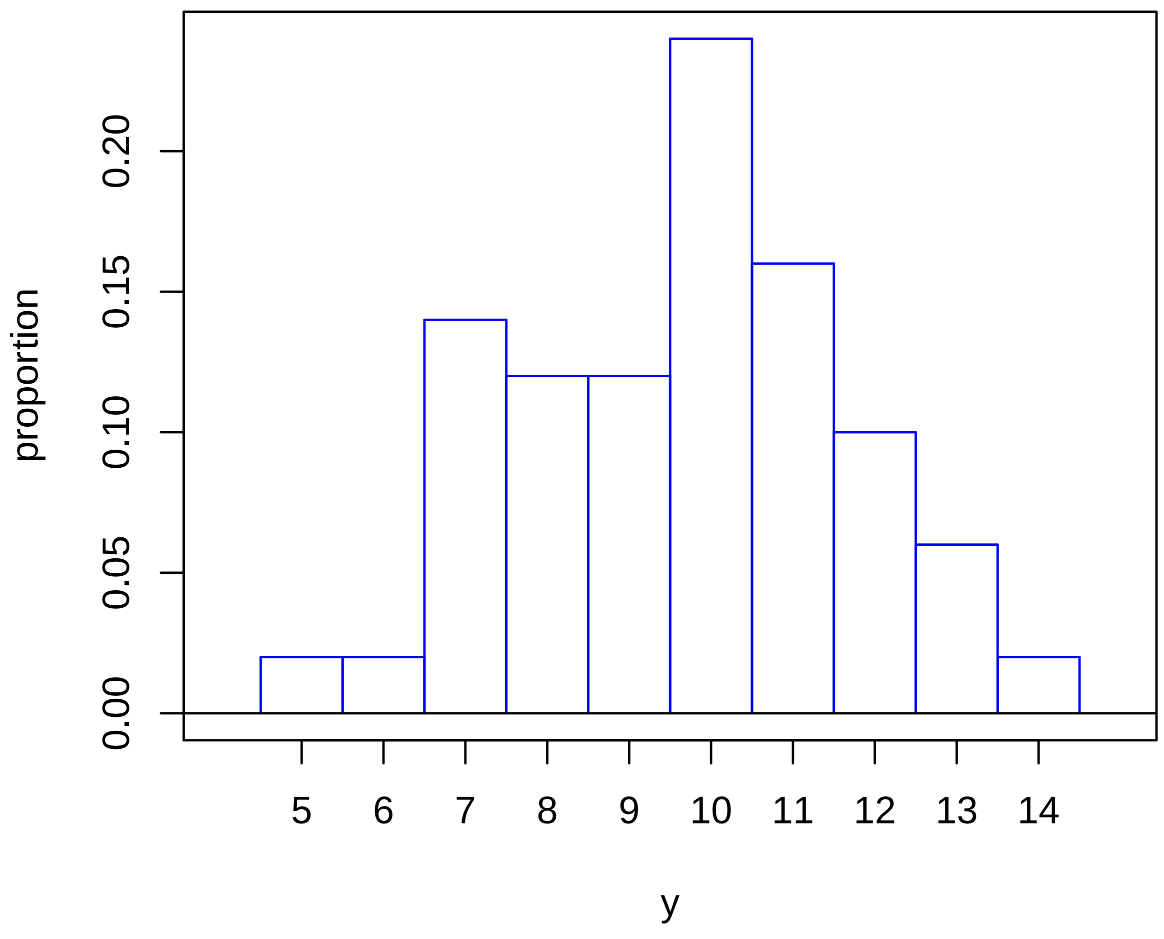
The posterior density, $p(\mu, \sigma^2|\boldsymbol{y})$, is proportional to

$$\prod_{i=1}^{n} \left[\Phi\left(\frac{y_i + 1/2 - \mu}{\sigma}\right) - \Phi\left(\frac{y_i - 1/2 - \mu}{\sigma}\right) \right]$$

$$\times \frac{1}{\sigma} \exp\left(-\frac{\mu^2}{2\sigma^2}\right) (\sigma^2)^{-2} e^{-1/\sigma^2}.$$

This is a decidedly nonstandard distribution. Neither ordinary Monte Carlo nor Gibbs sampling seem feasible here.

Data distribution



The data may be found at eCampus.

I used the Metropolis algorithm to generate observations from the posterior.

I took the proposal to be the following bivariate normal distribution:

$$J(\boldsymbol{u}|\boldsymbol{v}) = \frac{1}{2\pi s_1 s_2} \exp\left[-\frac{1}{2}(\boldsymbol{u}-\boldsymbol{v})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{u}-\boldsymbol{v})\right],$$

where

$$\boldsymbol{\Sigma} = \begin{bmatrix} s_1^2 & 0 \\ 0 & s_2^2 \end{bmatrix}.$$

The first and second components of $\boldsymbol{u}$ (or $\boldsymbol{v}$) correspond to μ and σ^2, respectively.

A seeming problem with this proposal distribution is that $\sigma^2 > 0$, and yet at least a small fraction of the generated values for σ^2 will be negative.

However, note that $p(\mu, \sigma^2|\boldsymbol{y}) = 0$ whenever $\sigma^2 < 0$, and therefore *a negative value of σ^2 will never be accepted in the Metropolis algorithm.*

This illustrates that it is ok for a Metropolis algorithm to use a proposal distribution whose support *contains* the support of the distribution of interest.

The only potential drawback of doing so is that the algorithm may be inefficient, i.e., it may generate too many values that are not accepted.

Remarks:

- After some experimentation I used $s_1 = 0.8$ and $s_2 = 1$.
- The two chains mixed reasonably well, although both exhibited autocorrelation.
- The autocorrelation was more pronounced for the σ^2 chain, but was essentially 0 after 50 lags.

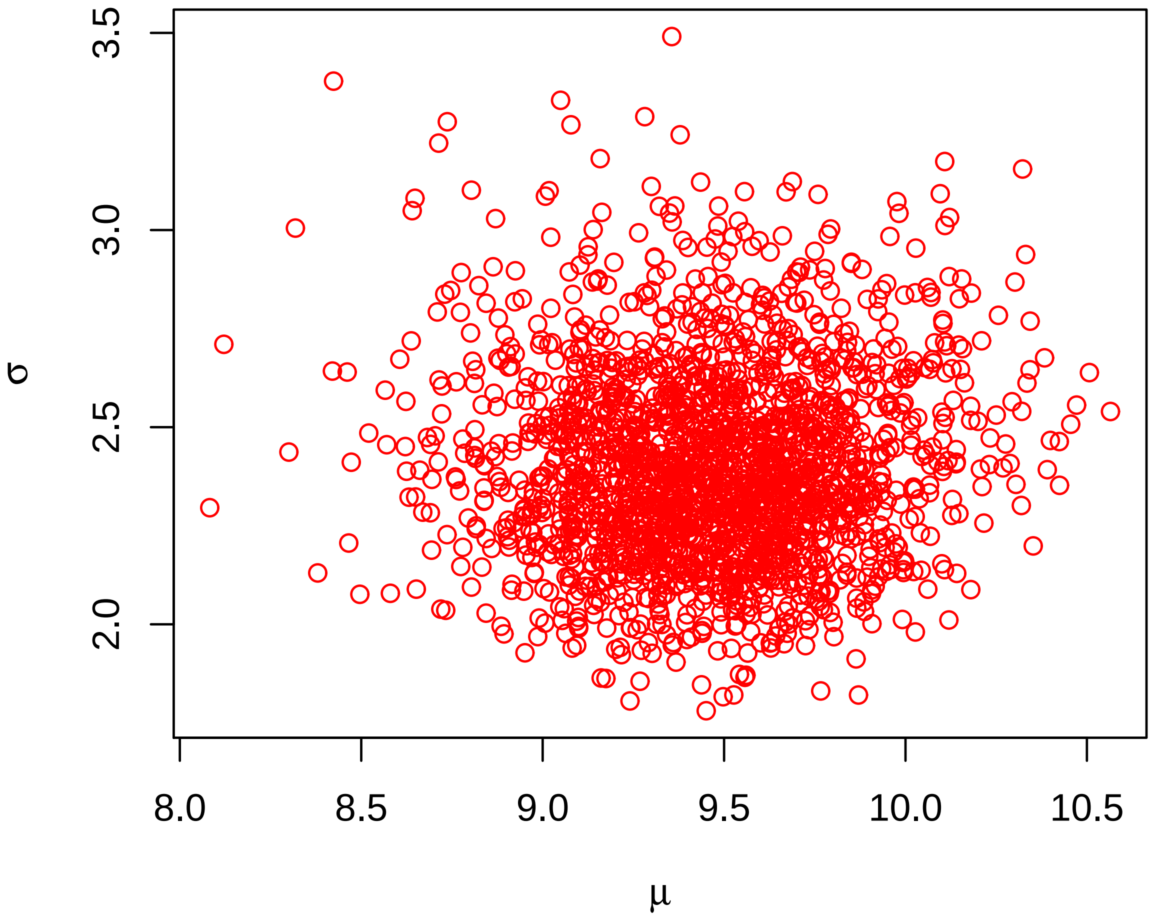
I generated a total of 100,000 observations and thinned them, using every 50th one.

The results are summarized below.

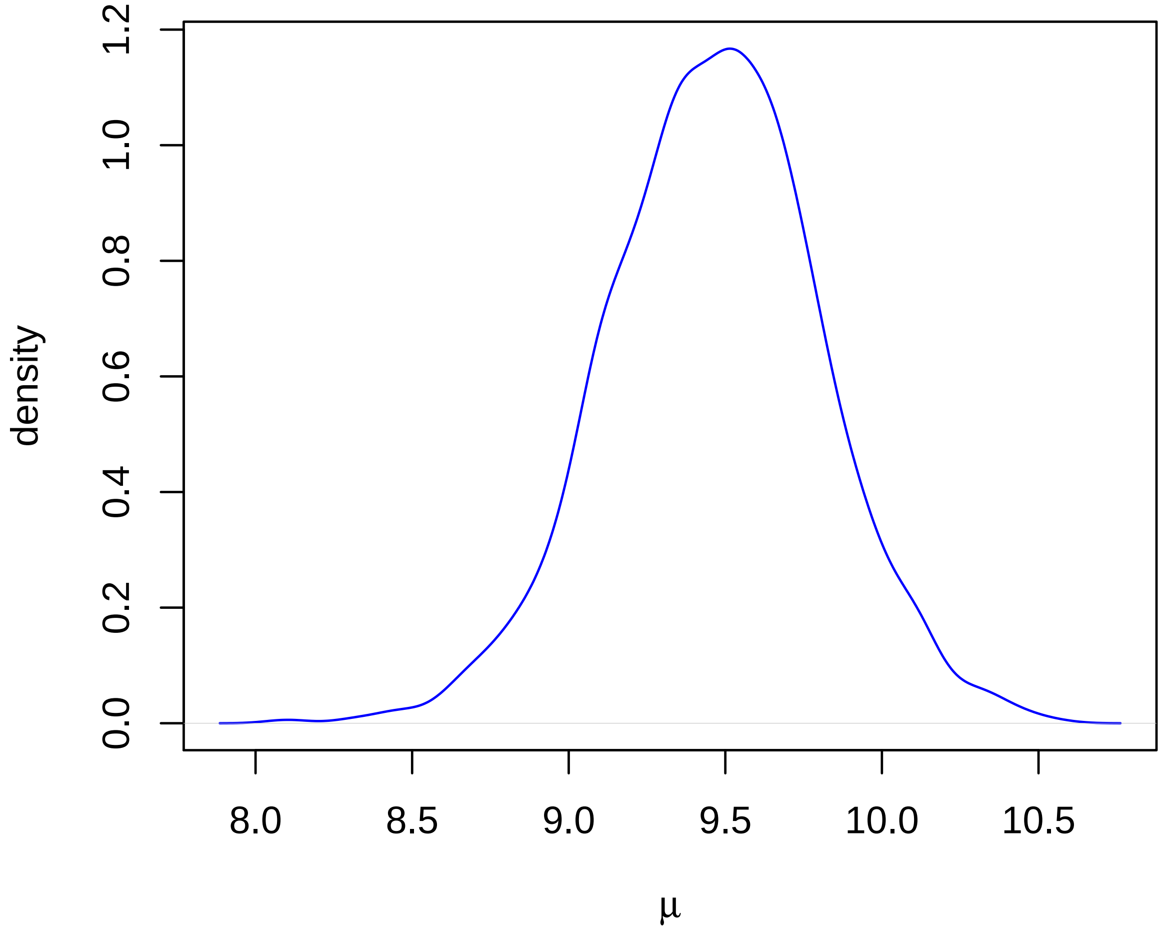
Parameter	Posterior mean	Posterior sd
μ	9.46	0.34
σ	2.40	0.25

The original data were generated from a normal distribution with mean 10 and standard deviation 2.

Scatterplot of σ vs. μ



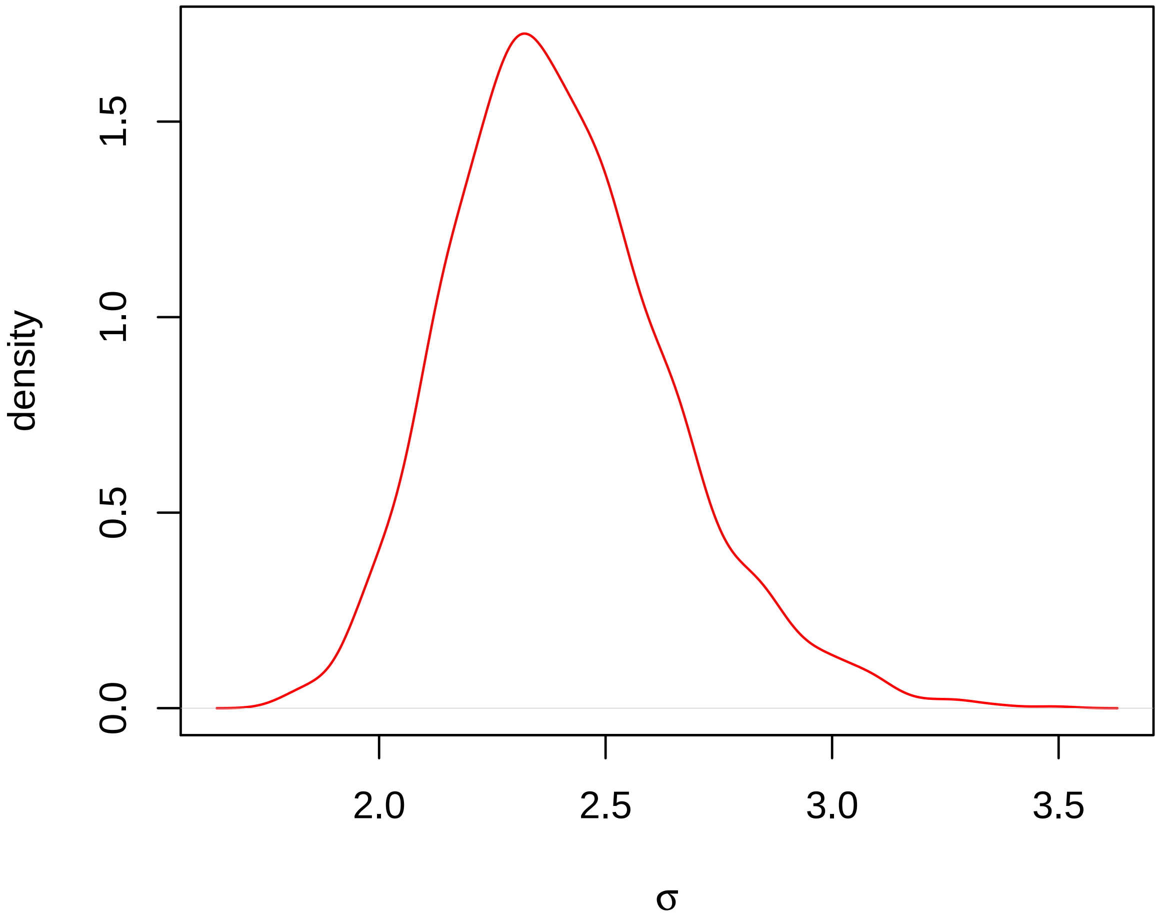
Kernel estimate of $p(\mu|\boldsymbol{y})$



Approximate 95% credible interval:

$(8.77, 10.13)$

Kernel estimate of $p(\sigma|\boldsymbol{y})$



Approximate 95% credible interval: $(1.99, 2.96)$

Metropolis-Hastings algorithm

The Metroplis-Hastings algorithm is the same as Metropolis except for two things:

- The proposal distribution $J(\boldsymbol{u}|\boldsymbol{v})$ does not have to be symmetric.

- If the current value of the chain is $\boldsymbol{\theta}^{(s)}$ and the proposed value $\boldsymbol{\theta}^*$, the acceptance ratio is
$$r = \frac{f(\boldsymbol{\theta}^*)}{f(\boldsymbol{\theta}^{(s)})} \cdot \frac{J(\boldsymbol{\theta}^{(s)}|\boldsymbol{\theta}^*)}{J(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(s)})}.$$

It is clear from the acceptance ratio that if the proposal is symmetric, then $r = f(\boldsymbol{\theta}^*)/f(\boldsymbol{\theta}^{(s)})$, and hence the Metropolis algorithm is a special case of Metropolis-Hastings.

Single-component Metropolis-Hastings (scMH)

One can also update components of $\boldsymbol{\theta}$ one at a time, as one does in Gibbs sampling.

We'll illustrate the idea in the case where $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$. As usual let f be the density of $\boldsymbol{\theta}$. The three full conditionals of f will be denoted f_1, f_2 and f_3.

There are three proposal distributions, q_1, q_2 and q_3, each of which is a conditional density.

Suppose the current value of the chain is $(\theta_1, \theta_2, \theta_3)$. We generate a value θ_1^* from $q_1(\,\cdot\,|\theta_1, \theta_2, \theta_3)$, and compute the acceptance ratio

$$r_1 = \frac{f_1(\theta_1^*|\theta_2, \theta_3)}{f_1(\theta_1|\theta_2, \theta_3)} \cdot \frac{q_1(\theta_1|\theta_1^*, \theta_2, \theta_3)}{q_1(\theta_1^*|\theta_1, \theta_2, \theta_3)}.$$

We decide whether or not to accept θ_1^* in the usual way.

Call the updated first component $\tilde{\theta}_1$ (which could be either θ_1^* or θ_1).

We now generate θ_2^* from $q_2(\,\cdot\,|\tilde{\theta}_1, \theta_2, \theta_3)$ and compute

$$r_2 = \frac{f_2(\theta_2^*|\tilde{\theta}_1, \theta_3)}{f_2(\theta_2|\tilde{\theta}_1, \theta_3)} \cdot \frac{q_2(\theta_2|\tilde{\theta}_1, \theta_2^*, \theta_3)}{q_2(\theta_2^*|\tilde{\theta}_1, \theta_2, \theta_3)}.$$

Now decide if we accept θ_2^* and call the updated value $\tilde{\theta}_2$.

Finally, generate θ_3^* from $q_3(\,\cdot\,|\tilde{\theta}_1, \tilde{\theta}_2, \theta_3)$ and compute

$$r_3 = \frac{f_3(\theta_3^*|\tilde{\theta}_1, \tilde{\theta}_2)}{f_3(\theta_3|\tilde{\theta}_1, \tilde{\theta}_2)} \cdot \frac{q_3(\theta_3|\tilde{\theta}_1, \tilde{\theta}_2, \theta_3^*)}{q_3(\theta_3^*|\tilde{\theta}_1, \tilde{\theta}_2, \theta_3)}.$$

Decide if we accept θ_3^* and call the updated value $\tilde{\theta}_3$.

All three components have been updated and we now iterate the process.

Suppose that *for the ith proposal we use the ith full conditional of f.* In other words,

$$q_i(u|\theta_1, \ldots, \theta_p) = f_i(u|\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_p).$$

Then each of the acceptance ratios is 1 and scMH becomes Gibbs sampling!

So, Gibbs sampling is a special case of Metropolis-Hastings.

In particular, *it is the special case of scMH where each proposal is a full conditional of the density from which we want to generate data.*

The independence sampler

The independence sampler is a special case of Metropolis-Hastings in which $J(\boldsymbol{u}|\boldsymbol{v}) = J(\boldsymbol{u})$.

In other words, the proposal distribution does not depend on the current value of the chain.

The value $\boldsymbol{\theta}^{(s+1)}$ depends on $\boldsymbol{\theta}^{(s)}$ only through the acceptance ratio

$$r = \frac{f(\boldsymbol{\theta}^*)}{f(\boldsymbol{\theta}^{(s)})} \cdot \frac{J(\boldsymbol{\theta}^{(s)})}{J(\boldsymbol{\theta}^*)}.$$

If J is a good match to f, then using the independence sampler will be close to generating independent observations directly from f.

On the other hand, if J is not a good match to f, then the independence sampler can perform very poorly.

Example 20 An application of MCMC to a problem in astronomy

Variable stars are characterized by brightness changes over time. *Long period variables* have substantial brightness changes, which are roughly sinusoidal with periods between 100 and 300 days.

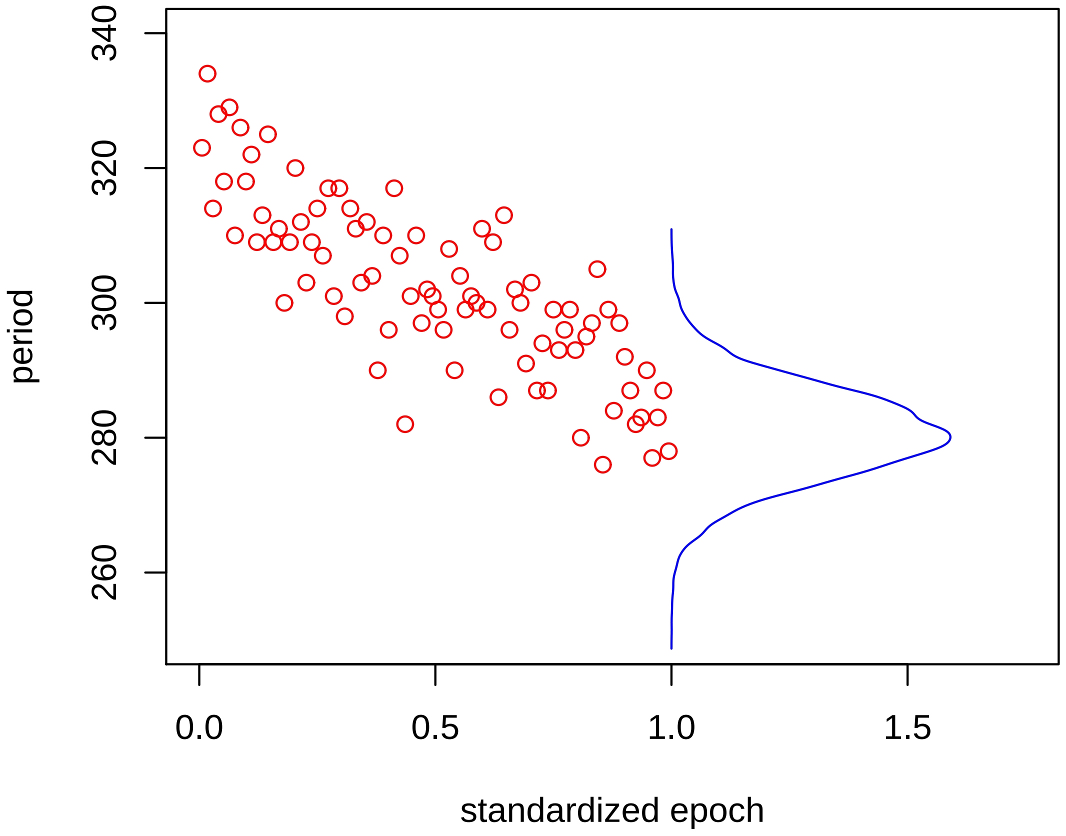
Period changes are deemed important by astronomers as they reflect changing conditions in a star.

We'll analyze *recorded times between maximum brightness* for the long period variable *R Aquilae*. The long-run average of a series of such times represents the period of the star.

See Hart, Koen and Lombard (2007), *Journal of the Royal Statistical Society, Series C* for an analysis of a collection of variable stars.

Of interest is detecting any systematic change that may occur in the period.

Observed times between maximum brightnesses for R Aquilae



The number of observations is $n = 86$. The average time between maximum brightnesses is about 302 days. Hence, these observations were taken over a time period of about

$$(302)86/365 = 71 \text{ years.}$$

A model proposed by astronomers

Let $Y_1, \dots, Y_n$ denote the series of observed times between max brightnesses. Our model is

$$Y_j = \mu(x_j) + I_j + \epsilon_j - \epsilon_{j-1}, \quad j = 1, \dots, n.$$

- $x_j = (j - 1/2)/n$, $j = 1, \dots, n$. These are standardized *epochs*. The epochs represent a chronological ordering of the observations.

- μ is a smooth function accounting for systematic changes in the period. In the absence of any change, μ is a constant equal to the star's period.

- $I_1, \ldots, I_n$ are *intrinsic* errors peculiar to the star. These account for random deviations from "perfect periods".

- $\epsilon_0, \ldots, \epsilon_n$ are measurement errors made in recording times of maximum brightness.

- The measurement errors are independent of the intrinsic errors.

We will model μ as a third degree polynomial:

$$\mu(x; \boldsymbol{\theta}) = \theta_0 + \theta_1 u_1(x) + \theta_2 u_2(x) + \theta_3 u_3(x),$$

where u_1, u_2 and u_3 are known orthogonal polynomials.

We model the measurement errors as independent normal random variables with common mean 0 and variance structure as follows:

$$\text{Var}(\epsilon_j) = v_j(\boldsymbol{\beta}), \quad j = 0, \ldots, n,$$

where $v_j(\boldsymbol{\beta}) = \exp(\beta_0 + \beta_1 x_j)$, $j = 0, \ldots, n$.

We assume $I_1, \ldots, I_n$ are i.i.d. $N(0, r\,\exp(\beta_0))$.

The parameter r has the interpretation of being the ratio of the variance of an intrinsic error to the variance of a measurement error at the earliest observation time.

Now, reparameterize by defining $\rho = \log(r)$, which means that

$$\text{Var}(I_j) = \sigma_I^2 = \exp(\rho + \beta_0).$$

Choice of prior

- We'll assume that a priori $\exp(\beta_0/2)$ and σ_I are independent and that each has prior

$$g(s) = \frac{2}{\Gamma(1/2)} s^{-2} \exp(-s^{-2}) I_{(0,\infty)}(s).$$

- We assume that β_1 is a priori independent of ρ and β_0 and has prior $N(0, 16)$.

- Finally, we assume that the regression coefficients are independent of the other parameters and have a uniform prior.

So, the prior ends up having the form

$$\begin{aligned}\pi(\rho, \beta_0, \beta_1, \boldsymbol{\theta}) \;\; \propto \;\; & g(\sigma_I) g(\sigma_\epsilon) \exp(\beta_0 + \rho/2) \\ & \times \phi(\beta_1/4),\end{aligned}$$

where $\sigma_\epsilon = \exp(\beta_0/2)$.

Now, the likelihood is

$$f(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\beta}, \rho) \propto$$

$$|\boldsymbol{\Sigma}|^{-1/2} \exp\left[-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{\mu}(\boldsymbol{\theta}))^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} - \boldsymbol{\mu}(\boldsymbol{\theta}))\right],$$

where

$$\boldsymbol{\mu}(\boldsymbol{\theta}) = (\mu(x_1; \boldsymbol{\theta}), \ldots, \mu(x_n; \boldsymbol{\theta}))^T$$

and the matrix $\boldsymbol{A} = \exp(-\beta_0)\boldsymbol{\Sigma}$ is symmetric and has elements

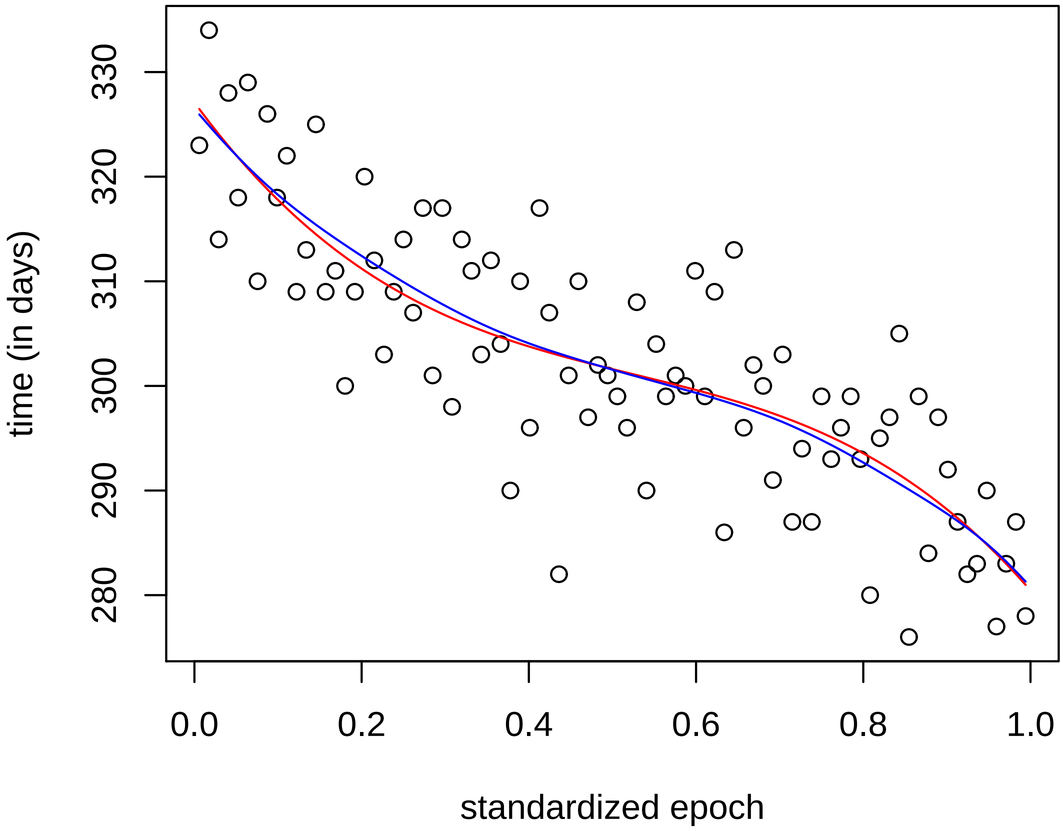
$$\boldsymbol{A}_{ii} = \exp(\rho) + \exp(\beta_1 x_i) + \exp(\beta_1 x_{i-1}),$$

$i = 1, \ldots, n$,

$$\boldsymbol{A}_{i(i+1)} = -\exp(\beta_1 x_i), \ i = 1, \ldots, n-1,$$

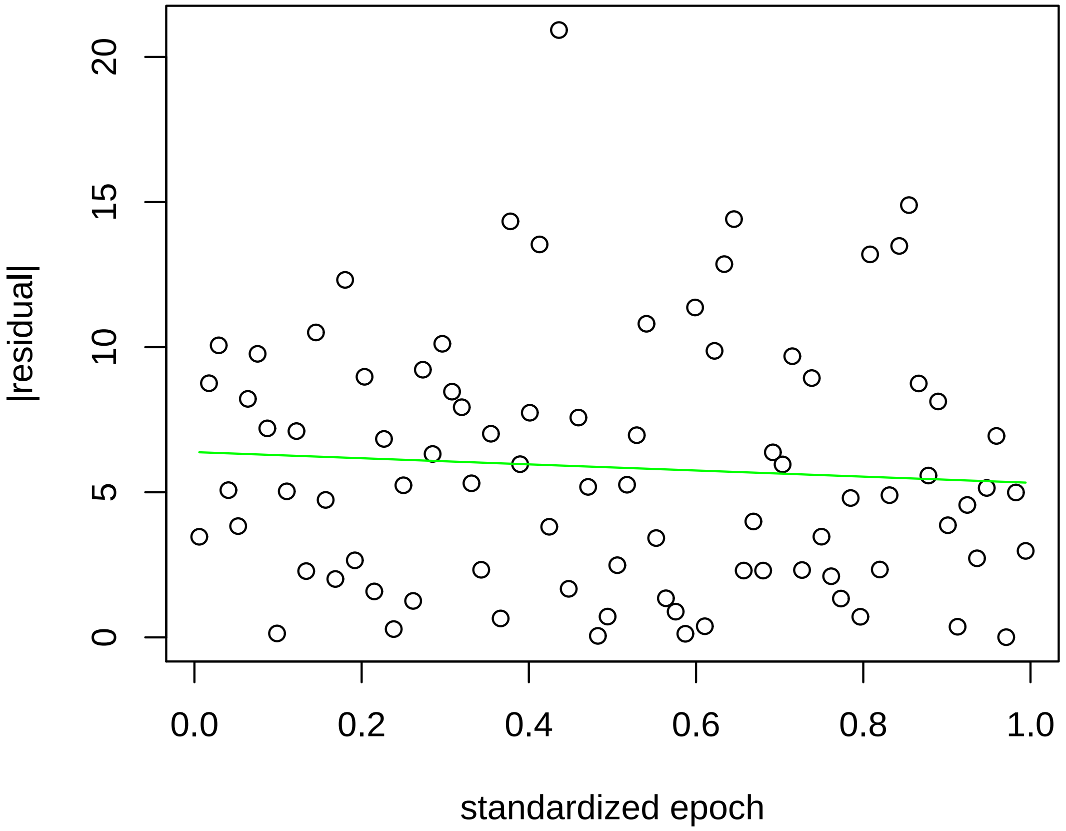
and $\boldsymbol{A}_{ij} = 0$ for all i and $j > i + 1$.

Before proceeding to a Bayesian analysis, let's look at the data a bit more.



* Bandwidth was chosen using a version of one-sided cross-validation that takes into account correlation.

Absolute residuals from cubic fit



For some stars, there is an apparent decrease in variance over the years, because of increased precision in measurement methods. In this case, however, there's not much evidence of a decrease.

MCMC

There are seven parameters to estimate in our model. My first try for the proposal distribution was of the form

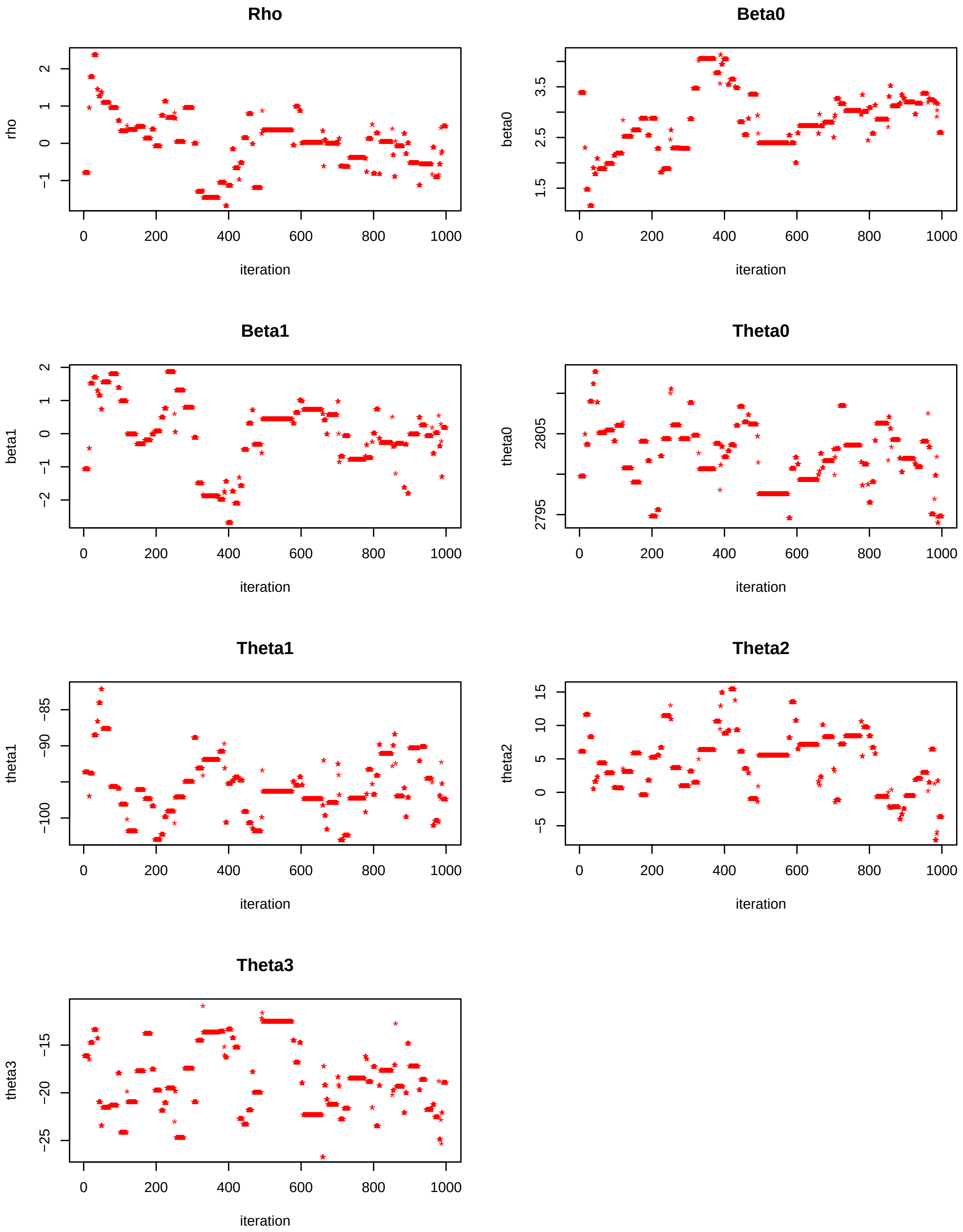
$$J(\boldsymbol{u}|\boldsymbol{v}) = \prod_{i=1}^{7} \frac{1}{\sigma_i} \phi\left(\frac{u_i - v_i}{\sigma_i}\right),$$

where ϕ is the standard normal density.

I started each chain at what I had previously determined to be the MLE for the parameter vector. Initial experiments often produced chains that moved only a few times out of 1000 iterations.

After several attempts I found $\sigma_1, \ldots, \sigma_7$ that produced the output on the following page.

Output from first try at proposal distribution



The previous output is not so good since the chains are mixing very slowly.

A potential problem with the proposal density on p. 283N is that the components are independent. We know that for large enough n, posteriors are approximately multivariate normal, but the covariance matrix might be far from diagonal.

A seemingly good proposal would be a multivariate normal with mean vector equal to the MLE and covariance matrix equal to the inverse of the Fisher information evaluated at the MLE.

But what if it's not so easy to determine the Fisher information? (This happens to be the case here!)

The output on p. 284 is perhaps not so good, but *maybe* it's good enough for getting "ball park" estimates of the mean and covariance matrix of the posterior distribution.

From the output on p. 284, we have 1000 (correlated) observations of $(\boldsymbol{\theta}, \boldsymbol{\beta}, \rho)$. I computed the sample mean vector and the 7×7 sample covariance matrix for these observations. Call these two objects $\widehat{\boldsymbol{\mu}}$ and $\boldsymbol{S}$.

I then used an *independence sampler* with proposal distribution equal to a multivariate normal with mean $\widehat{\boldsymbol{\mu}}$ and covariance matrix $\boldsymbol{S}$.

In this case, if the chain is at state $\boldsymbol{u}$, then a candidate $\boldsymbol{v}$ for $(\boldsymbol{\theta}, \boldsymbol{\beta}, \rho)$ is accepted with probability

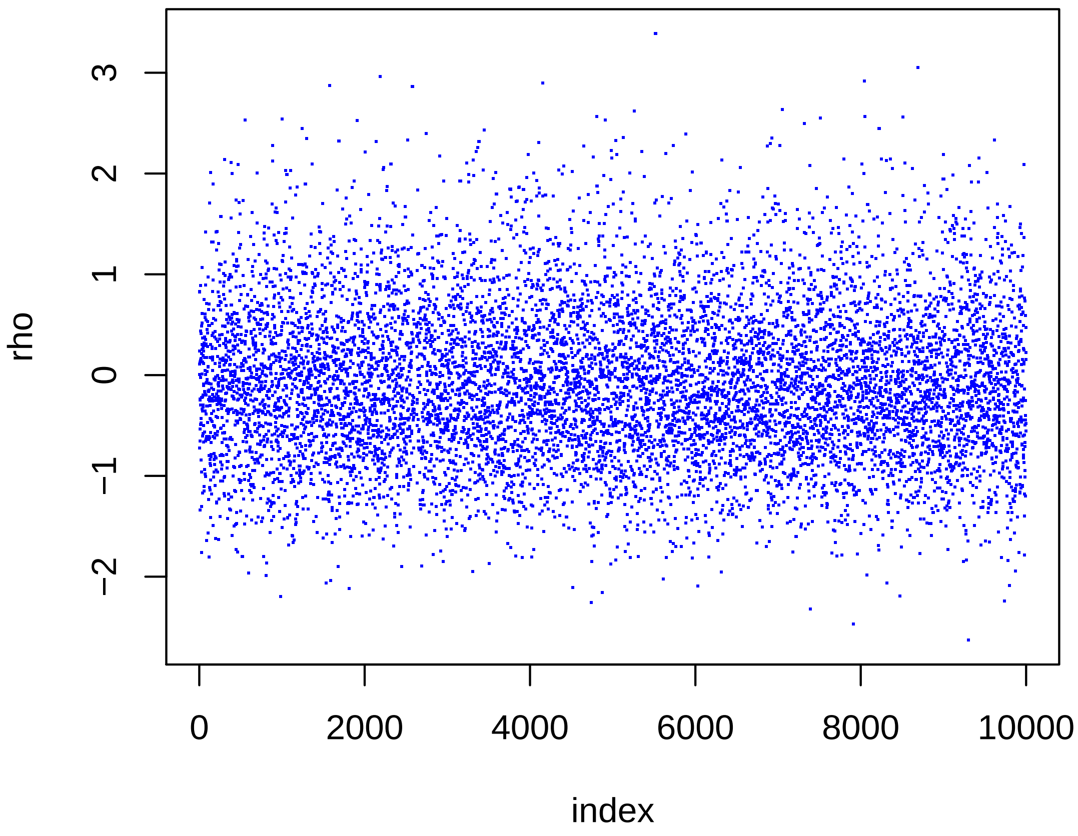
$$\min\left(1, \frac{h(\boldsymbol{v}, \boldsymbol{u})}{h(\boldsymbol{u}, \boldsymbol{v})}\right),$$

where

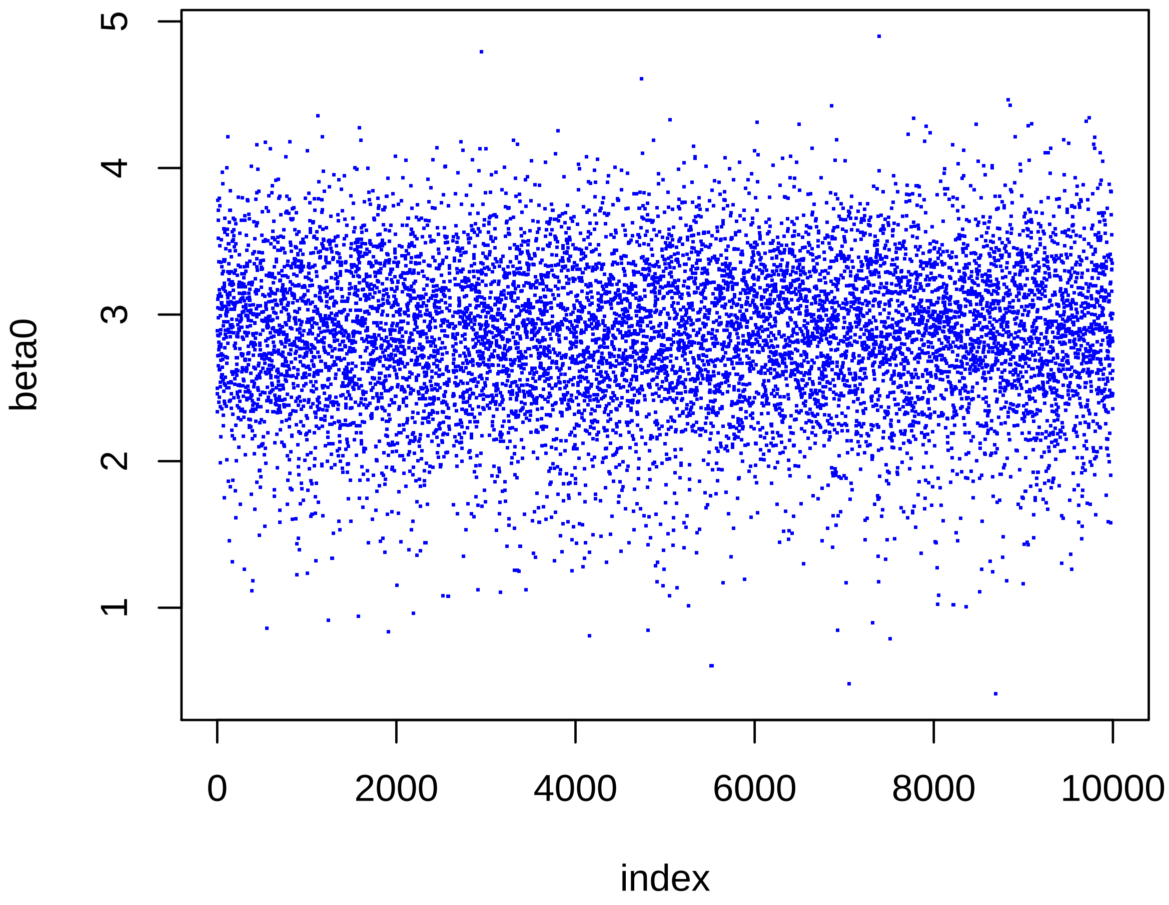
$$h(\boldsymbol{u}, \boldsymbol{v}) = \pi(\boldsymbol{u}|\boldsymbol{y}) \exp\left[-\frac{1}{2}(\boldsymbol{v} - \widehat{\boldsymbol{\mu}})^T \boldsymbol{S}^{-1}(\boldsymbol{v} - \widehat{\boldsymbol{\mu}})\right].$$

It turns out that this proposal worked very well, as can be seen from the output given on the next few pages.

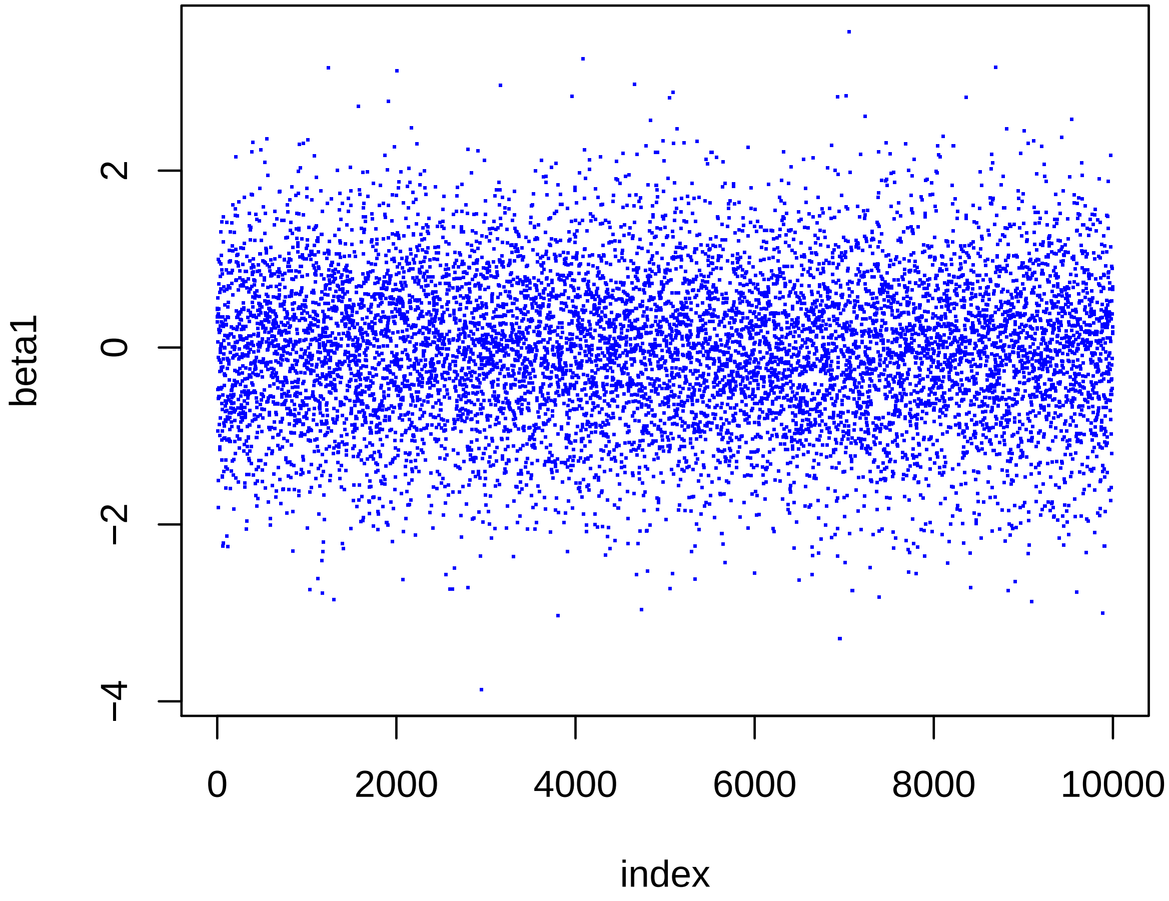
Output for ρ



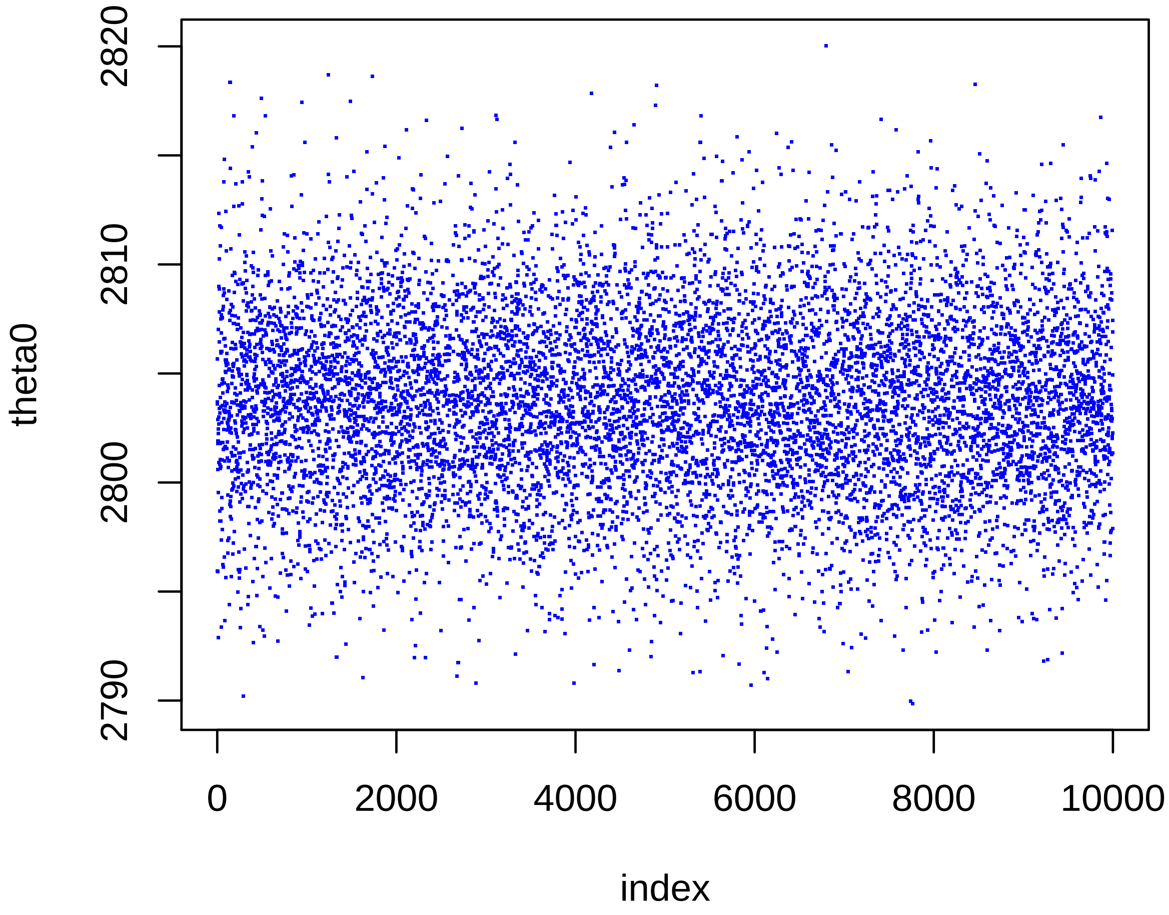
Output for β_0



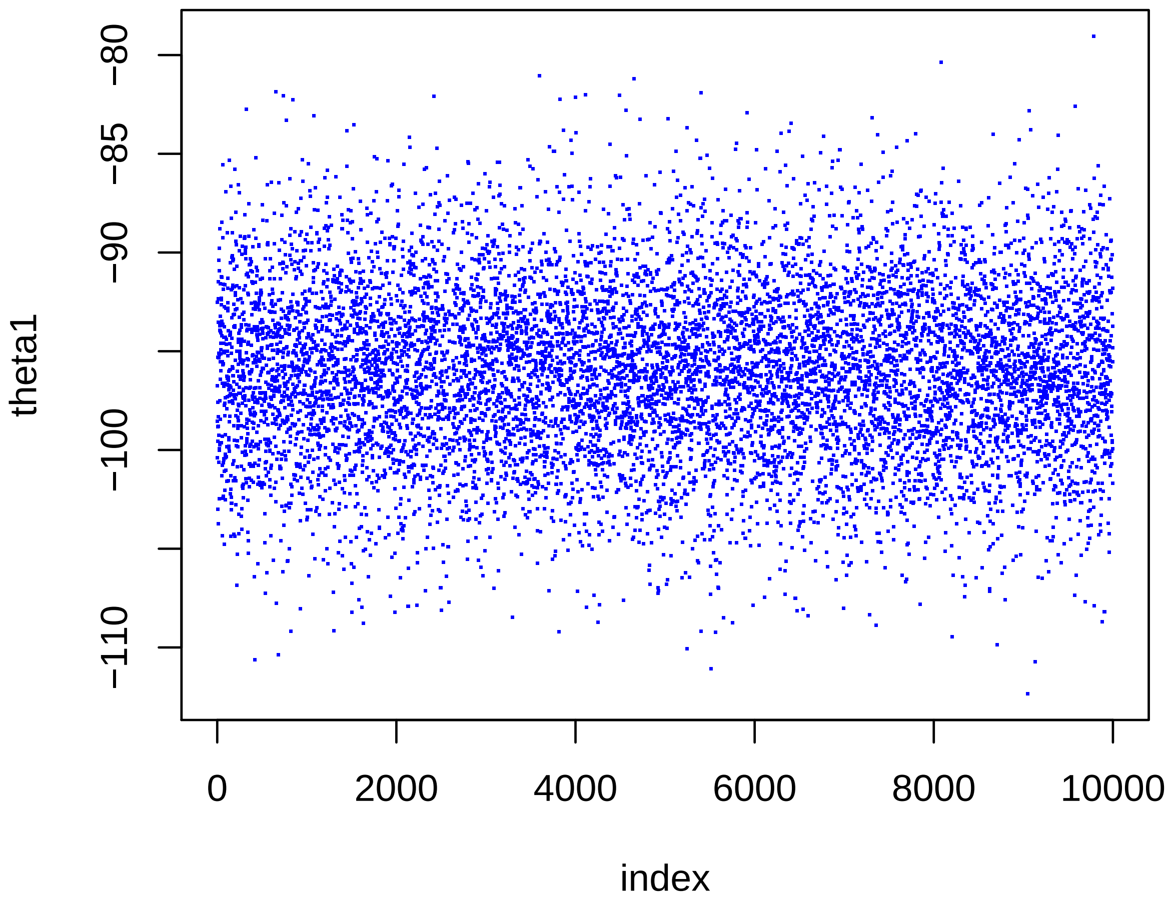
Output for β_1



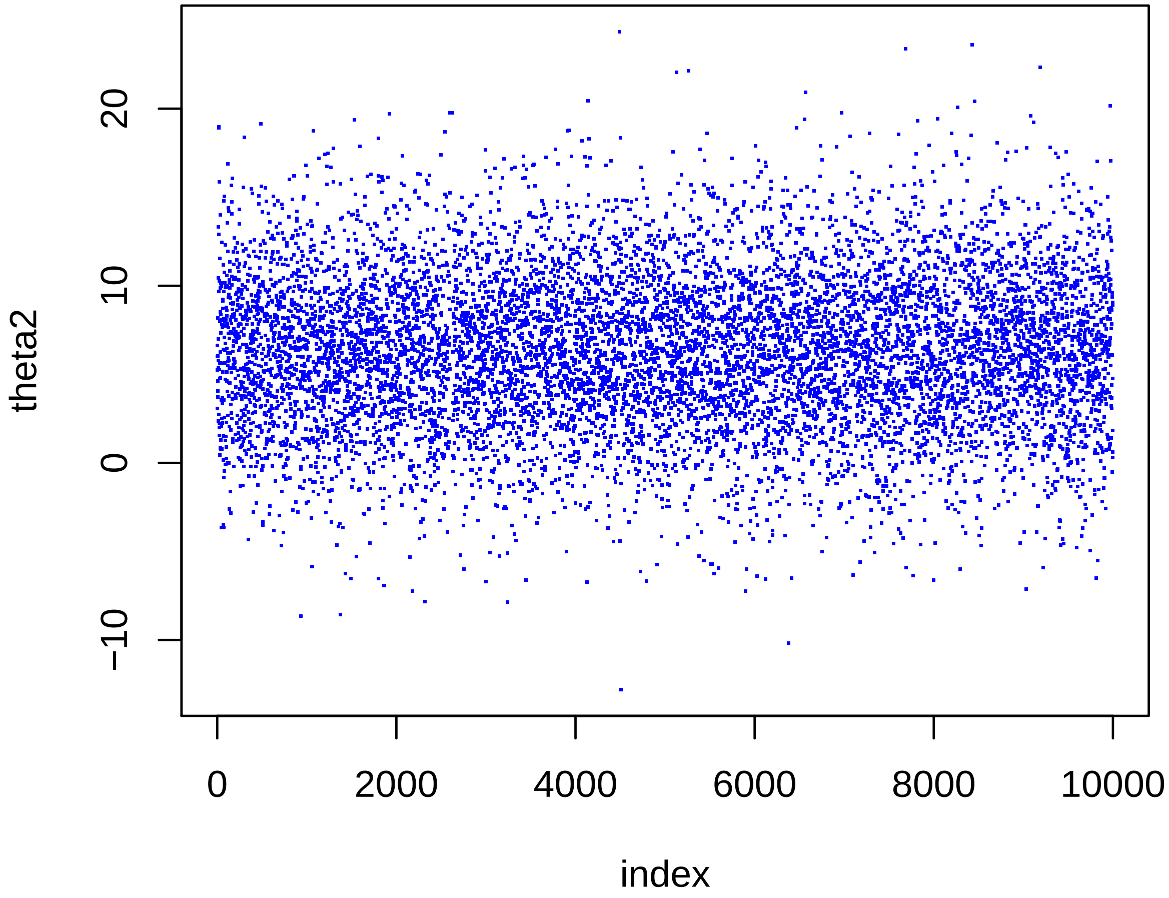
Output for θ_0



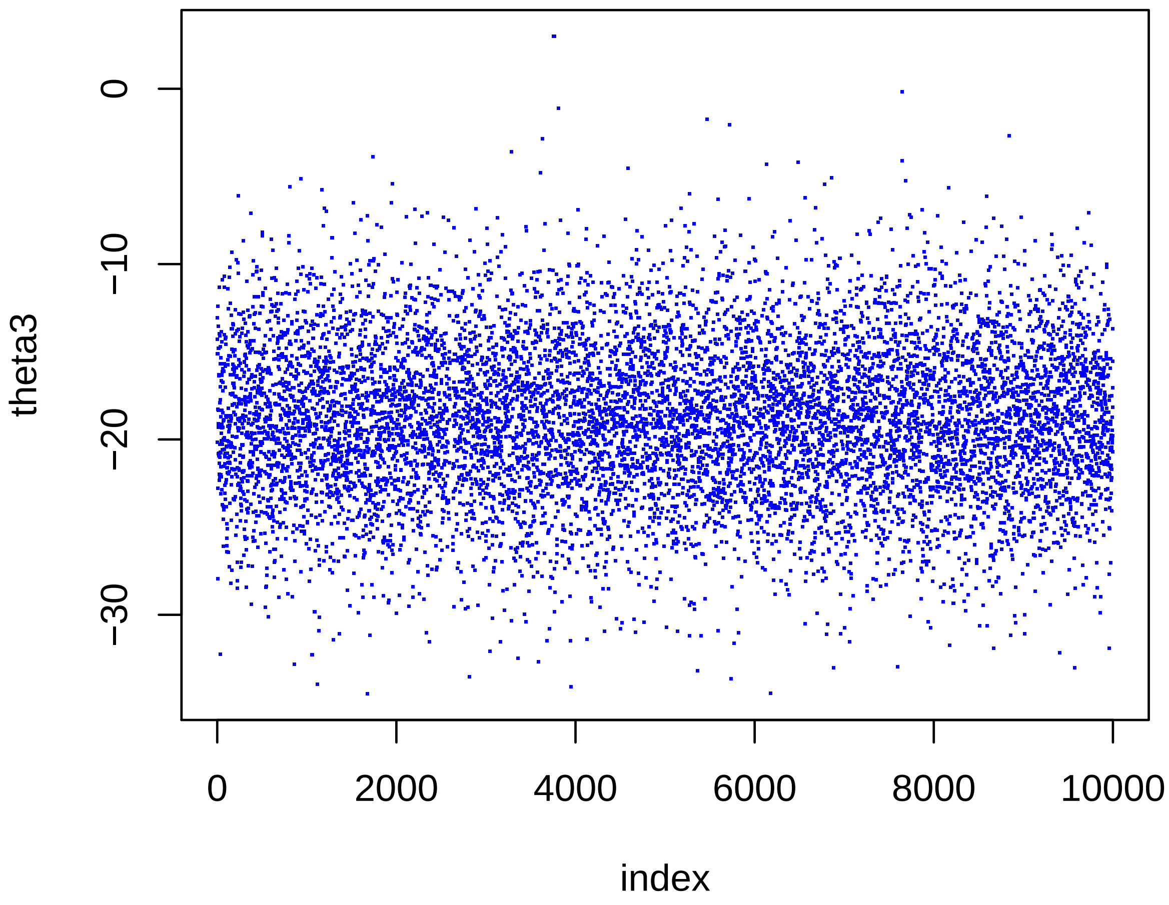
Output for θ_1



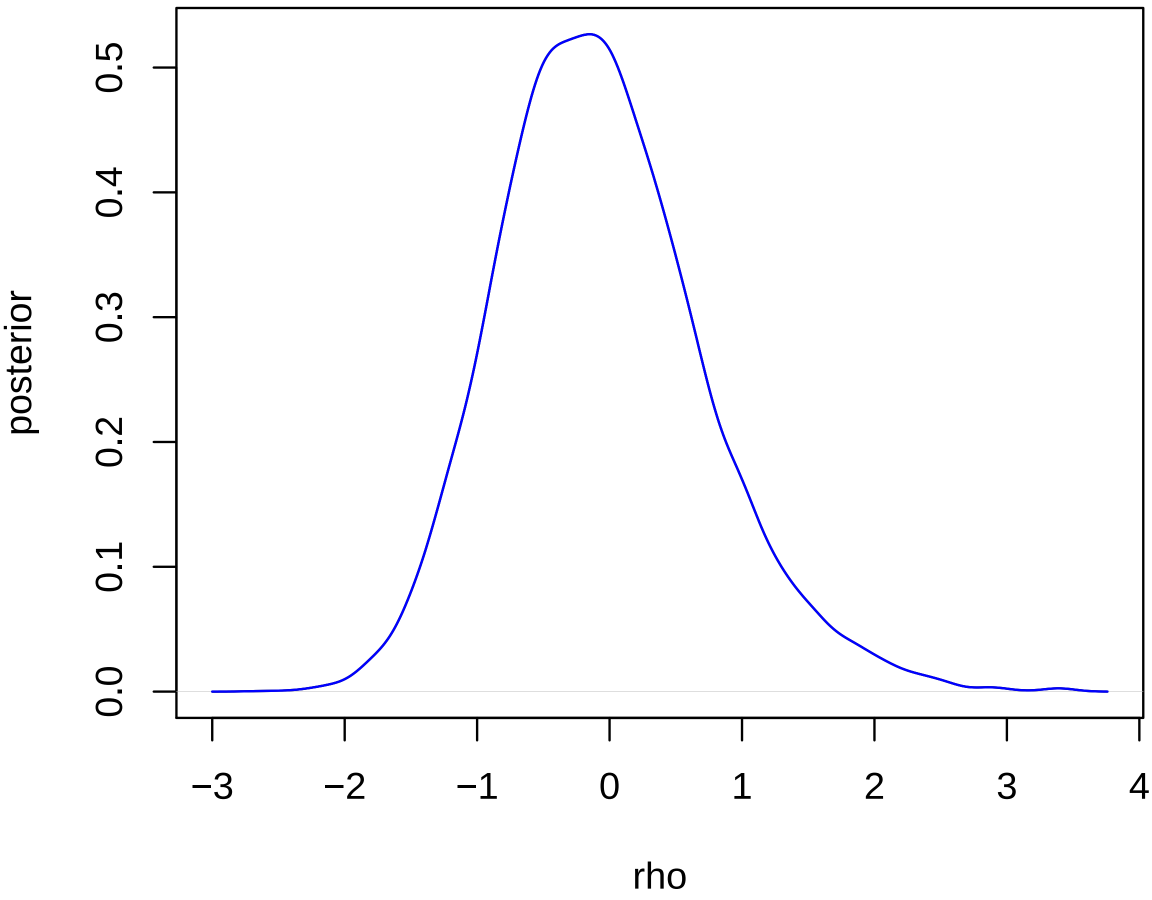
Output for θ_2



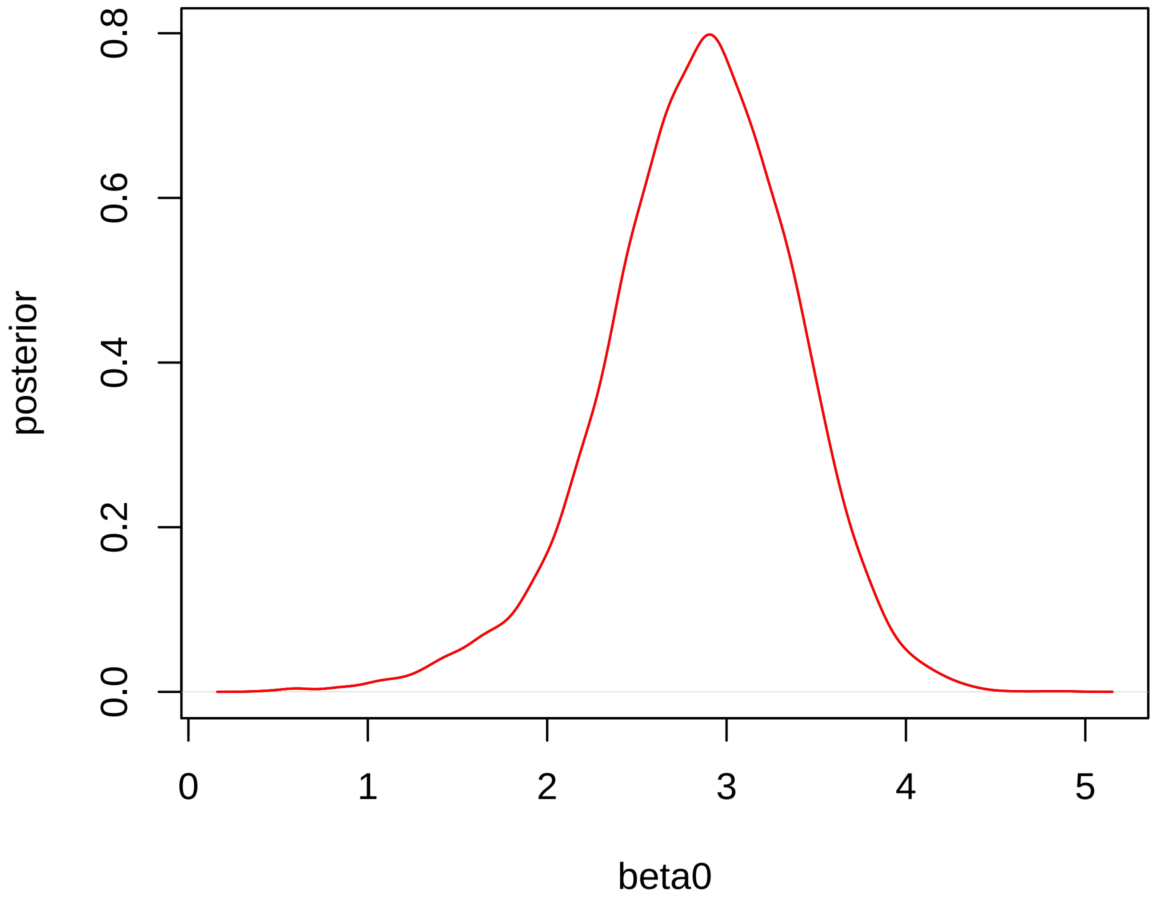
Output for θ_3



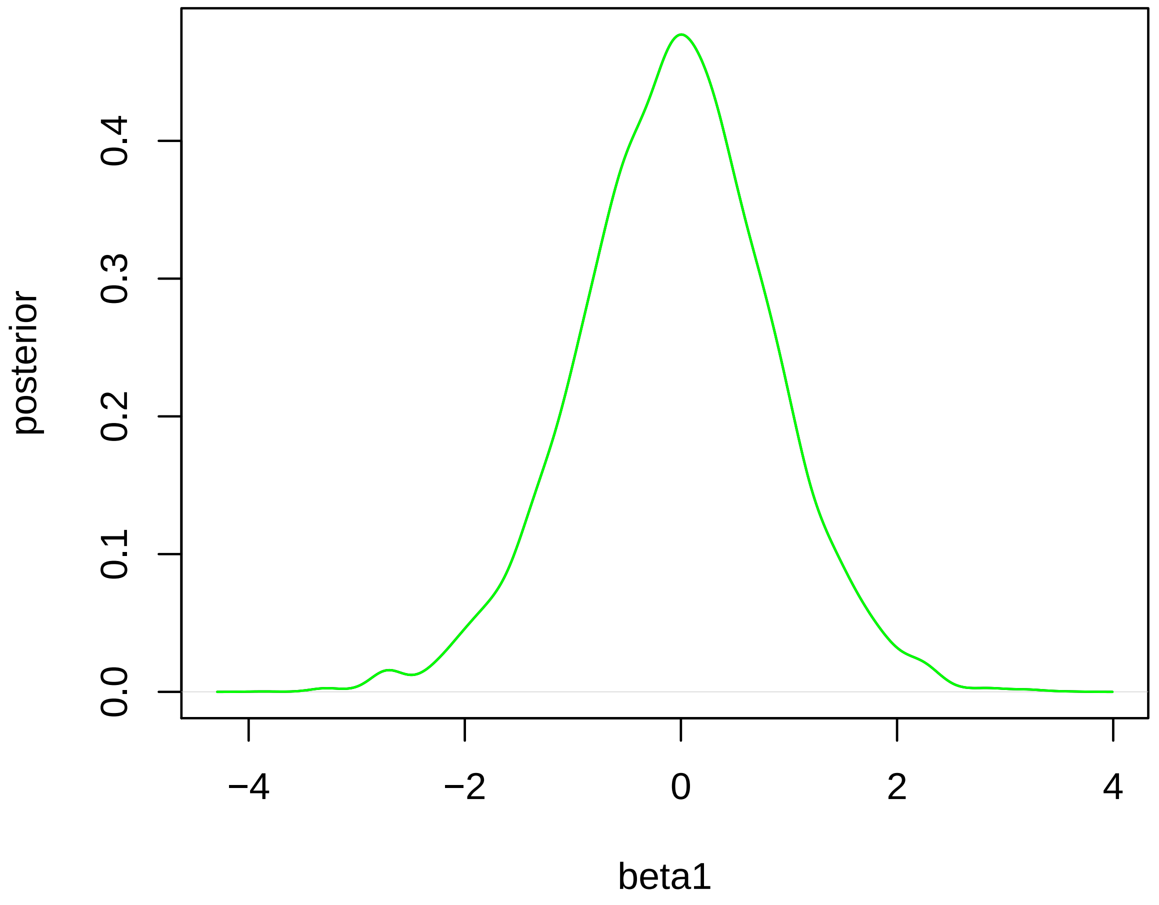
Density estimate of marginal posterior for ρ



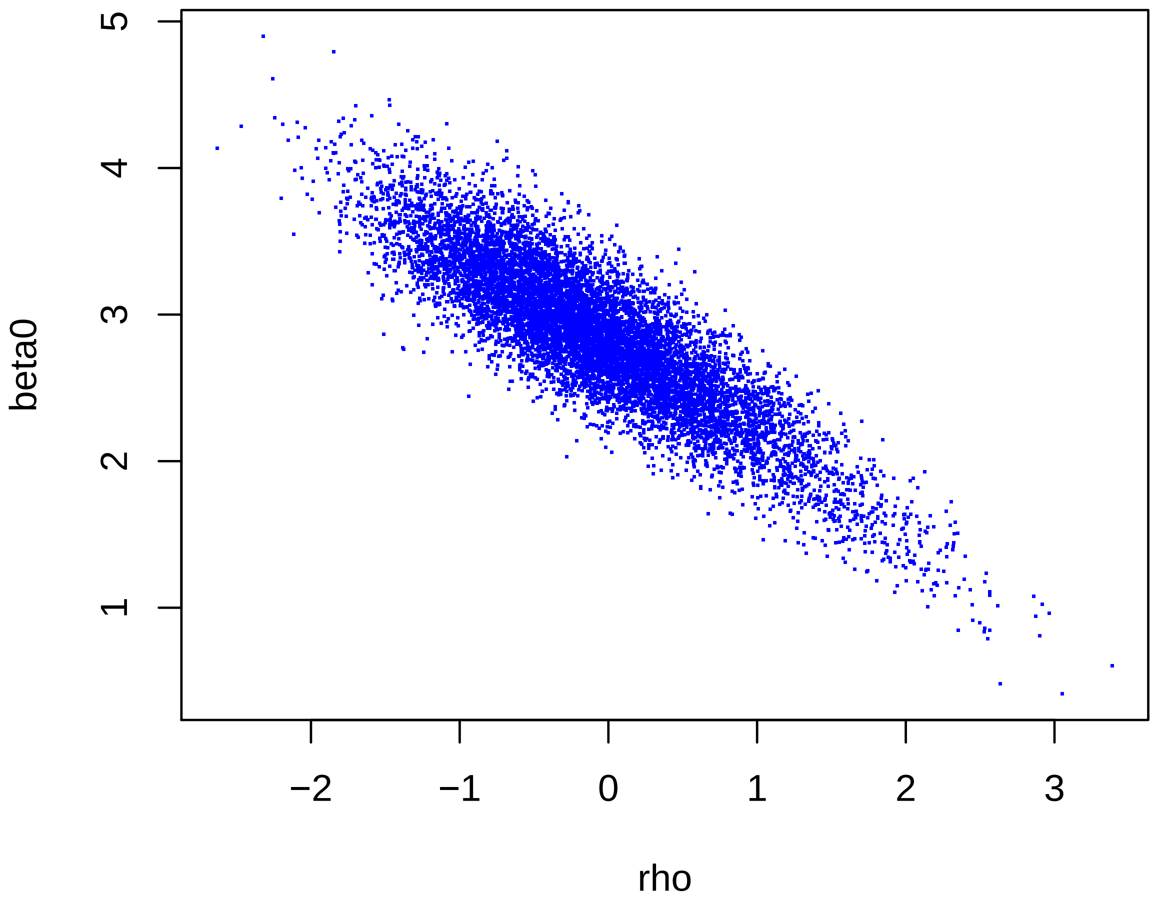
Density estimate of marginal posterior for β_0

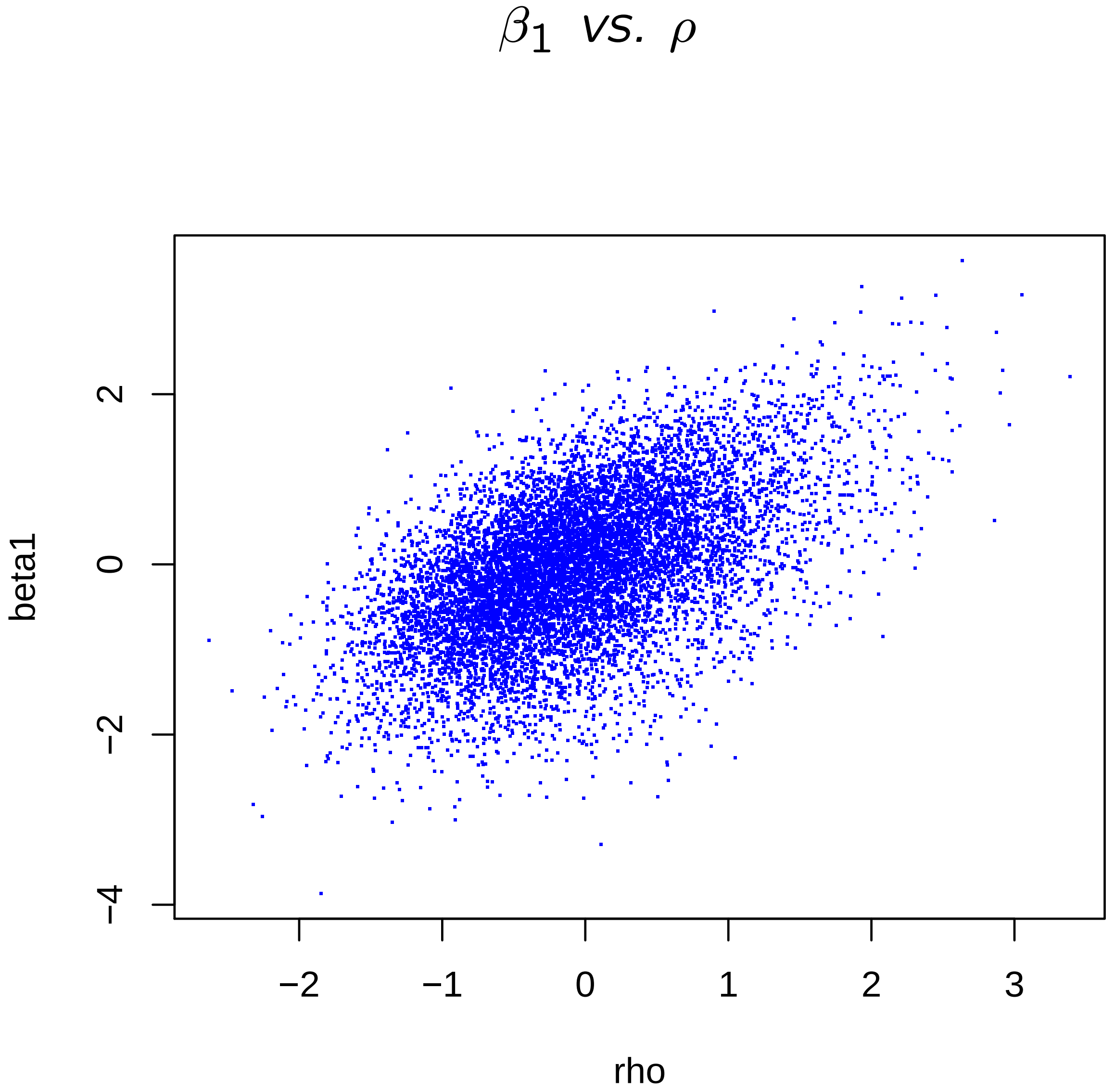


Density estimate of marginal posterior for β_1

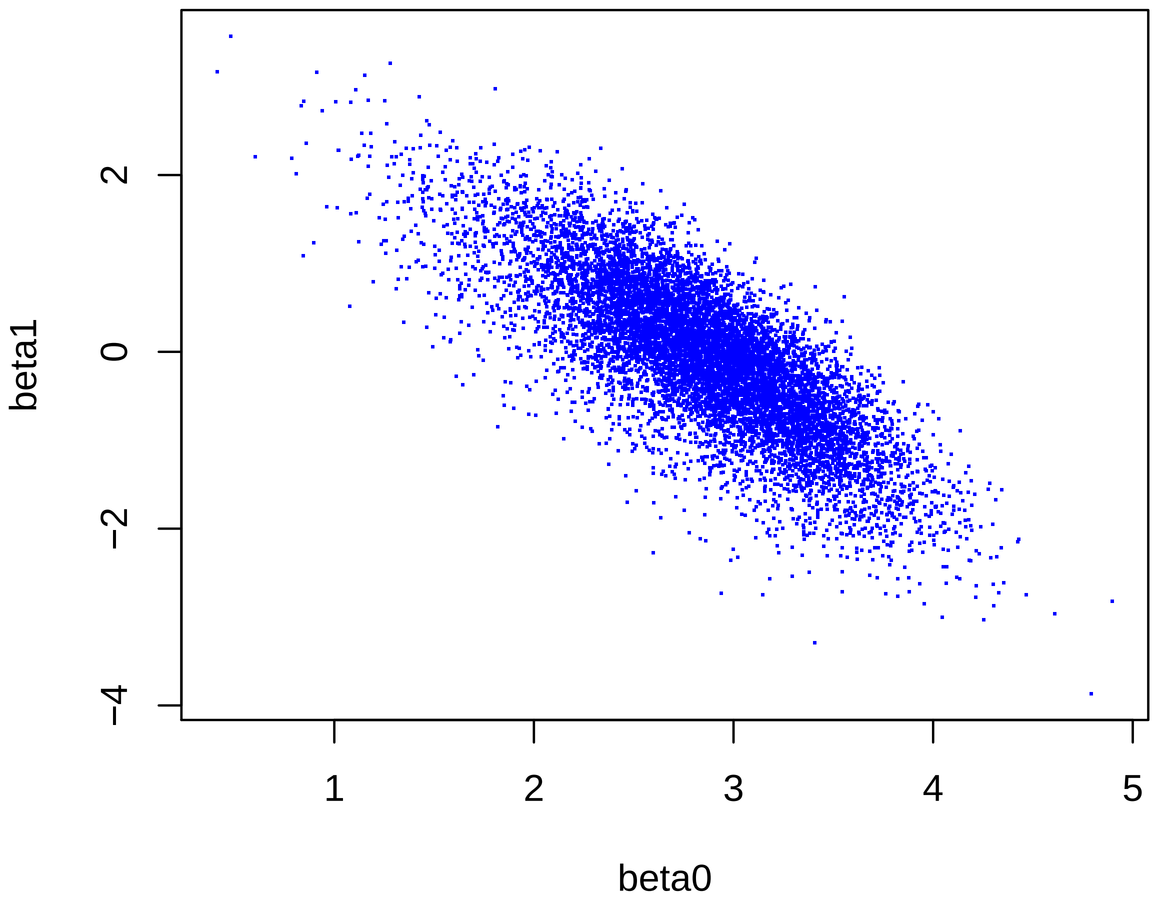


β_0 *vs.* ρ




β_1 vs. ρ
beta1
2
0
−2
−4
−2
−1
0
1
2
3
rho

β_1 *vs.* β_0



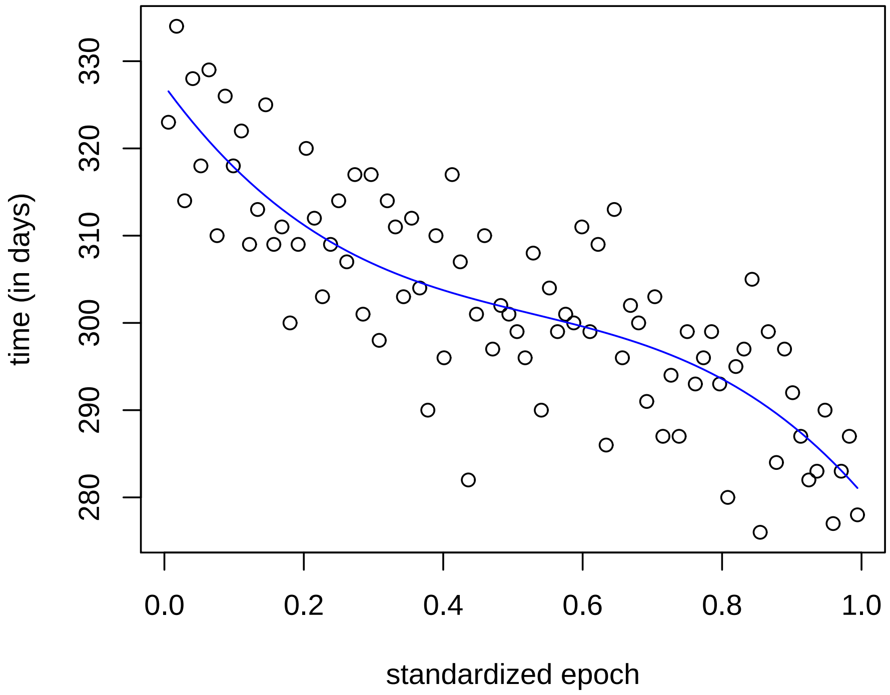
Values of variance parameters that maximize the posterior

These were determined by the search routine `nlm` in `R`.

$$\widehat{r} \approx e^{-0.595} = 0.552 \quad \widehat{\beta}_0 \approx 2.995$$

$$\widehat{\beta}_1 \approx -0.027$$

Maximum posterior estimate of cubic


time (in days)
standardized epoch