Assignment 5

Jeff Gould

4/8/2021

Theoretical Exercises

1

Define $s^2 = \frac{1}{n-1}\sum(y_i - \bar{y})^2$

$$\pi(\mu, \sigma^2, \alpha) \propto (1)((\sigma^2)^{-(c+1)}\exp(-d/\sigma^2))((\alpha)^{-(c+1)}\exp(-d/\alpha))$$
$$P(\mu, \sigma^2, \alpha|y) = \mathcal{L}(y|\mu, \sigma^2, \alpha)\pi(\mu, \alpha, \sigma^2)$$
$$\propto (2\pi(\alpha\sigma^2))^{-n/2}\exp\left(\frac{-\sum(y_i - \mu)^2}{2(\alpha\sigma^2)}\right)((\sigma^2)^{-(c+1)}\exp(-d/\sigma^2))((\alpha)^{-(c+1)}\exp(-d/\alpha))$$
$$\propto (\sigma^2)^{(-(c+n/2+1))}\alpha^{-(a+n/2+1)}\exp\left(-\frac{1}{(2\alpha\sigma^2)}(n-1)s^2\right)\exp\left(-\frac{1}{2(\alpha\sigma^2)/n}(\mu - \bar{y})^2\right)\exp\left(-\frac{b}{\alpha}\right)\exp\left(\frac{-d}{\sigma^2}\right)$$

Now find the conditional distributions for each parameter:

$$P(\mu|y, \sigma^2, \alpha) \propto \exp\left(-\frac{1}{2(\alpha\sigma^2)/n}(\mu - \bar{y})^2\right)$$

Which is the kernel for a normally distributed r.v with mean $\bar{y}$ and variance $\frac{\alpha\sigma^2}{n}$

$$(\sigma^2)^{(-(c+n/2+1))}\alpha^{-(a+n/2+1)}\exp\left(-\frac{b}{\alpha}\right)\exp\left(\frac{-d}{\sigma^2}\right)\exp\left(\frac{-\sum(y_i - \mu)^2}{2(\alpha\sigma^2)}\right)$$
$$\Rightarrow P(\alpha|\mu, \sigma^2, y) \propto \alpha^{-(a+n/2+1)}\exp\left(-\frac{b}{\alpha}\right)\exp\left(\frac{-\sum(y_i - \mu)^2}{2(\alpha\sigma^2)}\right)$$
$$= \alpha^{-(a+n/2+1)}\exp\left[-\left(\frac{b}{\alpha} + \frac{\sum(y_i - \mu)^2}{2\alpha\sigma^2}\right)\right]$$
$$= \alpha^{-(a+n/2+1)}\exp\left[-\left(\frac{b}{\alpha} + \frac{(1/2\sigma^2)\sum(y_i - \mu)^2}{\alpha}\right)\right]$$
$$= \alpha^{-(a+n/2+1)}\exp\left[-\left(\frac{b + (1/2\sigma^2)\sum(y_i - \mu)^2}{\alpha}\right)\right]$$

This is the kernel for a r.v. with an Inverse Gamma distriburion with parameters $a + n/2$ and $b + (\sum(y_i - \mu)^2)/(2\sigma^2)$

$$(\sigma^2)^{(-(c+n/2+1))}\alpha^{-(a+n/2+1)}\exp\left(-\frac{b}{\alpha}\right)\exp\left(\frac{-d}{\sigma^2}\right)\exp\left(\frac{-\sum(y_i-\mu)^2}{2(\alpha\sigma^2)}\right)$$

$$\Rightarrow P(\sigma^2|\mu,\alpha,y)\propto(\sigma^2)^{(-(c+n/2+1))}\exp\left(\frac{-d}{\sigma^2}\right)\exp\left(\frac{-\sum(y_i-\mu)^2}{2(\alpha\sigma^2)}\right)$$

$$=(\sigma^2)^{(-(c+n/2+1))}\exp\left[-\left(\frac{d}{\sigma^2}+\frac{\sum(y_i-\mu)^2}{2\alpha\sigma^2}\right)\right]$$

$$=(\sigma^2)^{(-(c+n/2+1))}\exp\left[-\left(\frac{d}{\sigma^2}+\frac{(1/2\alpha)\sum(y_i-\mu)^2}{\alpha}\right)\right]$$

$$=(\sigma^2)^{(-(c+n/2+1))}\exp\left[-\left(\frac{d+(1/2\alpha)\sum(y_i-\mu)^2}{\sigma^2}\right)\right]$$

This is the kernel for a r.v. with an Inverse Gamma distriburion with parameters $c+n/2$ and $d+(\sum(y_i-\mu)^2)/(2\alpha)$

Now the Gibbs Sampler:

Let's say we want to iterate 10,000 times, `B = 10000`. And suppose `a`, `b`, `c`, and `d` have already been defined

```
mu <- matrix(0, B)
sigma2 <- matrix(1, B)
alpha <- matrix(1, B)

n <- length(Y)
y_bar <- mean(Y)

mu[1] = mean(Y)
sigma2[1] = rinvgamma(1, c, d)
alpha = rinvgamma(1, a, b)

for(i in 2:B){

mu[i] <- rnorm(1, mean = y_bar, sd = sqrt(alpha[i-1] * sigma2[i-1] / n))
sigma2[i] <- rinvgamma(1, shape1 = c + n/2, shape2 = d + sum((Y - mu[i-1])^2) / (2 * alpha[i-1]))
alpha[i] <- rinvgamma(1, shape1 = a + n/2, shape2 = b + sum((Y - mu[i-1])^2) / (2 * sigma2[i]))


}

theta <- matrix(c(mu, alpha, sigma2), ncol = 3)
```

2

$\pi(\mu)\propto 1$

$\pi(\sigma^2)\propto(\sigma^2)^{-(\nu/2+1)}\exp(\frac{\nu/\alpha}{\sigma^2})$

$\pi(\alpha)\propto\alpha^{-(1/2+1)}\exp(-\frac{1/A^2}{\alpha})$

$\mathcal{L}(x|\mu,\sigma^2,\alpha)\propto(\sigma^2)^{-n/2}\exp(-\frac{\sum(x_i-\mu)^2}{2\sigma^2})$

$$\mathcal{L}(x|\mu,\sigma^2,\alpha)\pi(\alpha)\pi(\sigma^2)\pi(\mu)\propto(\sigma^2)^{-n/2}\exp\left(-\frac{\sum(x_i-\mu)^2}{2\sigma^2}\right)\alpha^{-(1/2+1)}\exp\left(-\frac{1/A^2}{\alpha}\right)(\sigma^2)^{-(\nu/2+1)}\exp\left(\frac{\nu/\alpha}{\sigma^2}\right)$$

$$P(\mu|x, \sigma^2, \alpha) \propto \exp\left(-\frac{\sum(x_i - \mu)^2}{2\sigma^2}\right)$$

$$= \exp\left(-\frac{1}{2\sigma^2/n}(n-1)s^2\right)\exp\left(-\frac{1}{2(\sigma^2/n)}(\mu - \bar{x})^2\right)$$

$$\propto \exp\left(-\frac{1}{2(\sigma^2/n)}(\mu - \bar{x})^2\right)$$

$$\therefore \mu \sim N(\bar{x}, \sigma^2/n)$$

$$P(\sigma^2|\mu, x, \alpha) \propto (\sigma^2)^{-n/2}\exp\left(-\frac{\sum(x_i - \mu)^2}{2\sigma^2}\right)(\sigma^2)^{-(\nu/2+1)}\exp\left(\frac{\nu/\alpha}{\sigma^2}\right)$$

$$= (\sigma^2)^{-((n+\nu)/2+1)}\exp\left(-\frac{(1/2)\sum(x_i - \mu)^2 + \nu/\alpha}{\sigma^2}\right)$$

$$\therefore \sigma^2 \sim IG\left(\frac{n+\nu}{2}, \frac{1}{2}\sum(x_i - \mu)^2 + \nu/\alpha\right)$$

$$P(\alpha|\sigma^2, \mu, x) \propto \alpha^{-(1/2+1)}\exp\left(-\frac{1/A^2}{\alpha}\right)\exp\left(\frac{\nu/\alpha}{\sigma^2}\right)$$

$$= \alpha^{-(1/2+1)}\exp\left(-\frac{1/A^2}{\alpha}\right)\exp\left(\frac{\nu/\sigma^2}{\alpha}\right)$$

$$= \alpha^{-(1/2+1)}\exp\left(\frac{\frac{\nu}{\sigma^2} + \frac{1}{A^2}}{\alpha}\right)$$

$$\therefore \alpha \sim IG\left(\frac{1}{2}, \frac{\nu}{\sigma^2} + \frac{1}{A^2}\right)$$

Now the Gibbs Sampler:

Let's say we want to iterate 10,000 times, `B = 10000`. And suppose `nu` and `A` have already been defined

```
mu <- matrix(0, B)
sigma2 <- matrix(1, B)
alpha <- matrix(1, B)

n <- length(X)
x_bar <- mean(X)

mu[1] = mean(Y)
alpha[1] = rinvgamma(1, 1/2, 1/A^2)
sigma2[1] = rinvgamma(1, nu / 2, nu/alpha[1])


for(i in 2:B){

mu[i] <- rnorm(1, mean = x_bar, sd = sqrt(sigma2[i-1] / n))
alpha[i] <- rinvgamma(1, shape1 = 1/2, shape2 = nu/sigma2[i-1] + 1/A^2)
sigma2[i] <- rinvgamma(1, shape1 = (n + nu) / 2,
shape2 = (1/2) * sum((X - mu[i-1])^2) + nu / alpha[i-1])
```
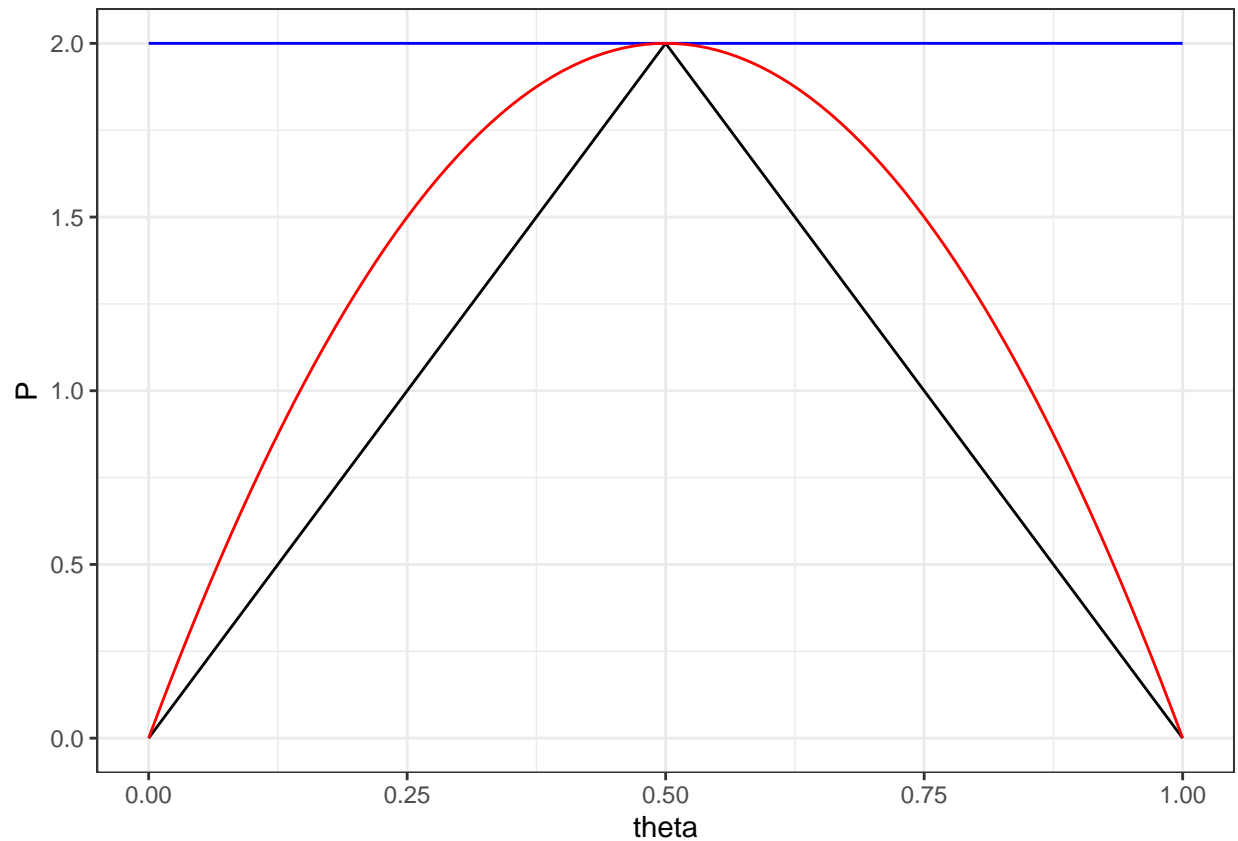
```
}

theta <- matrix(c(mu, alpha, sigma2), ncol = 3)
```

Computing Exercises

1

For the Uniform problem, set $M = 2$. For the beta, set $M = 4/3$:

```
preview <- data.frame(theta = seq(0, 1, 0.01)) %>%
  mutate(P = case_when(
    theta < 1/2 ~ 4 * theta,
    theta >= 1/2 ~ 4 - 4 * theta
  ),
  U = dunif(theta),
  B = dbeta(theta, shape1 = 2, shape2 = 2))

ggplot(preview, aes(x = theta)) +
  geom_line(aes(y = P)) +
  geom_line(aes(y = 2*U), col = "blue") +
  geom_line(aes(y = 4/3 * B), col = "red") +
  theme_bw()
```

Uniform Density:

```r
p_theta <- function(x){
  case_when(
    x < 1/2 ~ 4 * x,
    x >= 1/2 ~ 4 - 4 * x
  )
}



M <- 2
B <- 10000
theta <- vector(length = B)

b       <- 1
count   <- 1

set.seed(52918)
while(b < (B + 1)){
  ## step 1 ##
  tb    <- runif(1) # proposal
  U    <- runif(1) # always U(0,1)

  ## step 2 ##
  r <- p_theta(tb)/(M*dunif(tb)) # importance ratio scaled by 1/M
  if(U < r){
```

```
    theta[b]      <- tb
    b                 <- b + 1
  }
  count    <- count + 1
}

b/count
```
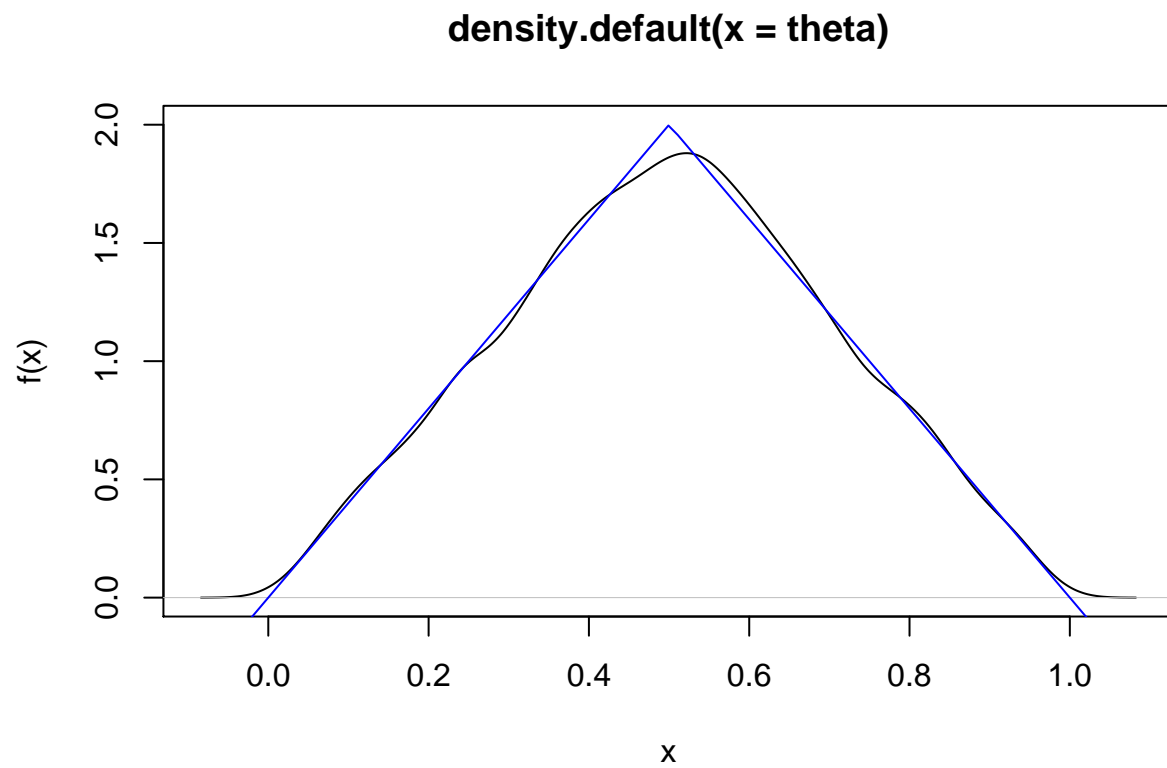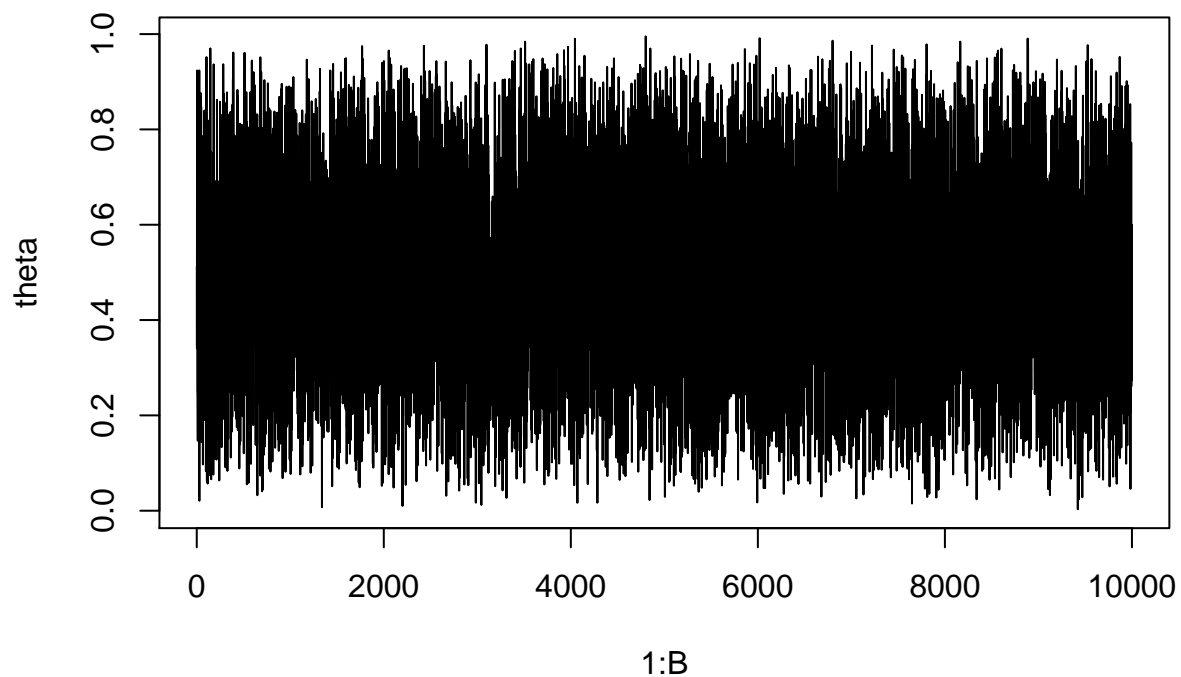
```
## [1] 0.4998501
```

```
plot(density(theta), ylab = 'f(x)', xlab = 'x', ylim = c(0,2))
curve(p_theta, add = TRUE, col = 'blue')
```

density.default(x = theta)


```
plot(1:B, theta, type = 'l')
```

Beta(2,2) Density:

```r
p_theta <- function(x){
  case_when(
    x < 1/2 ~ 4 * x,
    x >= 1/2 ~ 4 - 4 * x
  )
}
M <- 4/3
B <- 10000
theta <- vector(length = B)


b        <- 1
count    <- 1

set.seed(52918)
while(b < (B + 1)){
  ## step 1 ##
  tb    <- rbeta(1, 2, 2) # proposal
  U   <- runif(1) # always U(0,1)

  ## step 2 ##
  r <- p_theta(tb)/(M*dbeta(tb, 2, 2)) # importance ratio scaled by 1/M
  if(U < r){
    theta[b]    <- tb
```

```
    b                <- b + 1
  }
  count   <- count + 1
}

b/count
```
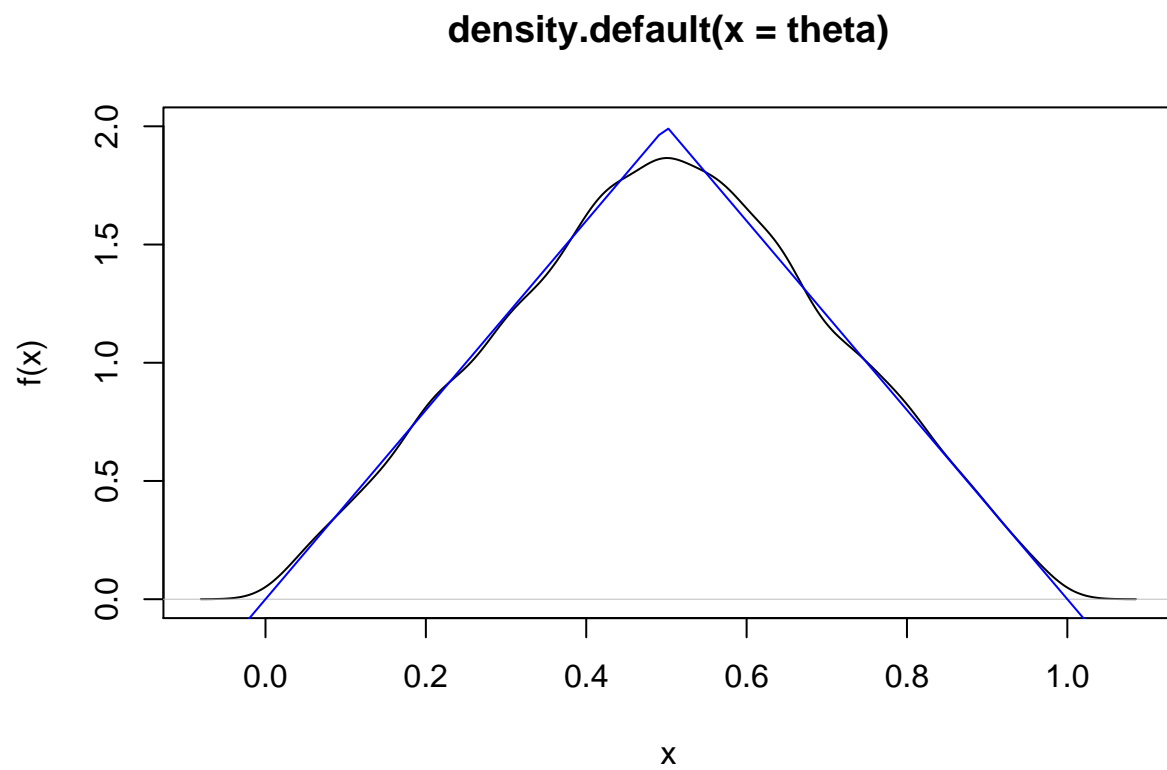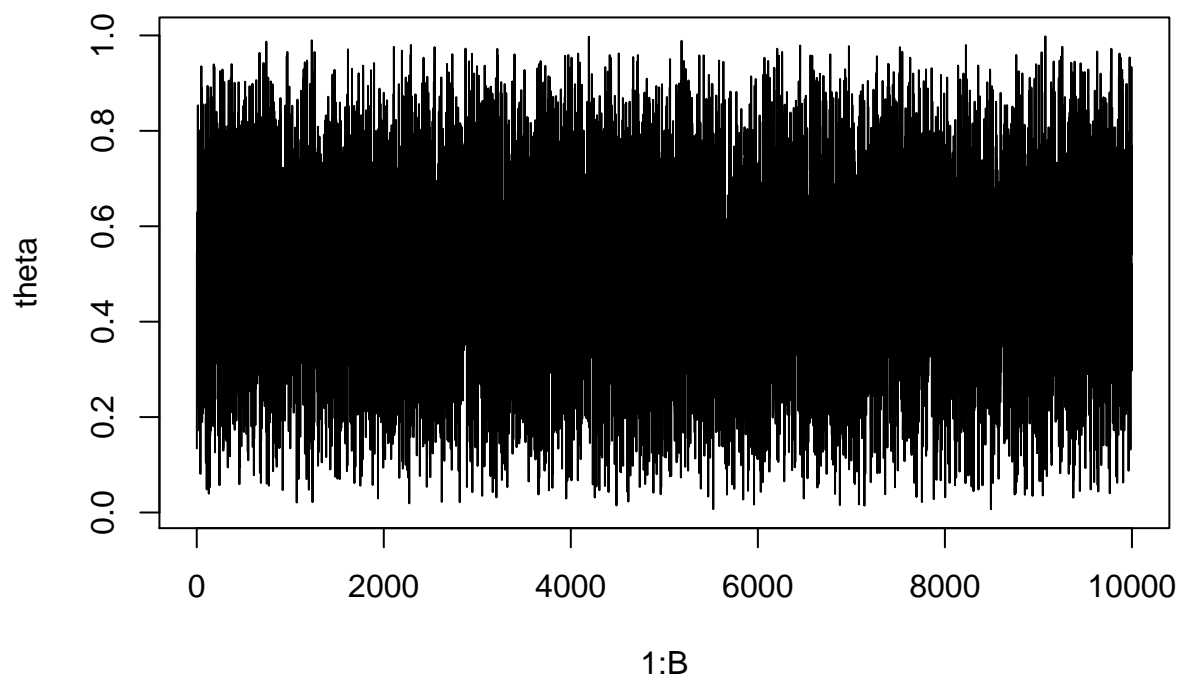
```
## [1] 0.7453421
```

```
plot(density(theta), ylab = 'f(x)', xlab = 'x', ylim = c(0,2))
curve(p_theta, add = TRUE, col = 'blue')
```

density.default(x = theta)


```
plot(1:B, theta, type = 'l')
```

Both options capture the underlying pretty similarly, however the beta proposal distribution has an acceptance rate roughly 50% higher, 75% to 50%. This makes intuituve sense as the beta distribution is more closely shaped to the target distribution than a blanket uniform distribution.

Analysis

```
hers    <- read.table('hersreg.txt', header = TRUE)

params <- hers %>% group_by(treatment)  %>%
  summarise(df = n() -1,
            avg = mean(chtchol),
            med = median(chtchol))
```

1.

Acceptance ratio: $\frac{P(\mu|X)}{Mg(\mu)}$

Using the pdf in Gelman:

$$P(\mu|X) \propto \mathcal{L}(X|\mu)\pi(\mu)$$

$$\propto \prod \left(1 + \frac{1}{\nu}\frac{(x_i - \mu)^2}{\sigma^2}\right)^{-(\nu+1)/2}$$

$$\log \mathcal{L} \propto \sum -\frac{(\nu+1)}{2}\log\left(1 + \frac{1}{\nu}\frac{(x_i - \mu)^2}{\sigma^2}\right)$$

This is not a distribtion with a closed solution. So we must define a function that calculates the log-likelihood:

Define Function to calc the log posterior for μ:

```r
P_t <- function(x, Y = hers[hers$treatment == 0, 1]){
  nu <- length(Y) - 1
  sigma <- sd(Y)
  P <- -((nu+1)/2)*log(1 + (1/nu) * (Y - x)^2 / sigma)
  return(sum(P))
}
```

Now utilize rejection sampling

Placebo Group:

```r
mus <- seq(-30, 10, .1)
P_mus <- sapply(mus, P_t)

log_M        <- max(
  P_mus - log(dcauchy(seq(-30, 10, .1), location = params$med[1], scale = 1))
    )


B      <- 10000
theta   <- vector(length = B)
b         <- 1
count    <- 1

set.seed(50814)
while(b < (B + 1)){
  ## step 1 ##
  tb     <- rcauchy(1, location = params$med[1], scale = 1) # proposal
  U    <- runif(1) # always U(0,1)

  ## step 2 ##
  r <- exp(
    P_t(x = tb) -  log_M - log(dcauchy(tb, location = params$med[1]))
  )
  if(U < r){
    theta[b]    <- tb
    b                <- b + 1
  }
  count    <- count + 1
  if(count >= 500000){break}
}

b/count
```
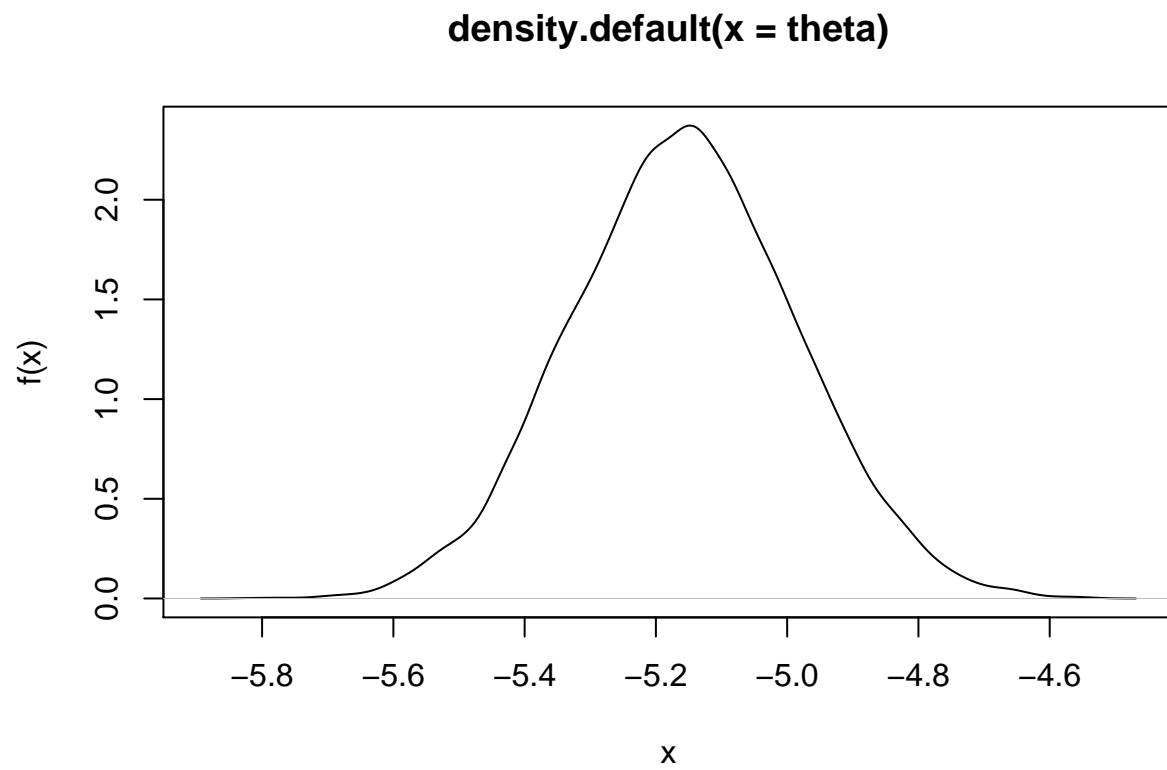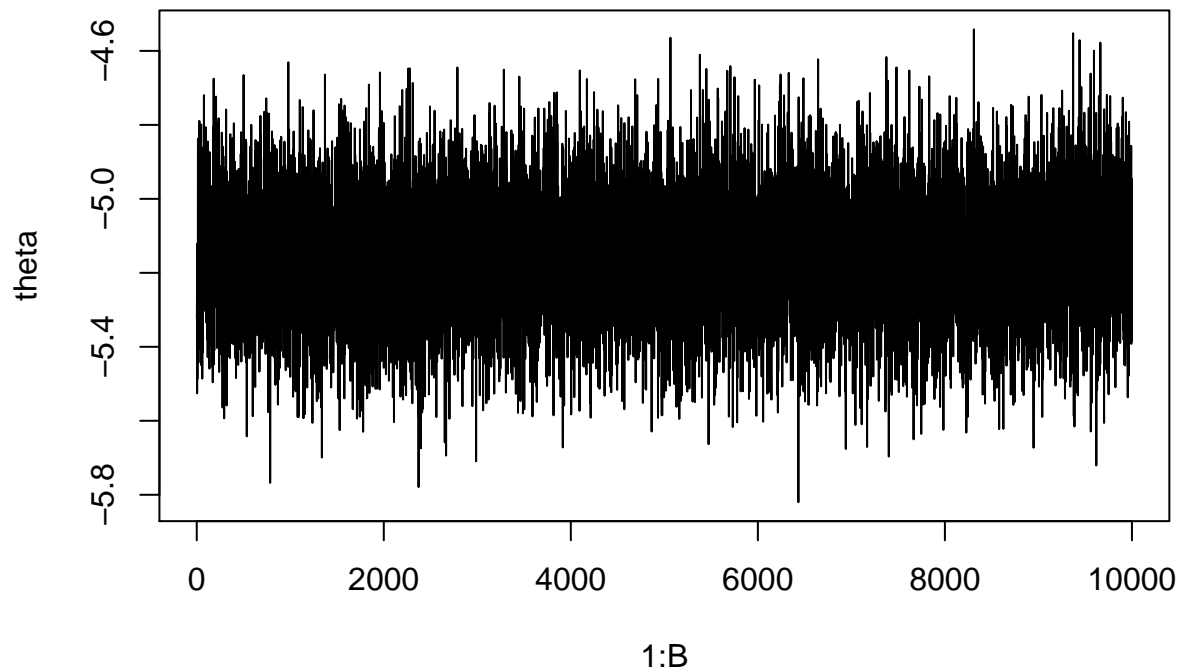
```
## [1] 0.1350921
```

```
plot(density(theta), ylab = 'f(x)', xlab = 'x')
```

density.default(x = theta)


```
plot(1:B, theta, type = 'l')
```

```
theta_placebo <- theta
```

Treatment Group:

```
P_t <- function(x, Y = hers[hers$treatment == 1, 1]){
  nu <- length(Y) - 1
  sigma <- sd(Y)
  P <- -((nu+1)/2)*log(1 + (1/nu) * (Y - x)^2 / sigma)
  return(sum(P))
}

mus <- seq(-30, 10, .1)
P_mus <- sapply(mus, P_t)

log_M        <- max(
  P_mus - log(dcauchy(seq(-30, 10, .1), location = params$med[2], scale = 1))
    )

# data.frame(cauchy = log_M + log(dcauchy(seq(-30, 10, .1), location = params$med[2], scale = 1)),
#            P_t = P_mus,
#            x = seq(-30, 10, .1)) %>%
#   ggplot(aes(x = x)) +
#   geom_line(aes(y = cauchy)) +
#   geom_line(aes(y = P_t), col = "blue", linetype = 2)
```

```r
B       <- 10000
theta   <- vector(length = B)
b         <- 1
count   <- 1

set.seed(50814)
while(b < (B + 1)){
  ## step 1 ##
  tb    <- rcauchy(1, location = params$med[2], scale = 1) # proposal
  U   <- runif(1) # always U(0,1)

  ## step 2 ##
  r <- exp(
    P_t(x = tb) -  log_M - log(dcauchy(tb, location = params$med[2]))
  )
  if(U < r){
    theta[b]    <- tb
    b                <- b + 1
  }
  count   <- count + 1
  if(count >= 500000){break}
}

b/count
```
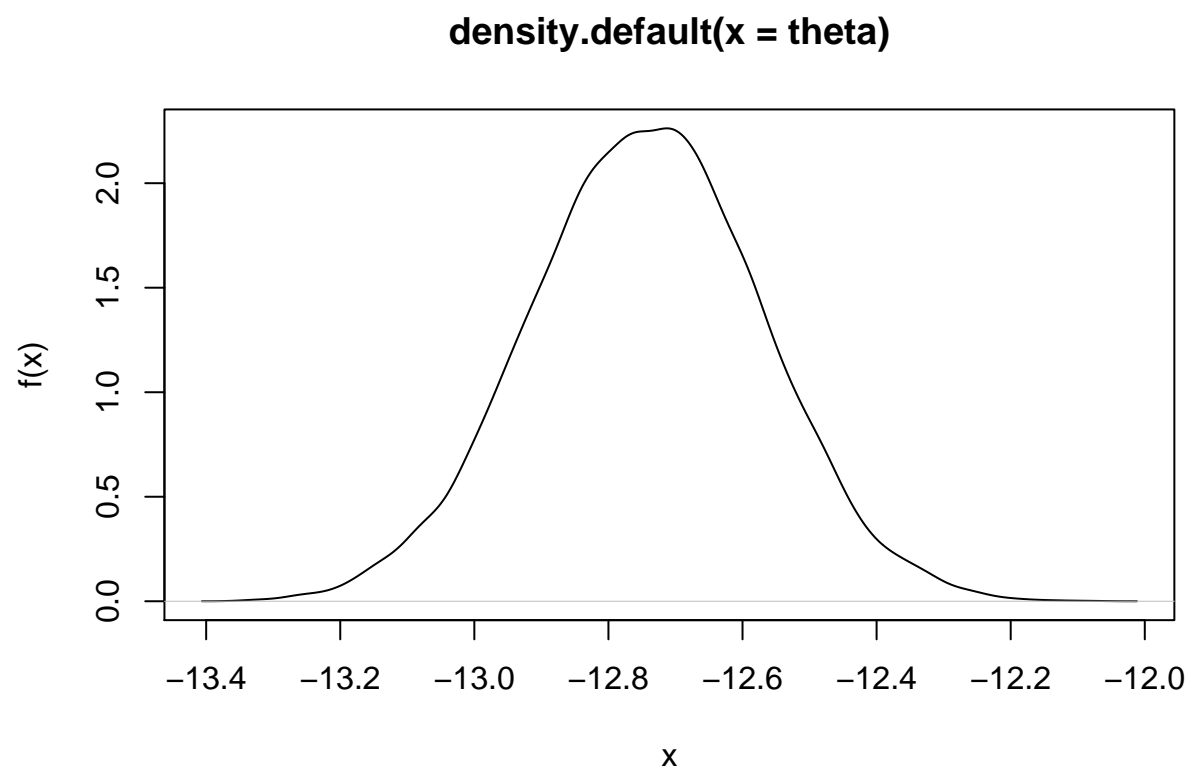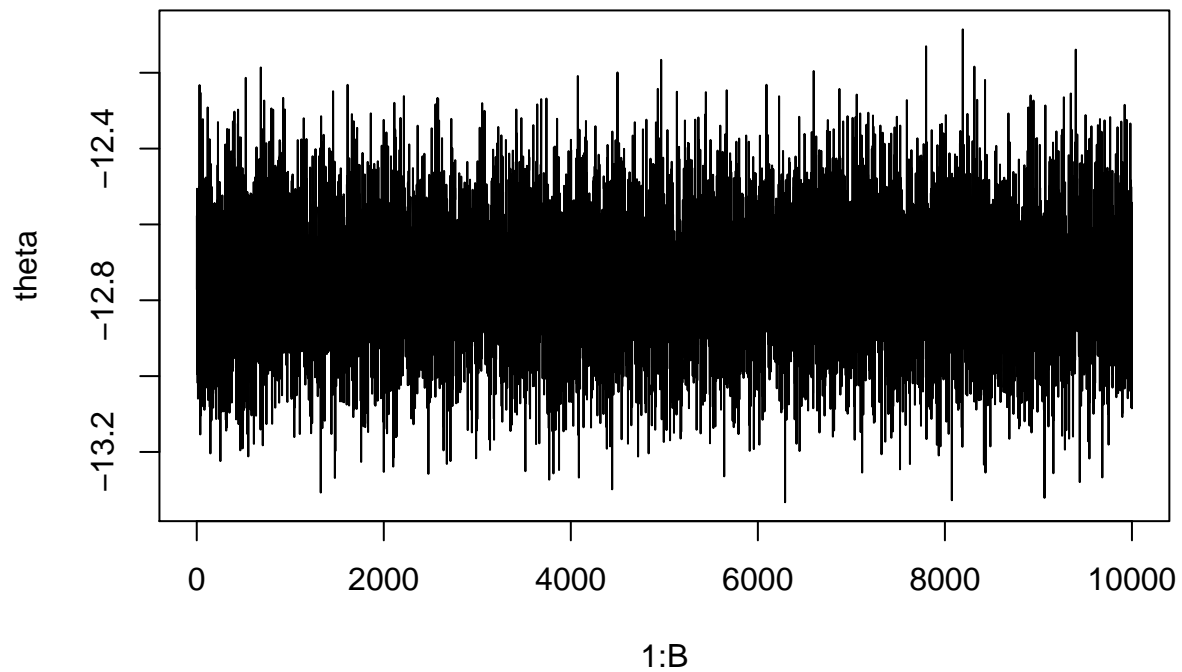
```
## [1] 0.1296642
```
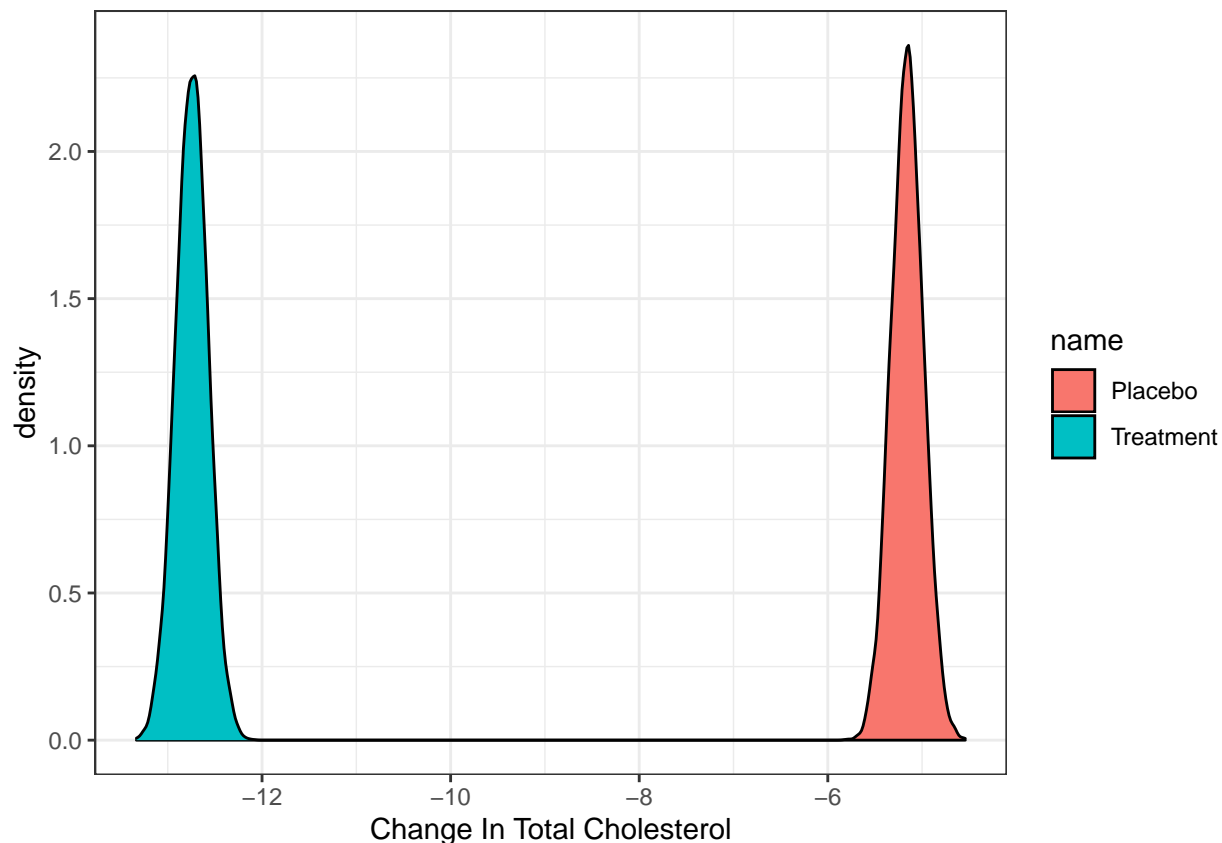
```r
plot(density(theta), ylab = 'f(x)', xlab = 'x')
```

density.default(x = theta)

```r
plot(1:B, theta, type = 'l')
```

```
theta_treatment <- theta
```

```
theta <- data.frame(Treatment = theta_treatment,
                    Placebo = theta_placebo) %>%
  pivot_longer(cols = everything())

theta %>%
  ggplot() +
  geom_density(aes(x = value, fill = name)) +
  theme_bw() +
  labs(x = "Change In Total Cholesterol")
```

```
theta %>% group_by(name) %>%
  summarise(Q.025 = quantile(value, prob = 0.025),
            Median = median(value),
            Q97.5 = quantile(value, prob = 0.975),
            Avg = mean(value))
```

```
## # A tibble: 2 x 5
##   name       Q.025 Median  Q97.5    Avg
## * <chr>      <dbl>  <dbl>  <dbl>  <dbl>
## 1 Placebo    -5.49  -5.16  -4.82  -5.16
## 2 Treatment -13.1  -12.7  -12.4  -12.7
```

We see that it is a statistically significant difference in change of total cholesterol between the treatment group and the placebo group

Diagnostics:

We see that the Geweke Diagnostic is more than tolerable, with the absolute value under 1 for both. Additionally, the mcmc plots also pass the visual tests

```
library(mcmcplots)
mcmcplots::mcmcplot1(theta_placebo)
```

```
## Error in apply(mco, 2, function(y) cumsum(y)/seq_along(y)): dim(X) must have a positive length
```
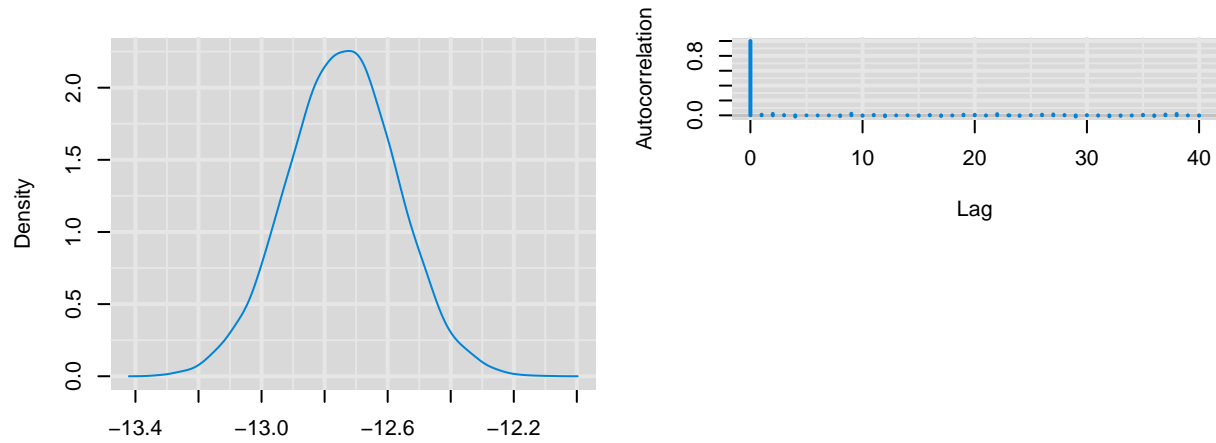
16
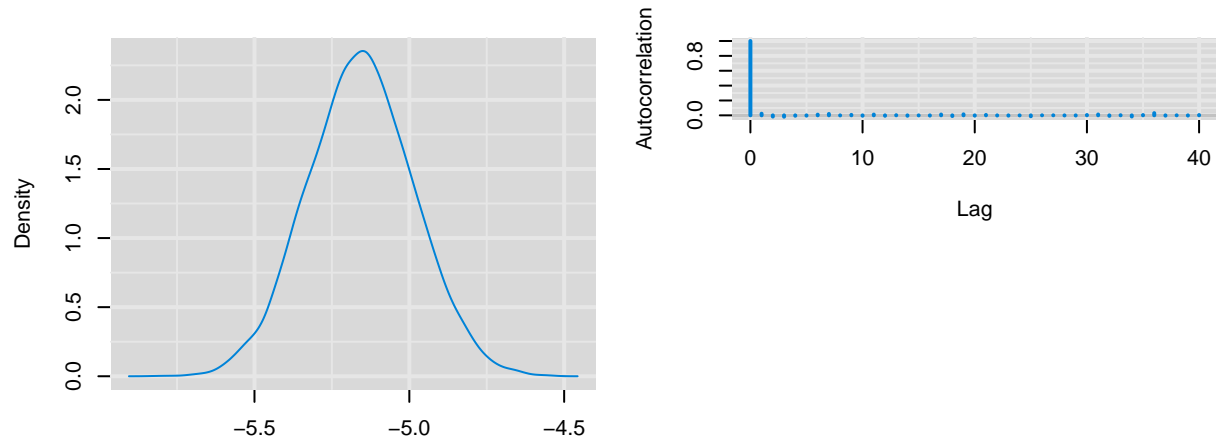
28383088, −5.18238263271404, −5.17954698159721, −5.176711:

```r
coda::geweke.diag(theta_placebo)
```

```
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
##
##    var1
## 0.06315
```

```r
mcmcplots::mcmcplot1(theta_treatment)
```

```
## Error in apply(mco, 2, function(y) cumsum(y)/seq_along(y)): dim(X) must have a positive length
```

570148, −12.7104565198665, −12.7076713827182, −12.7048862

```
coda::geweke.diag(theta_treatment)
```

```
##
## Fraction in 1st window = 0.1
## Fraction in 2nd window = 0.5
##
## 	   var1
## -0.3255
```