

# Homework 6, STA 360

Aakash Kothapally

I used the code given to me in class below.

```
set.seed(5983)
knitr::opts_chunk$set(cache=FALSE)
library(xtable)
# Samples from a truncated gamma with
# truncation (t, infty), shape a, and rate b
# Input: t,a,b
# Output: truncated Gamma(a,b)
sampleTrunGamma <- function(t, a, b){
  # This function samples from a truncated gamma with
# truncation (t, infty), shape a, and rate b
  p0 <- pgamma(t, shape = a, rate = b)
  x <- runif(1, min = p0, max = 1)
  y <- qgamma(x, shape = a, rate = b)
  return(y)
}
# Gibbs sampler for censored data
# Inputs:
# this function is a Gibbs sampler
# z is the fully observe data
# c is censored data
# n.iter is number of iterations
# init.theta and init.miss are initial values for sampler
# r,a, and b are parameters
# burnin is number of iterations to use as burnin
# Output: theta, z
sampleGibbs <- function(z, c, n.iter, init.theta, init.miss, r, a, b, burnin = 1){
  z.sum <- sum(z)
  m <- length(c)
  n <- length(z) + m
  miss.vals <- init.miss
  res <- matrix(NA, nrow = n.iter, ncol = 1 + m)
  for (i in 1:n.iter){
    var.sum <- z.sum + sum(miss.vals)
    theta <- rgamma(1, shape = a + n*r, rate = b + var.sum)
    miss.vals <- sapply(c, function(x) {sampleTrunGamma(x, r, theta)})
    res[i,] <- c(theta, miss.vals)
  }
  return(res[burnin:n.iter,])
}
```

2a. We produce trace plots and running average plots for theta and the censored values for 200 iterations. First, we run the Gibbs sampler with 200 iterations.

```
# set parameter values
r <- 10
a <- 1
b <- 1
# input data
z <- c(3.4,2.9,1.4,3.2,1.8,4.6,2.8)
c <- c(1.2,1.7,2.0,1.4,0.6)
n.iter <- 200
init.theta <- 1
init.missing <- rgamma(length(c), shape = r, rate = init.theta)
# run sampler
res <- sampleGibbs(z, c, n.iter, init.theta, init.missing, r, a, b)
```

Then, we produce the trace plots.

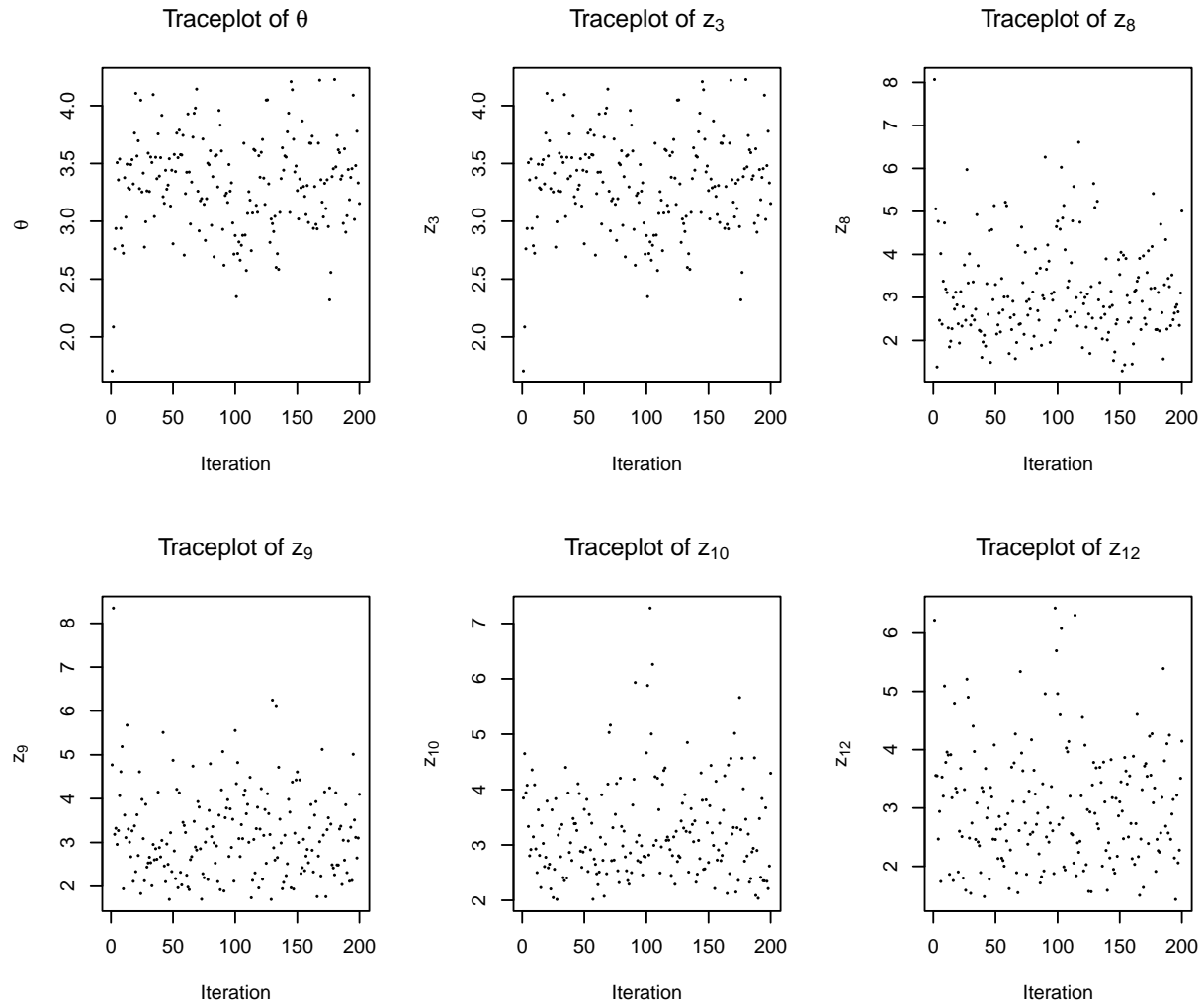


Figure 1: Traceplot of theta,  $z_3$ ,  $z_8$ ,  $z_9$ ,  $z_{10}$ ,  $z_{12}$ .

Although the trace plots show that there is a random mix of points, it is not clear whether the sampler is eventually able to converge. We can use the running average plots to help us with this.

```
# get running averages
run.avg <- apply(res, 2, cumsum)/(1:n.iter)
```

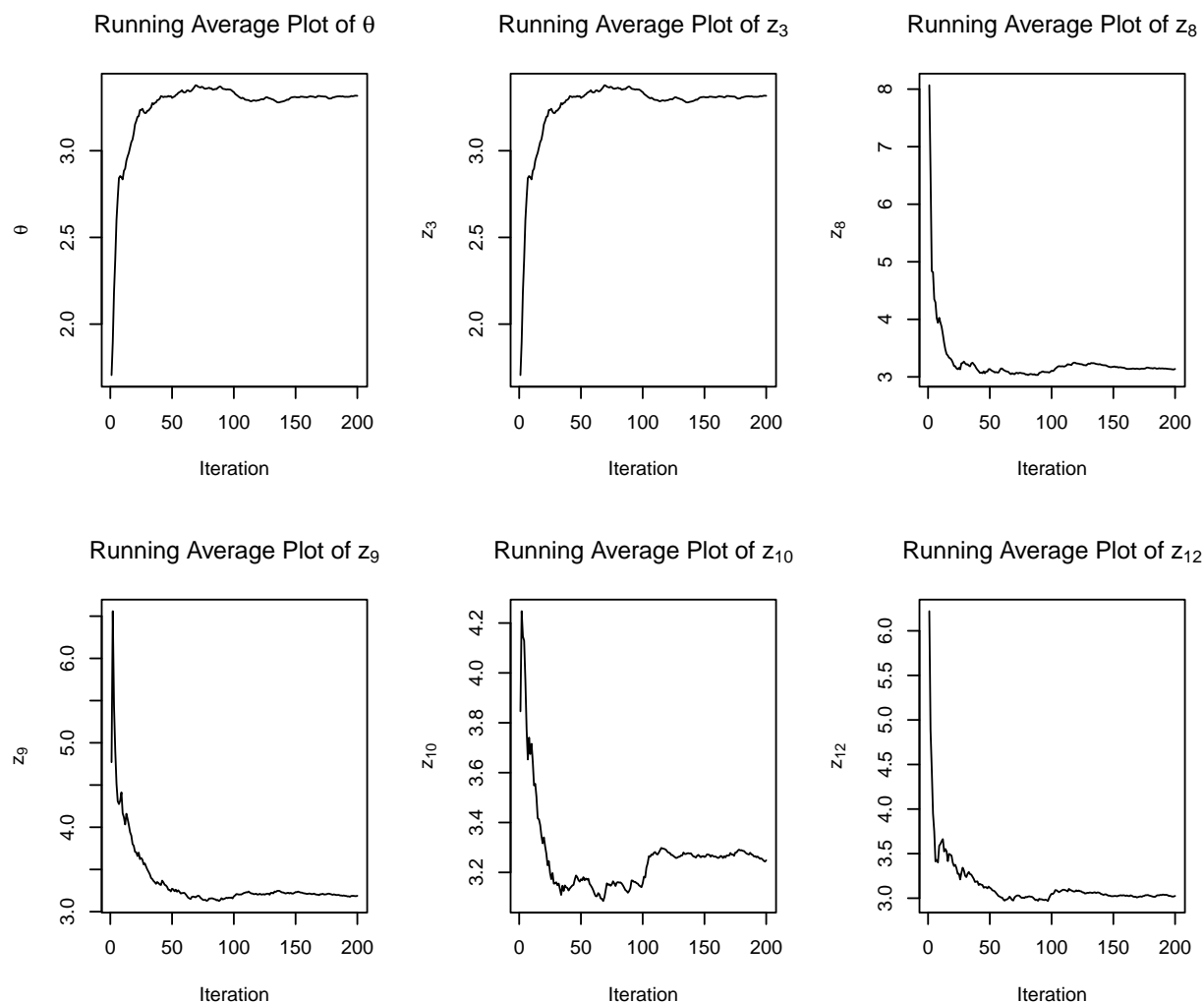


Figure 2: Running average plots of  $\theta$ ,  $z_3$ ,  $z_8$ ,  $z_9$ ,  $z_{10}$ ,  $z_{12}$ .

We can see for  $\theta$  and censored values, the running average does not converge onto a value for any of them. In other words, the sampler has mixing issues and needs to be run for longer to see if it will converge for that larger number of iterations. As a result, I do not believe these diagnostic plots suggest we have run the sampler long enough.

2b. We will now re-run the trace plots and running average plots for theta and the censored values, except we will use 10,000 iterations to see if this is enough runs to have a well-mixed sampler.

```
# set parameter values
r <- 10
a <- 1
b <- 1
# input data
z <- c(3.4,2.9,1.4,3.2,1.8,4.6,2.8)
c <- c(1.2,1.7,2.0,1.4,0.6)
n.iter <- 10000
init.theta <- 1
init.missing <- rgamma(length(c), shape = r, rate = init.theta)
# run sampler
res <- sampleGibbs(z, c, n.iter, init.theta, init.missing, r, a, b)
```

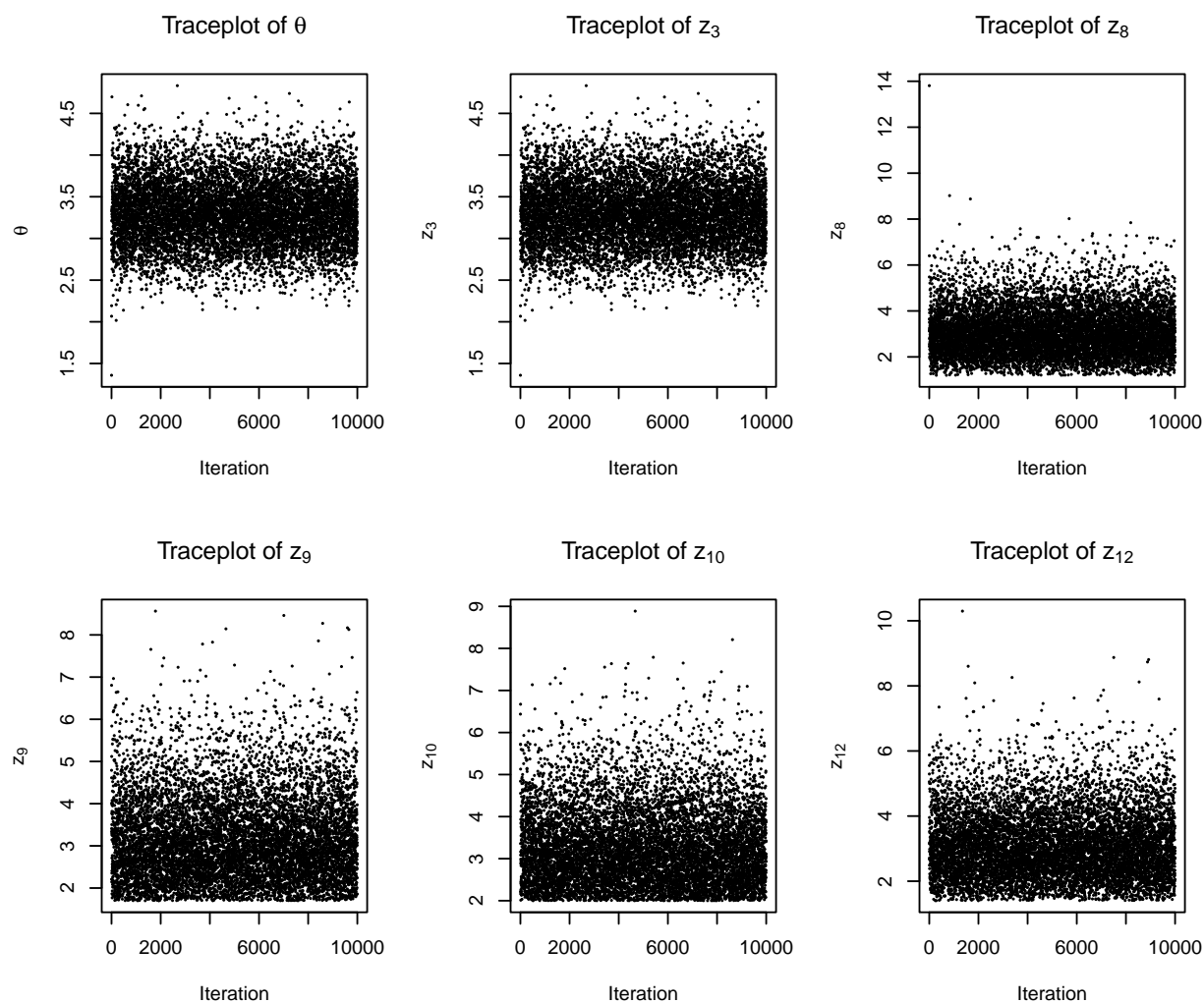


Figure 3: Traceplot of theta,  $z_3$ ,  $z_8$ ,  $z_9$ ,  $z_{10}$ ,  $z_{12}$ .

Already, this looks like a stronger case for a well-mixed sampler than the 200 iteration case, because we can

see the majority of points for each trace plot being within a certain range of values. However, this is not enough evidence to see if the values converged, so we need to look at the running average plots.

```
# get running averages
run.avg <- apply(res, 2, cumsum)/(1:n.iter)
```

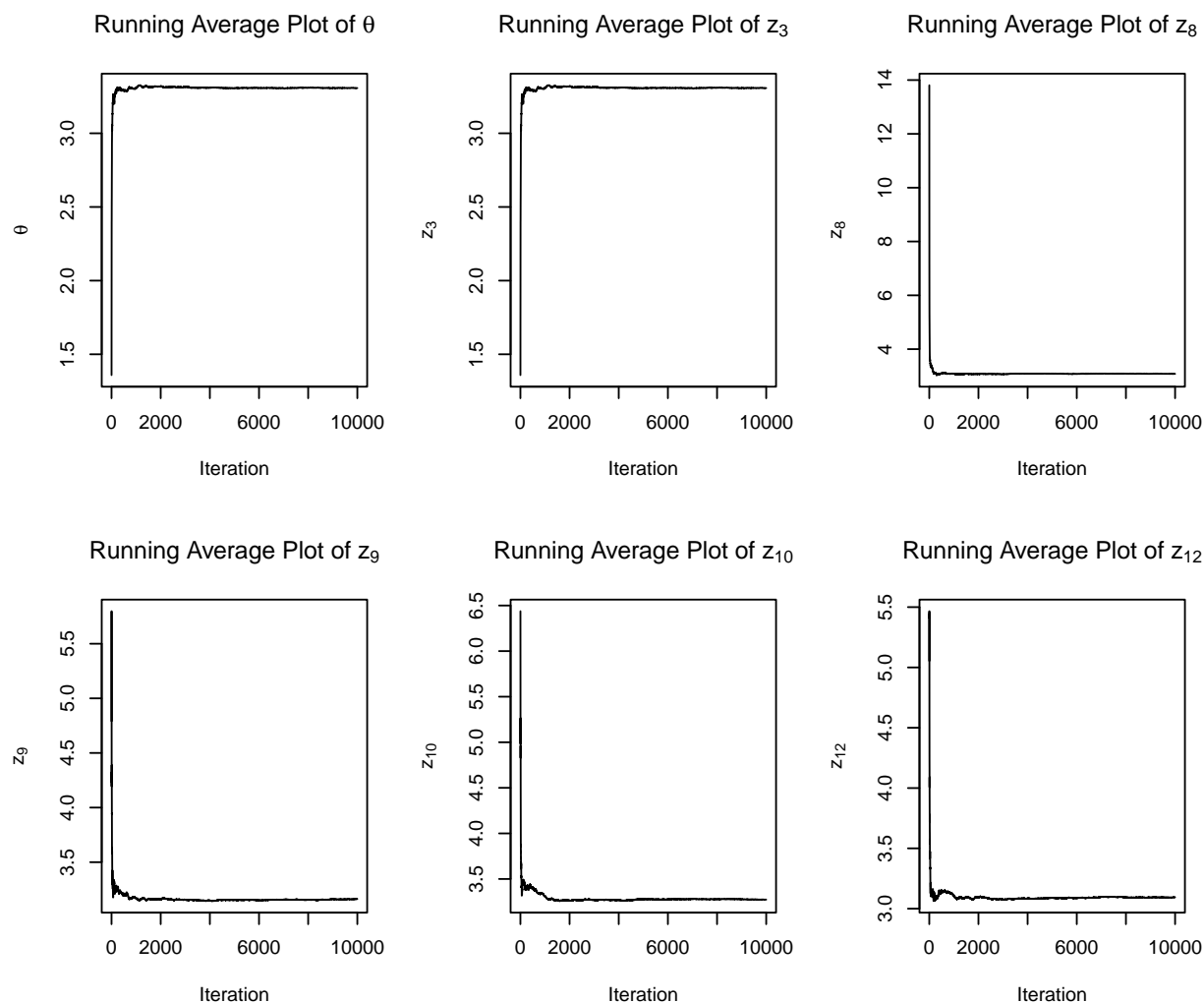


Figure 4: Running average plots of  $\theta$ ,  $z_3$ ,  $z_8$ ,  $z_9$ ,  $z_{10}$ ,  $z_{12}$ .

We can see for  $\theta$  and censored values, the running average converges better than the 200 iteration case. However, the running averages are only meandering by exceptionally small amounts. As a result, I believe these diagnostic plots suggest we have run the sampler long enough, although subject to interpretation, someone could determine they require more iterations. The results are better than the results for the 200 iteration case.

2c. These are the plots of the estimated density for theta and  $z_9$  (for 10,000 iterations).

We see that these plots are not as wonky or rough as the plots we saw for a lower iteration estimated density plot in the slides from class.

Specifically, theta appears to be roughly normally distributed around a posterior mean close to 3.3.

$z_9$  appears to be right-skewed, with a median lower than the posterior mean (close to 3.3).

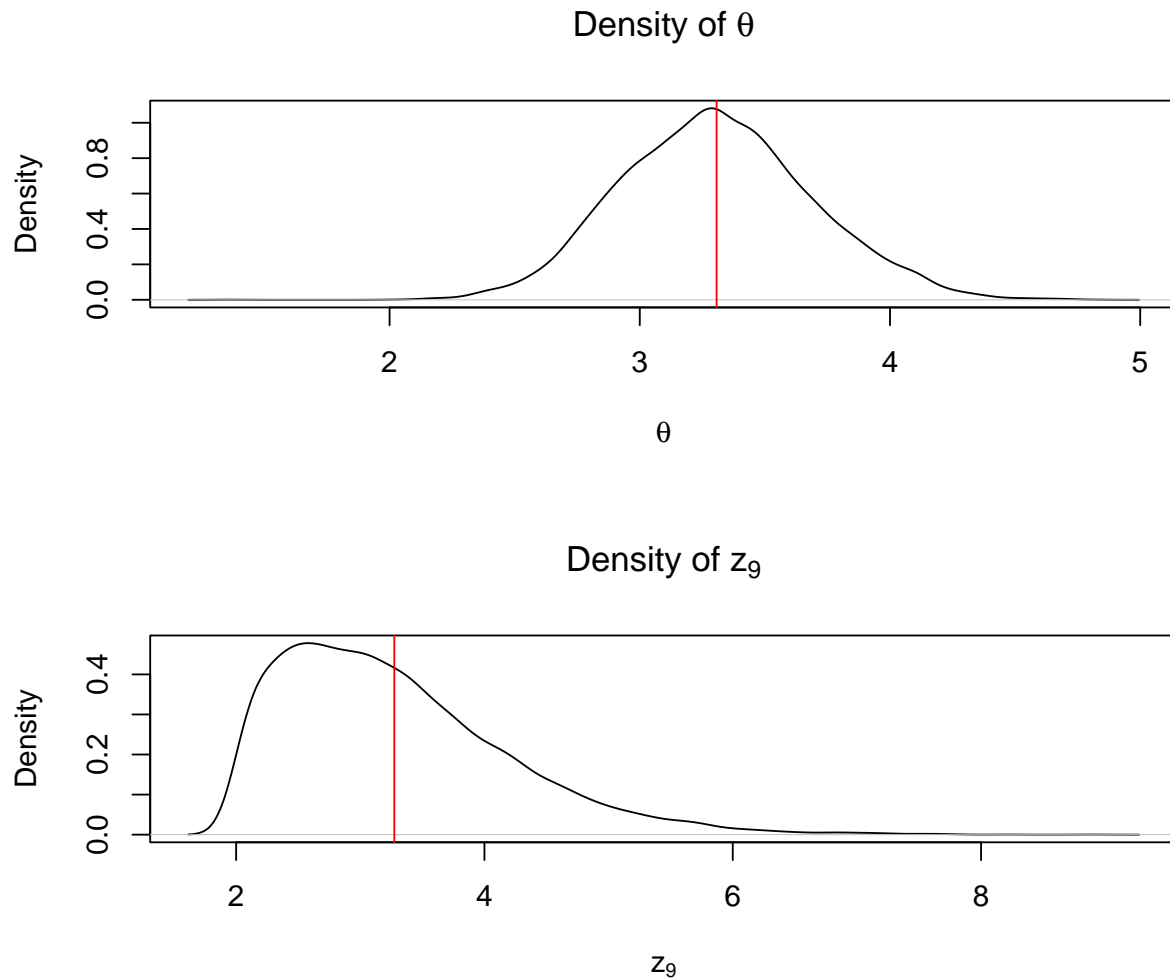


Figure 5: Estimated posterior density of theta,  $z_9$

2d. We can consider the posterior densities when altering the a and b values.

We expect the theta and z9 estimated density plots to have roughly the same shape when altering the parameters, but we also expect the posterior mean for the plots to be altered for both cases. The posterior densities will change for both cases.

First, we consider  $r = 10$ ,  $a = 1$ ,  $b = 100$ .

```
# set parameter values
r <- 10
a <- 1
b <- 100
# input data
z <- c(3.4,2.9,1.4,3.2,1.8,4.6,2.8)
c <- c(1.2,1.7,2.0,1.4,0.6)
n.iter <- 10000
init.theta <- 1
init.missing <- rgamma(length(c), shape = r, rate = init.theta)
# run sampler
res <- sampleGibbs(z, c, n.iter, init.theta, init.missing, r, a, b)
```

Next, we consider  $r = 10$ ,  $a = 100$ ,  $b = 1$ .

```
# set parameter values
r <- 10
a <- 100
b <- 1
# input data
z <- c(3.4,2.9,1.4,3.2,1.8,4.6,2.8)
c <- c(1.2,1.7,2.0,1.4,0.6)
n.iter <- 10000
init.theta <- 1
init.missing <- rgamma(length(c), shape = r, rate = init.theta)
# run sampler
res <- sampleGibbs(z, c, n.iter, init.theta, init.missing, r, a, b)
```

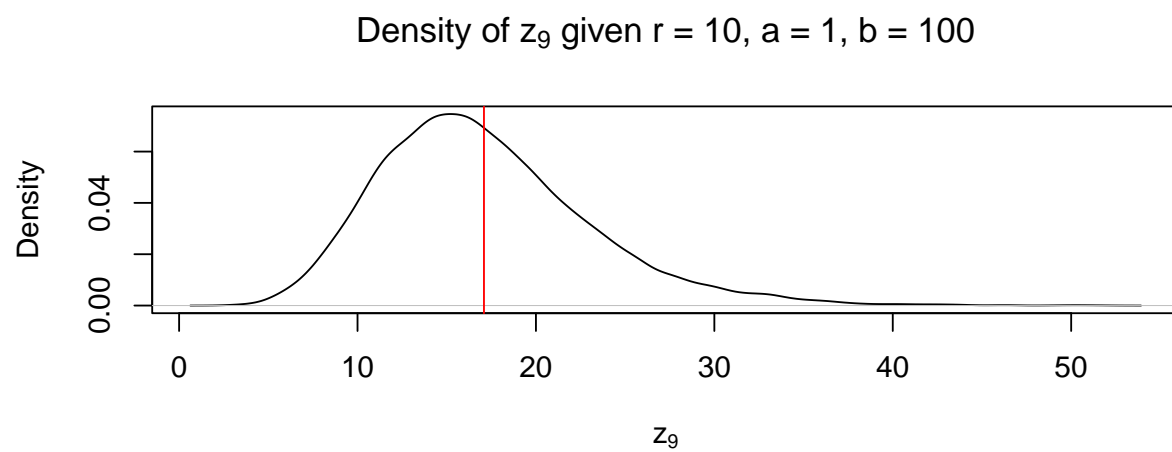
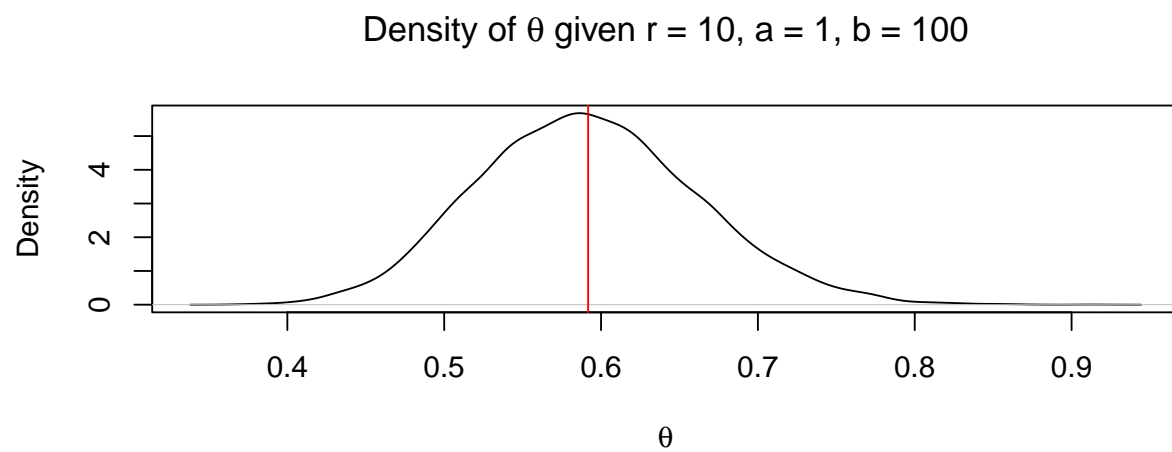


Figure 6: Estimated posterior density of  $\theta$ ,  $z_9$



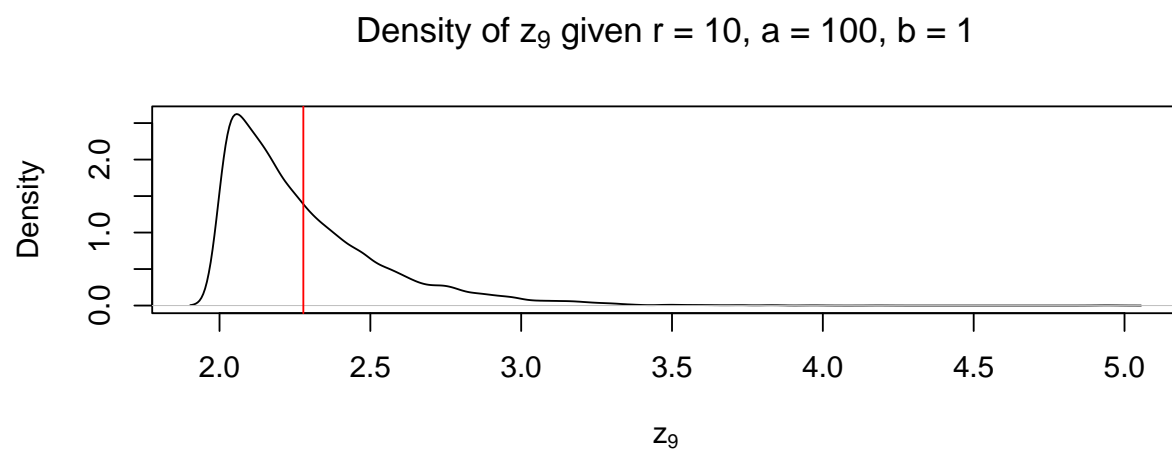
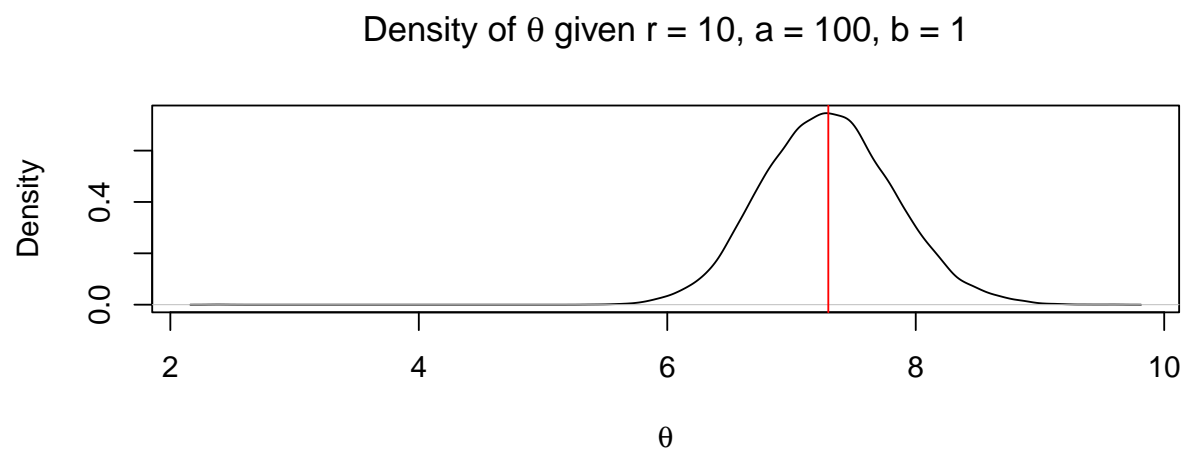


Figure 7: Estimated posterior density of  $\theta$ ,  $z_9$