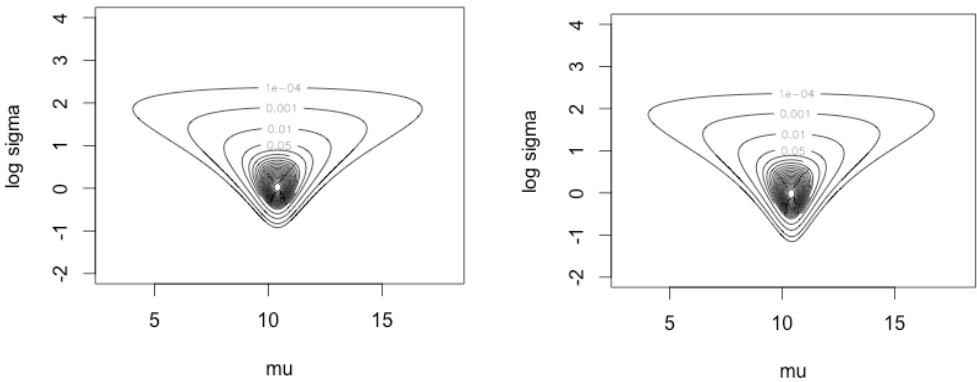


3.5  
(a)

(b)

(c)



			2.5%	25%	50%	75%	97.5%
[1,]	10.426601	0.6933751	9.1517278	10.054718	10.412948	10.807167	11.766502
[2,]	1.412234	0.6710720	0.6724254	0.986343	1.249111	1.641871	2.929167
			2.5%	25%	50%	75%	97.5%
[1,]	10.411018	0.6716420	9.1055276	10.0100503	10.386935	10.763819	11.819095
[2,]	1.349437	0.6486137	0.6111198	0.9320659	1.186318	1.603778	3.020818

```

# 3.5c
post.a <- function(mu, sd, y){
  ldens <- 0
  for (i in 1:length(y)){
    ldens <- ldens + log(dnorm(y[i], mu, sd))
  }
  return(ldens)
}

post.b <- function(mu, sd, y){
  ldens <- 0
  for (i in 1:length(y)){
    ldens <- ldens + log(pnorm(y[i] + 0.5, mu, sd) - pnorm(y[i] - 0.5,
mu, sd))
  }
  return(ldens)
}

summ <- function(x){
  return(c(mean(x), sqrt(var(x)), quantile(x,
c(.025, .25, .5, .75, .975))))
}

#ignoring rounding
nsim <- 2000
y <- c(10, 10, 12, 11, 9)
n <- length(y)
ybar <- mean(y)
sample_variance <- sum((y - ybar) ^ 2) / (n - 1)
mugrid <- seq(3, 18, length=200)
logsdgrid <- seq(-2, 4, length=200)
contours <- c(.0001, .001, .01, seq(.05, .96, .05))
logdens <- outer(mugrid, exp(logsdgrid), post.a, y)
dens <- exp(logdens - max(logdens))
contour(mugrid, logsdgrid, dens, levels=contours, xlab='mu', ylab='log
sigma', title="Posterior Density, Ignoring Rounding", labelx=0, cex=2)
sd <- sqrt((n - 1) * sample_variance / rchisq(nsim, 4))
mu <- rnorm(nsim, ybar, sd / sqrt(n))
print(rbind(summ(mu), summ(sd)))

# Consider rounding
logdens <- outer(mugrid, exp(logsdgrid), post.b, y)
dens <- exp(logdens - max(logdens))
contour(mugrid, logsdgrid, dens, levels=contours, xlab='mu', ylab='log
sigma', title="Posterior Density, Considering Rounding", labelx=0, cex=2)
dens.mu <- apply(dens, 1, sum)
muindex <- sample(1:length(mugrid), nsim, replace = T, prob = dens.mu)
mu <- mugrid[muindex]
sd <- rep(NA, nsim)
for (i in 1:nsim) {
  sd[i] <- exp(sample(logsdgrid, 1, prob = dens[muindex[i],]))
}
print(rbind(summ(mu), summ(sd)))

```

```
(d)
> mean((z[, 1] - z[, 2]) ^ 2)
[1] 0.1610319
```

```
# 3.5d
z <- matrix(NA, nsim, length(y))
for (i in 1:length(y)){
  lower <- pnorm(y[i] - .5, mu, sd)
  upper <- pnorm(y[i] + .5, mu, sd)
  z[, i] <- qnorm(lower + runif(nsim) * (upper - lower), mu,
sd)
}
mean((z[, 1] - z[, 2]) ^ 2)
```