Bayesian Learning

Lab 1

Emil K Svensson and Rasmus Holm

2017-04-11

Question 1

In this assignment we are given data coming from a Bernoulli distribution with unknown parameter θ and we use the conjugate prior

$$\theta \sim \text{Beta}(\alpha_0, \beta_0)$$

where $\alpha_0 = \beta_0 = 2$. The sample we get consists of 14 successes out of 20 trials which gives us the posterior distribution

$$\theta|y \sim \text{Beta}(\alpha_0 + 14, \beta_0 + 6).$$

```
n <- 20
s <- 14
f <- n - s

alpha_prior <- 2
beta_prior <- 2
```

a

Here we are sampling from the posterior distribution to verify that as the number of samples increases, the mean and standard deviation converges to the true values. In this particular case the true mean is $\alpha/(\alpha + \beta) = 2/3$ and standard deviation is $\sqrt{(\alpha\beta)/((\alpha + \beta)^2(\alpha + \beta + 1))} = \sqrt{2/225}$.

```
drawSamples <- function(nDraws, alpha, beta) {
    rbeta(nDraws, shape1=alpha, shape2=beta)
}

nDraws <- 10000
alpha <- alpha_prior + s
beta <- beta_prior + f

set.seed(123)
samples <- drawSamples(nDraws, alpha, beta)

true_mean <- alpha / (alpha + beta)
estimated_mean <- mean(samples)

true_sd <- sqrt((alpha * beta) / ((alpha + beta)^2 * (alpha + beta + 1)))
estimated_sd <- sd(samples)

set.seed(123)
```
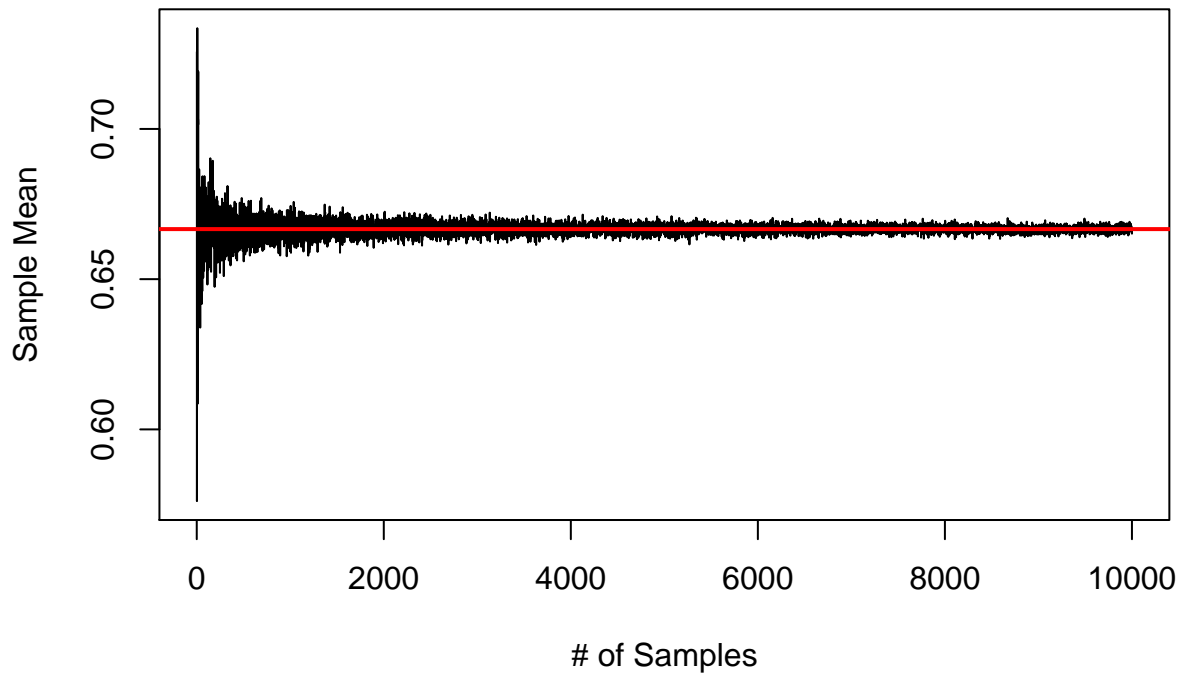
```r
betamean <- sapply(1:nDraws, FUN = function(X) {
    mean(drawSamples(nDraws = X, alpha = alpha , beta = beta))
})

set.seed(123)
betasd <- sapply(1:nDraws, FUN = function(X) {
    sd(drawSamples(nDraws = X, alpha = alpha , beta = beta))
})
```

```r
plot(x = 1:nDraws, y = betamean, type = "l",
     xlab="# of Samples", ylab="Sample Mean")
abline(h = true_mean, col = "red", lwd = 2)
```
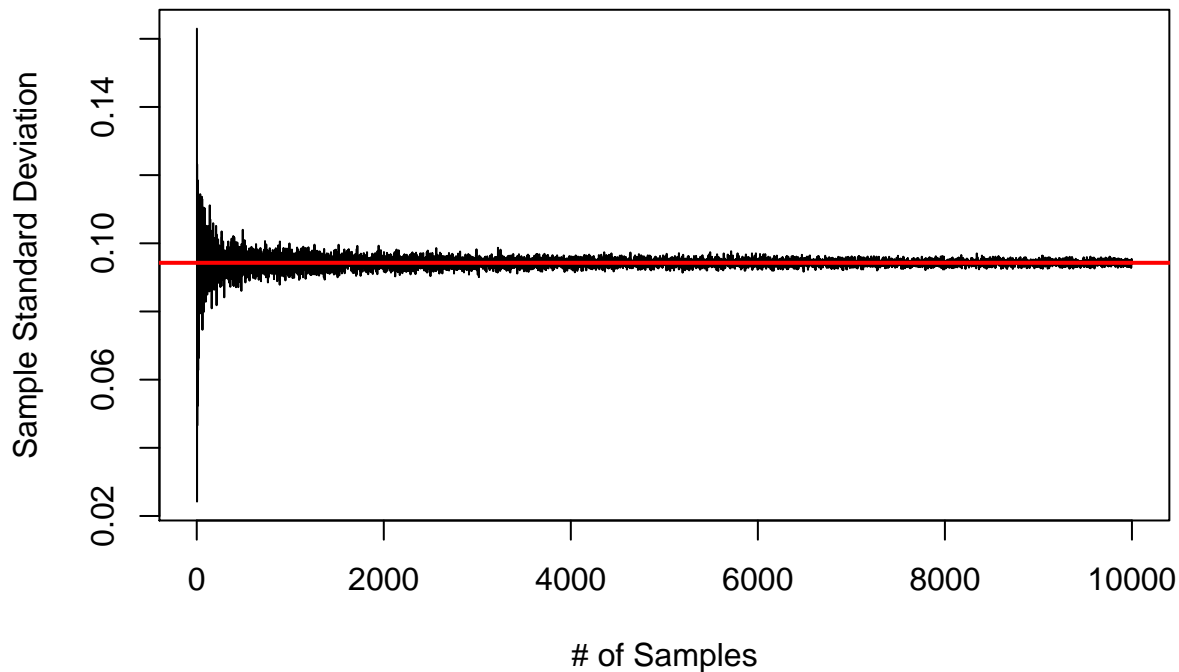

The plot above shows how the sample mean changes with the number of drawn samples. The red line is the true mean and we can see that as the sample increases it converges to the true mean.

```r
plot(x = 1:nDraws, y = betasd, type = "l",
     xlab="# of Samples", ylab="Sample Standard Deviation")
abline(h = true_sd, col = "red", lwd = 2)
```

Similar, the plot above shows how the standard deviation from the samples changes with the number of drawn samples. The red line shows the true standard deviation and we can see that the sample standard deviation converges to it.

b

```
est_04 <- sum(samples < 0.4) / length(samples)
true_04 <- pbeta(0.4, shape1 = alpha, shape2 = beta)
```

The posterior probability for theta being below 0.4 is calculated as the share of values under 0.4 in our drawn samples. We drew 10000 samples and got $\Pr(\theta < 0.4|y) \approx 0.0035$ while the true value is $\Pr(\theta < 0.4|y) = 0.0039727$ which is fairly close.
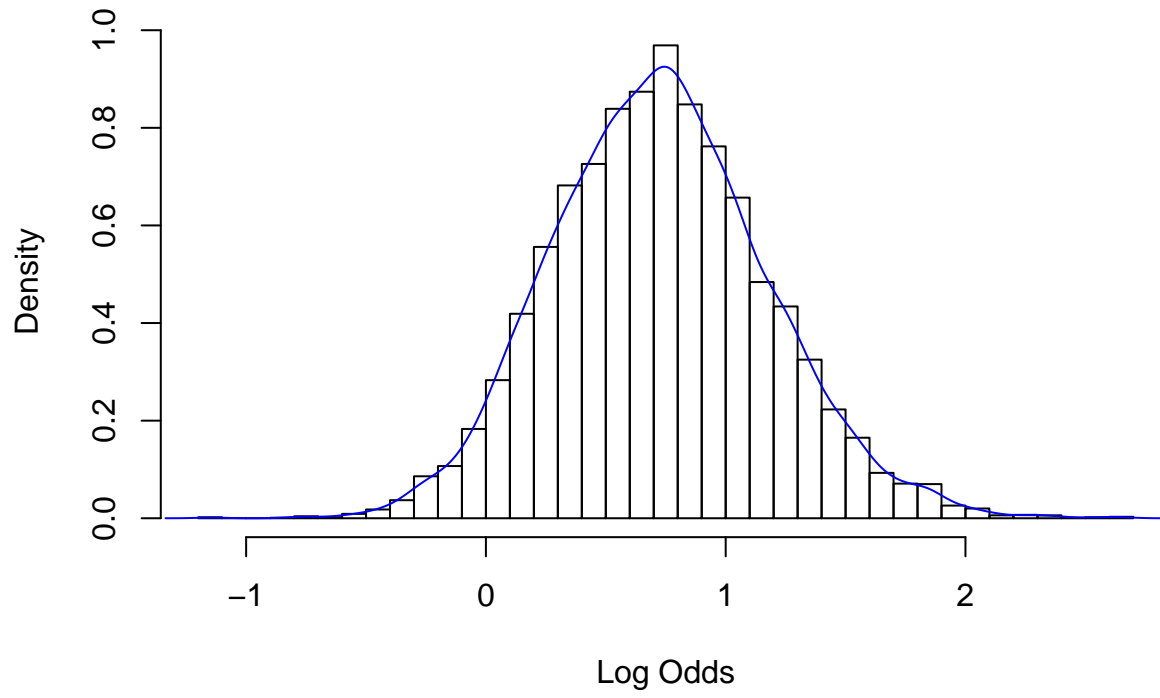
c

We can also use the samples to get the distribution of a function of those samples. In this case we take $\text{logit}(\theta) = \log(\theta/(1-\theta))$ and get the histogram below.

```
log_odds <- function(theta){
  log(theta / (1 - theta))
}

sample_odds <- log_odds(samples)

hist(sample_odds, breaks=50, freq = FALSE, main="Histogram of Log Odds", xlab="Log Odds")
lines(density(sample_odds), col = "blue")
```

Histogram of Log Odds

Question 2

a

```r
library(geoR)
sek <- c(14, 25, 45, 25, 30, 33, 19, 50, 34 ,67)

n <- length(sek)
mu <- 3.5
tausq <- sum((log(sek)- mu)^2) / n

set.seed(123)
chi_samples <- rinvchisq(nDraws, df = n, scale = tausq)

invchisq_pdf <- function(x, df, scale) {
    factor1 <- (scale * df / 2)^(df / 2) / gamma(df / 2)
    factor2 <- exp(-(df * scale) / (2 * x)) / x^(1 + df / 2)
    factor1 * factor2
}

xs <- seq(0.01, 2, 0.01)
ys <- invchisq_pdf(xs, 10, tausq)

hist(chi_samples, breaks = 100, freq=FALSE, xlim=c(0, 2))
lines(xs, ys, col="red", lwd=2)
```
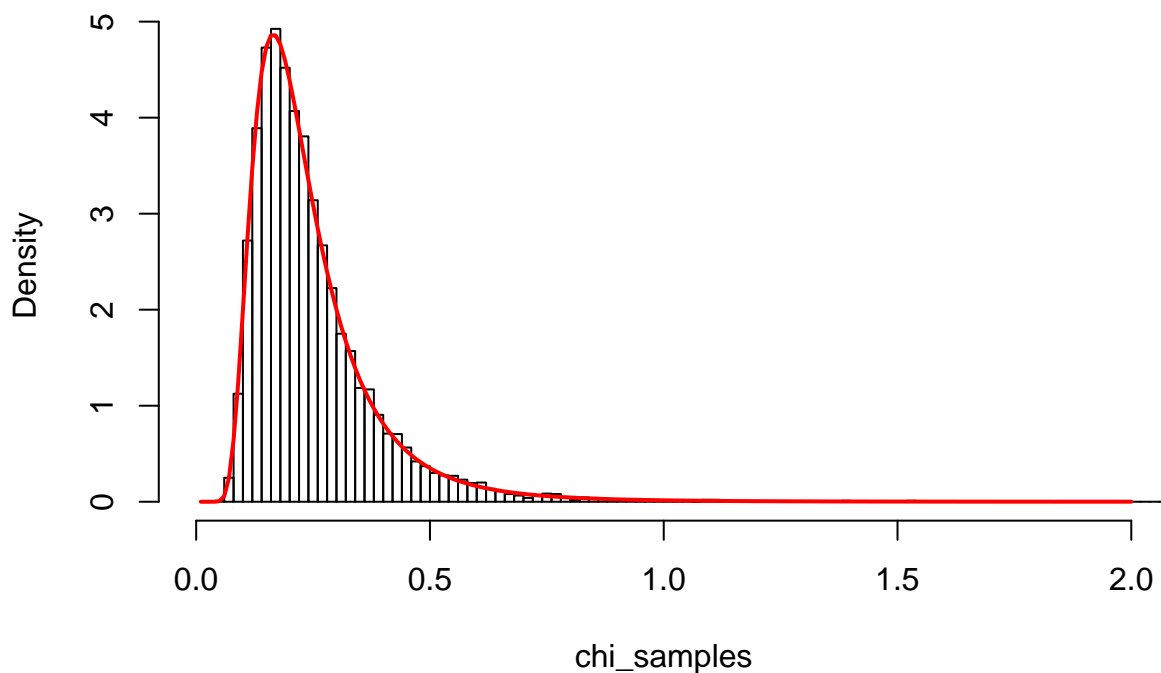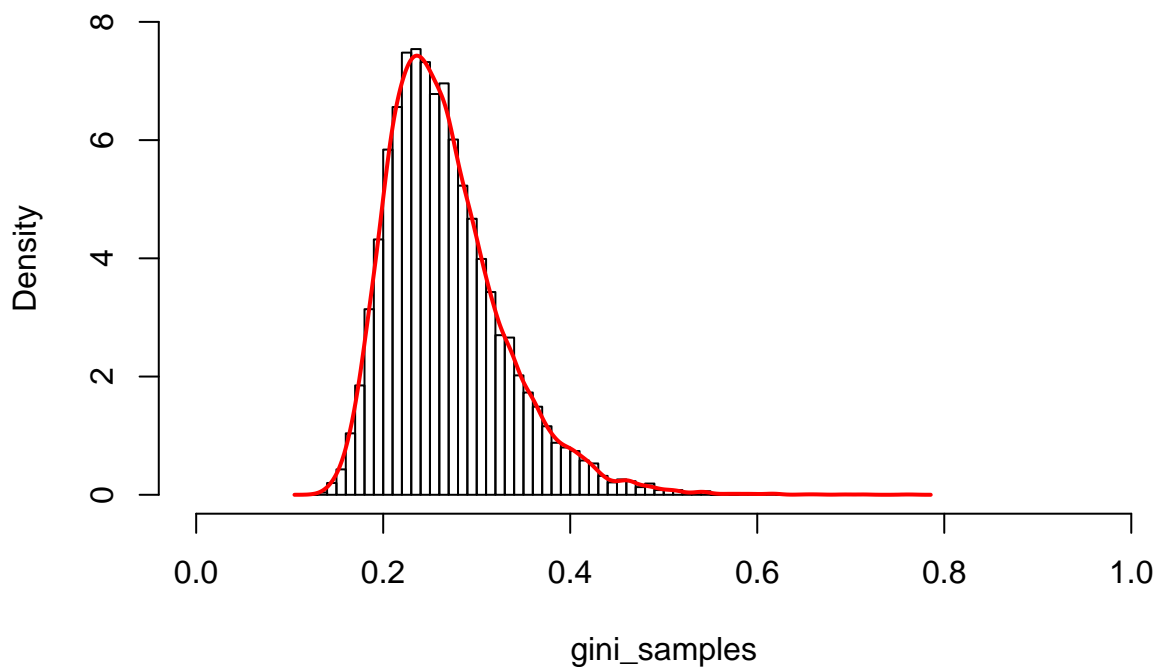
Histogram of chi_samples

The red line representing the theoretical distribution. I looks like a realy nice fit to the posterior in the histogram.

b

```
gini <- function(sigma){
    2 * pnorm(sigma / sqrt(2), mean = 0 , sd = 1) - 1
}

gini_samples <- gini(sqrt(chi_samples))

hist(gini_samples, breaks = 50, freq = FALSE,
     ylim = c(0,8), xlim=c(0, 1))
lines(density(gini_samples), col = "red", lwd = 2)
```


Histogram of gini_samples

c

```
gini_cred <- quantile(gini_samples, probs = c(0.025, 1 - 0.025))

dense_gini <- density(gini_samples)
dense_gini_cdf <- dense_gini$y/sum(dense_gini$y)
index_gini <- order(dense_gini_cdf)
dense_gini_cdf <- dense_gini_cdf[index_gini]

while(sum(dense_gini_cdf) > 0.95){
    dense_gini_cdf <- dense_gini_cdf[-1]
    index_gini <- index_gini[-1]
}

lower <- min(dense_gini$x[index_gini])
upper <- max(dense_gini$x[index_gini])
```
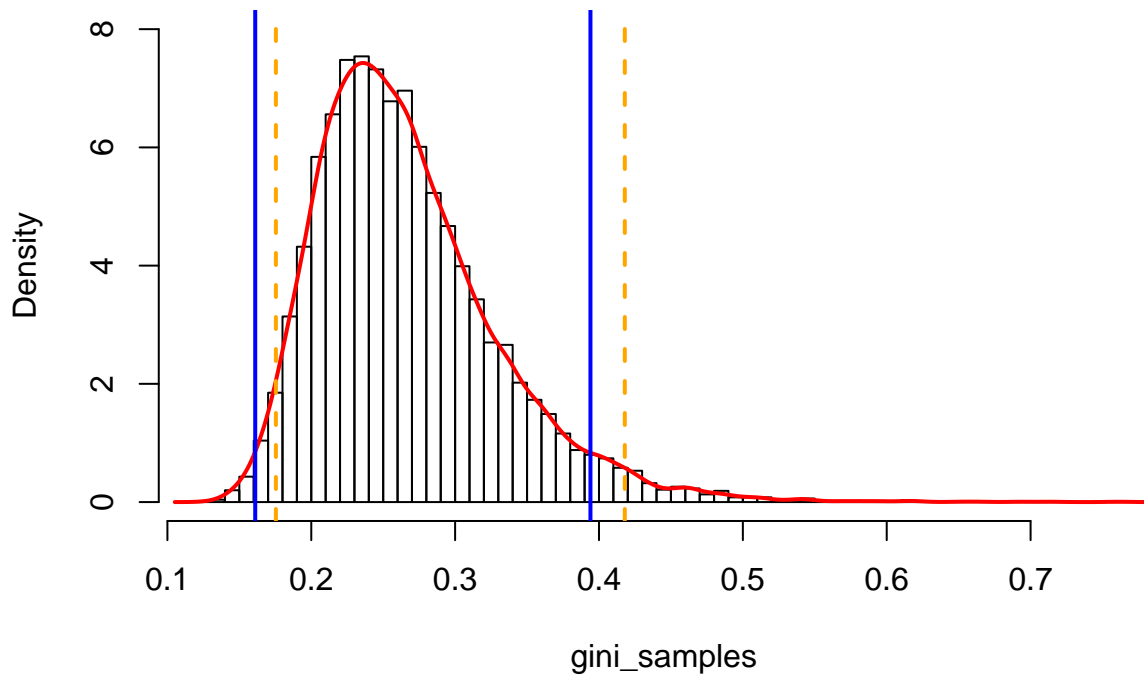
```
gini_HPD <- c(lower, upper)
```

```
hist(gini_samples, breaks = 50, freq = FALSE, ylim = c(0, 8))
lines(density(gini_samples), col = "red", lwd = 2)
abline(v = gini_HPD, col = "blue", lwd = 2)
abline(v = gini_cred, col = "orange", lwd = 2, lty = 2)
```

Histogram of gini_samples

The credible-interval is represented by the dashed orange line and contains with 95 % probability the parameter G lies within this interval. So we can say it is more probable that the income-distribution is equal than inequal.

For the Highest Posterior Density (HPD) interval we are with 95 probability certain that the most probable G-value.

The difference between the two intervalls is that the HPD is more shifted to the left since the distributions is positively skewed.

Question 3

For this exercise we have been given the likelihood

$$p(y|\mu, \kappa) = \frac{\exp\left[\kappa \cos(y - \mu)\right]}{2\pi I_0(\kappa)}, -\pi \leq y \leq \pi$$

where $I_0(\kappa)$ is the modified Bessel function of the first kind of order zero. We have also been given that $\kappa \sim \text{Exponential}(\lambda = 1)$. We want to approximate the marginal posterior distribution $p(\kappa|\mu, y)$ and we assume that $\mu = 2.39$. In order to this we generate samples of κ that we assume comes from the Exponential distribution, in our case 1000 values equally spaces between 0.01 to 10. We can then compute

$$p(\kappa|\mu, y) \propto p(y|\mu, \kappa)p(\kappa)$$

and get an approximation of the marginal posterior distribution. This method usually goes by the name of *grid approximation* and is appropriate for low number of parameters.

a

```
vonMises <- function(y,K,mu){
    exp(K * cos(y - mu)) /(2 * pi * besselI(K, nu = 0))
}

y <- c(-2.44, 2.14, 2.54, 1.83, 2.02, 2.33, -2.79, 2.23, 2.07, 2.02)

sample_exp <- seq(0.01, 10, 0.01)
sample_mises <- t(sapply(y, function(X) vonMises(y = X, K = sample_exp, mu = 2.39)))

prior <- dexp(sample_exp, rate = 1)
likelihood <- apply(sample_mises, MARGIN = 2, FUN = prod)

unnorm_posterior <- likelihood * prior
posterior <- unnorm_posterior / sum(unnorm_posterior)

plot(sample_exp, posterior, type="l", ylab="Density", xlab="k")
```
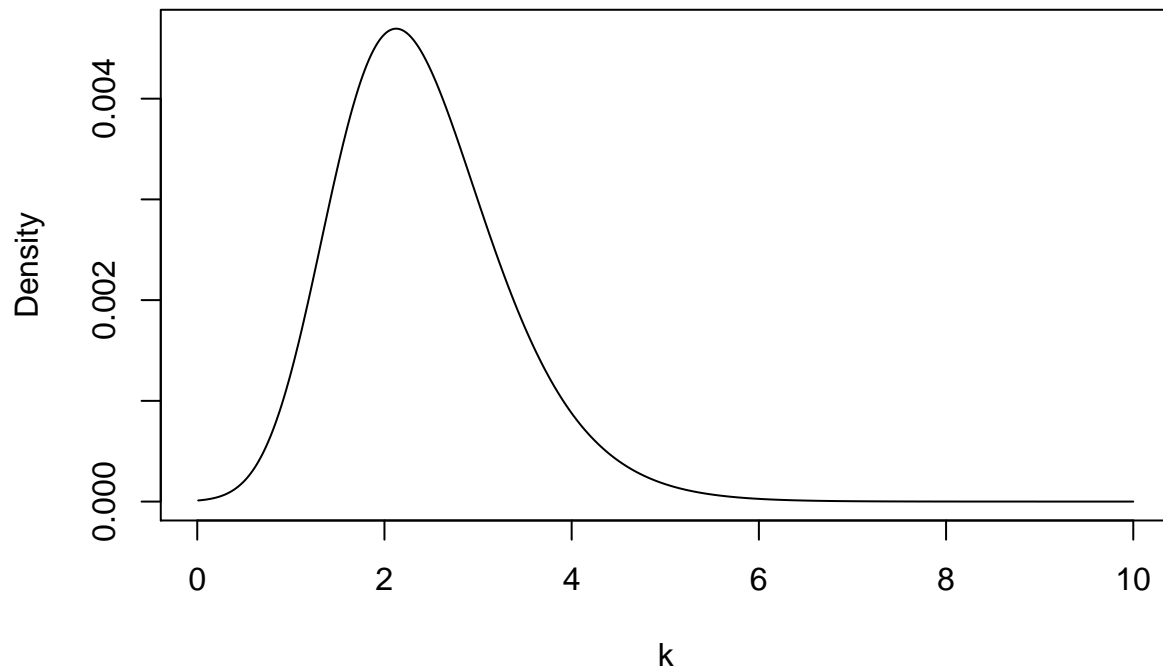
We can see from the distribution above that the most density is located between 0.5 and 4.

b

The posterior mode of κ is approximately 2.12 given by our approximation of the distribution.