Lab1

aswma317,varsi146

2023-04-14

Question 1.

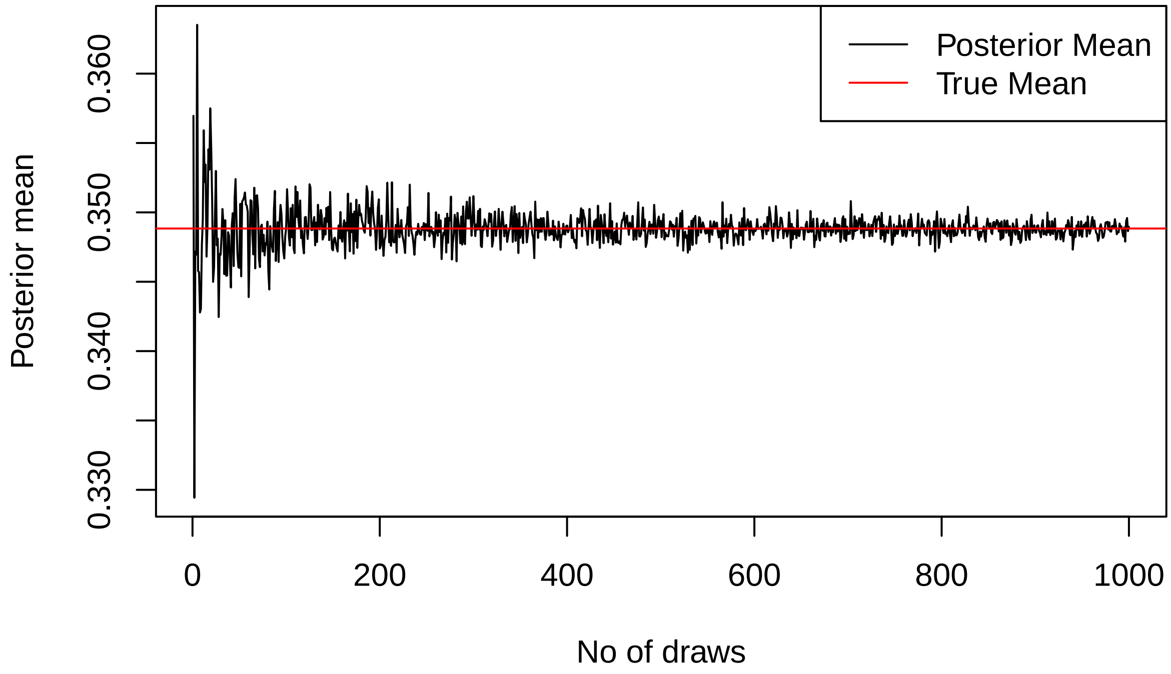
1 a).

```
#Question 1
#Bernoulli Data
n <- 70 #Total trials
s <- 22 #Successes
f <- n-s #Failures

#Prior
pr_alpha <- pr_beta <- 8

#Posterior
pos_alpha <- pr_alpha + s
pos_beta <- pr_beta + f
```

```
#1(a)
pos_mean <- c()
pos_sd <- c()
n_draws <- seq(from = 10, to = 10000, by = 10)
for (i in n_draws) {
  #Draw random var for posterior
  pos_rv <- rbeta(n = i, pos_alpha, pos_beta)
  pos_mean <- c(pos_mean, mean(pos_rv))
  pos_sd <- c(pos_sd, sd(pos_rv))
}
```

As we can see below, we can verify graphically that as the number of draws increases, the posterior mean E[θ|y] converges to the true value of mean.

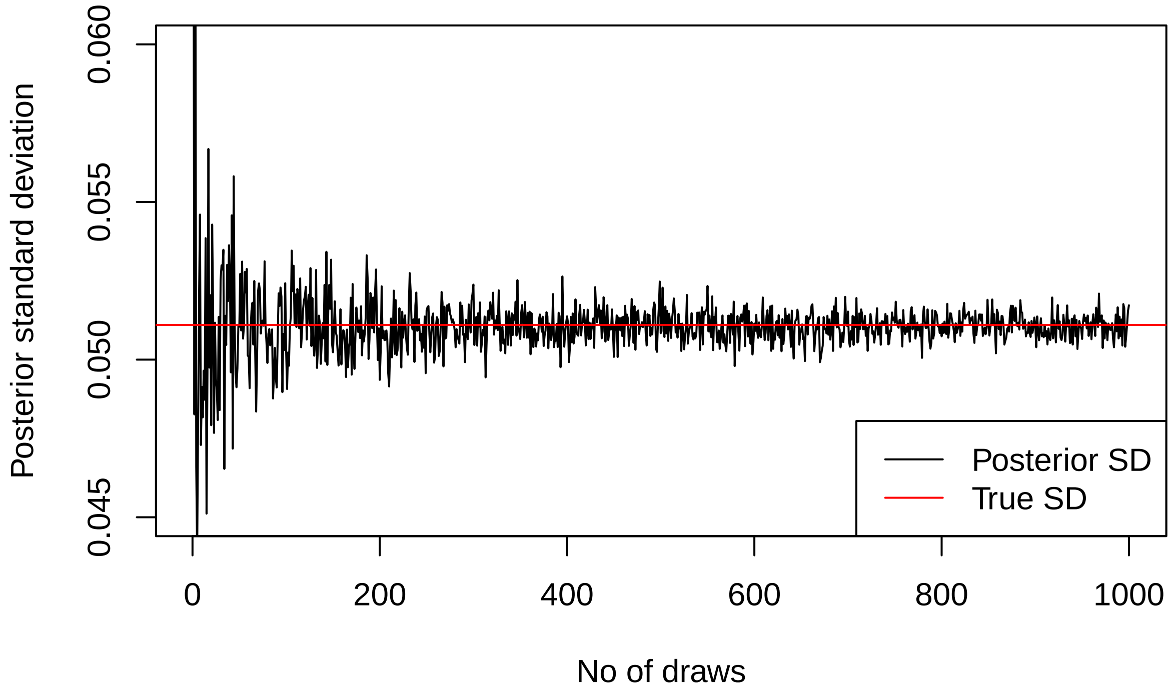
```
plot(pos_mean, type = 'l',
     xlab = 'No of draws', ylab = 'Posterior mean')
legend(x = 'topright', legend = c('Posterior Mean', 'True Mean'), col = c('black', 'red'),
     lty = c(1,1))
true_mean <- pos_alpha/(pos_alpha+pos_beta)
abline(h = true_mean, col = 'red')
```



```
#The pos_mean converges to true_mean as draws increases
```

As we can see below, we can verify graphically that as the number of draws increases, the posterior standard deviation SD[θ|y] converges to the true value of standard deviation.

```
plot(pos_sd, type = 'l',
     xlab = 'No of draws', ylab = 'Posterior standard deviation', ylim = c(0.045, 0.06))
legend(x = 'bottomright', legend = c('Posterior SD', 'True SD'), col = c('black', 'red'),
     lty = c(1,1))
true_sd <- sqrt((pos_alpha * pos_beta)/((pos_alpha + pos_beta)^2 * (pos_alpha + pos_beta + 1)))
abline(h = true_sd, col = 'red')
```



```
#The pos_sd converges to true_mean as draws increases
```

1 b).

```
pos_rv <- rbeta(n = 10000, pos_alpha, pos_beta)

#Filter out all theta's greater than 0.3
filt_pos_rv <- pos_rv[pos_rv>0.3]

#Getting the probability of theta>0.3
p_gt0.3 <- length(filt_pos_rv)/length(pos_rv)

#To get the exact/true value, using the pbeta to get the CDF.
cdf_0.3 <- pbeta(0.3, pos_alpha, pos_beta, lower.tail = FALSE)
```

We can see below the comparison of posterior probability Pr(θ>0.3|y) and the exact value from the Beta posterior.

```
print(paste('The posterior of probability that theta > 0.3 is:', round(p_gt0.3,2)))
```

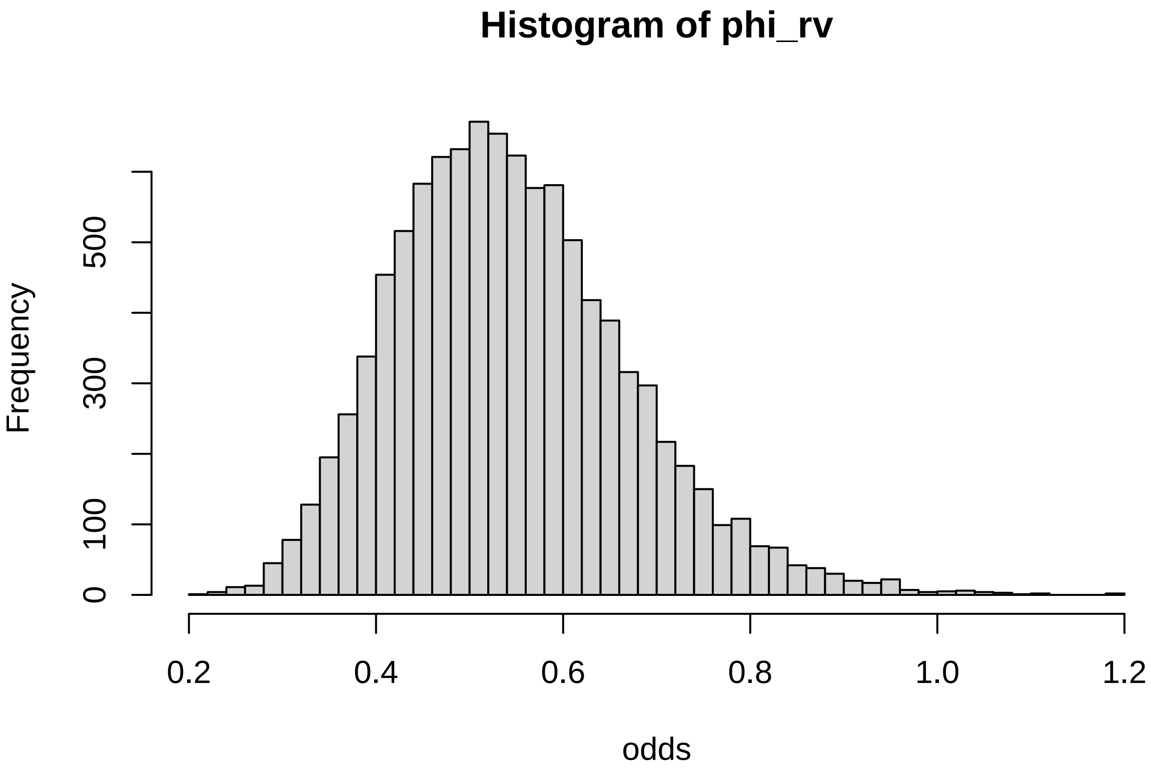
```
## [1] "The posterior of probability that theta > 0.3 is: 0.83"
```

```
print(paste('Exact value from beta posterior that theta > 0.3 is:', round(cdf_0.3,2)))
```

```
## [1] "Exact value from beta posterior that theta > 0.3 is: 0.83"
```

1 c).

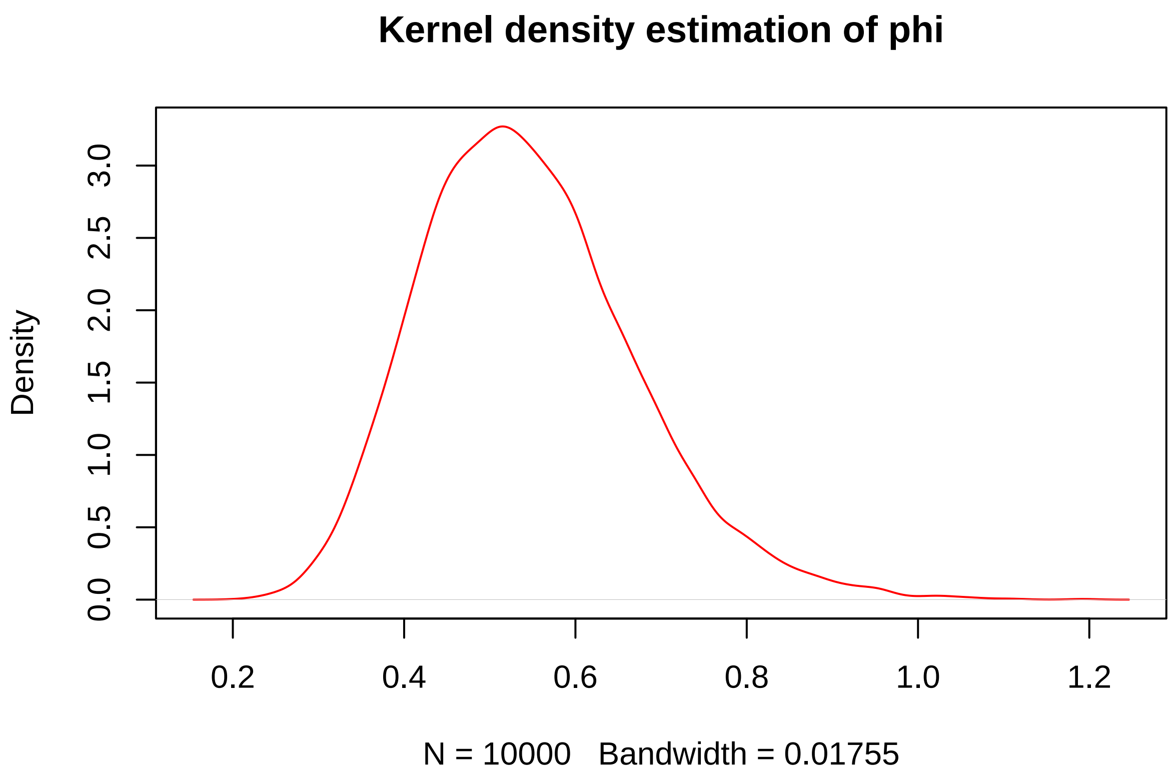
```
pos_rv <- rbeta(n = 10000, pos_alpha, pos_beta)

#Calculating the phi values from theta values
phi_rv <- pos_rv/(1-pos_rv)
hist(phi_rv, breaks = 40, xlab = "odds")
```



```
#Plotting the distribution of phi
phi_den <- density(phi_rv)
```

Below is the plot of the posterior distribution of ϕ

```
plot(phi_den, col = 'red', main = "Kernel density estimation of phi")
```



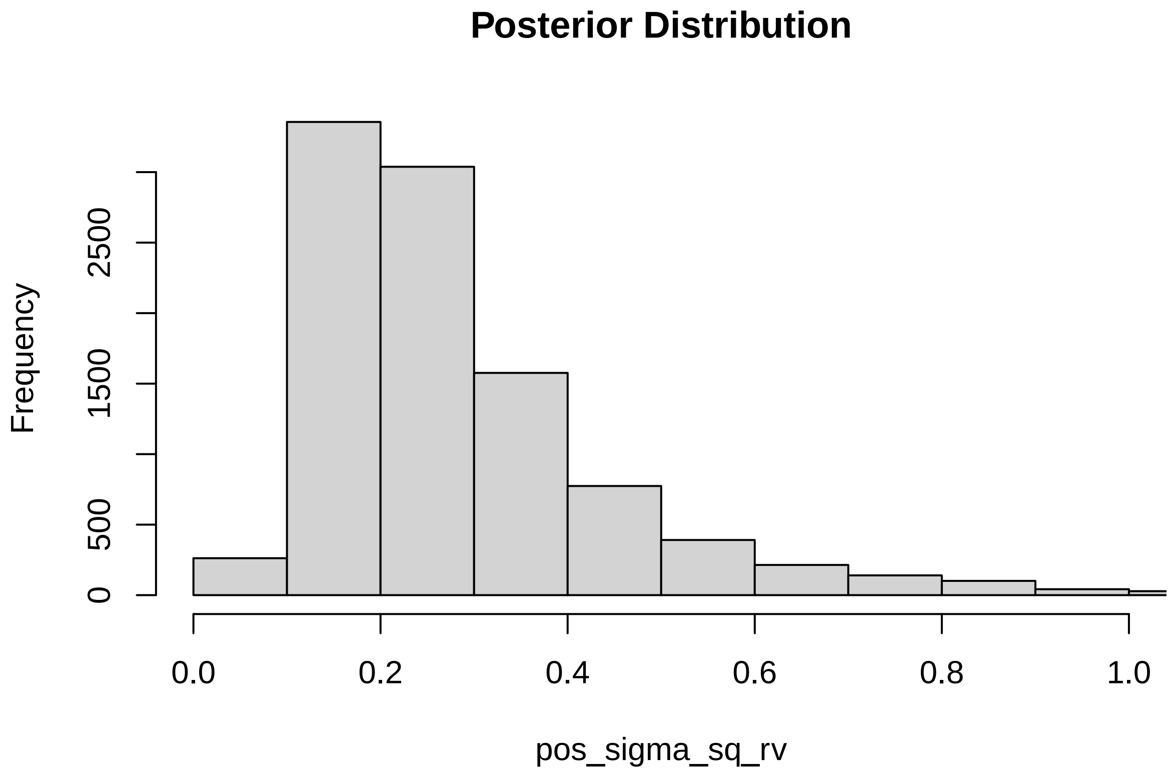
Question 2

2 a).

```r
#Normal model with unknown variance
n_draws <- 10000
data_mean <- 3.6
Y <- c(33, 24, 48, 32, 55, 74, 23, 17)
n <- length(Y)

#Step1: draws from chi squared distribution
X <- rchisq(n_draws, df = n)

#Step2: Compute sigma^2 - draw from inverse chi-squared
taosq <- sum((log(Y)-data_mean)^2)/n
pos_sigma_sq_rv <- (n * taosq)/X
```

We can see below the plot for the posterior of σ^2 by assuming μ=3.6

```r
hist(pos_sigma_sq_rv, breaks = 50, main = 'Posterior Distribution',
     xlim = c(0,1))
```

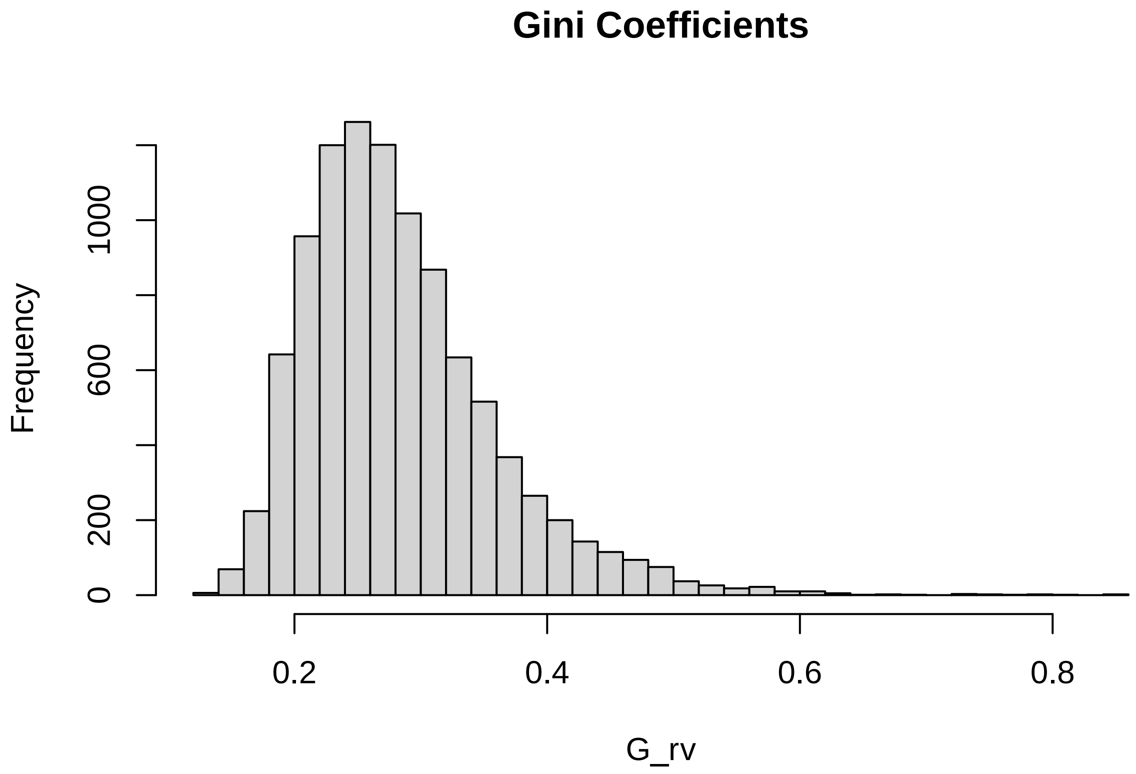


2 b).

```
#phi is CDF for normal standard normal with 0 mean and unit variance - hence pnorm
phi <- pnorm(sqrt(pos_sigma_sq_rv/2), mean = 0, sd = 1)
G_rv <- 2 * phi - 1
```

Below is the posterior distribution of Gini coefficient G using the posterior draws from 2a, where G = 2*$\phi(\frac{\sigma}{\sqrt{2}})$-1

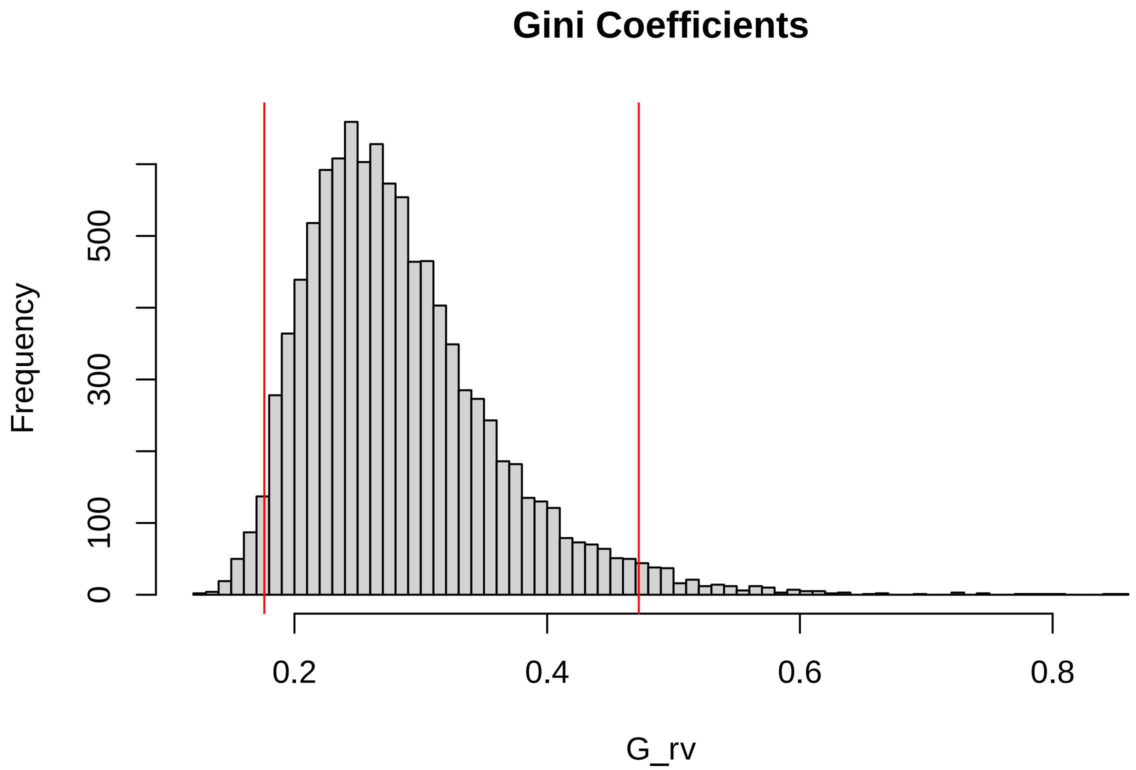
```
hist(G_rv, breaks = 40, main = 'Gini Coefficients')
```



2 c).

In the below histogram, the two vertical lines represents the computed 95% equal tail credible interval for G.

```
cred_int <- quantile(G_rv, probs = c(0.025, 0.975))
hist(G_rv, breaks = 100, main = 'Gini Coefficients')
abline(v = c(cred_int[1], cred_int[2]), col = 'red')
```



2 d).

```
#Computing HPDI
ker_den <- density(G_rv)

#Calculating the total area under the curve
f <- approxfun(ker_den$x, ker_den$y)
tot_area <- integrate(f, min(ker_den$x), max(ker_den$x))
print(tot_area)
```

```
## 1.00098 with absolute error < 1e-04
```

```
#Sort the densities and the x-values based on descending order of density values
sorted_index <- order(ker_den$y, decreasing = TRUE)
ker_den_y <- ker_den$y[sorted_index]
ker_den_x <- ker_den$x[sorted_index]

#Finding the indexes which gives the area <= 0.95
cum_area <- 0
indx <- 0

while (cum_area <= .95) {
  f <- approxfun(ker_den_x, ker_den_y)
  indx <- indx + 1
  #Getting the area of the sorted pairs of indexes
  area <- integrate(f, ker_den_x[indx], ker_den_x[indx+1])
  cum_area <- area$value/tot_area$value
  # print(paste("Index:", indx, " cum_area:", cum_area))
}
print(paste("Lower limit:", ker_den_x[indx]))
```

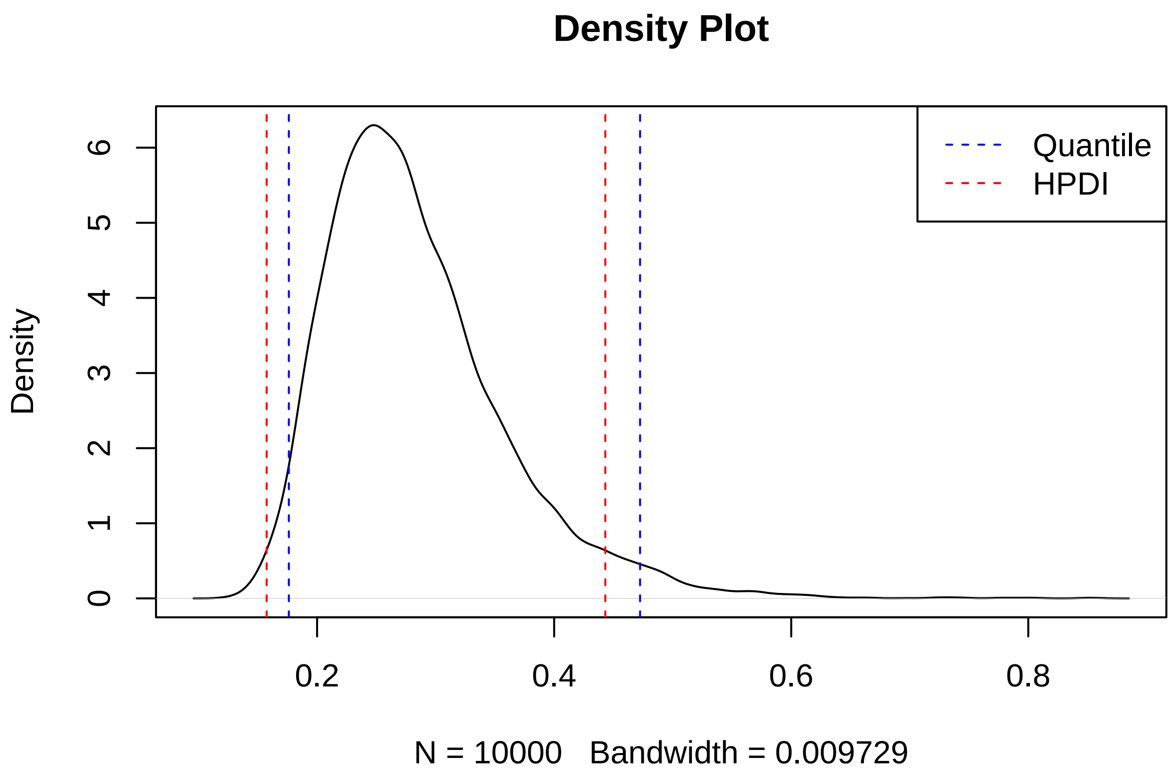
```
## [1] "Lower limit: 0.157460895430131"
```

```
print(paste("Upper limit:", ker_den_x[indx+1]))
```

```
## [1] "Upper limit: 0.443197715104105"
```

In the plot below, we can see the difference between the 95% credible interval region(indicated by blue dotted lines) and the Highest Posterior Density(HPD) Interval region(indicated by red dotted lines). It is clear that the HPD intervals have moved to the left, thereby indicating the region with the highest posterior density.

```
plot(ker_den, main = 'Density Plot')
legend(x = 'topright', legend = c('Quantile', 'HPDI'), col = c('blue', 'red'),
     lty = c(2,2))
abline(v = c(ker_den_x[indx], ker_den_x[indx+1]), col = 'red', lty = 2)
abline(v = c(cred_int[1], cred_int[2]), col = 'blue', lty = 2)
```



Question 3

3 a).

From the question, we have:

$$p(y|\mu,k) = \frac{exp[k.cos(y-\mu)]}{2\pi I_0(k)}$$

The likelihood of the log-normal distribution is given by:

$$\prod_{i=1}^{n} p(y_i|\mu,k)$$

That is,

$$p(y|\mu,\kappa) = \frac{\exp\left[\kappa \cdot \sum_{i=1}^{n} \cos\left(y_i-\mu\right)\right]}{(2\cdot\pi\cdot I_0(\kappa))^n}$$

The prior is defined as: $\kappa \sim Exponential(\lambda = 0.5)$, with mean $1/\lambda$. That is, from the question:

$$p(\kappa) = 0.5exp^{-0.5\kappa}$$

We know that the posterior distribution by Bayes theorem is written as $p(\kappa|y,\mu) \propto p(y|\mu,\kappa) \cdot p(\kappa)$

Therefore the posterior distribution can then be derived to be the following:

$$p(\kappa|y,\mu) \propto \frac{1}{(I_o(\kappa))^n} \cdot exp[k \cdot \sum_{i=1}^{n} \cos\left(y_i-\mu\right) - \frac{k}{2}]$$

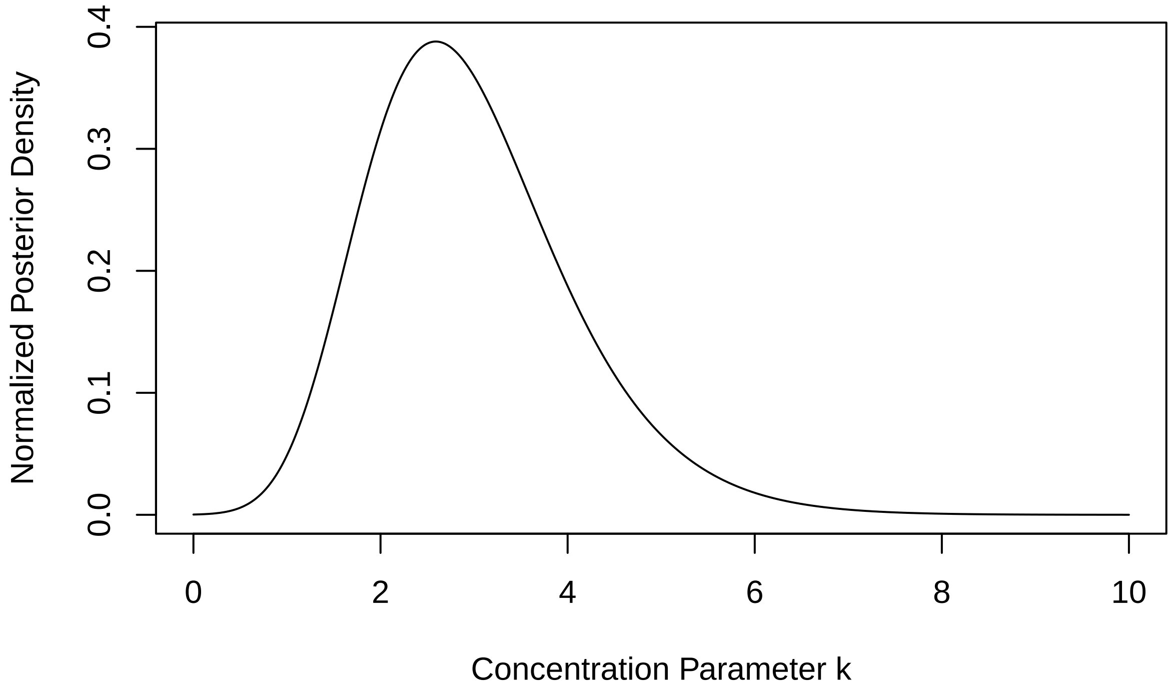
```
#Posterior
pos_pdf <- function(k){
  #Data
  Y <- c(-2.79, 2.33, 1.83, -2.44, 2.23, 2.33, 2.07, 2.02, 2.14, 2.5)
  data_mean <- 2.4
  n <- length(Y)
  bes_val <- besselI(k, nu = 0)
  return((1/(bes_val)^n) * exp(k * sum(cos(Y-data_mean)) - (k/2)))
}
k = seq(0, 10, 0.01)
```

We then compute the normalizing constant in order to normalize the posterior density.

```
#Calculating the normalizing constant for given unknown distribution
norm_const <- integrate(pos_pdf, lower = min(k), upper = max(k))

#Getting the normalized pdf
norm_pos_pdf <- pos_pdf(k)/norm_const$value
```

Below we can see the plot of the posterior distribution of κ for the wind direction data over a fine grid of κ values.

```
plot(k, norm_pos_pdf, type = 'l', xlab = 'Concentration Parameter k'
     ,ylab = 'Normalized Posterior Density')
```



```
#Confirming that the area is 1
f <- approxfun(k, norm_pos_pdf)
tot_area <- integrate(f, min(k), max(k))
print(paste("CDF:", tot_area$value))
```

```
## [1] "CDF: 1.00000140350999"
```

3 b).

The approximate posterior mode is calculated to be:

```
#The mode is the k with the maximum frequency/probability
print(paste("The mode:", k[which.max(norm_pos_pdf)]))
```

```
## [1] "The mode: 2.59"
```