Laboration report in Bayesian Statistics

Bayesian Learning, Computer Lab 1

732A91

Duc Tran
William Wiik

Division of Statistics and Machine Learning
Department of Computer Science
Linköping University

11 April 2024

Question 1: Daniel Bernoulli

Let $y_1, ..., y_n|\theta \sim Bern(\theta)$, and assume that you have obtained sample with $s = 22$ successes in $n = 70$ trials. Assume a Beta(α_0, β_0) prior for θ and let $\alpha_0 = \beta_0 = 8$.

a)

Question: Draw 10000 random values (nDraws = 10000) from the posterior $\theta|y \sim Beta(\alpha_0 + s, \beta_0 + f)$, where $y = (y_1, ..., y_n)$, and verify graphically that the posterior mean $E[\theta|y]$ and standard deviation $SD[\theta|y]$ converges to the true values as the number of random draws grows large.

Answer: The code to sample from the posterior is presented as follows:

```r
set.seed(13)
alpha <- 22+8
beta <- 70-22+8

# Draw 10000 values from posterior
sim_data <- rbeta(10000, alpha, beta)

# Cumulative sum for mean
cum_mean <- cumsum(sim_data) / 1:10000
cum_sd <- c()

# Cumulative sum for sd
for (n in 2:length(sim_data)){
  cum_sd[n] <- sd(sim_data[1:n])
}

# Expected values according to theory
expected_mean <- alpha / (alpha + beta)
expected_sd <- sqrt(alpha*beta / ( (alpha+beta)^2 * (alpha+beta+1) ))
```

In figure 1, the cumulative mean and the expected value of mean is presented.

```r
# Plot for cumulative mean
plot_data <- data.frame(cum_mean, cum_sd)
ggplot(plot_data, aes(x=1:10000)) +
  geom_point(aes(y = cum_mean), col="chartreuse4") +
  theme_bw() +
  geom_hline(aes(yintercept = expected_mean), col="black", lty=2, linewidth=1) +
  labs(x = "Number of draws",
       y = "Cumulative mean")
```

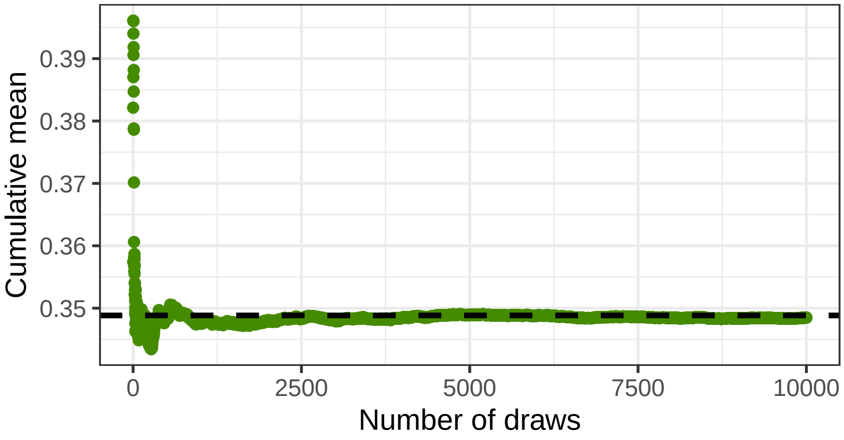

Figure 1: Mean of θ as a function of the accumulating number of drawn values

From figure 1, we can see that after around 2500 draws, the cumulative mean has stabilized around the expected mean.

In figure 2, the cumulative standard deviation and the expected value is presented.

```r
# Plot for cumulative sd
ggplot(plot_data[-1, ], aes(x=2:10000)) +
  geom_point(aes(y = cum_sd), col="red3") +
  theme_bw() +
  geom_hline(aes(yintercept = expected_sd), col="black", lty=2, linewidth=1) +
  labs(x = "Number of draws",
       y = "Cumulative standard deviation")
```

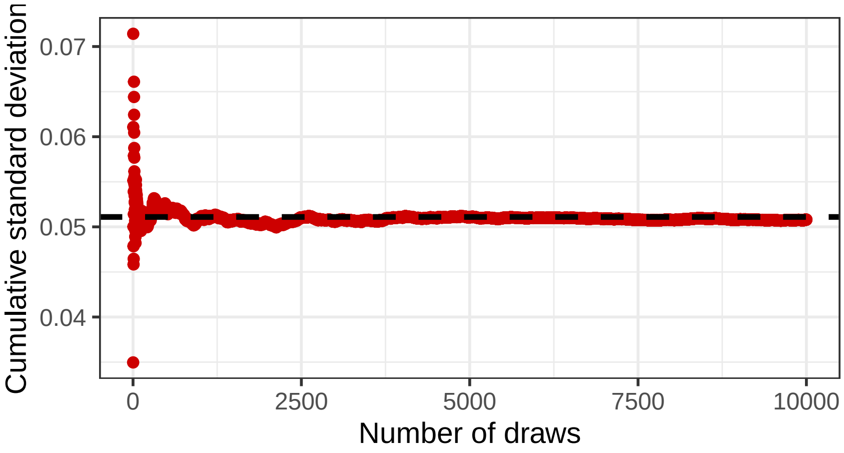

Figure 2: Standard deviation of θ as a function of the accumulating number of drawn values

From figure 2, we can see that after around 2500 draws, the cumulative sd has stabilized around the expected sd.

b)

Question: Draw 10000 random values from the posterior to compute the posterior probability $\Pr(\theta > 0.3|y)$ and compare with the exact value from the Beta posterior.

Answer:

```r
sum(sim_data > 0.3) / 10000
```

```
## [1] 0.827
```

```r
pbeta(0.3, alpha, beta, lower.tail = FALSE)
```

```
## [1] 0.8285936
```

The random values from posterior is 0.827 and the theoretical value is 0.829, a minor difference of 0.002. The minor difference is due to the random sampling of the posterior.

c)

Question: Draw 10000 random values from the posterior of the odds $\phi = \frac{\theta}{1-\theta}$ by using the previous random draws from the Beta posterior for θ and plot the posterior distribution of ϕ.

Answer: In figure 3, the histogram of the posterior of the odds is presented. In figure 4, the density of the posterior of the odds is presented.

```r
odds_ratio <- sim_data / (1-sim_data)

ggplot(data.frame(odds_ratio), aes(x = odds_ratio)) +
  geom_histogram(binwidth = 0.03, fill = "skyblue", color = "black") +
  labs(x = "Odds Ratio", y = "Frequency") +
  theme_bw()
```

```r
ggplot(data.frame(odds_ratio), aes(x = odds_ratio)) +
  geom_density(alpha = 0.5, fill = "skyblue") +
  labs(x = "Odds Ratio", y = "Density") +
  theme_bw()
```

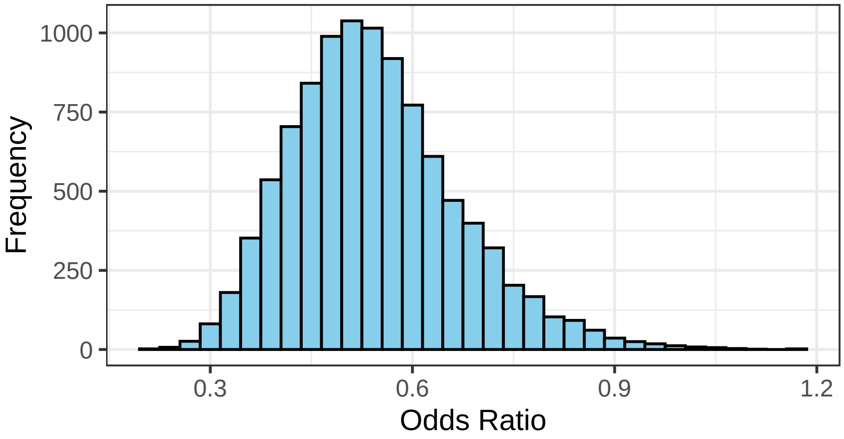

Figure 3: Posterior Distribution of Odds Ratio, histogram

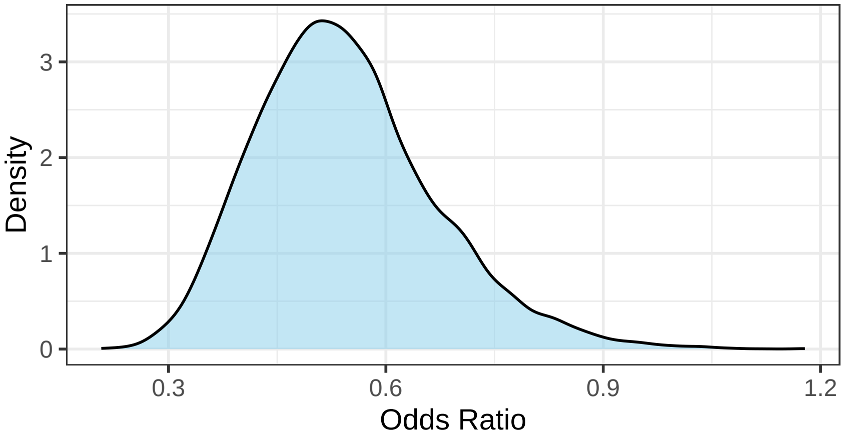

Figure 4: Posterior Distribution of Odds Ratio, density

Question 2: Log-normal distribution and the Gini coeffcient.

Assume that you have asked 8 randomly selected persons about their monthly income (in thousands Swedish Krona) and obtained the following eight observations: 33, 24, 48, 32, 55, 74, 23, and 17. A common model for non-negative continuous variables is the log-normal distribution. The log-normal distribution $\log \mathcal{N}(\mu, \sigma^2)$ has density function

$$p(y|\mu, \sigma^2) = \frac{1}{y \cdot \sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2} \log\ (y - \mu)^2 \right],$$

where $y > 0$, $-\infty < \mu < \infty$ and $\sigma^2 > 0$. The log-normal distribution is related to the normal distribution as follows: if $y \sim \log \mathcal{N}(\mu, \sigma^2)$ then $\log y \sim \mathcal{N}(\mu, \sigma^2)$. Let $y_1, ..., y_n|\mu, \sigma^2 \overset{\text{iid}}{\sim} \log \mathcal{N}(\mu, \sigma^2)$, where $\mu = 3.6$ is assumed to be known but σ^2 is unknown with non-informative prior $p(\sigma^2) \propto 1/\sigma^2$. The posterior for σ^2 is the $Inv - \chi^2(n, \tau^2)$ distribution, where

$$\tau^2 = \frac{\sum_{I=1}^{n} (\log\ y_i - \mu)^2}{n}.$$

a)

Question: Draw 10000 random values from the posterior of σ^2 by assuming $\mu = 3.6$ and plot the posterior distribution.

Answer: From lecture slides we can simulate draws from the posterior for a normal model with unknown mean and variance with the steps as follows:

1. Draw $x \sim \chi^2(n-1)$
2. Compute $\sigma^2 = \frac{(n-1) \cdot s^2}{X}$)(a draw from Inv-$\chi^2(n-1, s^2)$)
3. Draw a θ from $N\left(\bar{x}, \frac{\sigma^2}{n}\right)$
4. Repeat 1-3 many times.

Since we only need to sample σ^2, we only do step 1 and 2 many times. Furthermore, the mean is assumed to be known so we will use n instead of $n-1$ and the formula for τ^2 instead of sample variance.

```
data <- log(c(33, 24, 48, 32, 55, 74, 23, 17))
n <- length(data)
mu <- 3.6
tau_sq <- sum((data-mu)^2)/n

# Step 1
X <- rchisq(10000, n)
# Step 2
sample_sigma <- n*tau_sq/X

plot_data <- data.frame(sample_sigma)
```

```r
ggplot(plot_data, aes(x=sample_sigma)) +
  geom_histogram(bins=80, colour="black", fill="skyblue") +
  labs(x = expression("Sample"~sigma^2), y = "Frequency") +
  theme_bw()
```

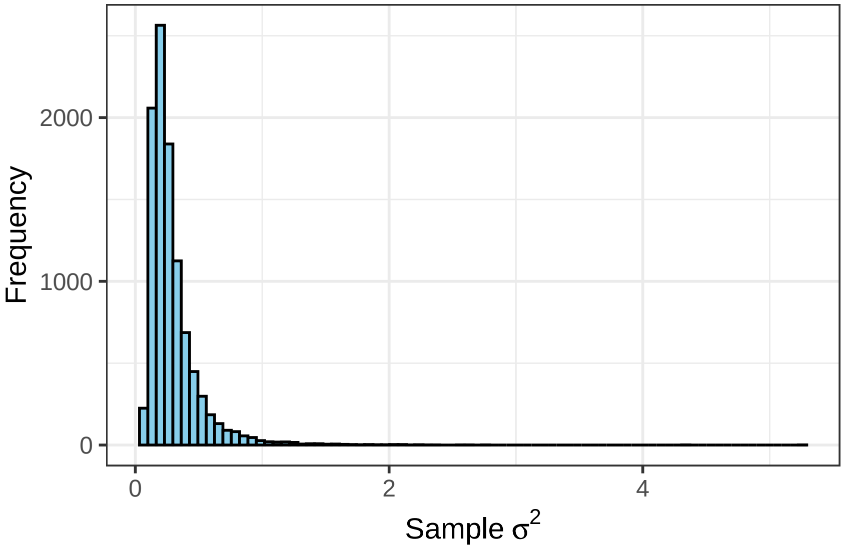

Figure 5: Posterior Distribution of σ^2

In figure 5, most of the posterior distribution is between the values 0 and 1.

b)

Question: The most common measure of income inequality is the Gini coeffcient, G, where $0 \leq G \leq 1$. $G = 0$ means a completely equal income distribution, whereas $G = 1$ means complete income inequality. It can be shown that $G = 2\Phi(\sigma/\sqrt{2}) - 1$ when incomes follow a $\log \mathcal{N}(\mu, \sigma^2)$ distribution. $\Phi(z)$ is the cumulative distribution function (CDF) for the standard normal distribution with mean zero and unit variance. Use the posterior draws in a) to compute the posterior distribution of the Gini coeffcient G for the current data set

Answer:

```r
gini <- 2 * pnorm((sqrt(sample_sigma) / sqrt(2)), mean = 0, sd = 1) - 1

ggplot(data.frame(gini), aes(x = gini)) +
  geom_histogram(binwidth = 0.02, fill = "skyblue", color = "black") +
  labs(x = "Gini coeffcient", y = "Frequency") +
  theme_bw()
```

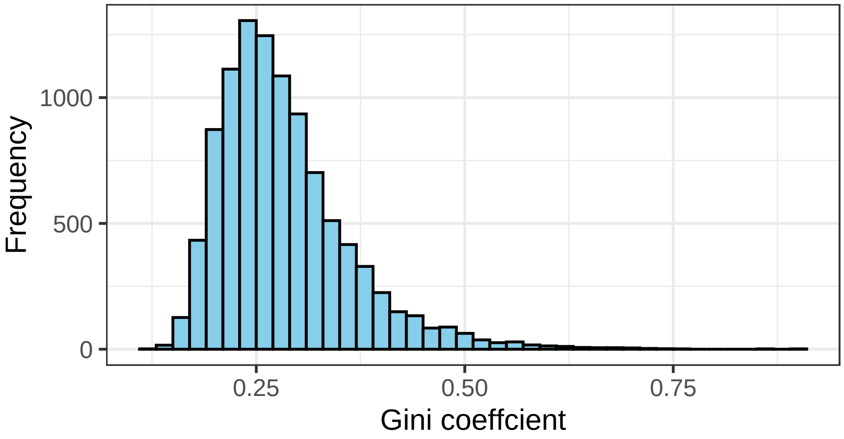

Figure 6: Posterior Distribution of gini coeffcient

c)

Question: Use the posterior draws from b) to compute a 95% equal tail credible interval for G. A 95% equal tail credible interval (a, b) cuts off 2.5% percent of the posterior probability mass to the left of a, and 2.5% to the right of b.

Answer:

```
# We can also sort samples in order and take index 251 as lower limit and 9750 as upper limit
interval_equal_tail <- quantile(sample_sigma, probs = c(0.025, 0.975))
cat("The 95% equal tail credible interval is: [", round(interval_equal_tail[1], 4), " ",
    round(interval_equal_tail[2], 4), "]", sep="")
```

```
## The 95% equal tail credible interval is: [0.1 0.845]
```

d)

Question: Use the posterior draws from b) to compute a 95% Highest Posterior Density Interval (HPDI) for G. Compare the two intervals in (c) and (d). [Hint: do a kernel density estimate of the posterior of G using the density function in R with default settings, and use that kernel density estimate to compute the HPDI. Note that you need to order/sort the estimated density values to obtain the HPDI].

Answer:

The code used is presented as follows.

```
# Note: "Posterior" is not proper
fit <- density(sample_sigma)
# Constant to divide each area so that the cumulative sum is 1
const = sum(fit$y)
# Cumulative sum for area, normalized
prob_mass <- cumsum(sort(fit$y, decreasing = TRUE)) / const
```

```r
# Index for which cumsum <= 0.95
index <- prob_mass <= 0.95
# Find index for original data
desc_order <- order(fit$y, decreasing = TRUE)
# Get index for original data
interval_HPDI <- sort(desc_order[index])

# Lag differences, if all == 1 => we have only 1 interval
all(diff(interval_HPDI) == 1)
```

```
## [1] TRUE
```

From the output we get that the interval is one continuous interval.

```r
cat("95% HPDI is [", round(fit$x[interval_HPDI[1]], 4), " ",
    round(fit$x[interval_HPDI[length(interval_HPDI)]], 4), "]", sep="")
```

```
## 95% HPDI is [0.0675 0.6589]
```

```r
ggplot(plot_data, aes(x=sample_sigma)) +
  geom_histogram(binwidth = 0.02, fill="skyblue") +
  theme_bw() +
  labs(x = expression("Sample"~sigma^2)) +
  geom_vline(xintercept = fit$x[interval_HPDI[1]], col="red3", linewidth=1) +
  geom_vline(xintercept = fit$x[interval_HPDI[length(interval_HPDI)]], col="red3", linewidth=1) +
  geom_vline(xintercept = interval_equal_tail[1], col="chartreuse4", linewidth=1) +
  geom_vline(xintercept = interval_equal_tail[2], col="chartreuse4", linewidth=1) +
  scale_x_continuous(limits = c(0, 2))
```

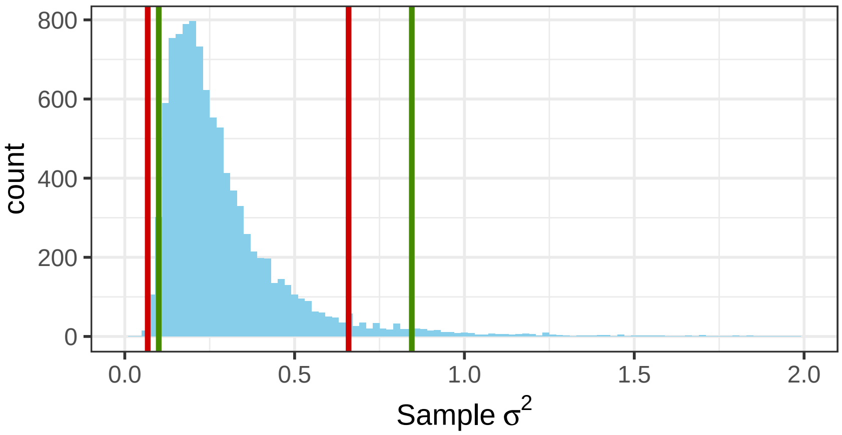

Figure 7: Posterior Distribution of gini coeffcient with equal tail interval (green) & HPDI interval (red)

From figure 7, we can see that the HPDI interval is more narrow than the equal tail interval, this is as expected.

Question 3, Bayesian inference for the concentration parameter in the von Mises distribution

This exercise is concerned with directional data. The point is to show you that the posterior distribution for somewhat weird models can be obtained by plotting it over a grid of values. The data points are observed wind directions at a given location on ten different days. The data are recorded in degrees:

$$(20, 314, 285, 40, 308, 314, 299, 296, 303, 326)$$

where North is located at zero degrees (see Figure 1 on the next page, where the angles are measured clockwise). To fit with Wikipedia's description of probability distributions for circular data we convert the data into radians $-\pi \leq y \leq \pi$. The 10 observations in radians are

$$(-2.79, 2.33, 1.83, -2.44, 2.23, 2.33, 2.07, 2.02, 2.14, 2.54)$$

Assume that these data points conditional on (μ, κ) are independent observations from the following von Mises distribution:

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

where $I_0(\kappa)$ is the modified Bessel function of the first kind of order zero. The parameter $\mu(-\pi \leq \mu \leq \pi)$ is the mean direction and $\kappa > 0$ is called the concentration parameter. Large κ gives a small variance around μ, and vice versa. Assume that μ is known to be 2.4. Let $\kappa \sim Exponential(\lambda = 0.5)$ a priori, where λ is the rate parameter of the exponential distribution (so that the mean is $1/\lambda$).

a)

Question: Derive the expression for what the posterior $p(\kappa|y,\mu)$ is proportional to. Hence, derive the function $f(\kappa)$ such that $p(\kappa|y,\mu) \propto f(\kappa)$. Then, plot the posterior distribution of κ for the wind direction data over a fine grid of κ values. [Hint: you need to normalize the posterior distribution of κ so that it integrates to one.]

Answer:

We have the following:

Prior:

$$p(\kappa) \sim Exp(0.5) = \begin{cases} 0.5e^{-0.5\kappa}, \kappa \geq 0 \\ 0 \qquad\quad, \kappa < 0 \end{cases}$$

Model:

$$p(y|\mu,\kappa) = \frac{exp\Big(\kappa \cdot cos(y-\mu)\Big)}{2\pi I_0(\kappa)}, -\pi \leq y \leq \pi$$

Likelihood:

$$p(y_1, y_2, ..., y_n|\mu,\kappa) = \prod_{i=1}^{n} \frac{exp\Big(\kappa \cdot cos(y_i-\mu)\Big)}{2\pi I_0(\kappa)} = \frac{exp\Big(\kappa \sum_{i=1}^{n} cos(y_i-\mu)\Big)}{(2\pi I_0(\kappa))^n} \propto \frac{exp\Big(\kappa \sum_{i=1}^{n} cos(y_i-\mu)\Big)}{I_0(\kappa)^n}$$

Posterior:

For $\kappa \geq 0$ we get:

$$p(\kappa|y,\mu) \propto p(y_1, y_2, ..., y_n|\mu,\kappa)\cdot p(\kappa) = \frac{exp\Big(\kappa\sum_{i=1}^{n} cos(y_i-\mu)\Big)}{I_0(\kappa)^n}\cdot 0.5exp(-0.5\kappa) \propto \frac{exp\Big(\kappa\Big(\sum_{i=1}^{n} cos(y_i-\mu)\Big)-0.5\kappa\Big)}{I_0(\kappa)^n}$$

The solution is implemented as follows:

```r
# Data
y <- c(-2.79, 2.33, 1.83, -2.44, 2.23, 2.33, 2.07, 2.02, 2.14, 2.54)
mu <- 2.4
n <- length(y)

# Fine grid of k from 0 to 10
k <- seq(from=0, to=10, length.out=10000)

# Posterior
calc_posterior <- function(k){
  exp( k*sum(cos(y-mu)) - k/2) / (besselI(k, nu=0))^n
}
posterior <- calc_posterior(k)

# Approximation of the area under the curve (height (posterior value) * width (how fine the grid is))
area = sum(posterior*k[2])
# Approximating a probability density curve of the posterior
posterior <- posterior/area

plot_data <- data.frame(posterior=posterior)
ggplot(plot_data, aes(x=k, y=posterior)) +
  geom_line() +
  geom_area(fill="skyblue") +
  theme_bw() +
  labs(x = expression(kappa), y = "Density")
```

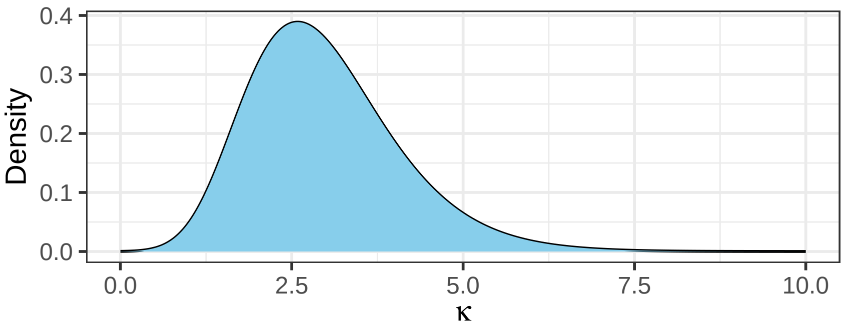

Figure 8: Posterior Distribution of κ for the wind direction data

b)

Question: Find the (approximate) posterior mode of κ from the information in a).

Answer:

```r
k[which.max(posterior)]
```

```
## [1] 2.586259
```

From the output the posterior mode is approximate 2.59.

Appendix

The code used in this laboration report are summarised in the code as follows:

```r
library(ggplot2)

knitr::opts_chunk$set(
  echo = TRUE,
  fig.width = 4.5,
  fig.height = 2.4)
set.seed(13)
alpha <- 22+8
beta <- 70-22+8

# Draw 10000 values from posterior
sim_data <- rbeta(10000, alpha, beta)

# Cumulative sum for mean
cum_mean <- cumsum(sim_data) / 1:10000
cum_sd <- c()

# Cumulative sum for sd
for (n in 2:length(sim_data)){
  cum_sd[n] <- sd(sim_data[1:n])
}

# Expected values according to theory
expected_mean <- alpha / (alpha + beta)
expected_sd <- sqrt(alpha*beta / ( (alpha+beta)^2 * (alpha+beta+1) ))
# Plot for cumulative mean
plot_data <- data.frame(cum_mean, cum_sd)
ggplot(plot_data, aes(x=1:10000)) +
  geom_point(aes(y = cum_mean), col="chartreuse4") +
  theme_bw() +
  geom_hline(aes(yintercept = expected_mean), col="black", lty=2, linewidth=1) +
  labs(x = "Number of draws",
       y = "Cumulative mean")
# Plot for cumulative sd
ggplot(plot_data[-1, ], aes(x=2:10000)) +
  geom_point(aes(y = cum_sd), col="red3") +
  theme_bw() +
  geom_hline(aes(yintercept = expected_sd), col="black", lty=2, linewidth=1) +
  labs(x = "Number of draws",
       y = "Cumulative standard deviation")
sum(sim_data > 0.3) / 10000
pbeta(0.3, alpha, beta, lower.tail = FALSE)
odds_ratio <- sim_data / (1-sim_data)

ggplot(data.frame(odds_ratio), aes(x = odds_ratio)) +
  geom_histogram(binwidth = 0.03, fill = "skyblue", color = "black") +
  labs(x = "Odds Ratio", y = "Frequency") +
  theme_bw()
ggplot(data.frame(odds_ratio), aes(x = odds_ratio)) +
  geom_density(alpha = 0.5, fill = "skyblue") +
  labs(x = "Odds Ratio", y = "Density") +
  theme_bw()
data <- log(c(33, 24, 48, 32, 55, 74, 23, 17))
n <- length(data)
mu <- 3.6
tau_sq <- sum((data-mu)^2)/n

# Step 1
X <- rchisq(10000, n)
# Step 2
sample_sigma <- n*tau_sq/X

plot_data <- data.frame(sample_sigma)

ggplot(plot_data, aes(x=sample_sigma)) +
  geom_histogram(bins=80, colour="black", fill="skyblue") +
  labs(x = expression("Sample"~sigma^2), y = "Frequency") +
  theme_bw()
gini <- 2 * pnorm((sqrt(sample_sigma) / sqrt(2)), mean = 0, sd = 1) - 1

ggplot(data.frame(gini), aes(x = gini)) +
  geom_histogram(binwidth = 0.02, fill = "skyblue", color = "black") +
  labs(x = "Gini coeffcient", y = "Frequency") +
  theme_bw()


# We can also sort samples in order and take index 251 as lower limit and 9750 as upper limit
interval_equal_tail <- quantile(sample_sigma, probs = c(0.025, 0.975))
cat("The 95% equal tail credible interval is: [", round(interval_equal_tail[1], 4), " ",
    round(interval_equal_tail[2], 4), "]", sep="")
# Note: "Posterior" is not proper
fit <- density(sample_sigma)
# Constant to divide each area so that the cumulative sum is 1
const = sum(fit$y)
# Cumulative sum for area, normalized
prob_mass <- cumsum(sort(fit$y, decreasing = TRUE)) / const

# Index for which cumsum <= 0.95
index <- prob_mass <= 0.95
# Find index for original data
desc_order <- order(fit$y, decreasing = TRUE)
# Get index for original data
interval_HPDI <- sort(desc_order[index])
```

```r
# Lag differences, if all == 1 => we have only 1 interval
all(diff(interval_HPDI) == 1)
cat("95% HPDI is [", round(fit$x[interval_HPDI[1]], 4), " ",
    round(fit$x[interval_HPDI[length(interval_HPDI)]], 4), "]", sep="")

ggplot(plot_data, aes(x=sample_sigma)) +
  geom_histogram(binwidth = 0.02, fill="skyblue") +
  theme_bw() +
  labs(x = expression("Sample"~sigma^2)) +
  geom_vline(xintercept = fit$x[interval_HPDI[1]], col="red3", linewidth=1) +
  geom_vline(xintercept = fit$x[interval_HPDI[length(interval_HPDI)]], col="red3", linewidth=1) +
  geom_vline(xintercept = interval_equal_tail[1], col="chartreuse4", linewidth=1) +
  geom_vline(xintercept = interval_equal_tail[2], col="chartreuse4", linewidth=1) +
  scale_x_continuous(limits = c(0, 2))

# Data
y <- c(-2.79, 2.33, 1.83, -2.44, 2.23, 2.33, 2.07, 2.02, 2.14, 2.54)
mu <- 2.4
n <- length(y)

# Fine grid of k from 0 to 10
k <- seq(from=0, to=10, length.out=10000)

# Posterior
calc_posterior <- function(k){
  exp( k*sum(cos(y-mu)) - k/2) / (besselI(k, nu=0))^n
}
posterior <- calc_posterior(k)

# Approximation of the area under the curve (height (posterior value) * width (how fine the grid is))
area = sum(posterior*k[2])
# Approximating a probability density curve of the posterior
posterior <- posterior/area

plot_data <- data.frame(posterior=posterior)
ggplot(plot_data, aes(x=k, y=posterior)) +
  geom_line() +
  geom_area(fill="skyblue") +
  theme_bw() +
  labs(x = expression(kappa), y = "Density")
k[which.max(posterior)]
```