

# Bayesian Learning Lab 1

Nazli Bilgic (nazbi056) & Simge Cinar (simci637)

2024-04-14

## Question 1: Daniel Bernoulli

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

```
alpha <- 8
beta <- 8
n <- 70
s <- 22
f <- n-s
nDraws <- 10000
```

### Part a)

**Question:** Draw 10000 random values ( $n\text{Draws} = 10000$ ) from the posterior  $\theta|y \sim \text{Beta}(\alpha_0 + s, \beta_0 + f)$ , where  $y = (y_1, \dots, y_n)$ , 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. [Hint: use `rbeta()` to draw random values and make graphs of the sample means and standard deviations of  $\theta$  as a function of the accumulating number of drawn values].

### Answer:

10000 random values are drawn from the posterior  $\theta|y \sim \text{Beta}(\alpha_0 + s, \beta_0 + f)$ . The mean and standard deviation formula for beta distribution is as follows:

$$E[X] = \frac{\alpha}{\alpha + \beta}, \quad SD(X) = \sqrt{\frac{\alpha\beta}{(\alpha + \beta + 1)(\alpha + \beta)^2}}$$

Then true mean and standard deviation should be as follows in the posterior distribution;

$$E[X] = \frac{\alpha + s}{\alpha + \beta + n} = \frac{8 + 22}{8 + 8 + 70} = \frac{30}{86} = 0.3488372$$

$$SD(X) = \sqrt{\frac{(\alpha + s)(\beta + f)}{(\alpha + \beta + n + 1)(\alpha + \beta + n)^2}} = \sqrt{\frac{(8 + 22)(8 + 48)}{(8 + 8 + 70 + 1)(8 + 8 + 70)^2}} = \sqrt{\frac{30 \cdot 56}{87 \cdot 86^2}} = 0.05109714$$

```
alpha_posterior <- alpha + s
beta_posterior <- beta + f

true_mean <- alpha_posterior / (alpha_posterior + beta_posterior)
true_variance <- (alpha_posterior * beta_posterior) / ((alpha_posterior+beta_posterior+1)*
```

```

(alpha_posterior+beta_posterior)^2)

cat("True mean:", true_mean, "\n")

## True mean: 0.3488372

cat("True SD:", sqrt(true_variance), "\n")

## True SD: 0.05109714

set.seed(123)
# Draws from posterior distribution
random_values_posterior <- rbeta(nDraws, alpha_posterior, beta_posterior)

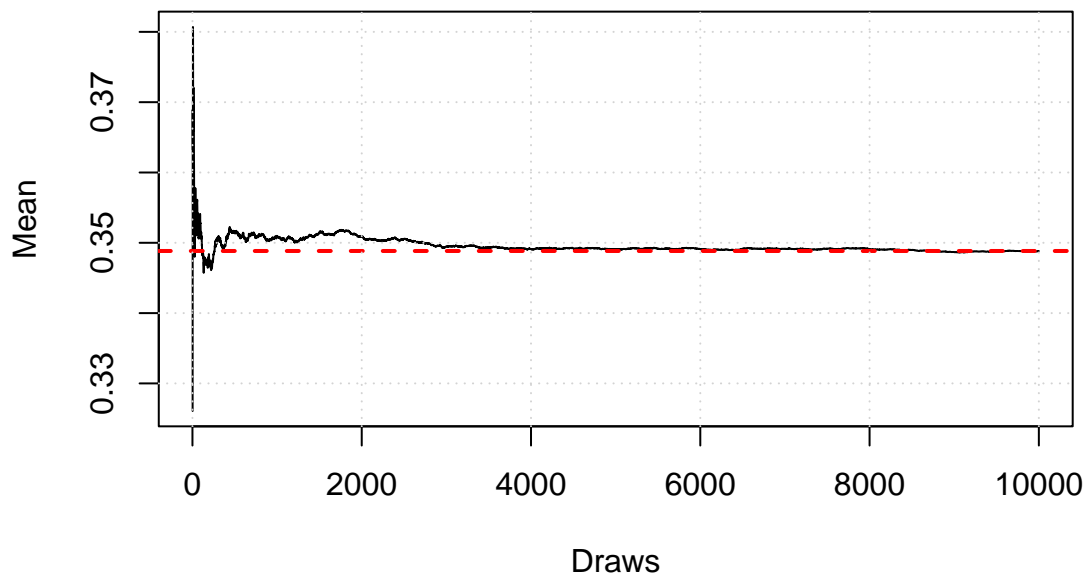
sample_means <- sapply(2:nDraws, function(i) mean(random_values_posterior[1:i]))
sample_stds <- sapply(2:nDraws, function(i) sd(random_values_posterior[1:i]))

mean_sd_data <- data.frame(Draws = 2:nDraws, Mean = sample_means, Std = sample_stds)

plot(mean_sd_data$Draws, mean_sd_data$Mean, type = 'l', xlab = "Draws", ylab = "Mean",
     main = "Convergence to True Mean")
abline(h = true_mean, col = "red", lwd = 2, lty = 2)
grid()

```

## Convergence to True Mean

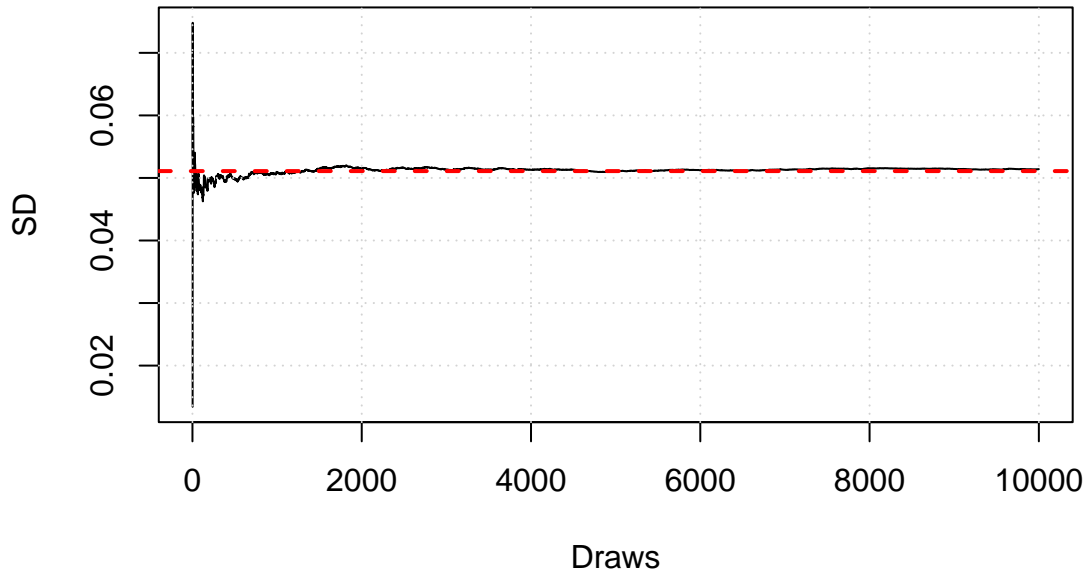


```

plot(mean_sd_data$Draws, mean_sd_data$Std, type = 'l', xlab = "Draws", ylab = "SD",
     main = "Convergence to True SD")
abline(h = sqrt(true_variance), col = "red", lwd = 2, lty = 2)
grid()

```

## Convergence to True SD



From the graphs above it can be observed that posterior mean and standard deviation approaches to true values as number of samples increases.

### Part 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. [Hint: use `pbeta()`].

**Answer:**

```
posterior_probability <- mean(random_values_posterior > 0.3)
exact_posterior_probability <- 1 - pbeta(0.3, alpha_posterior, beta_posterior)

cat("Posterior probability:", posterior_probability, "\n",
    "Exact Value:", exact_posterior_probability)
```

```
## Posterior probability: 0.8303
```

```
## Exact Value: 0.8285936
```

Exact value from the Beta posterior distribution for  $Pr(\theta > 0.3|y)$  is 0.8285936 and probability from the draws is 0.8303 which is close to exact value.

### Part c)

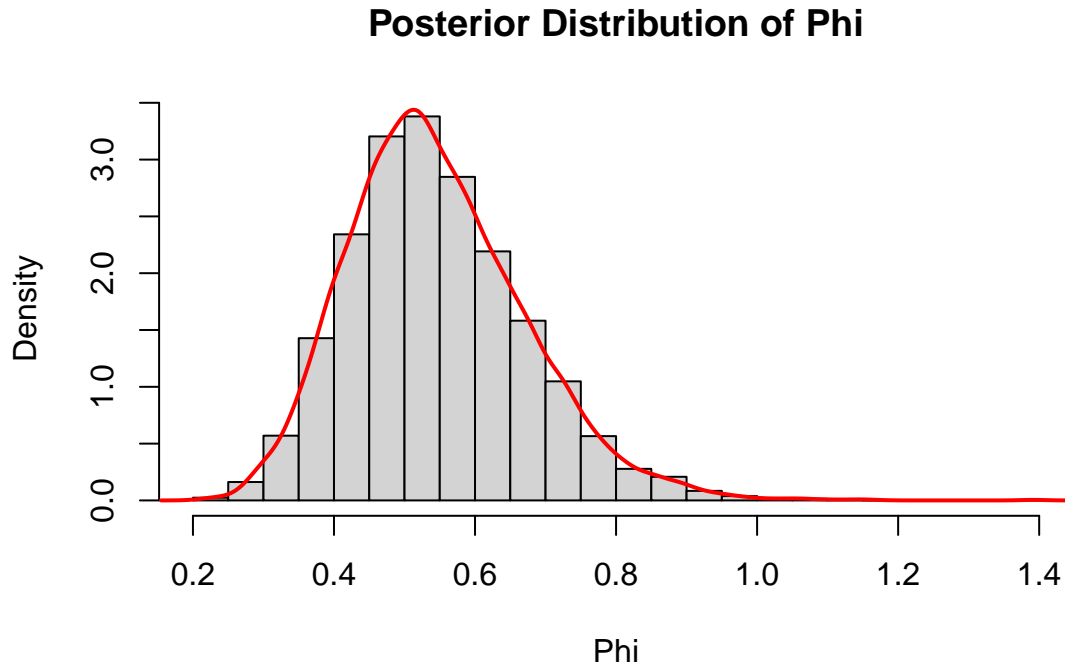
**Question:** Draw 10000 random values from the posterior of the odds  $\phi = \frac{\theta}{1-\theta}$  by using the previous random draws from the Beta posterior for  $\theta$  and plot the posterior distribution of  $\phi$ . [Hint: `hist()` and `density()` can be utilized].

**Answer:**

Now we calculated the posterior of the odds using  $\phi = \frac{\theta}{1-\theta}$  by using the previous random draws. The histogram for posterior distribution of  $\phi$  can be seen below

```
# Calculate the odds phi = theta / (1 - theta)
phi_samples <- random_values_posterior / (1 - random_values_posterior)
```

```
# Plot the posterior distribution of phi
hist(phi_samples, breaks = 30, freq = FALSE,
     main = "Posterior Distribution of Phi", xlab = "Phi")
lines(density(phi_samples), col = "red", lwd = 2)
```



## Question 2: Log-normal distribution and the Gini coefficient

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  $\log N(\mu, \sigma^2)$  has density function

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

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

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

### Part a)

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

**Answer:**

```

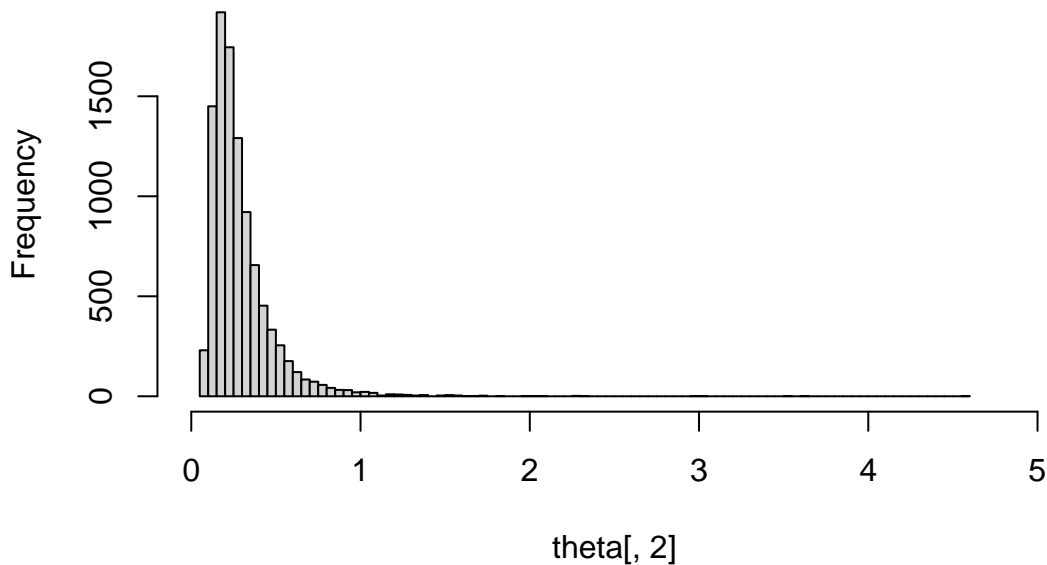
mu <- 3.6
obs <- c(33, 24, 48, 32, 55, 74, 23, 17)
n <- length(obs)
log_obs <- log(obs)
mean_data <- mean(log_obs) # mean of the observations
tau2 <- sum((log(obs) - mu)^2) / n # variance of the observations

nDraws <- 10000
theta <- matrix(0, nDraws, 2)
theta[,2] <- (n*tau2) / rchisq(nDraws, n)

# Plotting the histogram of sigma^2-draws
hist(theta[,2], breaks = 100, main = 'Histogram of sigma^2 draws', xlim = c(0,5))

```

**Histogram of sigma<sup>2</sup> draws**



#### Part b)

**Question:** The most common measure of income inequality is the Gini coefficient,  $G$ , where  $0 \leq G \leq 1$ .  $G = 0$  means a completely equal income distribution, whereas  $G = 1$  means complete income inequality (see e.g., Wikipedia for more information about the Gini coefficient). It can be shown that  $G = 2\Phi(\sigma/\sqrt{2}) - 1$  when incomes follow a  $\log N(\mu, \sigma^2)$  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 coefficient  $G$  for the current data set.

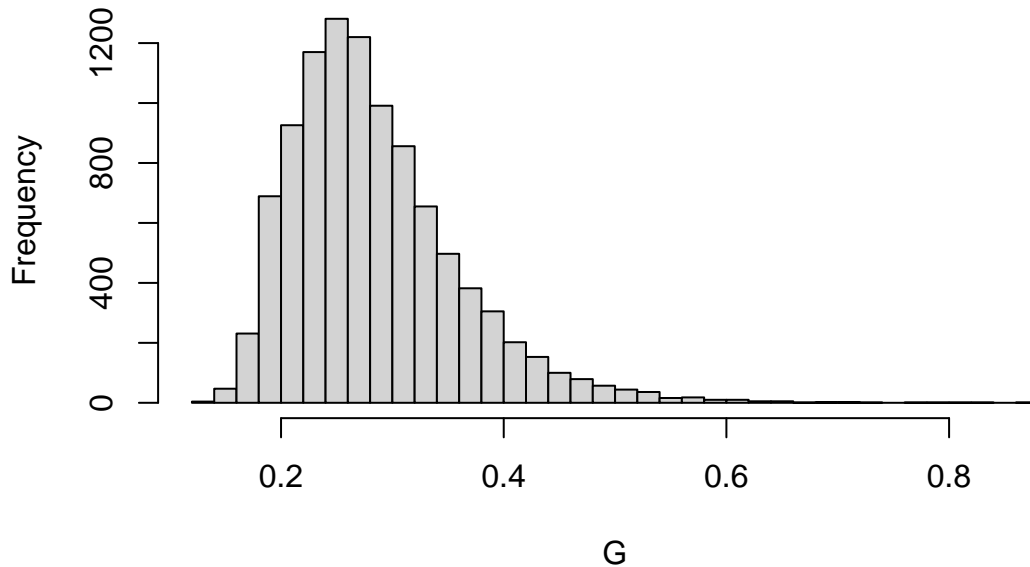
**Answer:** Now we calculated the Gini coefficient using formula  $G = 2\Phi(\sigma/\sqrt{2}) - 1$ . Note that  $G$  takes value between 0 and 1.  $G = 0$  means completely equal income distribution whereas  $G = 1$  means complete income inequality. Most of the values in the histogram below are between 0.05 and 0.15.

```

G <- 2 * pnorm(sqrt(theta[,2]) / sqrt(2)) - 1
hist(G, breaks = 50, main = "Histogram of Gini Index")

```

## Histogram of Gini Index



### Part 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% of the posterior probability mass to the left of  $a$ , and 2.5% to the right of  $b$ .

**Answer:** %95 equal tail credible interval for  $G$  can be seen below.

```
# Compute the lower and upper bounds of the credible interval
lower_bound <- quantile(G, 0.025)
upper_bound <- quantile(G, 0.975)

cat("95% Equal Tail Credible Interval for G:", round(lower_bound, 4), "-",
    round(upper_bound, 4), "\n")
```

```
## 95% Equal Tail Credible Interval for G: 0.1785 - 0.4686
```

```
cat("Difference between upper and lower bound:", upper_bound-lower_bound)
```

```
## Difference between upper and lower bound: 0.2900872
```

### Part 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:** HDPI interval for  $G$  can be seen below

```
density_estimate <- density(G)
sorted_density <- sort(density_estimate$y, decreasing = TRUE) # Sort density values
cumulative_density <- cumsum(sorted_density)

cdf <- cumulative_density[(cumulative_density < sum(sorted_density) * 0.95)]
idx <- density_estimate$x[order(density_estimate$y,decreasing=TRUE)][1:length(cdf)]
```

```

lower_bound_hpdi <- min(idx)
upper_bound_hpdi <- max(idx)

cat("HPDI for G:", round(lower_bound_hpdi, 4), "-", round(upper_bound_hpdi, 4), "\n")

## HPDI for G: 0.1601 - 0.4348

cat("Difference between upper and lower bound:", upper_bound_hpdi-lower_bound_hpdi)

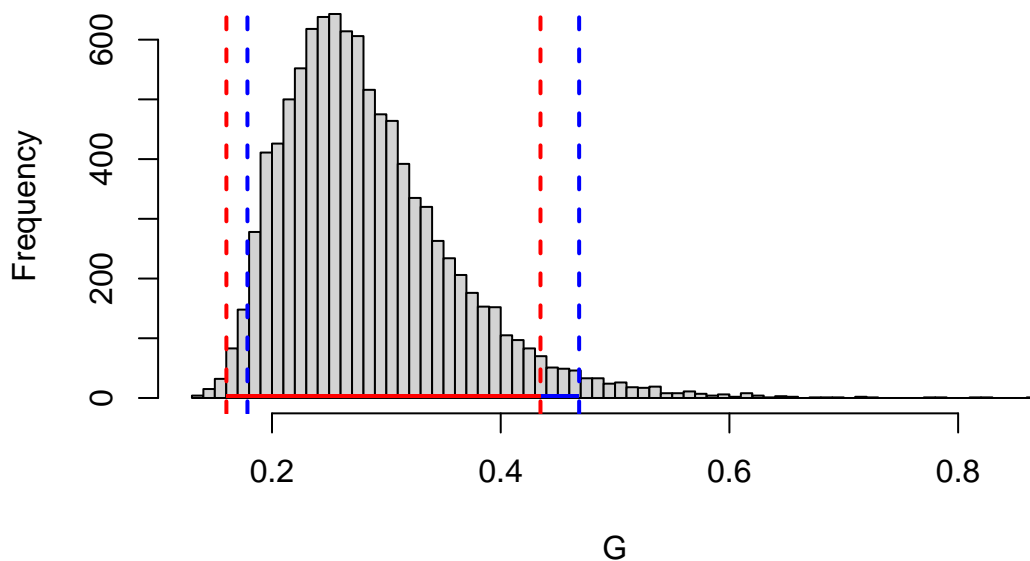
## Difference between upper and lower bound: 0.2746462

hist(G, breaks = 100, main = "Credible Interval and HPDI")
abline(v = lower_bound, col = "blue", lwd = 2, lty = 2)
abline(v = upper_bound, col = "blue", lwd = 2, lty = 2)
rect(lower_bound, 0, upper_bound,
      max(hist(G, breaks = 100, plot = FALSE)$density), col = "blue", border = NA)

abline(v = lower_bound_hpdi, col = "red", lwd = 2, lty = 2)
abline(v = upper_bound_hpdi, col = "red", lwd = 2, lty = 2)
rect(lower_bound_hpdi, 0, upper_bound_hpdi,
      max(hist(G, breaks = 100, plot = FALSE)$density), col = "red", border = NA)

```

### Credible Interval and HPDI



In the graph above, blue line shows the %95 credible interval and red line shows the HPDI. The difference between upper and lower bound for each interval is close to each other but HPDI is a little more narrower because posterior distribution is narrow.

### 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. 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  $(\mu, \kappa)$  are independent observations from the following von Mises distribution:

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

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

### Part 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  $\kappa$  for the wind direction data over a fine grid of  $\kappa$  values. [Hint: you need to normalize the posterior distribution of  $\kappa$  so that it integrates to one.]

**Answer:** The likelihood and prior function is follows;

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

$$p(\kappa) = \lambda \cdot \exp(-\lambda\kappa)$$

Then the posterior distribution is as follows;

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

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

We can eliminate  $(\frac{\lambda}{2\pi})^n$ . Posterior is distribution is proportional to the expression below

$$p(\kappa|y, \mu) \propto \frac{\exp(\sum_{i=1}^n \kappa \cdot \cos(y_i - \mu) - \kappa \cdot \lambda)}{(I_0(\kappa))^n}$$

We know that  $\mu = 2.4$  and  $\lambda = 0.5$ .

$$p(\kappa|y, \mu) \propto \frac{\exp(\sum_{i=1}^n \kappa \cdot \cos(y_i - 2.4) - \kappa \cdot 0.5)}{(I_0(\kappa))^n}$$

```
posterior_fnc <- function(k, y, mu, lambda) {
  n <- length(y)
  sum_cos <- sum(cos(y - mu))
  numerator <- exp(k * sum_cos - k * lambda)
  denominator <- (2*pi*besselI(k, 0))^n
  result <- numerator / denominator
  return(result)
}
```



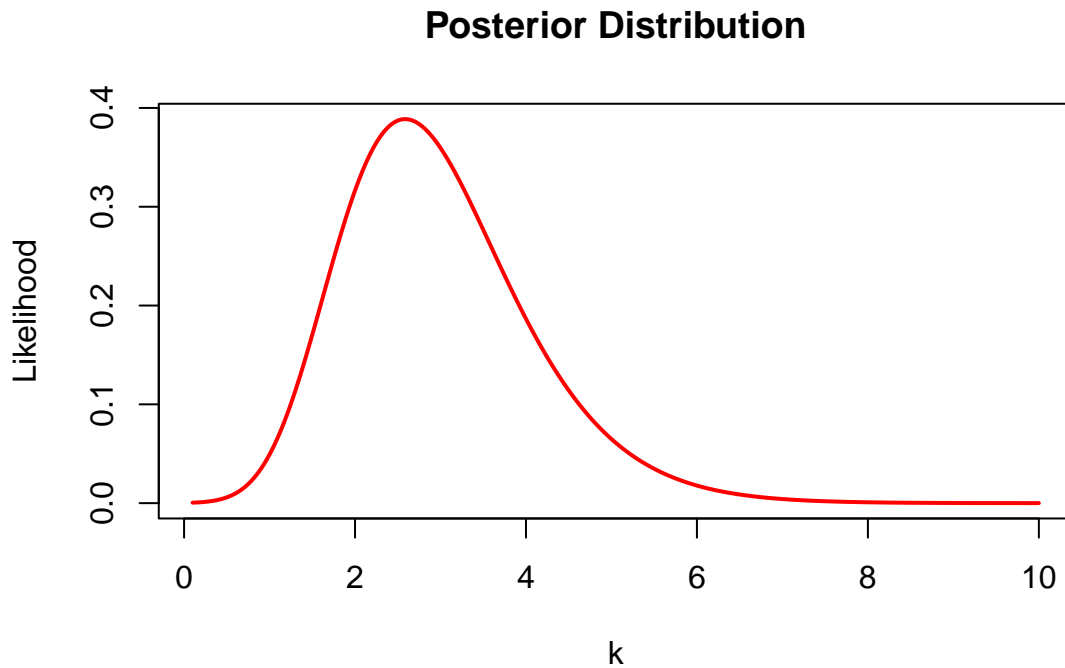
```

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
lambda <- 0.5
k <- seq(0.1, 10, length.out = 1000)
posterior_values <- posterior_fnc(k, y, mu, lambda)

# normalize the posterior values to integrate to one
posterior_values_normalized <- posterior_values / sum(posterior_values * diff(k[1:2]))

plot(k, posterior_values_normalized, type = 'l', lwd = 2, col = 'red',
     main = "Posterior Distribution", xlab = 'k', ylab = 'Likelihood')

```



Part b)

**Question:** Find the (approximate) posterior mode of  $\kappa$  from the information in (a).

**Answer:** The posterior mode refers to the value of the parameter that maximizes the posterior probability distribution.

```

max_posterior_index <- which.max(posterior_values_normalized)
cat("k value that maximizes the likelihood:", k[max_posterior_index])

```

## k value that maximizes the likelihood: 2.587387

$\kappa = 2.587387$  maximizes the posterior probability

```

plot(k, posterior_values_normalized, type = 'l', lwd = 2, col = 'red',
     main = "Posterior Distribution", xlab = 'k', ylab = 'Likelihood')
abline(v = 2.587387, col = "blue", lwd = 2, lty = 2)

```

## Posterior Distribution

