Section - B Summative Assessment

Thrisha Rajkumar

2024-11-28

Section B (30 marks)

B.1

Suppose a product is being sold in a supermarket. We are interested in knowing how quickly the product returns to the shelf again after it is sold out. Let X be a continuous random variable denoting the length of time between the time point at which it is sold out and the time point at which it is placed on the shelf again. So X should be a non-negative number, and $X = 0$ means that the product gets on the shelf immediately after it is sold out. Here, we assume that the probability density function of X is given by

$$p_\lambda(x) = \begin{cases} ae^{-\lambda(x-b)} & \text{if } x \geq b, \\ 0 & \text{if } x < b, \end{cases}$$

where $b > 0$ is a known constant, $\lambda > 0$ is a parameter of the distribution, and a is to be determined by λ and b.

(1) First, determine the value of a: derive a mathematical expression of a in terms of λ and/or b.

Answer (1)

Probability Density Function Analysis We need to determine the value of a and Also, Derive an expression of a in terms of λ and/or b

To determine the value of a using the probability density function firstly let us solve the Probability density function of X

Given:

According to the Question the Probability Density function is defined as:

$$p_\lambda(x) = \begin{cases} ae^{-\lambda(x-b)} & \text{if} x \geq b, \\ 0 & \text{if} x < b, \end{cases}$$

To check the validity of $p_\lambda(x)$ Probability density function, we need to make sure the integration is equal to 1 which is as below:

$$\int_{-\infty}^{\infty} p_\lambda(x)\, dx = 1$$

Now, as $p_\lambda(x) = 0$ for $x < b$ therefore we can make the limit $-\infty$ to b.

$$\int_b^\infty p_\lambda(x)\,dx = 1$$

By solving and simplifying the probability density function into the integral we get:

$$\int_b^\infty ae^{-\lambda(x-b)}\,dx = 1$$

Computing the integral by changing the variables ->

Changing the variables in the integral to simplify the integral Let $u = x - b$ i.e., $du = dx$ and, when $x = b$, $u = 0$.

Now the limits of the integral will become : 1. $x = b$, $u = 0$ 2. $x = \infty$, $u = \infty$

Therefore, the integral after changing the variables is:

$$\int_0^\infty ae^{-\lambda u}\,du$$

- Evaluating the integral

$$\int_0^\infty e^{-\lambda u}\,du = \left[-\frac{1}{\lambda}e^{-\lambda u}\right]_0^\infty = 0 - \left(-\frac{1}{\lambda}\right) = \frac{1}{\lambda}$$

$$\int_0^\infty e^{-\lambda u}\,du = \frac{1}{\lambda}$$

Multiplying by a, we get:

$$a \cdot \frac{1}{\lambda} = 1$$

Therefore, value of a is given by:

$$a = \lambda$$

Solution:

The value of a in terms of λ and/or b is:

$$\boxed{a = \lambda}$$

(2) Derive a formula for the population mean and standard deviation of the random variable X with parameter λ.

Answer (2) Derivation of Population Mean and Standard Deviation:

Population Mean

The population mean (expected value) of a continuous random variable X is given by the formula:

$$E(X) = \int_{-\infty}^{\infty} xp_\lambda(x)dx$$

Given the probability density function (PDF):

$$p_\lambda(x) = \begin{cases} \lambda e^{-\lambda(x-b)} & \text{if } x \geq b, \\ 0 & \text{if } x < b, \end{cases}$$

We can solve the value of PDF in Population Mean equation:

$$E(X) = \int_b^\infty x\lambda e^{-\lambda(x-b)}\, dx$$

Similarly like question 1, Changing of Variable in the integral in order to simplify the equation and solve accordingly.

Let us assume:

$$u = x - b \quad \text{thus} \quad x = u + b \quad \text{and} \quad dx = du$$

The limits of integration become:

1. When $x = b$, $u = 0$
2. When $x = \infty$, $u = \infty$

Substituting these into the integral gives:

$$E(X) = \int_0^\infty (u + b)\lambda e^{-\lambda u}\, du$$

Splitting the integral by expanding it from the addition of $u + b$

This integral can be split into two parts:

$$E(X) = \lambda \int_0^\infty u e^{-\lambda u}\, du + b\lambda \int_0^\infty e^{-\lambda u}\, du$$

Calculating the first part of the integral:

$\int_0^\infty u e^{-\lambda u}\, du$:

Let $v = u$ and $dw = e^{-\lambda u}\, du$. Then, $dv = du$ and $w = -\frac{1}{\lambda} e^{-\lambda u}$.

Using integration by parts:

$$\int u e^{-\lambda u}\, du = -\frac{1}{\lambda} u e^{-\lambda u}\Big|_0^\infty + \frac{1}{\lambda} \int e^{-\lambda u}\, du$$

The first term evaluates to zero as $ue^{-\lambda u}$ approaches zero as u approaches both 0 and ∞. The remaining integral is:

$$\int e^{-\lambda u}\, du = -\frac{1}{\lambda} e^{-\lambda u}\Big|_0^\infty = \frac{1}{\lambda}$$

Therefore, we have the first half of the integral as:

$$\int_0^\infty ue^{-\lambda u}\,du = \frac{1}{\lambda^2}$$

Calculating the second half of the integral in population mean:

$\int_0^\infty e^{-\lambda u}\,du$: - As computed earlier:

$$\int_0^\infty e^{-\lambda u}\,du = \frac{1}{\lambda}$$

Substituting the computations Back into the Mean i.e. adding the firt part of the integral and the second part for our expression for $E(X)$:

$$E(X) = \lambda \cdot \frac{1}{\lambda^2} + b\lambda \cdot \frac{1}{\lambda}$$

Simplifying gives:

$$E(X) = \frac{1}{\lambda} + b$$

Thus, the formula for the population mean is:

$$\boxed{E(X) = b + \frac{1}{\lambda}}$$

-(1) (equation 1)

Now, Deriving the Population Standard Deviation

The population standard deviation σ is defined as the square root of the variance $Var(X)$, which is computed using the formula for $Var(X)$ which is below:

$$Var(X) = E(X^2) - (E(X))^2$$

-(2) (equation 2)

Computing $E(X^2)$

To find $E(X^2)$, we use:

$$E(X^2) = \int_{-\infty}^{\infty} x^2 p_\lambda(x)\,dx$$

Computing the value of pdf in the $E(X^2)$

$$E(X^2) = \int_b^\infty x^2 \lambda e^{-\lambda(x-b)}\,dx$$

Changing the variables again in order to calculate the integral function.

Using the same change of variable $u = x - b$:

$$E(X^2) = \int_0^\infty (u+b)^2 \lambda e^{-\lambda u}\,du$$

Expanding $(u+b)^2$:

$$E(X^2) = \int_0^\infty (u^2 + 2bu + b^2)\lambda e^{-\lambda u}\, du$$

Splitting the integral into 3 parts as below and solving each and combining later for easier calculations

$$E(X^2) = \lambda \int_0^\infty u^2 e^{-\lambda u}\, du + 2b\lambda \int_0^\infty u e^{-\lambda u}\, du + b^2\lambda \int_0^\infty e^{-\lambda u}\, du$$

Calculating the first integral: $\int_0^\infty u^2 e^{-\lambda u}\, du$:

We can directly calculate it Using the formula:

$$\int_0^\infty x^n e^{-\lambda x}\, dx = \frac{n!}{\lambda^{n+1}}$$

For $n = 2$:

$$\int_0^\infty u^2 e^{-\lambda u}\, du = \frac{2!}{\lambda^3} = \frac{2}{\lambda^3}$$

Similarly,

- $\int_0^\infty u e^{-\lambda u}\, du = \frac{1}{\lambda^2}$
- $\int_0^\infty e^{-\lambda u}\, du = \frac{1}{\lambda}$

Combining the parts of the integral into $E(X^2)$

Now substituting back gives us the result as:

$$E(X^2) = \lambda \cdot \frac{2}{\lambda^3} + 2b\lambda \cdot \frac{1}{\lambda^2} + b^2\lambda \cdot \frac{1}{\lambda}$$

Simplifying this will give us:

$$E(X^2) = \frac{2}{\lambda^2} + \frac{2b}{\lambda} + b^2$$

-(3) (equation 3)

Now we can use this above equation -(3) and -(1) in equation -(2)

Now we can compute the variance as

$$Var(X) = E(X^2) - (E(X))^2$$

Where: For $(E(X))^2$:

$$(E(X))^2 = \left(b + \frac{1}{\lambda}\right)^2 = b^2 + \frac{2b}{\lambda} + \frac{1}{\lambda^2}$$

Substituting $E(X^2)$ in Equation - (2):

$$Var(X) = \left(\frac{2}{\lambda^2} + \frac{2b}{\lambda} + b^2\right) - \left(b^2 + \frac{2b}{\lambda} + \frac{1}{\lambda^2}\right)$$

Cancelling out terms gives us the result:

$$Var(X) = \frac{2}{\lambda^2} - \frac{1}{\lambda^2} = \frac{1}{\lambda^2}$$

Calculating the Standard Deviation:

The standard deviation is:

$$\sigma = \sqrt{Var(X)} = \sqrt{\frac{1}{\lambda^2}} = \frac{1}{\lambda}$$

Solution: Thus, the formulas for the population mean, standard deviation and Variance of the random variable X are:

$$\boxed{E(X) = b + \frac{1}{\lambda}}$$

$$\boxed{\sigma = \frac{1}{\lambda}}$$

$$\boxed{Var(X) = \frac{1}{\lambda^2}}$$

(3) Derive a formula for the cumulative distribution function and the quantile function for the random variable X with parameter λ.

Deriving the Quantile Function $Q(p)$

The quantile function $Q(p)$ represents the value of the random variable X that corresponds to a given cumulative probability p in the interval $[0, 1]$.

Given the cumulative distribution function (CDF) for X:

$$F(x) = 1 - e^{-\lambda(x-b)} \quad \text{for } x \geq b,$$

we solve for x in terms of p (where $F(x) = p$) to obtain the quantile function:

$$p = 1 - e^{-\lambda(x-b)}.$$

Rearranging to isolate x, we get:

$$p - 1 = -e^{-\lambda(x-b)}$$

$$1 - p = e^{-\lambda(x-b)}$$

$$\ln(1 - p) = -\lambda(x - b)$$

$$x = b - \frac{\ln(1-p)}{\lambda}.$$

Thus, the quantile function $Q(p)$ is:

$$Q(p) = b - \frac{\ln(1-p)}{\lambda}.$$

Now, analysing this function for different values of p in Quantile Function $Q(p)$.

- Case 1: $p = 0$

When $p = 0$, substitute $p = 0$ into the quantile function:

$$Q(0) = b - \frac{\ln(1-0)}{\lambda} = b - \frac{\ln(1)}{\lambda} = b - \frac{0}{\lambda} = b.$$

Therefore, $Q(0) = b$. This makes sense because $p = 0$ represents the minimum possible value, and in this distribution, X cannot be less than b.

- Case 2: $0 < p < 1$

For $0 < p < 1$, we use the general formula:

$$Q(p) = b - \frac{\ln(1-p)}{\lambda}.$$

Here, p is a cumulative probability, and $Q(p)$ is the value of X that corresponds to that percentile. Since p increases from 0 to 1, so does $Q(p)$, which further moves away from b as p approaches 1.

- Case 3: $p = 1$

When $p = 1$, substitute $p = 1$ into the quantile function:

$$Q(1) = b - \frac{\ln(1-1)}{\lambda} = b - \frac{\ln(0)}{\lambda}.$$

Since $\ln(0)$ tends toward $-\infty$, $Q(1)$ approaches infinity. This implies that as p gets closer and closer to 1, $Q(p)$ can get arbitrarily large. Therefore:

$$Q(1) \to \infty.$$

Solution:

The quantile function $Q(p)$ for the random variable X in each case is:

$$\boxed{Q(p) = \begin{cases} b & \text{if } p = 0, \\ b - \frac{\ln(1-p)}{\lambda} & \text{if } 0 < p < 1, \\ \infty & \text{if } p = 1. \end{cases}}$$

(4) Suppose that $X_1, \ldots, X_n$ are independent copies of X with the unknown parameter $\lambda > 0$. What is the maximum likelihood estimate λ_{MLE} for λ?

Problem 4: Deriving λ_{MLE}

Given a sample of independent observations $X_1, X_2, \ldots, X_n$, we need to find the Maximum Likelihood Estimate (MLE) for the parameter λ in the probability density function:

$$p_\lambda(x) = \begin{cases} \lambda e^{-\lambda(x-b)} & \text{if } x \geq b, \\ 0 & \text{if } x < b. \end{cases}$$

where b is a known constant (300 seconds) and λ is the parameter to estimate.

Deriving for λ_{MLE}

Likelihood Function derivation:

The likelihood function for λ, given observations $x_1, x_2, \ldots, x_n$, is:

$$L(\lambda) = \prod_{i=1}^{n} p_\lambda(x_i) = \prod_{i=1}^{n} \lambda e^{-\lambda(x_i - b)}$$

$$L(\lambda) = \lambda^n e^{-\lambda \sum_{i=1}^{n} (x_i - b)}$$

Taking the natural logarithm of $L(\lambda)$ from the above equation:

$$\ell(\lambda) = n \ln(\lambda) - \lambda \sum_{i=1}^{n} (x_i - b)$$

Differentiating and calculating for (:

$$\frac{d\ell(\lambda)}{d\lambda} = \frac{n}{\lambda} - \sum_{i=1}^{n} (x_i - b) = 0$$

Solving this gives us:

$$\lambda_{\text{MLE}} = \frac{n}{\sum_{i=1}^{n} (x_i - b)}$$

Solution:

The MLE for λ is:

$$\boxed{\lambda_{\text{MLE}} = \frac{n}{\sum_{i=1}^{n} (x_i - b)}}$$

Now download the .csv file entitled "supermarket_data_2024" from the Assessment section within Blackboard. The .csv file contains data on the length of time (in seconds) taken by a product to get on the shelf again after being sold out. So the sample is a sequence of time lengths. Let's model the sequence of time lengths in our sample as independent copies of X (where X is the random variable mentioned above) with parameter λ and known constant $b = 300$ (seconds).

(5) Given the sample, compute and display the maximum likelihood estimate λ_{MLE} of the parameter λ.

The Maximum Likelihood Estimate λ_{MLE} for the parameter λ can be computed by the formula which we calculated in the previous question:

$$\lambda_{\text{MLE}} = \frac{n}{\sum(X_i - b)}$$

Where:

- n is the number of observations (sample size),
- X_i is the recorded duration of each product,
- b is the known constant or is the fixed value (according to the question, $b = 300$ seconds).

The formula is calculating the rate at which products are restocked, dividing the number of observations by the sum of the differences between each observed time length and the constant b.

This formula gives us the estimated value of λ, which represents the rate at which the products are restocked after being sold out as per the question lets implement it in the supermarket_data_2024.csv fle:

```
# Loading the necesary readr library for reading and storing the data from the supermarket_data_2024.cs
library(readr)

# Storing the data from supermarket_data_2024.csv into the dataframe supermarket_data_2024_df
supermarket_data_2024_df <- read_csv("supermarket_data_2024.csv")
```

```
## Rows: 2500 Columns: 1
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (1): TimeLength
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# displaing the first few rows of the supermarket_data_2024_df dataset
head(supermarket_data_2024_df)
```

```
## # A tibble: 6 x 1
##   TimeLength
##        <dbl>
## 1       304.
## 2       504.
## 3       388.
## 4       327.
## 5       311.
## 6       340.
```

```
# According to the question the value of b is 300 and is set as a constant below:
b <- 300

# defining the function to calculate lambda MLE as per our calculation:
lambda_mle <- function(sample) {
  n <- length(sample)
  return(n / sum(sample - b))
}

# creating a data frame of the column TimeLength in supermarket_data_2024_df into time_lengths
```

```r
time_lengths <- supermarket_data_2024_df$TimeLength

# Calculating the lambda_MLE using the time lengths data
lambda_MLE <- lambda_mle(time_lengths)

# displaying the estimated value of the Lamba_MLE
print(lambda_MLE)
```

```
## [1] 0.01988426
```

(6) Apply the method of Bootstrap confidence interval to obtain a confidence interval for λ with a confidence level of 95%. To compute the Bootstrap confidence interval, the number of resamples (i.e., subsamples that are generated to compute the bootstrap statistics) should be set to 10000.

Next, conduct a simulation study to explore the behaviour of the maximum likelihood estimator:

```r
# loading ggplot2 library for plotting the behaviour of Maximum likelihood estimator:

library(ggplot2)

# Known constant b = 300 from question above.
b <- 300

# Computing the MLE - Maximum Likelihood Estimator for lambda using the given data
lambda_mle <- function(data, b) {
  n <- length(data)
  mle <- n / sum(data - b)
  return(mle)
}

# Bootstrap resampling function
bootstrap_ci <- function(data, b, n_resamples = 10000, conf_level = 0.95) {
  mle_values <- numeric(n_resamples)
  for (i in 1:n_resamples) {
    sample_resample <- sample(data, length(data), replace = TRUE)
    mle_values[i] <- lambda_mle(sample_resample, b)
  }

  # Calculating the lower_bound and upper_bound
  lower_bound <- quantile(mle_values, (1 - conf_level) / 2)
  upper_bound <- quantile(mle_values, 1 - (1 - conf_level) / 2)

  return(c(lower_bound, upper_bound))
}

# applying the bootstrap method for computing the 95% confidence interval for
bootstrap_result <- bootstrap_ci(supermarket_data_2024_df$TimeLength, b)
bootstrap_result
```

```
##       2.5%      97.5%
## 0.01911752 0.02071190
```

```r
# displaying the confidence interval
cat("Bootstrap 95% Confidence Interval for  : [", bootstrap_result[1], ", ", bootstrap_result[2], "]\n")
```

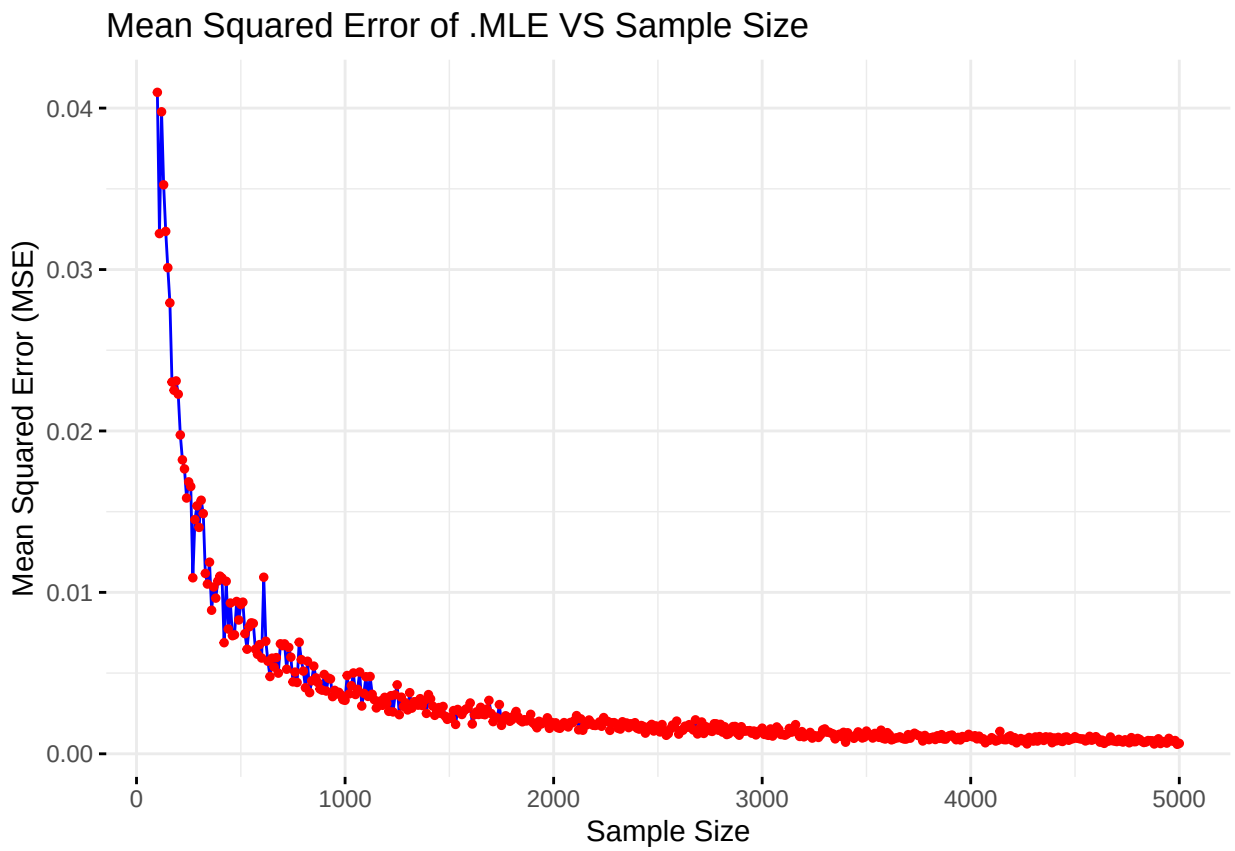
```
## Bootstrap 95% Confidence Interval for  : [ 0.01911752 ,  0.0207119 ]
```

(7) Conduct a simulation study to explore the behaviour of the maximum likelihood estimator λ_{MLE} for λ on simulated data $X_1, \ldots, X_n$ (as independent copies of X with parameter λ) according to the following instructions. Let $b = 0.01$ and the true parameter be $\lambda = 2$. Generate a plot of the mean squared error as a function of the sample size n. You should consider sample sizes from 100 to 5000 in increments of 10. For each sample size, consider 100 trials. In each trial, generate a random sample $X_1, \ldots, X_n$ (as independent copies of X with parameter $\lambda = 2$), and then compute the maximum likelihood estimate λ_{MLE} for λ based upon the sample. Display a plot of the mean square error of λ_{MLE} as an estimator for λ as a function of the sample size n.

```r
# loading ggplot2 library for the plotting of the mean squared error as a function of the sample size
library(ggplot2)

# setting the parameters as per the question
lambda_true <- 2    # True value of
b <- 0.01           # known constant of b = 0.01 as given in the question
sample_sizes <- seq(100, 5000, by = 10)  # the Sample size of values between 100 to 5000 by incrementin
n_trials <- 100     # Number of trials for each sample size = 100 given in the question

# creating a function generate_simulated_data which is calculating the simulation data from the values
generate_simulated_data <- function(n, lambda_true, b) {
  # Generating n random samples from the exponential distribution with rate
  simulated_data <- rexp(n, rate = lambda_true) + b
  return(simulated_data)
}

# Calculating the length of data nd mle using the Function lambda_mle
lambda_mle <- function(data, b) {
  n <- length(data)
  mle <- n / sum(data - b)
  return(mle)
}

# FComputing the Mean Squared Error (MSE) for a range of sample sizes using the function compute_mse_la
compute_mse_lambda_mle <- function(sample_sizes, n_trials, lambda_true, b) {
  mse_values <- numeric(length(sample_sizes))
  #outer looping through the sample sizes
  for (i in 1:length(sample_sizes)) {
    sample_size <- sample_sizes[i]
    mle_values <- numeric(n_trials)

    # generating and computing  MLE for each trial or data point by Inner Loop for Trials
    for (j in 1:n_trials) {
      #Generating Simulated Data using the generate_simulated_data function
      simulated_data <- generate_simulated_data(sample_size, lambda_true, b)
```

```
      #Initializing the Storage for MLE Values
      mle_values[j] <- lambda_mle(simulated_data, b)
    }

    # Computing MSE for the current sample size
    mse_values[i] <- mean((mle_values - lambda_true)^2)
  }

  return(mse_values)
}

# Computing MSE for different sample sizes
mse_values <- compute_mse_lambda_mle(sample_sizes, n_trials, lambda_true, b)

# Plotting the Mean Squared Error (MSE) against Sample Size using the line graph and point graph with t
ggplot(data = data.frame(SampleSize = sample_sizes, MSE = mse_values), aes(x = SampleSize, y = MSE)) +
  geom_line(color = "blue", size = 0.5) + #Using size =0.5 and colour blue for the line plot
  geom_point(color = "red", size = 1.0) +# using size = 1.0 and colour red for points plotting
  labs(title = "Mean Squared Error of  MLE VS Sample Size",
       x = "Sample Size",
       y = "Mean Squared Error (MSE)") +
  theme_minimal() +
  theme(
    #This is used to show th ticks or marks against each value in the data
    axis.ticks = element_line(color = "black")
  )
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



B.2 Consider a bag of a red balls and b blue balls (the bag has $a + b$ balls in total), where $a \geq 1$ and $b \geq 1$. We randomly draw two balls from the bag without replacement. That means, we draw the first ball from the bag and, WITHOUT returning the first ball to the bag, we draw the second one. Each ball has an equal chance of being drawn.

Now we record the colour of the two balls drawn from the bag, and let X denote the number of red balls minus the number of blue balls. So X is a discrete random variable. For example, if we draw one red ball and one blue ball, then $X = 0$.

(1) Give a formula for the probability mass function $p_X : R \to [0, 1]$ of X.

Answer (1)

Given in the Question: The bag contains a red balls and b blue balls, and $a \geq 1$ and $b \geq 1$

2 balls are drawn from the bag without replacement and each ball has each chance of being drawn.

Total no. of balls in the bag $= a + b$

- $X = a + a = 2$a: both balls drawn are red.
- $X = b + b = 2$b: both balls drawn are blue.

X is considered to be a number of red balls a minus the number of blue balls b.

The possible outcomes for X can be:

- $X = 2$: both balls drawn are red.
- $X = 0$: one red ball and one blue ball are drawn.
- $X = -2$: both balls drawn are blue.

-> Derivation of the Probability Mass Function

Calculating the probabilities for each possible outcomes of X

-> Probability of two red balls drawn is: $(X = 2)$

$$P(X = 2) = \frac{a}{a+b} \times \frac{a-1}{a+b-1}$$

the probability of a red balls divided by the total number of balls multiplied by the probability of a - 1 as the ball is drawn without replacement divided by total number of balls - 1 as the first ball is already taken.

-> probability of drawing two blue balls $(X = -2)$

$$P(X = -2) = \frac{b}{a+b} \times \frac{b-1}{a+b-1}$$

Again, similar to drawing a or two red balls, the probability of a blue ball divided total number of balls and multiplied by b-1 as one ball is drawn without replacement and divided by total number of balls which is a+b-1.

-> Probability of red ball and one blue ball $(X = 0)$

$$P(X = 0) = \frac{a}{a+b} \times \frac{b}{a+b-1} + \frac{b}{a+b} \times \frac{a}{a+b-1}$$

Logic behind this is one red ball a taken from the total number of balls which is a+b and then next ball drawn as a blue ball b without replacement in the total which is a+b-1. this is one case and the other case can be the first ball taken out from the bag can be blue ball b from the total a+b and the second ball taken out without replacement can be a which is a red ball from the total now which is a+b-1 Therefore, we are adding up the two cases to get the total probability of X=0.

$$P(X = 0) = \frac{a(a+b-1) \cdot b(a+b)}{(a+b)(a+b-1)} + \frac{b(a+b-1) \cdot a(a+b)}{(a+b)(a+b-1)}$$

Simplifying the equation of X= 0 we get:

$$P(X = 0) = \frac{2ab}{(a+b)(a+b-1)}$$

Solution : -> Therefore, Probability Mass Function

$$\boxed{p_X(x) = \begin{cases} \frac{a(a-1)}{(a+b)(a+b-1)} & \text{if } x = 2 \\ \frac{2ab}{(a+b)(a+b-1)} & \text{if } x = 0 \\ \frac{b(b-1)}{(a+b)(a+b-1)} & \text{if } x = -2 \\ 0 & \text{otherwise} \end{cases}}$$

-> Implementation of the Probability mass function for X in Code with example of a and b values as below:

```r
compute_pmf_X <- function(a, b, x) {
  if (x == 2) {
    return(a * (a - 1) / ((a + b) * (a + b - 1)))
  } else if (x == 0) {
    return(2 * a * b / ((a + b) * (a + b - 1)))
  } else if (x == -2) {
    return(b * (b - 1) / ((a + b) * (a + b - 1)))
  } else {
    return(0)
  }
}
a <- 6
b <- 10
pmf_2 <- compute_pmf_X(a, b, 2)
pmf_0 <- compute_pmf_X(a, b, 0)
pmf_minus2 <- compute_pmf_X(a, b, -2)

pmf_2
```

```
## [1] 0.125
```

```r
pmf_0
```

```
## [1] 0.5
```

```r
pmf_minus2
```

```
## [1] 0.375
```

The above function is the explaination of the derivation.

(2) Use the probability mass function p_X to obtain an expression of the expectation $E(X)$ of X (i.e., the population mean) in terms of a and/or b.

Required -> expression of the expectation $E(X)$ of X.

According to the question we are drawing two balls from the bag without replacement. As mentioned in the question (1)- X is considered to be a number of red balls a minus the number of blue balls b.

From above answer: possible values of X are: - $X = 2$ both balls are red. - $X = 0$one red and one blue ball - $X = -2$ both balls are blue.

Expectation $E(X)$ of a discrete random variable X

$$E(X) = \sum_x x \cdot p_X(x)$$

-> $p_X(x)$ is the probability mass function of X and x represents the possible outcomes of X

From the last step, we derived the probabilities for each possible outcome value of X:

-> $p_X(2) = \frac{a(a-1)}{(a+b)(a+b-1)}$ (the probability of drawing two red balls) - $p_X(0) = \frac{2ab}{(a+b)(a+b-1)}$ (the probability of drawing one red and one blue ball) - $p_X(-2) = \frac{b(b-1)}{(a+b)(a+b-1)}$ (the probability of drawing two blue balls)

Now lets calculate the expectation $E(X)$ As,

$$E(X) = \sum_x x \cdot p_X(x)$$

let us substitute the values derived above of the possible outcomes of X.

$$E(X) = 2 \cdot p_X(2) + 0 \cdot p_X(0) + (-2) \cdot p_X(-2)$$

Substituting the values of $p_X(2)$, $p_X(0)$, and $p_X(-2)$ in the above equation:

$$E(X) = 2 \cdot \frac{a(a-1)}{(a+b)(a+b-1)} + 0 \cdot \frac{2ab}{(a+b)(a+b-1)} + (-2) \cdot \frac{b(b-1)}{(a+b)(a+b-1)}$$

Since the middle term $0 \cdot p_X(0)$ is zero, we are left with:

$$E(X) = 2 \cdot \frac{a(a-1)}{(a+b)(a+b-1)} - 2 \cdot \frac{b(b-1)}{(a+b)(a+b-1)}$$

$$E(X) = \frac{2}{(a+b)(a+b-1)} [a(a-1) - b(b-1)]$$

$$a(a-1) = a^2 - a \quad \text{and} \quad b(b-1) = b^2 - b$$

Therefore after multiplying and deriving the expectation as $E(X)$

$$E(X) = \frac{2}{(a+b)(a+b-1)} [(a^2 - a) - (b^2 - b)]$$

$$E(X) = \frac{2}{(a+b)(a+b-1)} [a^2 - b^2 - a + b]$$

Solution : The Expression of the Expectation $E(X)$ of X:

$$\boxed{E(X) = \frac{2(a^2 - b^2 - a + b)}{(a+b)(a+b-1)}}$$

The following R code calculates the expectation $E(X)$ given values of a and b with a and b as a = 6 and b =10 as example values:

```
compute_expectation <- function(a, b) {
  numerator <- 2 * (a^2 - b^2 - a + b)
  denominator <- (a + b) * (a + b - 1)
  E_X <- numerator / denominator
  return(E_X)
}
a <- 6
b <- 10
expectation <- compute_expectation(a, b)
expectation
```

```
## [1] -0.5
```

(3) Give an expression of the variance Var(X) of X in terms of a and b.

Required -> the variance Var(X) of X in terms of a and b.

The variance of a discrete random variable X is defined as:

$$\text{Var}(X) = E(X^2) - (E(X))^2$$

-> $E(X^2)$ is the expected value of X^2 -> $E(X)$ is the expectation of X

We have derived the expectation of X that is $E(X)$ in the above question which is

$$E(X) = \frac{2(a^2 - b^2 - a + b)}{(a+b)(a+b-1)}$$

-> for $E(X^2)$

Formula to calculate $E(X^2)$

$$E(X^2) = \sum_x x^2 \cdot p_X(x)$$

In pmf of X the possible outcomes are as below and calculating for $E(X^2)$

-> $X = 2$ –> $x^2 = 4$ -> $X = 0$ –> $x^2 = 0$ -> $X = -2$–> $x^2 = 4$

Now lets calculate the $E(X^2)$ using the formula

$$E(X^2) = \sum_x x^2 \cdot p_X(x)$$

$$E(X^2) = 4 \cdot p_X(2) + 0 \cdot p_X(0) + 4 \cdot p_X(-2)$$

substituting the probabilities of p_X(2), p_X(0), and p_X(-2)

$$E(X^2) = 4 \cdot \frac{a(a-1)}{(a+b)(a+b-1)} + 4 \cdot \frac{b(b-1)}{(a+b)(a+b-1)}$$

$$E(X^2) = \frac{4}{(a+b)(a+b-1)} \left[a(a-1) + b(b-1)\right]$$

$$a(a-1) = a^2 - a \quad \text{and} \quad b(b-1) = b^2 - b$$

$$E(X^2) = \frac{4}{(a+b)(a+b-1)} \left[a^2 - a + b^2 - b\right]$$

So, the expression for $E(X^2)$ is:

$$E(X^2) = \frac{4(a^2 - a + b^2 - b)}{(a+b)(a+b-1)}$$

Lets substitute the values of $E(X^2)$ and $E(X)$ in the Varience formula:

$$\text{Var}(X) = E(X^2) - (E(X))^2$$

$$\text{Var}(X) = \frac{4(a^2 - a + b^2 - b)}{(a+b)(a+b-1)} - \left(\frac{2(a^2 - b^2 - a + b)}{(a+b)(a+b-1)}\right)^2$$

Simplifying for $E(X)^2$

$$\left(\frac{2(a^2 - b^2 - a + b)}{(a+b)(a+b-1)}\right)^2 = \frac{4(a^2 - b^2 - a + b)^2}{(a+b)^2(a+b-1)^2}$$

the variance $\text{Var}(X)$ of X in terms of a and b:

$$\boxed{\text{Var}(X) = \frac{4(a^2 - a + b^2 - b)}{(a+b)(a+b-1)} - \frac{4(a^2 - b^2 - a + b)^2}{(a+b)^2(a+b-1)^2}}$$

-> Just like above code implementation of variance -> $\text{Var}(X)$ of X in terms of a and b with examples of a and b values as a = 6 and b =10.

```
# function for computing the variance Var(X)
compute_variance <- function(a, b) {
  #E(X^2)
  E_X2 <- (4*(a^2-a+b^2-b))/((a+b)*(a+ b- 1))

  #E(X)
  E_X <- (2*(a^2- b^2-a+b))/ ((a+ b)*(a+ b- 1))

  #(E(X))^2
  E_X_squared <- E_X^2

  #Calculating the Variance
  variance_X <- E_X2-E_X_squared
  return(variance_X)
}
a <- 6
b <- 10
variance <- compute_variance(a, b)
variance
```

```
## [1] 1.75
```

(4) Write a function called `compute_expectation_X` that takes a and b as inputs and outputs the expectation $E(X)$. Write a function called `compute_variance_X` that takes a and b as input and outputs the variance Var(X). Display your code.

```
# function - `compute_expectation_X` to calculate the value of expectation of X
compute_expectation_X <- function(a,b) {
  # defining the probability of X = 2
  p_X_2 <- a*(a-1)/((a+b)*(a+b-1))

  # defining the probability of X = 0
  p_X_0 <- 2 *a*b/((a+b)*(a+b-1))
```

```
  # defining the probability of X = -2
  p_X_neg2 <-b *(b-1)/((a+b)*(a+b-1))

  # Expectation of X Formula calculation
  E_X <- 2 *p_X_2+0 *p_X_0 +(-2)* p_X_neg2
  return(E_X)
}

# Function "compute_variance_X" to calculate the variance (Var(X)) of X
compute_variance_X <- function(a, b) {
  # defining the probability of X = 2
  p_X_2 <- a*(a-1)/((a+b)*(a+b-1))

  # defining the probability of X = 0
  p_X_0 <- 2 *a*b/((a+b)*(a+b-1))

  # defining the probability of X = -2
  p_X_neg2 <-b *(b-1)/((a+b)*(a+b-1))

  # E(X^2) formula
  E_X2 <- 4*p_X_2 + 0*p_X_0 + 4*p_X_neg2

  # E(X)
  E_X<- 2 *p_X_2 +0*p_X_0+(-2)*p_X_neg2

  #Variance formula - Var(X)= E(X^2)-(E(X))^2
  variance_X <- E_X2 - E_X^2
  return(variance_X)
}

a <- 4
b <- 6

expectation_X <- compute_expectation_X(a, b)
cat("Expectation E(X) of X:", expectation_X, "\n")
```

```
## Expectation E(X) of X: -0.4
```

```
variance_X <- compute_variance_X(a, b)
cat("Variance Var(X) of X:", variance_X, "\n")
```

```
## Variance Var(X) of X: 1.706667
```

In the following questions, we additionally assume that $X_1, X_2, \dots, X_n$ are independent copies of X. So $X_1, X_2, \dots, X_n$ are i.i.d. random variables having the same distribution as that of X. Let $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$ be the sample mean.

(5) Give an expression of the expectation of the random variable $\bar{X}$ in terms of a, b.

-> Given: $X_1, X_2, \dots, X_n$ are independent variables of X

The sample mean $\overline{X}$:

$$(\overline{X}) = \left(\frac{1}{n}\sum_{i=1}^{n} X_i\right)$$

-> Required is the Expectation of Random variable $\overline{X}$ which is :

$$(E\overline{X}) = E\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right)$$

$$E(\overline{X}) = \frac{1}{n}\sum_{i=1}^{n} E(X_i) = \frac{1}{n} \cdot n \cdot E(X) = E(X)$$

the expectation of the sample mean $\overline{X}$ is:

$$E(\overline{X}) = E(X)$$

Solution :

Therefore, As we calculated the expectation of X above:

$$E(\overline{X}) = E(X) = 2 \cdot \frac{a(a-1)}{(a+b)(a+b-1)} - 2 \cdot \frac{b(b-1)}{(a+b)(a+b-1)}$$

OR

$$\boxed{E(\overline{X}) = E(X) = \frac{2(a^2 - b^2 - a + b)}{(a+b)(a+b-1)}}$$

(6) Give an expression of the variance of the random variable $\bar{X}$ in terms of a, b, and n.

Variance of the sample mean $\mathrm{Var}(\overline{X})$ formula is:

$$\mathrm{Var}(\overline{X}) = \mathrm{Var}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right)$$

the X_i's are independent and identically distributed therefore,

$$\mathrm{Var}(\overline{X}) = \frac{1}{n^2}\sum_{i=1}^{n} \mathrm{Var}(X_i)$$

$$\mathrm{Var}(\overline{X}) == \frac{1}{n^2} \cdot n \cdot \mathrm{Var}(X) = \frac{\mathrm{Var}(X)}{n}$$

Required -> expression of the variance of the random variable $\bar{X}$ in terms of a, b, and n is given by

$$\mathrm{Var}(\overline{X}) = \frac{\mathrm{Var}(X)}{n}$$

-> $\mathrm{Var}(X)$ in Terms of a and b from the above calculated results.

$$\text{Var}(X) = 4 \cdot \frac{a(a-1)+b(b-1)}{(a+b)(a+b-1)} - \left(2 \cdot \frac{a(a-1)-b(b-1)}{(a+b)(a+b-1)}\right)^2$$

Therefore the Variance of sample mean is

$$\text{Var}(\overline{X}) = \frac{4 \cdot \frac{a(a-1)+b(b-1)}{(a+b)(a+b-1)} - \left(2 \cdot \frac{a(a-1)-b(b-1)}{(a+b)(a+b-1)}\right)^2}{n}$$

(7) Create a function called `sample_Xs` which takes as inputs a, b, and n and outputs a sample $X_1, X_2, \ldots, X_n$ of independent copies of X.

```
# Creating the sample_Xs function to calculate the outputs of the samples
sample_Xs <- function(a, b, n) {

  p_X_2 <- a * (a - 1) / ((a + b) * (a + b - 1))
  p_X_0 <- 2 * a * b / ((a + b) * (a + b - 1))
  p_X_neg2 <- b * (b - 1) / ((a + b) * (a + b - 1))

  # defining the possible values of X and probabilies
  values <- c(2, 0, -2)
  probabilities <- c(p_X_2, p_X_0, p_X_neg2)
  #calculating the sample using the values and probability.
  sample <- sample(values, size = n, replace = TRUE, prob = probabilities)

  return(sample)
}
```

(8) Let $a = 3$, $b = 5$, and $n = 100000$. First, compute the numerical value of $E(X)$ using the function `compute_expectation_X` and compute the numerical value of $\text{Var}(X)$ using the function `compute_variance_X`. Second, use the function `sample_Xs` to generate a sample $X_1, X_2, \ldots, X_n$ of independent copies of X. With the generated sample, compute the sample mean $\bar{X}$ and sample variance. How close is the sample mean $\bar{X}$ to $E(X)$? How close is the sample variance to $\text{Var}(X)$? Explain your observation.

```
a <- 3
b <- 5
n <- 100000
 #using the previous functions expectation_X and variance_X for computing the expectation and variance
expectation_X <- compute_expectation_X(a, b)
variance_X <- compute_variance_X(a, b)

cat("E(X):", expectation_X, "\n")
```

```
## E(X): -0.5
```

```
cat("Var(X):", variance_X, "\n")
```

```
## Var(X): 1.607143
```

```r
#Using the function sample_Xs to generate a sample of independen copies of X
sample_data <- sample_Xs(a, b, n)
sample_mean <- mean(sample_data)
sample_variance <- var(sample_data)

# Calculating the sample mean X and sample variance using the sample_Xs function and displaying it .
cat("Sample Mean:", sample_mean, "\n")
```

```
## Sample Mean: -0.49936
```

```r
cat("Sample Variance:", sample_variance, "\n")
```

```
## Sample Variance: 1.608096
```

Moreover, let $\mu := E(X)$ and $\sigma := \sqrt{\mathbf{Var}(X)/n}$ (the random variable X is defined above), and let $f_{\mu,\sigma} : R \to [0, \infty)$ be the probability density function of a Gaussian random variable with distribution $N(\mu, \sigma^2)$, i.e., the expectation is μ and the variance is σ^2. Next, conduct a simulation study to explore the behaviour of the sample mean $\bar{X}$ by answering questions (9)-(11).

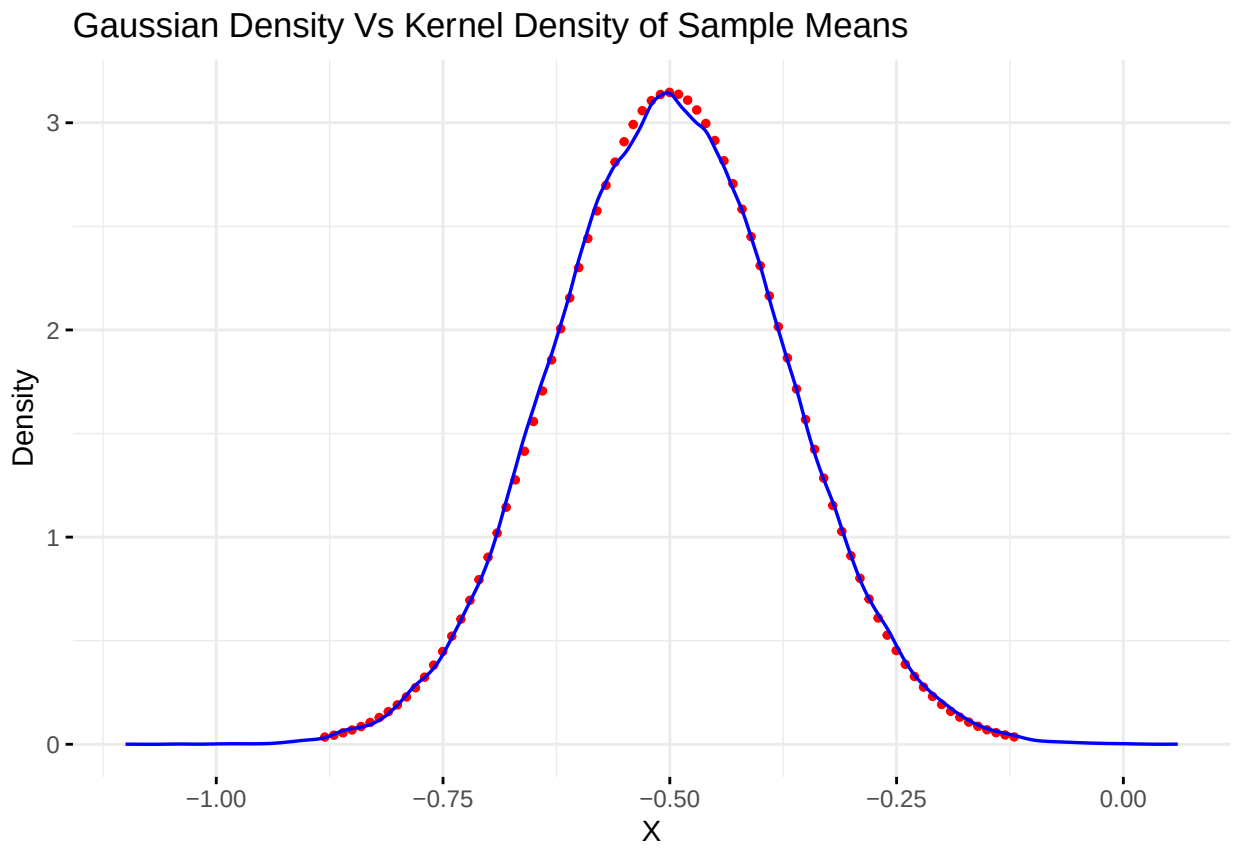
(9) Let $a = 3$, $b = 5$, and $n = 100$. Conduct a simulation study with 50000 trials. In each trial, generate a sample $X_1, \ldots, X_n$ of independent copies of X. For each of the 50000 trials, compute the corresponding sample mean $\bar{X}$ based on $X_1, \ldots, X_n$.

```r
#Below is the code for conducting the simulation study with 50000 trials
set.seed(123)
a <- 3
b <- 5
n_trials <- 50000
n <- 100

#generating 50000 trials of the sample mean based of the no of trials of independent copies of X
sample_means <- numeric(n_trials)

#For each trail looping or iterating and Calculating the sample mean for the sample_data

for (i in 1:n_trials) {
  sample_data <- sample_Xs(a, b, n)
  sample_means[i] <- mean(sample_data)
}

#Displaying the Mean of the Sample Means:
cat("Mean of Sample Means:", mean(sample_means), "\n")
```

```
## Mean of Sample Means: -0.4998036
```

(10) Create a scatter plot of the points $\{(x_i, f_{\mu,\sigma}(x_i))\}$ where $\{x_i\}$ are a sequence of numbers between $\mu - 3\sigma$ and $\mu + 3\sigma$ in increments of 0.01 (where 0.01 is the desired increment).

```r
# setting the values as per question
a <- 3
b <- 5
n <- 100
num_trials <- 50000

# Computing the expectation and variance for X
mu <- compute_expectation_X(a, b)
sigma <- sqrt(compute_variance_X(a, b) / n)

# Generating a sequence of x values for the Gaussian density plot
x_val <- seq(mu - 3 * sigma, mu + 3 * sigma, by = 0.01)

# Calculating the Gaussian density values for the x sequence using dnorm
f_val <- dnorm(x_val, mean = mu, sd = sigma)

# creating a vector to store sample means
sample_means <- numeric(num_trials)

# Conducting a simulation to compute sample means
set.seed(123)
for (t in 1:num_trials) {
  sample_data <- sample_Xs(a, b, n)
  sample_means[t] <- mean(sample_data)
}

# Creating the data frames for plotting the line and point graph
gaussian_df <- data.frame(x = x_val, f_mu_sigma = f_val)
sample_means_df <- data.frame(sample_means = sample_means)

# Plotting Gaussian density vs. kernel density of sample means
library(ggplot2)
ggplot() +
  geom_point(data = gaussian_df, aes(x = x, y = f_mu_sigma), color = "red", size = 1.0) +
  geom_density(data = sample_means_df, aes(x = sample_means), color = "blue", size = 0.6) +
  labs(title = "Gaussian Density Vs Kernel Density of Sample Means",
       x = "X", y = "Density") +
  scale_x_continuous(breaks = c(-1.00, -0.75, -0.50, -0.25, 0.00)) +
  theme_minimal()+
  theme(
    #This is used to show th ticks or marks against each value in the data
    axis.ticks = element_line(color = "black")
  )
```



(11) Describe the relationship between the density of $\bar{X}$ and the function $f_{\mu,\sigma}$ displayed in your plot. Try to explain the reason.

Explanation of the graph above:

The given graph shows the Kernel Density Estimate-KDE of sample means, "the red points in the graph", against the theoretical Gaussian density function plotting as a line. One can consider Kernel Density Estimation as an alternative to constructing histograms. Basically, KDE is a non-parametric method that infers the probability density function of a random variable taking the help of a sample. Here, the sample would include 50,000 sample means computed from 50,000 independent and identically distributed uniform random variables.

Overall Similarity in the graph, the KDE does resemble the Gaussian density function fairly well but according to the CLT, a sample of means should converge in distribution to a normal with increasing sample size, regardless of the underlying distribution of the individual observations.

There are minor discrepancies in the tails as well. Although the general shape of the two curves is somewhat similar, there are slight deviations in the tails. The KDE seems to have slightly heavier tails than the Gaussian density function. This discrepancy can be attributed to factors such as Sampling Variability and Kernel Density Estimation. Since sample means are random variables themselves, the distribution is affected by the samples taken. For small samples, the distribution of sample means may be noticeably different from normal. KDE bears some sensitivities to the kernel function choice and bandwidth parameter. In fact, different choices will yield a variety of estimators of density, in particular in the tails of the distribution.