**Data Warehousing and Data Mining (CSA1622)**

**Vikas R (192311353).**

**17/02/2025**

**EXP1.The intervals and corresponding frequencies are as follows. age frequency**

**1-5. 200**

**5-15 450**

**15-20 300**

**20-50 1500**

**50-80 700**

**80-110 44**

**Compute an approximate median value for the data.**

**CODE:**

age\_intervals <- c("1-5", "5-15", "15-20", "20-50", "50-80", "80-110")

frequencies <- c(200, 450, 300, 1500, 700, 44)

cumulative\_frequency <- cumsum(frequencies)

total\_frequency <- sum(frequencies)

N\_by\_2 <- total\_frequency / 2

median\_class\_index <- which(cumulative\_frequency >= N\_by\_2)[1]

L <- as.numeric(strsplit(age\_intervals[median\_class\_index], "-")[[1]][1]) # Lower boundary of median class

CF <- ifelse(median\_class\_index == 1, 0, cumulative\_frequency[median\_class\_index - 1]) # Cumulative frequency of the class before median class

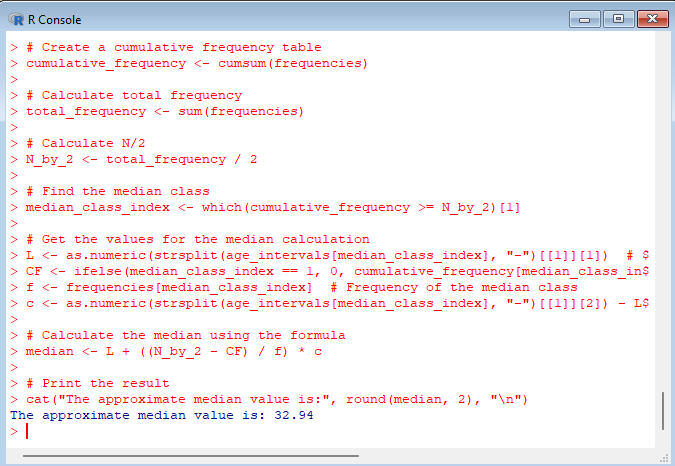
f <- frequencies[median\_class\_index] # Frequency of the median class

c <- as.numeric(strsplit(age\_intervals[median\_class\_index], "-")[[1]][2]) - L # Width of the median class

median <- L + ((N\_by\_2 - CF) / f) \* c

cat("The approximate median value is:", round(median, 2), "\n")

**OUTPUT:**



**EXP 2.Suppose that the data for analysis includes the attribute age. The age values for the data tuples are (in increasing order) 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70.**

**(a) What is the mean of the data? What is the median?**

**(b) What is the mode of the data? Comment on the data’s modality (i.e., bimodal, trimodal, etc.).**

**(c) What is the midrange of the data?**

**(d) Can you find (roughly) the first quartile (Q1) and the third quartile (Q3) of the data?**

**CODE:**

age\_data <- c(13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70)

mean\_age <- mean(age\_data)

median\_age <- median(age\_data)

uniq\_v <- unique(v)

uniq\_v[which.max(tabulate(match(v, uniq\_v)))]

}

mode\_age <- get\_mode(age\_data)

frequency\_table <- table(age\_data)

modality <- sum(frequency\_table > 1)

midrange\_age <- (min(age\_data) + max(age\_data)) / 2

Q1 <- quantile(age\_data, 0.25)

Q3 <- quantile(age\_data, 0.75)

cat("Mean of the data:", round(mean\_age, 2), "\n")

cat("Median of the data:", median\_age, "\n")

cat("Mode of the data:", mode\_age, "\n")

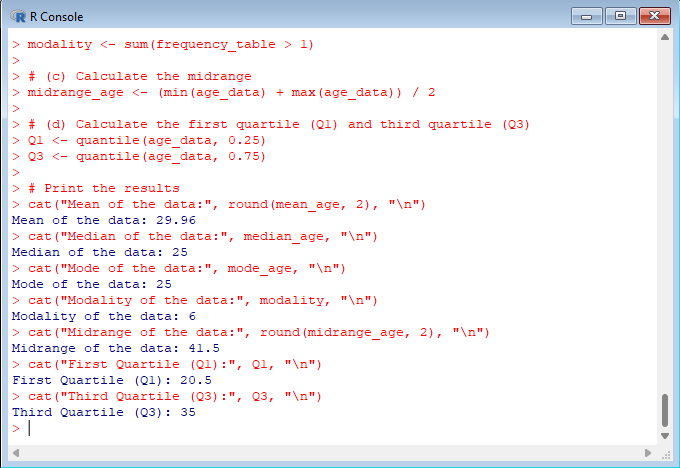
cat("Modality of the data:", modality, "\n")

cat("Midrange of the data:", round(midrange\_age, 2), "\n")

cat("First Quartile (Q1):", Q1, "\n")

cat("Third Quartile (Q3):", Q3, "\n")

**OUTPUT:**

****

**EXP 3.Data Preprocessing :Reduction and Transformation**

**Use the two methods below to normalize the following group of data: 200, 300, 400, 600, 1000 (a) min-max normalization by setting min = 0 and max = 1 (b) z-score normalization**

**CODE:**

data <- c(200, 300, 400, 600, 1000

min\_value <- min(data)

max\_value <- max(data)

min\_max\_normalized <- (data - min\_value) / (max\_value - min\_value)

cat("Min-Max Normalized Data:\n")

print(min\_max\_normalized)

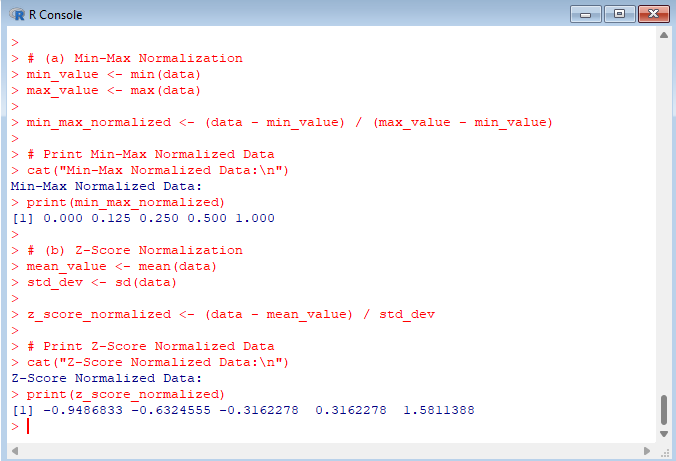
mean\_value <- mean(data)

std\_dev <- sd(data)

z\_score\_normalized <- (data - mean\_value) / std\_dev

cat("Z-Score Normalized Data:\n")

print(z\_score\_normalized)

**OUTPUT:**

**EXP 4.Data:11,13,13,15,15,16,19,20,20,20,21,21,22,23,24,30,40,45,45,45,71, 72,73,75**

**a) Smoothing by bin mean**

**b) Smoothing by bin median**

**c) Smoothing by bin boundaries**

**CODE:**

data\_smoothing <- c(11, 13, 13, 15, 15, 16, 19, 20, 20, 20, 21, 21, 22, 23, 24, 30, 40, 45, 45, 45, 71, 72, 73, 75)

bin\_mean\_smoothing <- function(data, bin\_size) {

binned\_data <- tapply(data, (seq\_along(data) - 1) %/% bin\_size, mean)

return(binned\_data)

}

bin\_median\_smoothing <- function(data, bin\_size) {

binned\_data <- tapply(data, (seq\_along(data) - 1) %/% bin\_size, median)

return(binned\_data)

}

bin\_boundaries\_smoothing <- function(data, bin\_size) {

binned\_data <- tapply(data, (seq\_along(data) - 1) %/% bin\_size, function(x) c(min(x), max(x)))

return(binned\_data)

}

# Set bin size

bin\_size <- 5

mean\_smoothed <- bin\_mean\_smoothing(data\_smoothing, bin\_size)

median\_smoothed <- bin\_median\_smoothing(data\_smoothing, bin\_size)

boundaries\_smoothed <- bin\_boundaries\_smoothing(data\_smoothing, bin\_size)

cat("Smoothing by Bin Mean:\n")

print(mean\_smoothed)

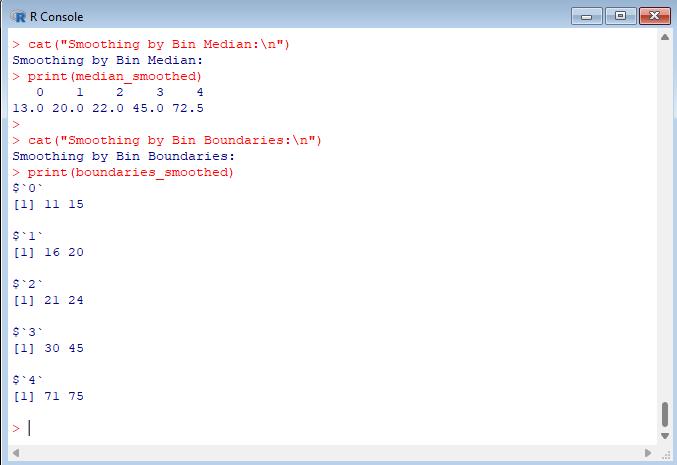
cat("Smoothing by Bin Median:\n")

print(median\_smoothed)

cat("Smoothing by Bin Boundaries:\n")

print(boundaries\_smoothed)

**OUTPUT:**



**5.Suppose that a hospital tested the age and body fat data for 18 randomly selected adults with the following results:**

****

**(a) Calculate the mean, median, and standard deviation of age and %fat.**

**(b) Draw the boxplots for age and %fat.  
(c) Draw a scatter plot and a q-q plot based on these two variables.**

**CODE:**

age <- c(25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110)

body\_fat <- c(15, 20, 22, 25, 30, 32, 35, 36, 40, 42, 45, 48, 50, 52, 55, 58, 60, 62)

mean\_age <- mean(age)

median\_age <- median(age)

sd\_age <- sd(age)

mean\_fat <- mean(body\_fat)

median\_fat <- median(body\_fat)

sd\_fat <- sd(body\_fat)

cat("Age - Mean:", mean\_age, "Median:", median\_age, "Standard Deviation:", sd\_age, "\n")

cat("Body Fat - Mean:", mean\_fat, "Median:", median\_fat, "Standard Deviation:", sd\_fat, "\n")

par(mfrow=c(1, 2)) # Set up the plotting area

boxplot(age, main="Boxplot of Age", ylab="Age")

boxplot(body\_fat, main="Boxplot of Body Fat %", ylab="% Fat")

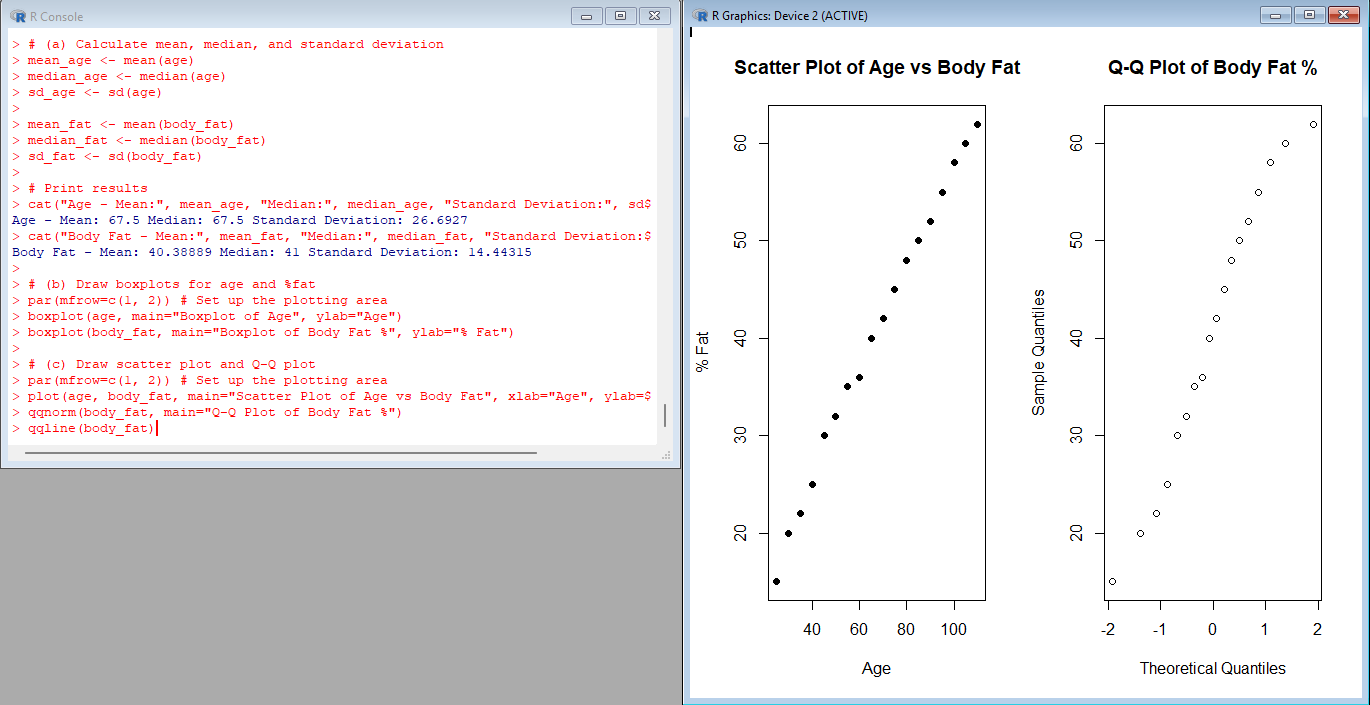
par(mfrow=c(1, 2)) # Set up the plotting area

plot(age, body\_fat, main="Scatter Plot of Age vs Body Fat", xlab="Age", ylab="% Fat", pch=19)

qqnorm(body\_fat, main="Q-Q Plot of Body Fat %")

qqline(body\_fat)

**OUTPUT:**

****

**6.Suppose that a hospital tested the age and body fat data for 18 randomly selected adults with the following results:**

**(i) Use min-max normalization to transform the value 35 for age onto the range [0.0, 1.0].  
(ii) Use z-score normalization to transform the value 35 for age, where the standard deviation of age is 12.94 yearS  
(iii) Use normalization by decimal scaling to transform the value 35 for age. Perform the above functions using R – tool**

**CODE:**

age <- c(25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110)

value\_to\_normalize <- 35

min\_age <- min(age)

max\_age <- max(age)

min\_max\_normalized <- (value\_to\_normalize - min\_age) / (max\_age - min\_age)

cat("Min-Max Normalized Value for 35 (Age):", min\_max\_normalized, "\n")

mean\_age\_value <- mean(age)

std\_dev\_age <- sd(age)

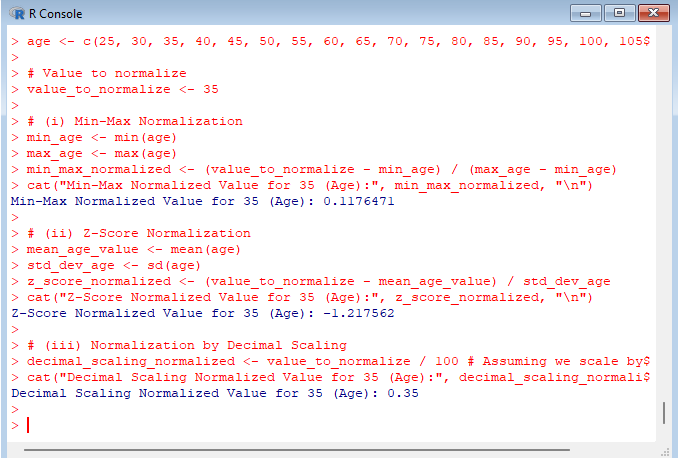
z\_score\_normalized <- (value\_to\_normalize - mean\_age\_value) / std\_dev\_age

cat("Z-Score Normalized Value for 35 (Age):", z\_score\_normalized, "\n")

decimal\_scaling\_normalized <- value\_to\_normalize / 100 # Assuming we scale by 100

cat("Decimal Scaling Normalized Value for 35 (Age):", decimal\_scaling\_normalized, "\n")

**OUTPUT:**

****

**7.The following values are the number of pencils available in the different boxes. Create a vector and find out the mean, median and mode values of set of pencils in the given data.**

**Box1 Box2 Box3 Box4 Box5 Box6 Box7 Box8 Box9 Box 10**

**9 25 23 12 11 6 7 8 9 10**

**CODE:**

pencils <- c(9, 25, 23, 12, 11, 6, 7, 8, 9, 10)

mean\_pencils <- mean(pencils)

median\_pencils <- median(pencils)

get\_mode <- function(v) {

uniq\_v <- unique(v)

uniq\_v[which.max(tabulate(match(v, uniq\_v)))]

}

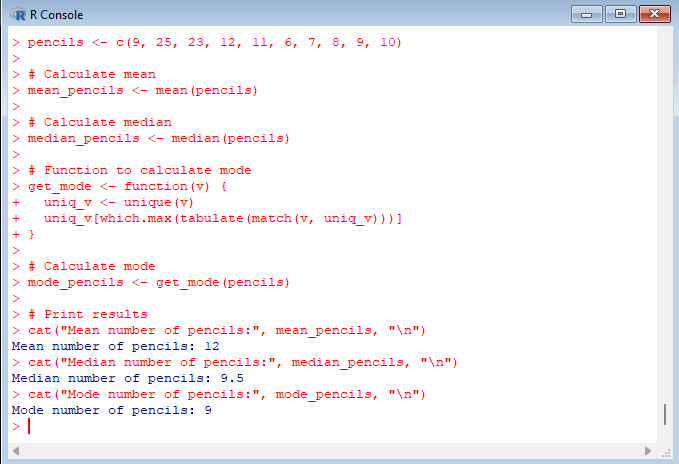
mode\_pencils <- get\_mode(pencils)

cat("Mean number of pencils:", mean\_pencils, "\n")

cat("Median number of pencils:", median\_pencils, "\n")

cat("Mode number of pencils:", mode\_pencils, "\n")

**OUTPUT:**

****

**8.the following table would be plotted as (x,y) points, with the first column being the x values as number of mobile phones sold and the second column being the y values as money. To use the scatter plot for how many mobile phones sold.**

**x :4 1 5 7 10 2 50 25 90 36**

**y :12 5 13 19 31 7 153 72 275 110**

**CODE:**

x <- c(4, 1, 5, 7, 10, 2, 50, 25, 90, 36)

y <- c(12, 5, 13, 19, 31, 7, 153, 72, 275, 110)

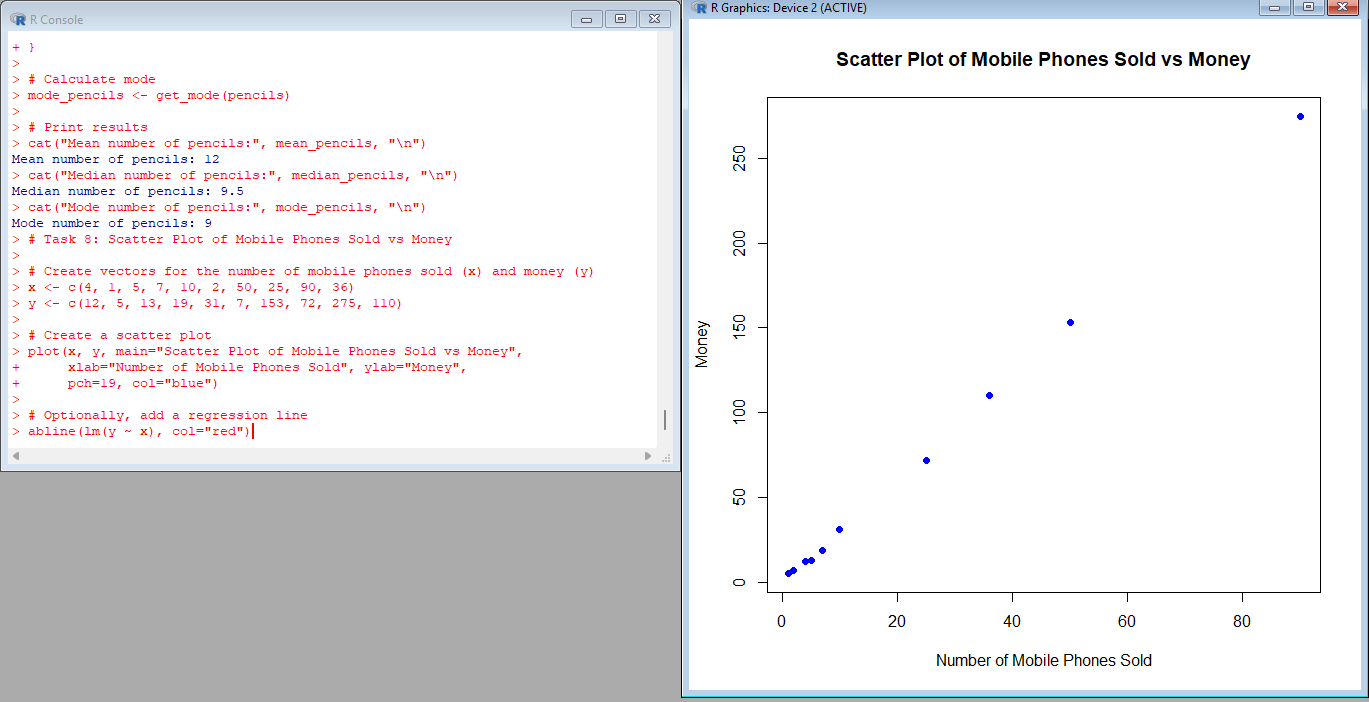
plot(x, y, main="Scatter Plot of Mobile Phones Sold vs Money",

xlab="Number of Mobile Phones Sold", ylab="Money",

pch=19, col="blue")

abline(lm(y ~ x), col="red")

**OUTPUT:**

****

**9.Implement of the R script using marks scored by a student in his model exam has been sorted as follows: 55, 60, 71, 63, 55, 65, 50, 55,58,59,61,63,65,67,71,72,75. Partition them into three bins by each of the following methods. Plot the data points using histogram.**

**(a) equal-frequency (equi-depth) partitioning (b) equal-width partitioning**

**CODE:**

marks <- c(55, 60, 71, 63, 55, 65, 50, 55, 58, 59, 61, 63, 65, 67, 71, 72, 75)

num\_bins <- 3

bin\_size <- length(marks) / num\_bins

sorted\_marks <- sort(marks)

bins\_eq\_freq <- cut(sorted\_marks, breaks = seq(min(sorted\_marks), max(sorted\_marks), length.out = num\_bins + 1), include.lowest = TRUE)

cat("Equal-Frequency Bins:\n")

print(table(bins\_eq\_freq))

hist(marks, breaks = seq(min(marks), max(marks), length.out = num\_bins + 1), main = "Histogram of Marks (Equal-Frequency)", xlab = "Marks", ylab = "Frequency", col = "lightblue")

bin\_width <- (max(marks) - min(marks)) / num\_bins

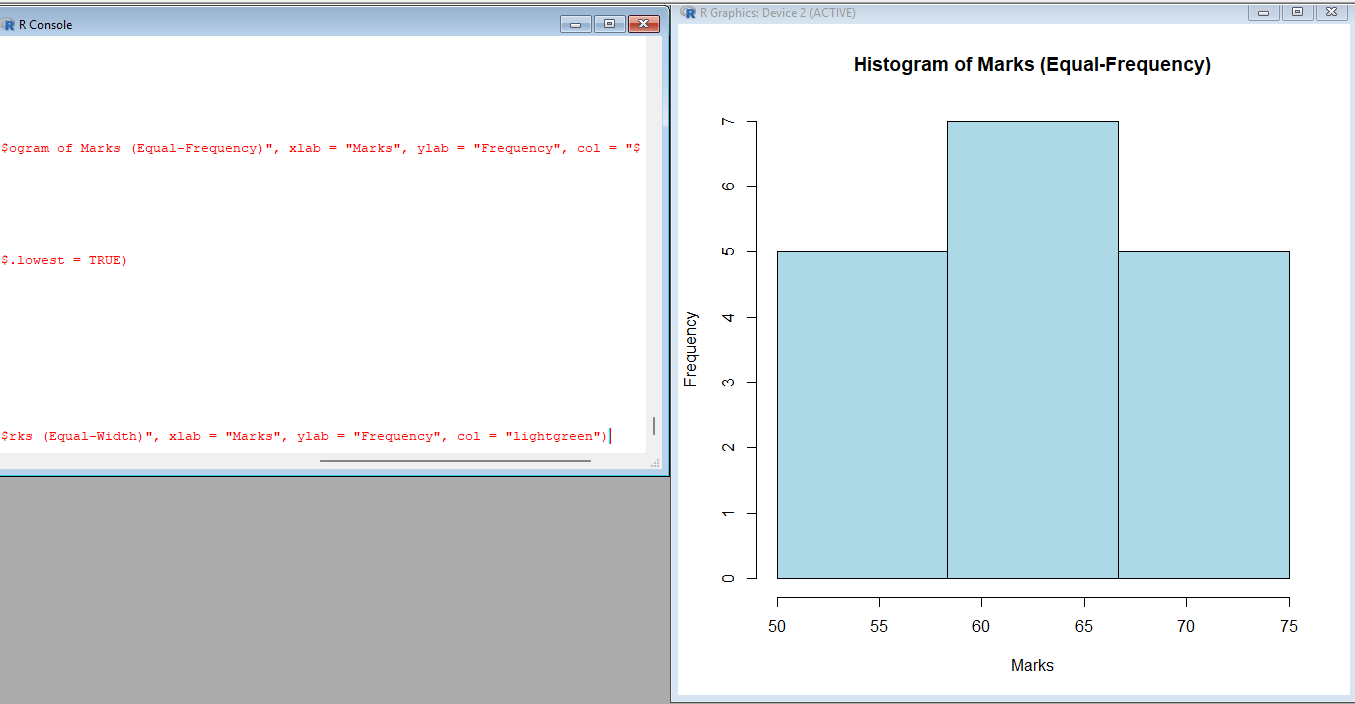
bins\_eq\_width <- cut(marks, breaks = seq(min(marks), max(marks), by = bin\_width), include.lowest = TRUE)

cat("Equal-Width Bins:\n")

print(table(bins\_eq\_width))

hist(marks, breaks = seq(min(marks), max(marks), by = bin\_width), main = "Histogram of Marks (Equal-Width)", xlab = "Marks", ylab = "Frequency", col = "lightgreen")

**OUTPUT:**

****

**10.Suppose that the data for analysis includes the attribute age. The age values for the data tuples are (in increasing order) 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70.**

**Can you find (roughly) the first quartile (Q1) and the third quartile (Q3) of the data?**

**CODE:**

age <- c(13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70)

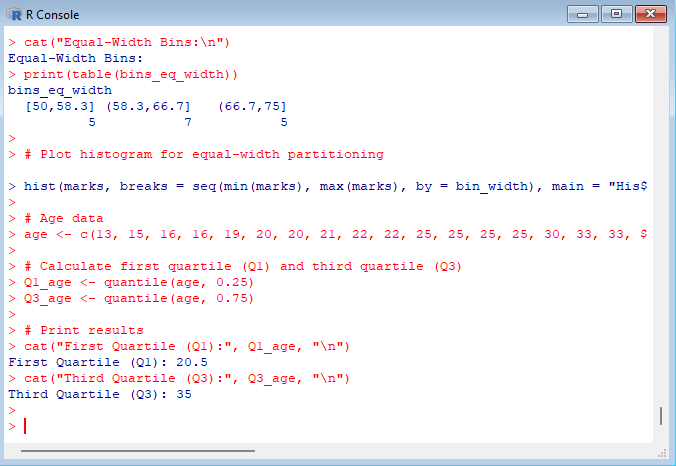
Q1\_age <- quantile(age, 0.25)

Q3\_age <- quantile(age, 0.75)

cat("First Quartile (Q1):", Q1\_age, "\n")

cat("Third Quartile (Q3):", Q3\_age, "\n")

**OUTPUT:**

****