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

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

midpoints <- c(3, 10, 17.5, 35, 65, 95)

N <- sum(frequencies)

cf <- cumsum(frequencies)

i <- which(cf >= N/2)[1]

L <- midpoints[i - 1]

F <- cf[i - 1]

f <- frequencies[i]

h <- midpoints[i] - midpoints[i - 1]

median <- L + ((N/2 - F) / f) \* h

cat("Approximate median:", round(median, 2), "\n")

2)# Points scored by tennis players

scores <- c(55, 60, 62, 65, 68, 70, 72, 75, 78, 80, 82, 85, 90, 95, 150)

# Create a boxplot

boxplot(scores,

main = "Boxplot of Tennis Players' Scores",

ylab = "Points Scored",

col = "skyblue",

border = "darkblue",

pch =11)

# Add grid for better readability

grid()

3) # Given data

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

# (a) Min-Max Normalization (min = 0, max = 1)

min\_val <- min(data)

max\_val <- max(data)

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

# (b) Z-score Normalization

mean\_val <- mean(data)

std\_dev <- sd(data)

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

# Printing results

cat("Original Data: ", data, "\n")

cat("Min-Max Normalized Data: ", min\_max\_normalized, "\n")

cat("Z-Score Normalized Data: ", z\_score\_normalized, "\n")

4) # Given data

data <- 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)

# Function to apply smoothing

smooth\_data <- function(data, bin\_size, method) {

bins <- split(data, ceiling(seq\_along(data) / bin\_size))

if (method == "mean") return(unlist(lapply(bins, function(bin) rep(mean(bin), length(bin)))))

if (method == "median") return(unlist(lapply(bins, function(bin) rep(median(bin), length(bin)))))

if (method == "boundary") return(unlist(lapply(bins, function(bin) rep(min(bin), length(bin)))))

}

# Set bin size

bin\_size <- 5

# Apply smoothing methods

cat("Smoothed by Bin Mean: \n", smooth\_data(data, bin\_size, "mean"), "\n")

cat("Smoothed by Bin Median: \n", smooth\_data(data, bin\_size, "median"), "\n")

cat("Smoothed by Bin Boundaries (Min): \n", smooth\_data(data, bin\_size, "boundary"), "\n")

5)