MOHAMMED IDHRIS

CSA1601

DATA WAREHOUSING AND DATA MINING

OBSERVATION PROGRAMS

1.APPROXIMATE MEDIAN VALUE

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_frequencies <- cumsum(frequencies)

N <- sum(frequencies)

median_class_index <- which(cumulative_frequencies >= N / 2)[1]

median_class <- age_intervals[median_class_index]

lower_boundary <- as.numeric(strsplit(median_class, "-")[[1]][1])

frequency_median_class <- frequencies[median_class_index]

cumulative_frequency_before <- ifelse(median_class_index == 1, 0,
cumulative_frequencies[median_class_index - 1])

median <- lower_boundary + ((N / 2 - cumulative_frequency_before) / frequency_median_class) *
(as.numeric(strsplit(median_class, "-")[[1]][2]) - lower_boundary)

median
```

2.AGE 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)

get_mode <- function(v) {

  uniqv <- unique(v)

  uniqv[which.max(tabulate(match(v, uniqv)))]

}

mode_age <- get_mode(age_data)
```

```
midrange_age <- (min(age_data) + max(age_data)) / 2

Q1 <- quantile(age_data, 0.25)

Q3 <- quantile(age_data, 0.75)

list(mean = mean_age, median = median_age, mode = mode_age, midrange = midrange_age, Q1 = Q1, Q3 = Q3)
```

3.DATA PREPROCESSING

CODE:

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

min_max_normalized <- (data - min(data)) / (max(data) - min(data))

z_score_normalized <- (data - mean(data)) / sd(data)

min_max_normalized

z_score_normalized
```

4.SMOOTHING

CODE:

```
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)

#a

library(dplyr)

bin_mean <- function(data, bin_size) {

 cut_data <- cut(data, breaks = seq(min(data), max(data), by = bin_size), include.lowest = TRUE)

 mean_data <- aggregate(data, by = list(cut_data), FUN = mean)

 return(mean_data)

}

mean_smooth <- bin_mean(data, 10)

#b

bin_median <- function(data, bin_size) {

 cut_data <- cut(data, breaks = seq(min(data), max(data), by = bin_size), include.lowest = TRUE)

 median_data <- aggregate(data, by = list(cut_data), FUN = median)

 return(median_data)

}

median_smooth <- bin_median(data, 10)

#c#  bin_boundaries <- function(data, bin_size) {

 cut_data <- cut(data, breaks = seq(min(data), max(data), by = bin_size), include.lowest = TRUE)

 boundaries_data <- data.frame(table(cut_data))

 return(boundaries_data)

}

boundaries_smooth <- bin_boundaries(data, 10)
```

5.HOSPITAL TEST

CODE:

```
library(ggplot2)

age <- c(23, 23, 27, 27, 39, 41, 47, 49, 50, 52, 54, 54, 56, 57, 58, 58, 60, 61)

fat <- c(9.5, 26.5, 7.8, 17.8, 31.4, 25.9, 27.4, 27.2, 31.2, 34.6, 42.5, 28.8, 33.4, 30.2, 34.1, 32.9, 41.2, 35.7)

# (a) Calculate mean, median, and standard deviation

age_mean <- mean(age)

age_median <- median(age)

age_sd <- sd(age)

fat_mean <- mean(fat)

fat_median <- median(fat)

fat_sd <- sd(fat)

cat("Age - Mean:", age_mean, "Median:", age_median, "SD:", age_sd, "\n")

cat("Fat - Mean:", fat_mean, "Median:", fat_median, "SD:", fat_sd, "\n")

# (b) Draw boxplots

par(mfrow=c(1,2))

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

boxplot(fat, main="Boxplot of % Fat", ylab="% Fat")

# (c) Draw scatter plot

plot(age, fat, main="Scatter Plot of Age vs % Fat", xlab="Age", ylab="% Fat")

# Q-Q plot

qqnorm(fat)

qqline(fat, col = "red")
```

6.HOSPITAL TEST

CODE:

```
age_value <- 35

min_age <- 18  # Example minimum age

max_age <- 65  # Example maximum age

mean_age <- 40  # Example mean age

std_dev_age <- 12.94  # Standard deviation of age

min_max_normalized <- (age_value - min_age) / (max_age - min_age)
```

```
z_score_normalized <- (age_value - mean_age) / std_dev_age
decimal_scaling_normalized <- age_value / 100  # Assuming scaling by 100
min_max_normalized
z_score_normalized
decimal_scaling_normalized
```

7.VECTOR

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)
mean_pencils
median_pencils
mode_pencils
```

8.SCATTER PLOT FOR MOBILE PHONES SOLD

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", xlab="Number of Mobile Phones Sold", ylab="Money",
pch=19, col="blue")
```

9. Equal-Frequency (Equi-Depth) Partitioning

CODE:

```
scores <- c(55, 60, 71, 63, 55, 65, 50, 55, 58, 59, 61, 63, 65, 67, 71, 72, 75)
equi_depth_bins <- quantile(scores, probs = seq(0, 1, length.out = 4))
hist(scores, breaks = equi_depth_bins, main = "Equal-Frequency Partitioning", xlab = "Scores", col = "lightblue")
```

```
min_score <- min(scores)
max_score <- max(scores)
width <- (max_score - min_score) / 3
equi_width_bins <- seq(min_score, max_score, by = width)
hist(scores, breaks = equi_width_bins, main = "Equal-Width Partitioning", xlab = "Scores", col = "lightgreen")
```

10.INTER QUANTILE AND STANDARD DEVIATION

CODE:

```
speed <- c(78.3, 81.8, 82, 74.2, 83.4, 84.5, 82.9, 77.5, 80.9, 70.6)
iqr_value <- IQR(speed)
sd_value <- sd(speed)
iqr_value
sd_value
```

11.QUARTILE CALCULATION

CODE:

```
age_values <- 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 <- quantile(age_values, 0.25)
Q3 <- quantile(age_values, 0.75)
Q1
Q3
```