1. Implement of the R script using a group of 12 sales price records has been sorted as

follows: 5, 10, 11, 13, 15, 35, 50, 55, 72, 92, 204, 215. Partition them into three bins by each

of the following methods.

(a) equal-frequency (equi depth) partitioning

(b) equal-width partitioning

(c) clustering

PROGRAM:

marks<-c(5, 10, 11, 13, 15, 35, 50, 55, 72, 92, 204, 215)

binning1=c()

binning2=c()

binning3=c()

class=6

#binning partition

for(a in marks[1:class]){

binning1=append(binning1,a)

}

range1=range+1

range2=range\*2

for(b in marks[range1:range2])

{

binning2=append(binning2,b)

}

range3=range2+1

range4=range\*3

for(c in marks[range3:range4])

{

binning3=append(binning3,c)

}

print(binning1)

print(binning2)

print(binning3)

#9a

#equal-frequency

freq=length(marks)/range

print(freq)

#9b

#equal-width

min<-min(marks)

max<-max(marks)

result<-max-min

width<-result/range

cat("width is",width)

bin1=width+min

print(bin1)

bin2=2\*width+min

print(bin2)

bin3=3\*width+min

print(bin3)

----------------------------------------------------------------------------------------------------------------------------------------------------------------

4.Use following group of data: 200, 300, 400, 600, 1000

(a) min-max normalization by setting min = 0 and max = 1 (b)

(b) z-score normalization

(c) (c) z-score normalization using the mean absolute deviation instead of standard

deviation (d) normalization by decimal scaling

.PROGRAM

Coding:

#4a

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

min<-min(data)

max<-max(data)

for (i in data)

{

result1=i-min

result2=max-min

result3=result1/result2

print(result3)

}

OUTPUT:

[1] 0

[1] 0.125

[1] 0.25

[1] 0.5

[1] 1

#4b

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

mean1<-mean(data)

deviation<-sd(data)

for (i in data)

{

result1=i-mean1

result2=result1/deviation

print(result2)

}

OUTPUT:

[1] -0.9486833

[1] -0.6324555

[1] -0.3162278

[1] 0.3162278

[1] 1.581139

-------------------------------------------------------------------------------------------------------------------------------------------------------------

5.Implement using R language in which age group of people are affected byblood pressure

based on the diabetes dataset show it using scatterplot and bar chart (that is BloodPressure vs

Age using dataset “diabetes.csv”)

PROGRAM:

dia<-read.csv("C:/Users/haris/Downloads/diabetes.csv")

View(dia)

plot(dia$Age, dia$BloodPressure, xlab = "Age", ylab = "Blood Pressure", main = "Blood Pressure vs. Age", col = "blue",pch = 16)

barplot(dia$Age,dia$Blood\_Pressure)

-------------------------------------------------------------------------------------------------------------------------------------------------------------

6.Analysis the dataset “diabetes. csv” how the diabetes trend is for different age people, using linear regression and multiple regression.

Input:

data<-read.csv("C:/Users/Harshana B/Downloads/diabetes.csv")

data

relation<-lm(data$Age~data$Outcome)

relation

relation<-lm(data$Age~data$Outcome+data$BMI)

relation

----------------------------------------------------------------------------------------------------------------------------------------------------------------

9.

PROGRAM:

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?

Coding:

#9a

x<-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

mean(x)

#median

median(x)

output:

mean(x)

[1] 29.96296

> #median

> median(x)

[1] 25

CODING FOR 9b-

#mode

MultipleModes <- function(x) {

uniqx <- unique(x)

freq\_table <- tabulate(match(x, uniqx))

modes <- uniqx[freq\_table == max(freq\_table)]

modes

}

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)

multiple\_modes <- MultipleModes(age\_values)

print(multiple\_modes)

output:

25 35

CODING FOR 9c-

#midrange

c) 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)

median(age\_values)

OUTPUT-

25

CODING FOR 2d-

d) #quartile

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)

quantile(age\_values)

output: 0% 25% 50% 75% 100%

13.0 20.5 25.0 35.0 70.0

-------------------------------------------------------------------------------------------------------------------------------------------------------------

10. Download the Dataset "water" From R dataset Link.Find out whether there is a linear relation between attributes"mortality" and"hardness" by plot function.Fit the Data into the Linear Regression model. Predict the mortality for the hardness=88

# Install and load necessary packages (if not already installed)

if (!requireNamespace("datasets", quietly = TRUE)) {

install.packages("datasets")

}

# Load required libraries

library(datasets)

# Load the water dataset

data("water")

# Explore the structure of the dataset

str(water)

# Plotting the data to visualize the relationship between "mortality" and "hardness"

plot(water$hardness, water$mortality, main = "Scatter Plot: Mortality vs. Hardness",

xlab = "Hardness", ylab = "Mortality")

# Fit the Linear Regression model

linear\_model <- lm(mortality ~ hardness, data = water)

# Summary of the linear model

summary(linear\_model)

# Visualize the regression line on the scatter plot

abline(linear\_model, col = "blue")

# Predict mortality for hardness=88

new\_data <- data.frame(hardness = 88)

predicted\_mortality <- predict(linear\_model, newdata = new\_data)

# Print the predicted mortality for hardness=88

cat("Predicted mortality for hardness=88:", predicted\_mortality, "\n")

-------------------------------------------------------------------------------------------------------------------------------------------------------------

13.Imagine that you have selected data from the All Electronics data warehouse for analysis.

The data set will be huge! The following data are a list of All Electronics prices for

commonly sold items (rounded to the nearest dollar). The numbers have been sorted: 1, 1, 5,

5, 5, 5, 5, 8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15, 15, 15, 15, 15, 18, 18, 18, 18, 18,

18, 18, 18, 20, 20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 25, 25, 25, 25, 25, 28, 28, 30,

30, 30

(i) Partition the dataset using an equal-frequency partitioning method with bin equal to 3

(ii) apply data smoothing using bin means and bin boundary.

(iii) Plot Histogram for the above frequency division

PROGRAM:

data<-c(1, 1, 5, 5, 5, 5, 5, 8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15, 15, 15, 15, 15, 18, 18, 18, 18, 18, 18, 18, 18, 20, 20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 25, 25, 25, 25, 25, 28, 28, 30, 30, 30)

bin<-length(data)/3

bins<-cut(data, breaks = c(-Inf, quantile(data, probs = seq(0,1, 1/3)),Inf), include.lowest = TRUE)

tapply(data,bins,mean)

tapply(data, bins,function(x) c(min(x),max(x)))

hist(data,breaks = 3, main = "Histogram", xlab = "prices")

----------------------------------------------------------------------------------------------------------------------------------------------------------------

14. Two Maths teachers are comparing how their Year 9 classes performed in the end of year

exams. Their results are as follows:

Class A: 76, 35, 47, 64, 95, 66, 89, 36, 8476,35,47,64,95,66,89,36,84

Class B: 51, 56, 84, 60, 59, 70, 63, 66, 5051,56,84,60,59,70,63,66,50

(i) Find which class had scored higher mean, median and range.

(ii) Plot above in boxplot and give the inferences

PROGRAM

A <- c(76, 35, 47, 64, 95, 66, 89, 36, 84)

B <- c(51, 56, 84, 60, 59, 70, 63, 66, 50)

mean\_A <- mean(A)

median\_A <- median(A)

range\_A <- max(A) - min(A)

mean\_B <- mean(B)

median\_B <- median(B)

range\_B <- max(B) - min(B)

combined\_data <- data.frame(Class = c(rep("A", length(A)), rep("B", length(B))), Score = c(A, B))

boxplot(Score ~ Class, data = combined\_data, col = c("blue", "green"), xlab = "Class", ylab = "Scores", main = "Boxplot of Scores by Class")

(II)

combined\_data <- data.frame(Class = c(rep("A", length(A)), rep("B", length(B))), Score = c(A, B))

boxplot(Score ~ Class, data = combined\_data, col = c("blue", "green"), xlab = "Class", ylab = "Scores", main = "Boxplot of Scores by Class")

----------------------------------------------------------------------------------------------------------------------------------------------------------------

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

CODING-

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

body\_fat\_percent <- 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)

#5.a

mean(age)

mean(body\_fat\_percent)

median(age)

median(body\_fat\_percent)

sd(age)

sd(body\_fat\_percent)

#5.b

#create dataframe

df<-data.frame(age,body\_fat\_percent)

#box plot

boxplot(df)

#scatter plot

plot(df)

#qq plot

qqnorm(age)

qqline(age)

qqnorm(body\_fat\_percent)

qqline(body\_fat\_percent)

----------------------------------------------------------------------------------------------------------------------------------------------------------------

18.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

CODING-

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

new\_age<-c()

for(i in age){

if(i<=35){

new\_age=append(new\_age,i)

}

}

print(new\_age)

#18a

#min max normalization

min<-min(new\_age)

max<-max(new\_age)

for (i in new\_age)

{

result1=i-min

result2=max-min

result3=result1/result2

print(result3)

}

#18b

#z score normalization

mean1<-mean(new\_age)

for (i in new\_age)

{

result1=i-mean1

result2=result1/12.94

print(result2)

}

#18c

#decimal scaling

n=200

j=nchar(y)

scaling=n/10^j

print(scaling)

----------------------------------------------------------------------------------------------------------------------------------------------------------------

---------------

21.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

25 23 12 11 6 7 8 9 10

PROGRAM:

box\_no=c("box1","box2","box3","box4","box5","box6","box7","box8","box9")

pencil=c(25,23,12,11,6,7,8,9,10)

df<-data.frame(box\_no,pencil)

#dataframe

print(df)

#mean

mean(pencil)

#median

----------------------------------------------------------------------------------------------------------------------------------------------------------------

22. Assume the Tennis coach wants to determine if any of his team players are scoring

outliers. To visualize the distribution of points scored by his players, then how can he

decide to develop the box plot? Give suitable example using Boxplot visualization

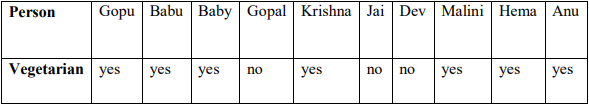
technique.

score <- c(20, 25, 30, 32, 35, 38, 40, 45, 50, 52, 55, 56, 58, 59, 60, 62, 65, 70, 75, 80, 85)

boxplot(score, col = "lightblue", main = "Box Plot of Points Scored by Tennis Players", ylab = "Points Scored")

----------------------------------------------------------------------------------------------------------------------------------------------------------------

25.The following list of persons with vegetarian or not details given in the table. How will you find out how many of them are vegetarian and how many of them are non-vegetarian? Which type of the person total count is greater value?



persons <- c("Gopu", "Babu", "Baby", "Gopal", "Krishna", "Jai", "Dev", "Malini", "Hema", "Anu")

vegetarian\_status <- c("yes", "yes", "yes", "no", "yes", "no", "no", "yes", "yes", "yes")

# Create a data frame

data <- data.frame(Person = persons, Vegetarian = vegetarian\_status)

# Count the number of vegetarians and non-vegetarians

vegetarian\_count <- sum(data$Vegetarian == "yes")

non\_vegetarian\_count <- sum(data$Vegetarian == "no")

# Print the counts

cat("Number of vegetarians:", vegetarian\_count, "\n")

cat("Number of non-vegetarians:", non\_vegetarian\_count, "\n")

# Determine which type of person has a greater total count

if (vegetarian\_count > non\_vegetarian\_count) {

cat("Vegetarians have a greater total count.\n")

} else if (non\_vegetarian\_count > vegetarian\_count) {

cat("Non-vegetarians have a greater total count.\n")

} else {

cat("The counts of vegetarians and non-vegetarians are equal.\n")

--------------------------------------------------------------------------------------------------------------------------------------------------------------

26. .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.

PROGRAM:

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,xlab='MOBILE PHONES SOLD',ylab='MONEY')

--------------------------------------------------------------------------------------------------------------------------------------------------------------

29.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

CODING-

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

binning1=c()

binning2=c()

binning3=c()

class=6

#binning partition

for(a in marks[1:class]){

binning1=append(binning1,a)

}

range1=range+1

range2=range\*2

for(b in marks[range1:range2])

{

binning2=append(binning2,b)

}

range3=range2+1

range4=range\*3

for(c in marks[range3:range4])

{

binning3=append(binning3,c)

}

print(binning1)

print(binning2)

print(binning3)

#histogram

hist(binning1)

hist(binning2)

hist(binning3)

#9a

#equal-frequency

freq=length(marks)/range

print(freq)

#9b

#equal-width

min<-min(marks)

max<-max(marks)

result<-max-min

width<-result/range

cat("width is",width)

bin1=width+min

print(bin1)

bin2=2\*width+min

print(bin2)

bin3=3\*width+min

print(bin3)

----------------------------------------------------------------------------------------------------------------------------------------------------------------

32.The given are the strike-rates scored by a batsman in season 1 in different tournaments. 100, 70, 60, 90, 90 (a) min-max normalization by setting min = 0 and max = 1 (b) z-score normalization (c) z-score normalization using the mean absolute deviation instead of standard deviation (d) normalization by decimal scaling

# Given data

strike\_rates <- c(100, 70, 60, 90, 90)

# Min-Max Normalization

min\_max\_normalized <- (strike\_rates - min(strike\_rates)) / (max(strike\_rates) - min(strike\_rates))

# Print the result

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

# Z-Score Normalization

z\_score\_normalized <- (strike\_rates - mean(strike\_rates)) / sd(strike\_rates)

# Print the result

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

# Z-Score Normalization using MAD

mad\_value <- mad(strike\_rates, constant = 1.4826)

z\_score\_normalized\_mad <- (strike\_rates - median(strike\_rates)) / mad\_value

# Print the result

cat("Z-Score Normalization using MAD:", z\_score\_normalized\_mad, "\n")

# Normalization by Decimal Scaling

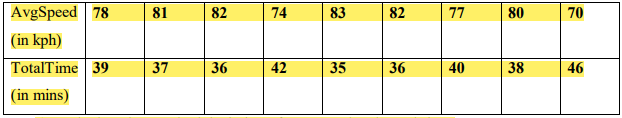
decimal\_scaled <- strike\_rates / 10^ceiling(log10(max(strike\_rates)))

# Print the result

cat("Normalization by Decimal Scaling:", decimal\_scaled, "\n")

----------------------------------------------------------------------------------------------------------------------------------------------------------------

33.Suppose some car is tested for the AvgSpeed and TotalTime data for 9 randomly selected car with the following result



a) Calculate the standard deviation of AvgSpeed and TotalTime. b) Calculate the Variance of AvgSpeed and TotalTime for the above dataset.

# Given data

avg\_speed <- c(78, 81, 82, 74, 83, 82, 77, 80, 70)

total\_time <- c(39, 37, 36, 42, 35, 36, 40, 38, 46)

# Calculate standard deviation

sd\_avg\_speed <- sd(avg\_speed)

sd\_total\_time <- sd(total\_time)

# Calculate variance

var\_avg\_speed <- var(avg\_speed)

var\_total\_time <- var(total\_time)

# Print the results

cat("Standard Deviation of AvgSpeed:", sd\_avg\_speed, "\n")

cat("Standard Deviation of TotalTime:", sd\_total\_time, "\n")

cat("Variance of AvgSpeed:", var\_avg\_speed, "\n")

cat("Variance of TotalTime:", var\_total\_time, "\n")

--------------------------------------------------------------------------------------------------------------------------------------------------------------

34.Consider a person want to take a censes / plot for the breast-cancer affected people through the years.Create a own dataset with this parameters age, tumorsize,inv-nodes [example between age 1-5 = no.of.count, 6-10=no.of.count,etc] Draw the Histogram, scatterplot,boxplot

# Load necessary libraries

if (!requireNamespace("ggplot2", quietly = TRUE)) {

install.packages("ggplot2")

}

if (!requireNamespace("dplyr", quietly = TRUE)) {

install.packages("dplyr")

}

# Load libraries

library(ggplot2)

library(dplyr)

# Set seed for reproducibility

set.seed(123)

# Create a synthetic dataset

n <- 100 # Number of samples

# Generate random ages between 1 and 80

age <- sample(1:80, n, replace = TRUE)

# Define age groups

age\_groups <- cut(age, breaks = seq(0, 80, by = 5), include.lowest = TRUE, labels = FALSE)

# Generate random tumor sizes between 1 and 10

tumor\_size <- rnorm(n, mean = 5, sd = 2)

# Generate random number of involved nodes between 0 and 20

inv\_nodes <- rpois(n, lambda = 5)

# Create a dataframe

cancer\_data <- data.frame(Age = age, AgeGroup = age\_groups, TumorSize = tumor\_size, InvNodes = inv\_nodes)

# Display the first few rows of the dataset

head(cancer\_data)

# Histogram of Age

ggplot(cancer\_data, aes(x = Age)) +

geom\_histogram(binwidth = 5, fill = "skyblue", color = "black", alpha = 0.7) +

labs(title = "Histogram of Age", x = "Age", y = "Frequency")

# Scatterplot of Age vs. TumorSize

ggplot(cancer\_data, aes(x = Age, y = TumorSize)) +

geom\_point(color = "darkred", alpha = 0.7) +

labs(title = "Scatterplot of Age vs. TumorSize", x = "Age", y = "Tumor Size")

# Boxplot of AgeGroups and TumorSize

ggplot(cancer\_data, aes(x = factor(AgeGroup), y = TumorSize)) +

geom\_boxplot(fill = "lightgreen", color = "darkgreen", alpha = 0.7) +

labs(title = "Boxplot of Age Groups and TumorSize", x = "Age Group", y = "Tumor Size")

--------------------------------------------------------------------------------------------------------------------------------------------------------------

37.a) Suppose that the “Diabetes data set ” data for analysis includes the attribute age. The age values for the data are (in increasing order) 30, 57, 68, 96, 39, 40, 20, 19, 42, 12, 25, 25, 65, 35, 30, 23, 23, 35, 45, 85. What is the mean? b) Suppose that the speed car is mentioned in different driving style.



Calculate the Inter quantile and standard deviation of the given data.

a)

age <- c(30, 57, 68, 96, 39, 40, 20, 19, 42, 12, 25, 25, 65, 35, 30, 23, 23, 35, 45, 85)

mean\_age <- mean(age)

print(mean\_age)

b)speed<-c(78.3 ,81.8 ,82 ,74.2 ,83.4 ,84.5 ,82.9 ,77.5 ,80.9 ,70.6 )

IQR(speed)

sd(speed)

----------------------------------------------------------------------------------------------------------------------------------------------------------------

38. a)Let us consider one example to make the calculation method clear. Assume that the minimum and maximum values for the feature F are $50,000 and $100,000 correspondingly. It needs to range F from 0 to 1. In accordance with min-max normalization, v = $80, b) 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 <- 50000

max\_value <- 100000

v <- 80

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

min\_max\_normalized

mean\_value <- mean(data)

standard\_deviation <- sd(data)

z\_score\_normalized <- (v - mean\_value) / standard\_deviation

z\_score\_normalized

----------------------------------------------------------------------------------------------------------------------------------------------------------------