DWDM R PROGRAMMING-PRACTICALS

1.List of Programs:

1The 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

**CODING-**

intervals <- c("1-5", "5-15", "15-20", "20-50", "50-80", "80-110")

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

cumulative\_freq <- cumsum(frequencies)

median\_interval\_index <- which(cumulative\_freq >= sum(frequencies)/2)[1]

lower\_bound <- as.numeric(strsplit(intervals[median\_interval\_index], "-")[[1]][1])

upper\_bound <- as.numeric(strsplit(intervals[median\_interval\_index], "-")[[1]][2])

cumulative\_freq\_before <- cumulative\_freq[median\_interval\_index] - frequencies[median\_interval\_index]

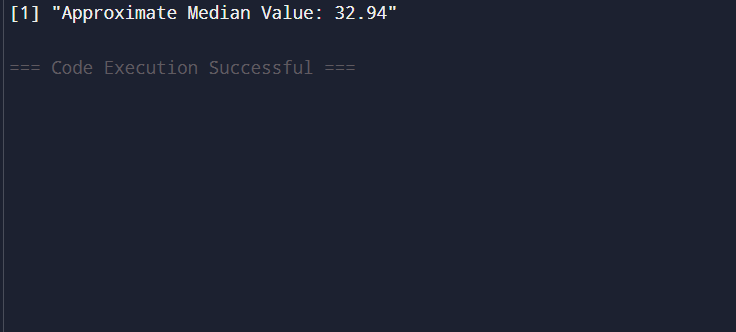
frequency\_median <- frequencies[median\_interval\_index]

width <- upper\_bound - lower\_bound

median\_value <- lower\_bound + ((sum(frequencies)/2 - cumulative\_freq\_before) / frequency\_median) \* width

print(paste("Approximate Median Value:", median\_value))

**OUTPUT-**



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.

(1) What is the mean of the data? What is the median?

**Coding:**

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

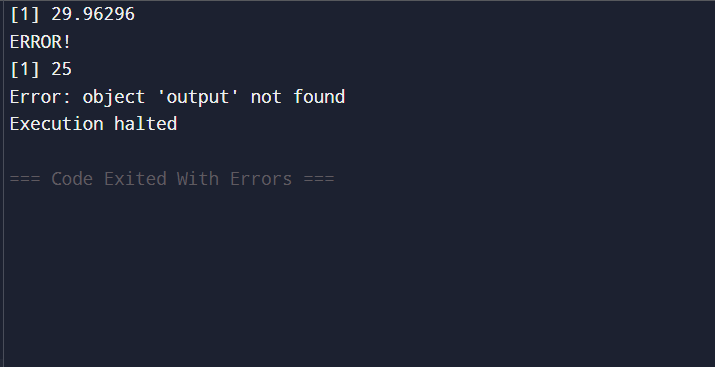
> #median

> median(x)

[1] 25

CODING FOR 2b-

**Output:**



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

**Coding:**

**#3a**

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

min<-min(data)

max<-max(data)

for (i in data)

{

result1=i-min

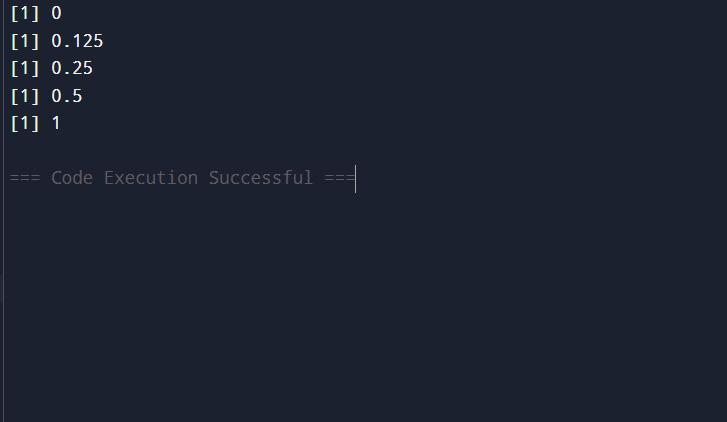
result2=max-min

result3=result1/result2

print(result3)

}

**OUTPUT:**



**#3b**

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

mean1<-mean(data)

deviation<-sd(data)

for (i in data)

{

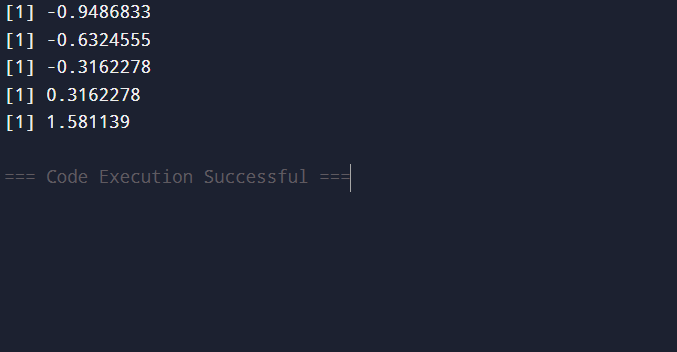
result1=i-mean1

result2=result1/deviation

print(result2)

}

**OUTPUT:**

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

**CODING-**

#binning

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)

range=6

bin1=c()

bin2=c()

bin3=c()

bin4=c()

for(i in data[1:range]){

bin1=append(bin1,i)

}

range1=range+1

range2=range\*2

for(j in data[range1:range2])

{

bin2=append(bin2,j)

}

range3=range2+1

range4=range\*3

for(k in data[range3:range4])

{

bin3=append(bin3,k)

}

range5=range4+1

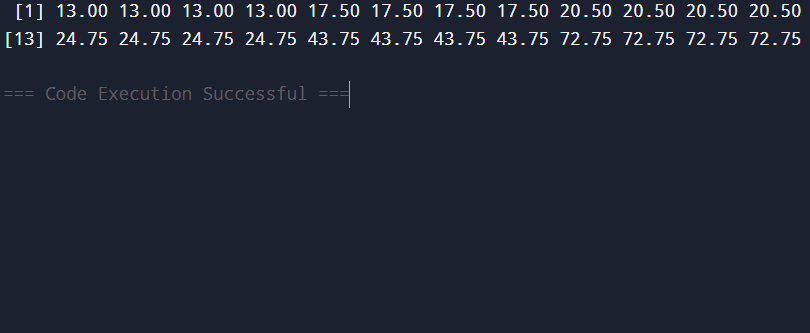
range6=range\*4

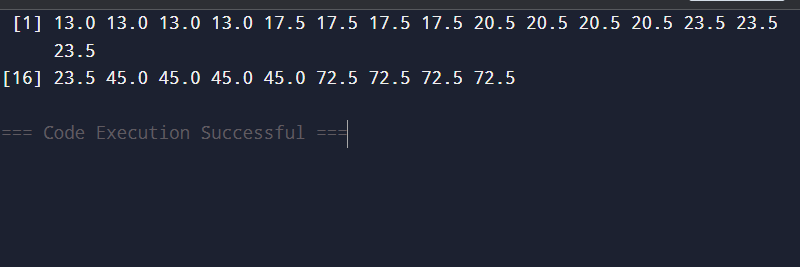
for(l in data[range5:range6]){

bin4=append(bin4,l)

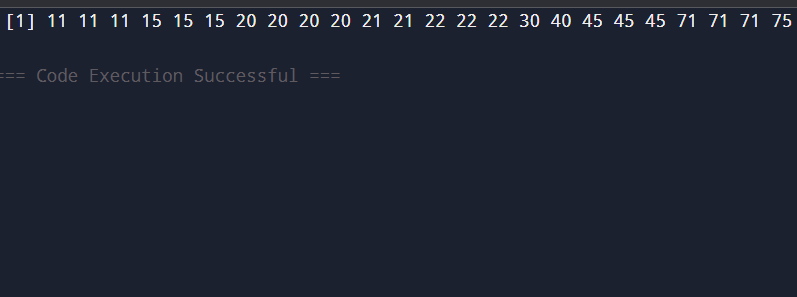
}

**#4a**

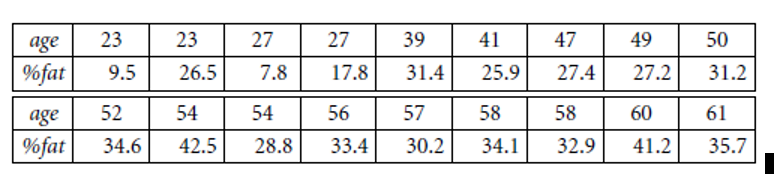
**#4b**



**#4c**

****

5) 5. 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)

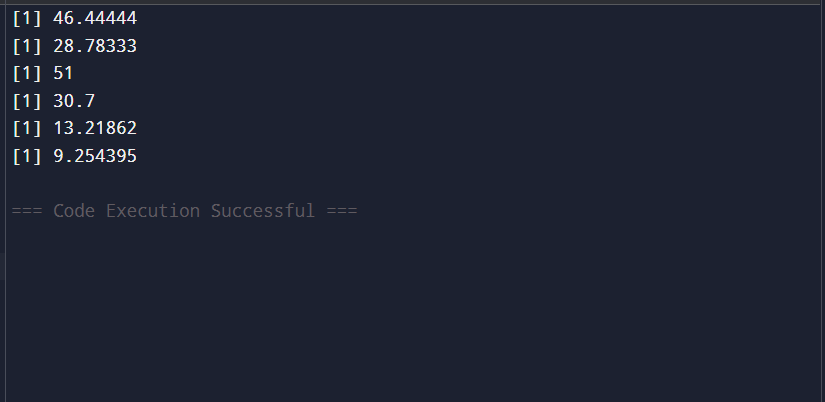
qqline(age)

qqnorm(body\_fat\_percent)

qqline(body\_fat\_percent)

**OUTPUT-**

**#5a**



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

**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)

**#6a**

#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)

}

**#6b**

#z score normalization

mean1<-mean(new\_age)

for (i in new\_age)

{

result1=i-mean1

result2=result1/12.94

print(result2)

}

**#6c**

#decimal scaling

n=200

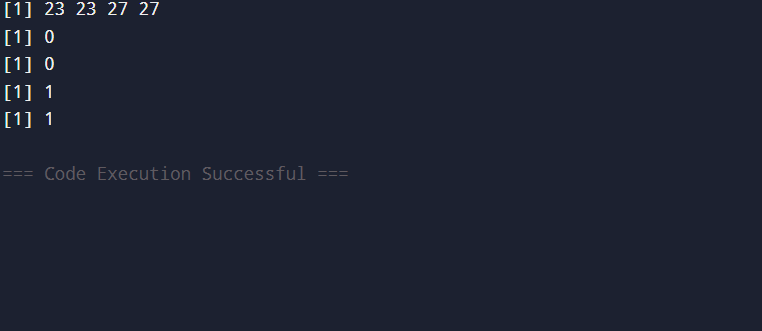
j=nchar(y)

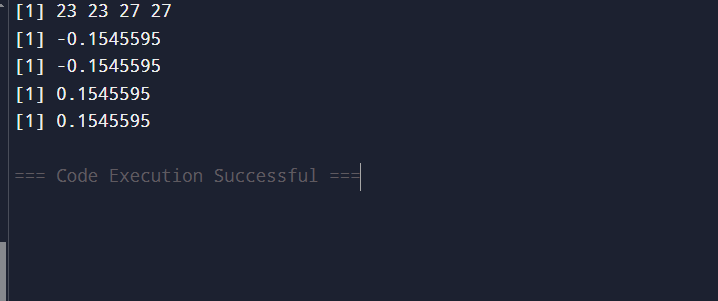
scaling=n/10^j

print(scaling)

**OUTPUT-**

6.a MIN MAX NORMALIZATION

6.b Z SCORE NORMALIZATION

6.c

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

**CODING-**

box\_no=c("box1","box2","box3","box4","box5","box6","box7","box8","box9","box10")

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

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

#dataframe

print(df)

#mean

mean(pencil)

#median

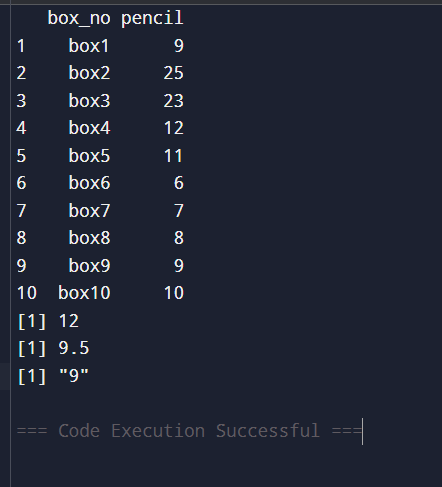
median(pencil)

#mode

mode=names(which.max(table(pencil)))

print(mode)

**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

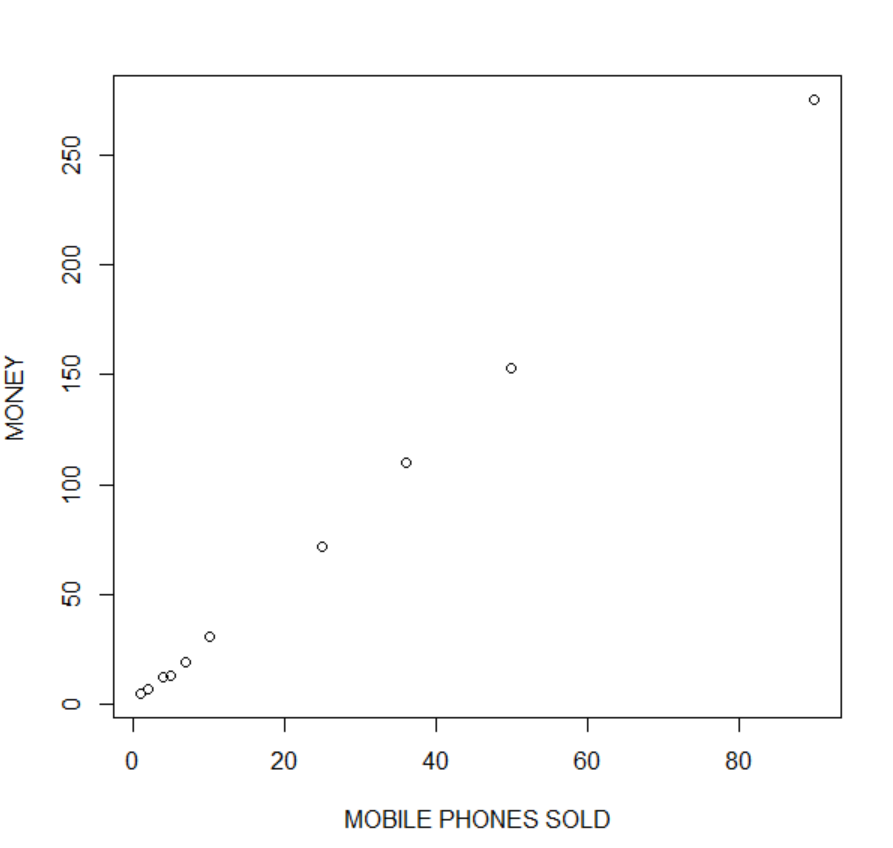
CODING-

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')

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

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)

OUTPUT-

> print(binning1)

[1] 55 60 71 63 55 65

> print(binning2)

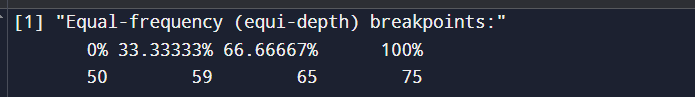
[1] 50 55 58 59 61 63

> print(binning3)

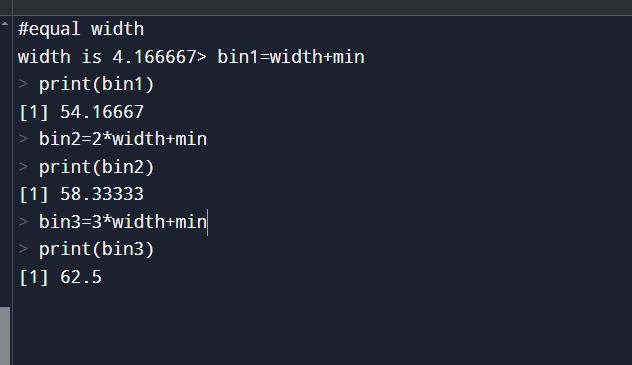
[1] 65 67 71 72 75 NA

Output:

#9a



**#9b**



10. Suppose that the speed car is mentioned in different driving style.

Regular 78.3 81.8 82 74.2 83.4 84.5 82.9 77.5 80.9 70.6 Speed

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

CODING-

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

#interquartile

IQR(speed)

#standard deviation

sd(speed)

OUTPUT-

