experiment - 14

ODD or even, ADDITION and subtraction programs.

Aim: To find odd or even, Addition and subtraction of number using R.

problem statement:

Download the data set from the UCI repository (or) any other appropriate website and perform (or) implement the central tendency measures. (mean, median, mode and midrange) and Data dispersion technize including summary.

Description:

Central Tendency:

i. odd or Even: odd numbers are those numbers that cannot be divided into two equal parts.

even numbers are those numbers that can be divided into two equal parts.

ii. Addition: Adding, something especially two or more number

iii subtraction: subtracting something, especially two or more numbers.

Inputs and outputs of program:

ODD or even:

```
num = as.integer(readline(prompt = 'Enter a number: '))
if ((num %% 2) == 0) {
  Print('Number is even')
} else {
  Print ('Number is odd')
}
```

output:

enter a number: 4
[1] "Number is even"

enter a number: 5
[1] "Number is odd"

ADDITION:

Input:

```
num1 = as.integer(readline(prompt="Enter a number1:"))
num2 = as.integer(readline(prompt="Enter a number2:"))
num3 = num1 + num2
print(num3)
```

output:

enter a number1: 2
enter a number2: 2
[1] 4

subtraction:

```
num1 = as.integer(readline(prompt="Enter a number1:"))
num2 = as.integer(readline(prompt="Enter a number2:"))
num3 = num1 - num2
print(num3)
```

output:

enter a number1: 4
enter a number2: 2
[1] 2.

Result: Thus the basic programs like odd or even addition and subtraction are executed successfully.

Carol

Experiment - 15

central Tendency and Data Dispersion measures.

Aim: Central Tendency and data Dispersion measures using R-TOOL.

problem statement:

Dowlad the dataset from the UCI respsitory (or) any other appropriate website and perform (or) implement the central tendency measures / mean, median, mode and mid range).

Central Tendency:

i. mean: The mean is the average of the numbers: a calculated "central" value of a set of numbers

ii median: The median is a statistical term that is one way of finding the average of a set of data points.

iii) Mode: The mode of a set of data values is the value that appears most often.

Inputs and outputs of central Tendency and Data Dispersion:

Mean:

Input:

```
names<-c("Ram", "shyam", "kumar")
age<-c(23,24,35)
marks<-c(88,78,25)
df<-data.frame(names,age,marks)
mean(df$age)
write.csv(df,"datafr.csv")
```

output:

```
>median (df $age)
[1] 24
```

MODE:

Input:

```
names <- c("Ram","shyam","kumar")
age <- c(23, 24, 35)
marks <- c(88, 78, 25)
df <- data.frame(names, age, marks)
mode (df $age)
write.csv (df, "datafr.csv").
```

output:

```
>mode (df $age)
[1] "numeric
```

Input:

above code'

```
summary (df $age)
write.csv(df, "datafr.csv")
```

output:

```
> summary
Min    1st Qu   median   mean   3rd Qu   max.
23.00  23.50    24.00    27.33  29.50    35.00
```

Result:

Thus the central tendency and measures of dispersion have been executed successfully. The outlier values are more than upper fence than are no lower fence values.

Creal

Experiment - 16

Central Tendency and data Dispersion measures

Aim: To central Tendency and data Dispersion measures using R-TOOL.

Measures of DISPERSION:

i. Inter Quartile Range: The interquartile range (IQR) is a measure of variability based on dividing a data set into quartiles. Quartiles divide a rank-ordered data set into four equal parts.

ii. Quartiles: A quartile is a statistical term describing a division of observations into four defined intervals based upon the values of the data and how they compare to the entire set of observations.

iii. Mid Range:
The arithmetic mean of the largest and the smallest values in sample or other group.

Input / outputs:

IQR:

Input:

```
names <- c("Ram", "shyam","kumar")
age<-c(23,24,35)
marks <-c (88,78,25)
df<-data.frame(names,age,marks)
IQR(df $age)
write.csv(df, "datafr.csv").
```

output:

```
> IQR (df $age)
[1] 6
```

Quantile:

Input:
```
names<-c("Ram","shyam","kuma")
age<-c(23,24,35)
marks<-c(88,78,25)
df<-data.frame(names,age,marks)
quantile(df$age)
write.csv(df,"datafr.csv").
```

output:
```
>quantile(df$age)
 0%   25%   50%   75%   100%
23.0  23.5  24.0  29.5  35.0
```

Range:

Input:
```
names<-c("Ram","shyam","kumar")
age<-c(23,24,35)
marks<-c(88,78,25)
df<-data.frame(names,age,marks)
range(df$age)
write.csv(df,"datafr.csv")
```

output:
```
>range(df$age)
[] 23 35
```

Result: Thus the program executed successfully.

Experiment - 17

PLOTTING GRAPHS USING R-TOOL

Aim: To find the plotting graphs using the R-Tool.

Problem statement:

Plot the boxplot, barplot and horizontal barplot for the dataset which was taken in the previous exercise.

Description:

consider a dataset diabetes, csv, where it contains the attributes are presgancies, Glucose, BloodPressure, skin Thickness, Insulin, BMI, DiabetesPedigree Function, Age, outcomes.

Implementation:

i. Box Plot

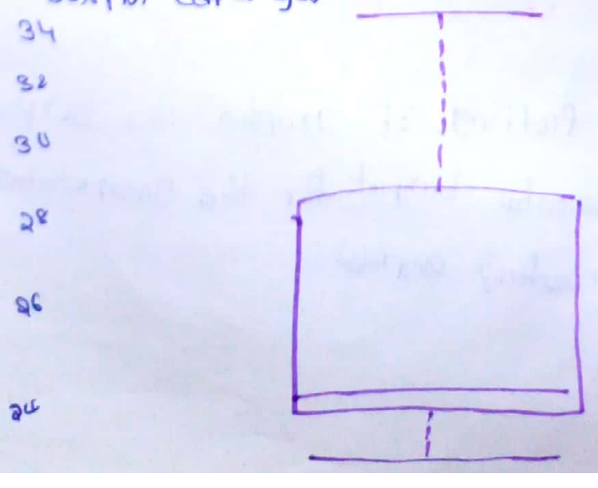
ii. BarPlot

Box Plot: A box plot is a graphical rendition of statistical data based on the minimum, first quartile, median, third quartile, and maximum.

Input:

```
names<-c("Ram", "shyam", "kumar")
age<-c(23,24,35)
marks<-c(88,78,25)
df<-data.frame(names,age,marks)
hist(df$age)
boxplot(df$age)
```

output:



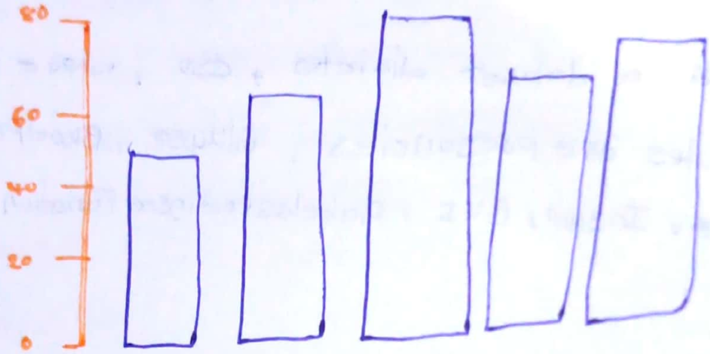
BARPLOT:

A barplot is one of the most common types of graphic. It shows tea relationship between a numeric and a categoric variable.

input:

```
a<-c(55,67,89,80,90)
barplot(a))
```

output:

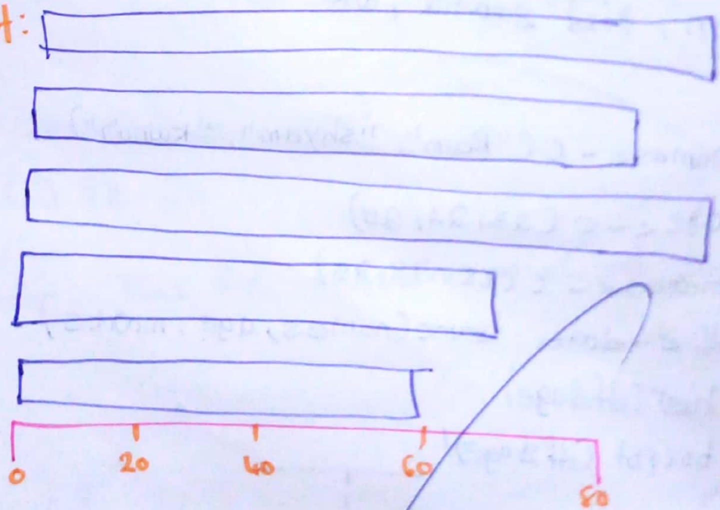


HORIZONTAL BARPLOT:

Input:

```
a<-c (55,67,89,80,90)
barplot (a)
barplot (a, horiz = TRUE)
```

output:



Result: Thus, the plotting of graphs like boxplot, barplot and horizontal barplot for the given dataset has been succesfully completed.

Experiment - 18

Plotting GRAPS using R-TOOL

Aim: To plot the histogram and scatterplot for the dataset using R-TOOL.

Problem statement:

Plot the histogram and scatterplot for the dataset which was taken in the previous exercise.

Implementation:

i. Histogram
ii. scatterplot (scatter smooth .

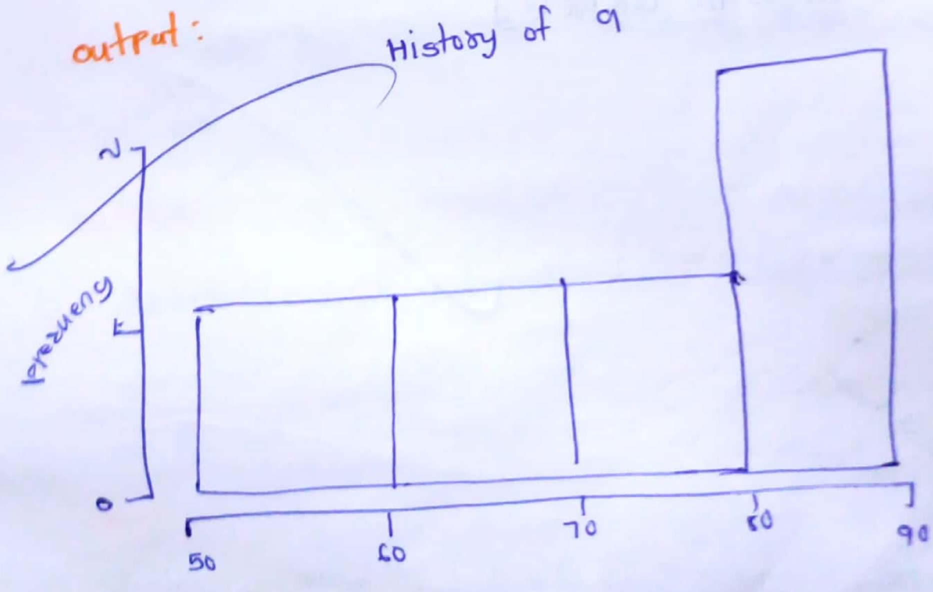
Histogram:

A diagram consisting of rectangles whose area is proportional to the frequency of a variable and whose width is equal to the class interval.

Input:

```
a<-c(55,67,89,80,90)
hist(a)
```

output:

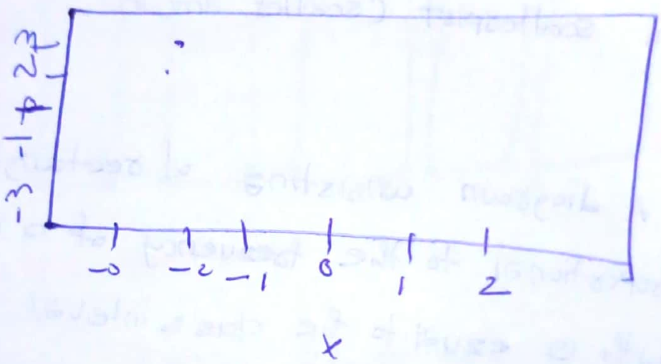


Scatterplot:

Input:

```
#Data
x<- rnorm(1000)
x<- rnorm (1000)
# smooth scatter plot
smoothscatter (x~x)
# ezuivalent to:
smoothscatter (x,y)
```

Output:



Result: Thus, the program plotting like histogram and scattempt for the given dataset successful completed.

Experiment - 19

perform correlation analysis and normalization.

Aim: To perform correlation analysis and normalization using R-Tool.

problem statement:

perform the correlation analysis for the numerical attribute using Pearson coefficient and for categorical attribute using chi-square and also perform the normalization technique using z score for the given data frames of particular dataset

Description:

A dataset of name diabetes.csv is given for the correlation analysis to calculate or to correlate between age and Insulin and the same dataset for the performance of normalization technique.

- **correlation Analysis:**

Steps involved:

i. create a new table with required data frames.

ii. After that apply the formula or query for the chi-square test.

Queries:

```
diabetes1<-table (diabetes$Age, diabetes$Insulin)
diabetes1
chi sq.test (diabetes1)
```

Input:

```
diabetes<-read.csv ("D:\\folders\\DWHDM\\diabetes.csv")
#step1
diabetes1<-table(diabates1 $Age, diabetes1 $Insulin)
diabetes1
#step2
```

chisq. test(diabetes1)

output:

```
> diabetes1

       0  14 15 16 18 22 23 25 29 32 36 37 38 40 41 42 43 44 45 46 49
21 28  0  0  0  1  0  1  1  0  0  0  0  0  1  0  0  0  0  1  0  0  0
22 29  0  0  1  0  0  0  0  0  1  1  1  0  0  0  0  0  1  1  0  0  0
23 10  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1  1  0  2  2
24 15  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  1  0  0  0  0  0
25 18  1  0  0  1  0  0  0  0  0  0  0  1  0  1  0  0  0  0  0  0  0

> chisq.test(diabetes1)

        Pearson's chi-squared test

data: diabets1 1
X-squared = 7561.7, df = 9435, p-value = 1
```

- z score normalization:

```
A<-c(diabetes$Age)
mean<-mean(A)
std<-sd(A)
zscore<-(A-mean)/std
zscore.
```

input:

```
diabetes<-read.csv("D:\\folders\\DWDM\diabetes.csv")
A<-c(diabetes$Age)
mean<-mean(A)
std<-sd(A)
zscore<-(A-mean)/std
zscore
```

output:

```
> sd(A)
[1] 11.76023
>
```

Result: Thus, the correlation analysis and normalization for the given dataset has been successfully execute and observed.

perform correlation analysis and Normalization using R-Tool

Aim: To perform correlation analysis and Normalization using R-Tool.

problem statement:

perform the correlation analysis for to perform the normalization technisu for the given data frames of particular dataset.

Description:

A dataset of name diabetes.csv is given for the correlation analysis, to calculate or to correlate between Age and Insulin and the same dataset for the performance of normalization technisue.

Normalization:

i. Mean Normalization:

- A<-c(diabetes$Age)
- Mean<-mean(A)

Input:

```
diabetes<-read.csv("D:\\folders\\DWHDM\\diabetes.csv")
A<-c (diabetes$Age)
#step
Mean<-mean(A)
```

output:

```
>mean(A)
[1] 33.24089
>
```

ii. Minimum NORMALIZATION.

- A<-c(diabetes$ Age)
- Minimum<-min(diabetes$Age)

Input:

diabetes<-read.csv("D:\\folders\\DWHDM\\diabetes.csv")

A<-c(diabetes$Age)

#step2

minimum<-min(diabetes$Age)

output:

>minimum

[1] 21

>

iii) MAXIMUM Normalization:

- A<-c(diabetes$Age)
- Maximum<-max(diabetes$Age)

input:

diabetes<-read.csv("D:\\folders\\DWHDM\\diabetes.csv")

A<-c(diabetes$Age)

#step 3

Maximum<-max(diabetes$Age)

output:

>Maximum

[1] 81

iv) MINMAX NORMALIZATION

- A<-c(diabetes$Age)
- MinMax<-(A-minimum)/(maximum-minimum)
- Minmax

Input: diabetes<-read.csv("D:\\folders\\DWHDM\\diabetes.csv")

A<-c(diabetes$Age)

Minmax<-(A-minimum)/(Maximum-minimum)

Minmax

output:

Min Max

[1] 0.48333333

[6] 0.15000000 08

[11] 0.15000000

[16] 0.18333333

[21] 0.10000000

v) Decimal scaling Normalization:

. A=c

Experiment : 21

Aim: To find mean, median, mode of numbers using R tool.

Algorith:

step1: open Rstudio application

step2: write a program for mean, median and mode.

step3: Get the required answers

program:

```
names <- c("Ram", "shyam", "kuma")
age <- c (23, 24, 25)
marks <- c (88, 78, 25)
df <- data.frame (names, age, marks)
mean (df$age)
median(df $age)
mode (df $age)
summary (df $age)
```

outputs :

mean :- 27.33

median : 24

mode : numeric

Summary :

Min	1st Qu	Med	Mean	3rd Qu	Max
23.00	23.50	24.00	27.33	29.50	35.00