INDEX

Practical 1

Using 'R' execute the basic array,list and frames.

Array: While matrices are confined to two dimensions,arrays can be of any number of dimensions.The array function takes a dim attribute which creates the required number of dimension.

Example:-

```
a=array(c("yes","no"),dim=c(3,3,2))

> print(a)
```

Output:

```
, , 1

     [,1]   [,2]   [,3]
[1,] "yes" "no"   "yes"
[2,] "no"   "yes" "no"
[3,] "yes" "no"   "yes"
, , 2

     [,1]   [,2]   [,3]
[1,] "no"   "yes" "no"
[2,] "yes" "no"   "yes"
[3,] "no"   "yes" "no"
```

Frames: Data frames are tabular data objects.Unlike a matrix in data fram each column can contain different modes of data.The first column can be numeric while the second column can be character and third column can be logical.It is a list of vectors of equal length. Data frames are created using the data.frame() function.

Example:-

```
> BMI=data.frame(gender=c("Female","Female","Male","Female"),
```

```
+ height=c(152,155,165.2,170.0),

+ weight=c(35,40,60,55),

+ age=c(14,16,20,17))

> print(BMI)
```

Output:

	Gender	Height	Weight	Age
1	Female	152.0	35	14
2	Female	155.0	40	16
3	Male	165.2	60	20
4	Female	170.0	55	17

List: A list is an R-object which can contain many different types of elements inside it like vectors, functions and even another list inside it.

Example:

```
>list1=list(c(2,6.5,sin))

> print(list1)

> print(class(list1))
```

Output:

```
[[1]]

[[1]][[1]]

[1] 2

[[1]][[2]]

[1] 6.5

[[1]][[3]]

function (x)   .Primitive("sin")
```

Practical 2

Create a Matrix using R and Perform the operation addition, inverse, transpose and multiplication operations.

```
> A<-matrix(c(2,3,-2,1,2,2),3,2)
> print(A)
     [,1] [,2]
[1,]    2    1
[2,]    3    2
[3,]   -2    2
> B<-matrix(c(1,4,-2,1,2,1),3,2)
> print(B)
     [,1] [,2]
[1,]    1    1
[2,]    4    2
[3,]   -2    1
```

(1) Addition matrix:

```
> C<-A+B
> print(C)
     [,1] [,2]
[1,]    3    2
[2,]    7    4
[3,]   -4    3
```

(2) <u>Inverse matrix:</u>

\> E<-matrix(c(2,1,6,1,3,4,6,4,-2),3,3)

\> print(E)

```
      [,1] [,2] [,3]
[1,]    2    1    6
[2,]    1    3    4
[3,]    6    4   -2
```

\> AI<-solve(E)

\> print(AI)

```
            [,1]          [,2]          [,3]
[1,]   0.2156863  -0.25490196   0.13725490
[2,]  -0.2549020   0.39215686   0.01960784
[3,]   0.1372549   0.01960784  -0.04901961
```

(3) <u>Transpose matrix:</u>

\> AT<-(A)

\> print(AT)

```
      [,1] [,2]
[1,]    2    1
[2,]    3    2
[3,]   -2    2
```

\> ATT<-t(AT)

\> print(ATT)

```
      [,1] [,2] [,3]
```

```
[1,]      2      3     -2
[2,]      1      2      2
```

(4)   <u>**Multiplication matrix:**</u>

```
> D<-A*B
> print(D)
      [,1] [,2]
[1,]    2    1
[2,]   12    4
[3,]    4    2
```

Practical 3

Using R execute the statistical functions mean,median,mode,quartile,range,Inter-quartile range,histogram.

(1)Mean

Example:

```
> x<-c(12,7,3,4,2,18,2,54,-21,8,-5)

> mean<-mean(x)

> print(mean)
```

Output:

```
[1] 7.636364
```

(2)Median

Example:

```
> x<-c(12,7,3,4,2,18,2,54,-21,8,-5)

> median<-median(x)

> print(median)
```

Output:

```
[1] 4
```

(3)Mode

Example:

```
>getmode<- function(v) {

+     uniqv<- unique(v)

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

```
+ }

> v <- c(2,1,2,3,1,2,3,4,1,5,5,3,2,3)

> result <- getmode(v)

> print(result)
```

Output:

```
     [1] 2
```

(4)Quartile

Example:

```
> nuclear <- c(7, 20, 16, 6, 58, 9, 20, 50, 23, 33, 8, 10, 15, 16, 104)

> quantile(nuclear)
```

Output:

```
   0%    25%    50%    75%   100%

   6.0    9.5   16.0   28.0 104.0
```

(5)Range and histogram

Example:

```
> x<-c(1,2.3,2,3,4,8,12,43,-4,-1)

> r<-range(x)

> print(r)

[1] -4 43

> diff(r)

[1] 47

> hist(r)
```

Output:

Histogram of r
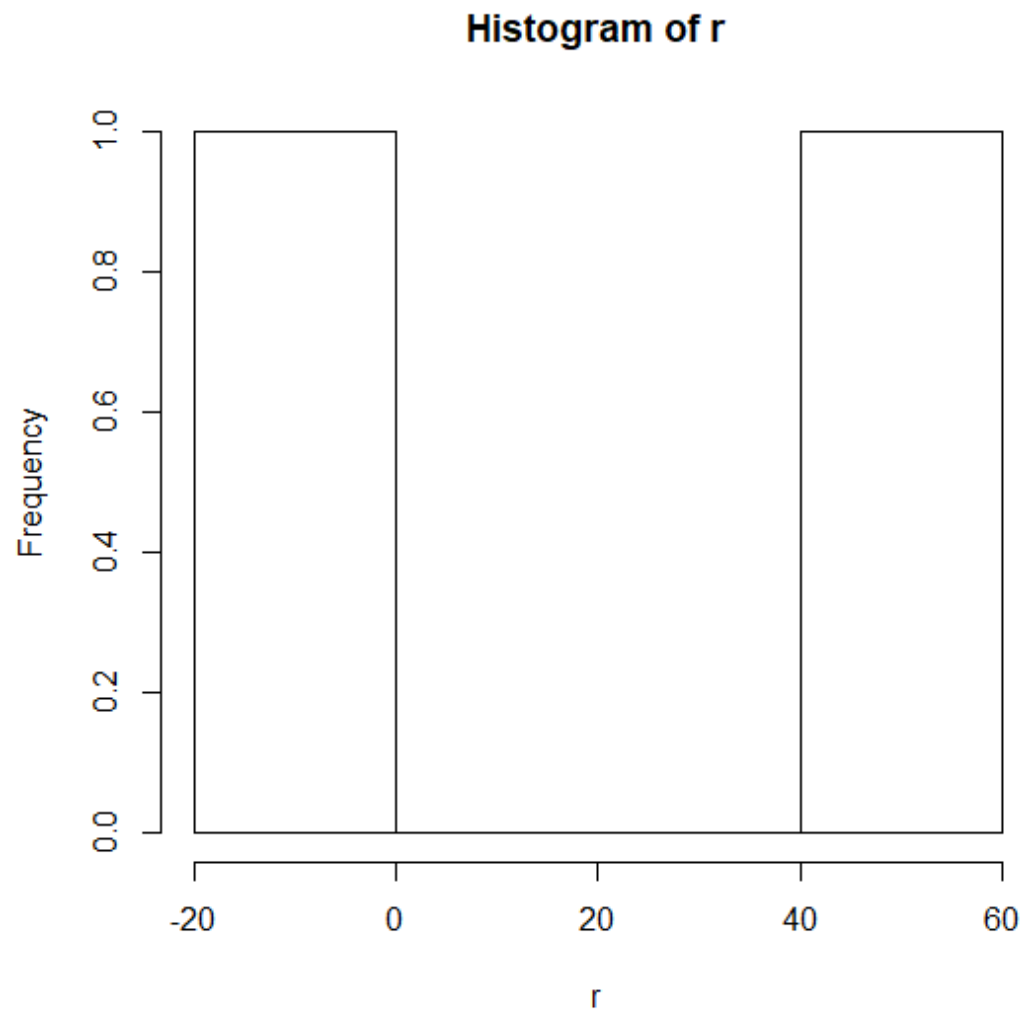
(6)Inter-Quartile Range

> x<-c(12,19,21,24,26,29,33,35,36)

> IQR(x)

[1] 12

Practical 4

Using R import the data from excel /.CSC file and perform the above function.

Example:

> data1<-read.csv(file.choose(),header=T)

> data1

	No.	x
1	2	12
2	2	7
3	3	3
4	4	4
5	5	2
6	6	18
7	7	2
8	8	54
9	9	-21
10	10	8
11	11	-5

> mean(x)

[1] 7.03

> median(x)

[1] 2.65

Practical 5

Using R import the data from excel /.CSC file and calculate the standard deviation, variance and co-variance.

(1) Standard Deviation:

Example:

```
        x
1       2
2       3
3       7
4       8
5      10
```

data1<-read.csv(file.choose(),header=T)

> data1

Output:

>sd(data1$x)

[1] 1.599731

(2)Variance

Example:

```
      x
1     1
2     2
3     3
4     4
5     5
6     6
7     7
8     8
9     9
```

Output: data1<-

read.csv(file.choose(),header=T) >

data1

Practical 6

Using R import the data from Excel /.CSV and draw the skewnessand kurtosis.

Q. Seema is interested on the elapse time (in minutes.). She spends on riding a tricycle from home to school for three weeks(excluding weekends).She obtain the following data:

19.09,19.55,17.69,17.63,25.15,27.27,25.24,21.65,20.92,22.61,15.71,22.04,22.60,24.25

Compute and interpret the skewness and kurtosis.

Solution:

>Data1<-read.CSV(file.choose(),header=T)

>Data1

>time<-c(19.09,19.55,17.69,17.63,25.15,27.27,25.24,21.65,20.92,22.61,15.71,22.04,22.60,24.25)

> library(moments)

> skewness(time)

[1] -0.05336739

> kurtosis(time)

[1] 2.131211

Practical 7

Import the data from excel /.csv and perform the hypothetical testing.

Problem: Suppose the manufacturer claims that the mean lifetime of a light bulb is more than 10,000 hours. In a sample of 30 light bulbs, it was found that they only last 9,900 hours on average. Assume the population standard deviation is 120 hours . At .05 significance level, can we reject the claim by the manufacturer?

```
xbar=9900     //Sample mean
mu0=10000     //hypothesized value
sigma=120     //Populatios standard deviation
n=30          //sample size
z=(xbar-mu0)/(sigma/sqrt(n))
z             //test static
```

[1] -4.564355

We then compute the critical value at .05 significance level

> alpha=.05

>t.alpha=qt(1-alpha,df=n-1)

>t.alpha   //critical value

[1] 1.699127

Problem: Suppose the food label on a cookie bag states that there is at most 2 grams of saturated fat in a single cookie. In a sample of 35 cookie, it is found that the mean amount of saturated fat per cookie is 2.1grams. Assume that the population standard deviation is 0.25 grams. At 0.05 significance level,can we reject the claim on food label.

Soln:

>xbar<-2.1

```
> mu0<-2
> Sigma<-0.25
> n<-35
> z=(xbar-mu0)/(Sigma/sqrt(n))
> z
[1] 2.366432
```

We then compute the critical value at .05 significance level

```
>z.alpha<-qnorm(1-alpha)
> alpha<-0.05
>z.alpha
[1] 1.644854
```

Practical 8

Import the data from Excel/.CSV and perform the Chi-squared Test.

>Load Packages (MASS)

> library("MASS")

>   local({pkg<-   select.list(sort(.packages(all.available   = TRUE)),graphics=TRUE)

+ if(nchar(pkg)) library(pkg, character.only=TRUE)})

> print(str(Cars93))

Output:

'data.frame':   93 obs. of   27 variables:

 $ Manufacturer   : Factor w/ 32 levels "Acura","Audi",..: 1 1 2 2 3 4 4 4 4 5 ...

 $ Model   : Factor w/ 93 levels "100","190E","240",..: 49 56 9 1 6 24 54 74 73 35 ...

 $ Type   : Factor w/ 6 levels "Compact","Large",..: 4 3 1 3 3 3 2 2 3 2 ...

 $ Min.Price   : num   12.9 29.2 25.9 30.8 23.7 14.2 19.9 22.6 26.3 33 ...

 $ Price   : num   15.9 33.9 29.1 37.7 30 15.7 20.8 23.7 26.3 34.7 ...

 $ Max.Price   : num   18.8 38.7 32.3 44.6 36.2 17.3 21.7 24.9 26.3 36.3 ...

 $ MPG.city   : int   25 18 20 19 22 22 19 16 19 16 ...

 $ MPG.highway   : int   31 25 26 26 30 31 28 25 27 25 ...

 $ AirBags   : Factor w/ 3 levels "Driver & Passenger",..: 3 1 2 1 2 2 2 2 2 2 ...

 $ DriveTrain   : Factor w/ 3 levels "4WD","Front",..: 2 2 2 2 3 2 2 3 2 2 ...

```
 $ Cylinders          : Factor w/ 6 levels "3","4","5","6",..: 2 4 4 4 2 2 4 4
4 5 ...
 $ EngineSize         : num    1.8 3.2 2.8 2.8 3.5 2.2 3.8 5.7 3.8 4.9 ...
 $ Horsepower         : int    140 200 172 172 208 110 170 180 170 200
...
 $ RPM                : int   6300 5500 5500 5500 5700 5200 4800
4000 4800 4100 ...
 $ Rev.per.mile       : int   2890 2335 2280 2535 2545 2565 1570 1320
1690 1510 ...
 $ Man.trans.avail    : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 1 1 1 1 1 ...
 $ Fuel.tank.capacity: num    13.2 18 16.9 21.1 21.1 16.4 18 23 18.8 18 ...
 $ Passengers         : int   5 5 5 6 4 6 6 6 5 6 ...
 $ Length             : int    177 195 180 193 186 189 200 216 198 206
...
 $ Wheelbase          : int    102 115 102 106 109 105 111 116 108 114 ...
 $ Width              : int    68 71 67 70 69 69 74 78 73 73 ...
 $ Turn.circle        : int    37 38 37 37 39 41 42 45 41 43 ...
 $ Rear.seat.room     : num    26.5 30 28 31 27 28 30.5 30.5 26.5 35 ...
 $ Luggage.room       : int    11 15 14 17 13 16 17 21 14 18 ...
 $ Weight             : int    2705 3560 3375 3405 3640 2880 3470 4105
3495 3620 ...
 $ Origin             : Factor w/ 2 levels "USA","non-USA": 2 2 2 2 2 1
1 1 1 1 ...
 $ Make               : Factor w/ 93 levels "Acura Integra",..: 1 2 4 3 5 6
7 9 8 10 ...
NULL
```

Chisquare Code:

```
> library("MASS")
```

```
>car.data<-data.frame(Cars93$AirBags,Cars93$Type)

>car.data=table(Cars93$AirBags,Cars93$Type)

> print(car.data)

> print(chisq.test(car.data))
```

Output:

	Compact	Large	Midsize	Small	Sporty	Van
Driver & Passenger	2	4	7	0	3	0
Driver only	9	7	11	5	8	3
None	5	0	4	16	3	6

Pearson's Chi-squared test

data: car.data

X-squared = 33.001, df = 10, p-value = 0.0002723

Q. Suppose there are 12 multiple types question in an English class quiz each question has 5 possible answers, and only 1 of them is correct. Find the probability of having 4 or less correct answers if a student attempt to every question at random.

Soln: Since only one out of 5 possible answers

```
>dbinom(4,size=12,prob=0.2)
```
[1] 0.1328756

To find the probability of having 4 or less correct answers by random attempts, we apply the function dbinom with x=0,…,4.

```
>dbinom(0,size=12,prob=0.2)+
```

+ dbinom(1,size=12,prob=0.2)+

+ dbinom(2,size=12,prob=0.2)+

+ dbinom(3,size=12,prob=0.2)+

+ dbinom(4,size=12,prob=0.2)

[1] 0.9274445

Alternatively, we can use the cumulative probability function for binomial distribution pbinom.

>pbinom(4,size=12,prob=0.2)

[1] 0.9274445

Normal distribution:

Q.Assume that the test scores of a college entrance exam fits a normal distribution. Furthermore, the mean test score is 72 and the S.D is 15.2. What is the percentage of students scoring 84 or more in the exam?

Soln.:We apply the function pnorm of the normal distribution with mean 72 and   S.D 15.2.Since we are looking for the percentage of students scoring higher than 84.

Practical 9

Using R perform the binomial and normal distribution on the data.

x<-seq(0,50,by=1)

y<-dbinom(x,50,0.5)

png(file="dbinom.png")

dev.off()

null device

           1

> plot(x,y)
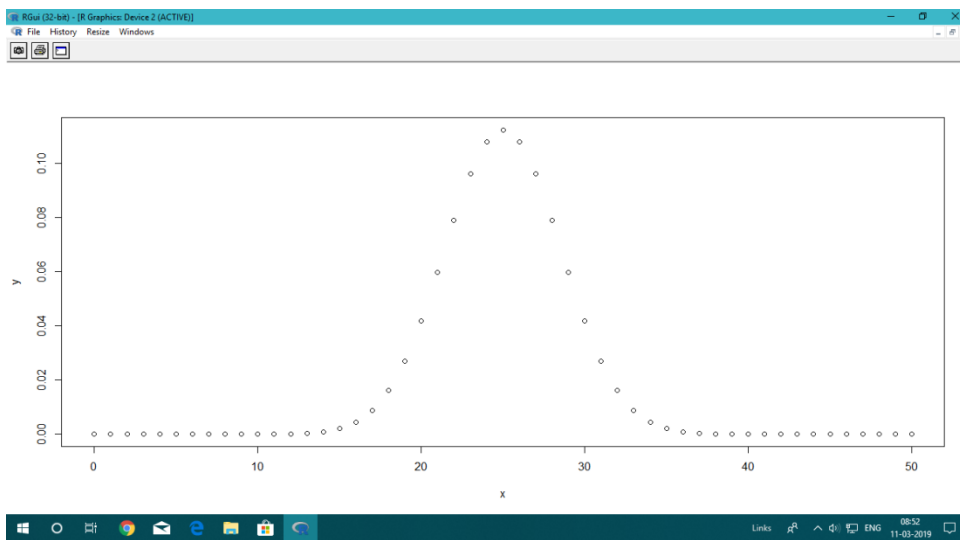
Practical 10

Perform the Linear Regression using R.

> x<-c(151,174,138,186,128,136,179,163,152,131)

> y<-c(63,81,56,91,47,57,76,72,62,48)

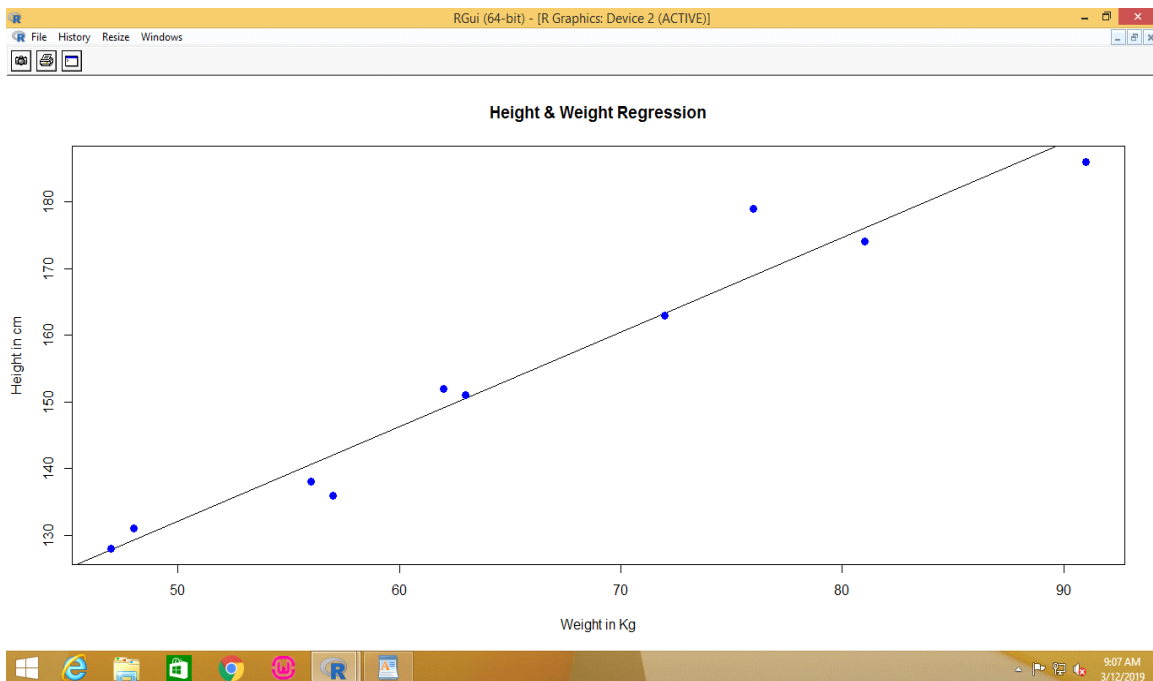> relation<-lm(y~x)

>png(file="linearregression.png")

>dev.off()

null device

      1

> plot(y,x,col="blue",main="Height & Weight Regression",

+ abline(lm(x~y)),cex=1.3,pch=16,xlab="Weight in Kg",ylab="Height in cm")

Output:

Practical 11

Perform the correlation using R tool.

year=c(2000,2001,2002,2003,2004)

> rate=c(9.34,8.50,7.62,6.93,6.60)

> plot(year,rate)

>cor(year,rate)

[1] -0.9880813

Output: