Statistics for Engineers Prashanth.S

19MID0020

14th July 2020

```
> # Power of a number
> a=8
> print(a**2)  # 8*8-->64
[1] 64
> print(a^2)   # 8*8-->64
[1] 64
>
> print(c(1,2,3,4,5)^2)      # All the values are powered in terms of 2
[1]  1  4  9 16 25
> print(c(1,2,3,4)*c(3,4))  # 3  8  9 16
[1]  3  8  9 16
> print(c(2,4,6,8)*c(-2,-4,-6,-8))  # The values are multiplied element-wise -4 -16 -36 -64
[1]  -4 -16 -36 -64
> print(c(1,2,3,4)+5)        # All the values are added with 5
[1] 6 7 8 9
>
> # Integer division (quotient)
> print(c(2,4,6,8)%/%c(2,3))  # 1 1 3 2
[1] 1 1 3 2
>
> # Modulo division (remainder)
> print(c(2,4,6,8)%%c(2,3))   # 0 1 0 2
[1] 0 1 0 2
> # Maximum and minimum function
> print(max(c(2,4,5,1))) # 5
[1] 5
> print(min(c(2,4,5,1))) # 1
[1] 1
>
> # abs(),round(),sqrt(),sum(),prod()
> print(abs(-2))           # 2
[1] 2
> print(round(12.78))      # 13
[1] 13
> print(sqrt(c(2,4,6,8)))  # 1.414214 2.000000 2.449490 2.828427
[1] 1.414214 2.000000 2.449490 2.828427
> print(sum(c(2,4,6,8)))   # 20
[1] 20
> print(prod(c(2,4,6,8)))  # 384
[1] 384
```

```
> ############### lograthemic function #################s
> # Natural log (ln --> log to the base e)
> print(log(5))   # 1.609438
[1] 1.609438
>
> # Common log (log --> log to the base 10)
> print(log10(5))        # 0.69897
[1] 0.69897
> print(log(5,base=10))  # 0.69897
[1] 0.69897
>
> # log(number,base=<number>) --> We can find the log of any number with any base
> print(log(9,base=4))  # 1.584963
[1] 1.584963
```

```
> ################ Complex functions #########################
> a = 3+5i
> print(Re(a))    # real part of a --> 3
[1] 3
> print(Im(a))    # imaginary part of a --> 5
[1] 5
> print(Conj(a))  # conjugate of a --> 3-5i
[1] 3-5i
> print(Mod(a))   # modulus of a  -->  5.830952
[1] 5.830952
> print(Arg(a))   # argument of a --> 1.030377
[1] 1.030377
```

```
> ###############  Matrix  ##########################
> x = matrix(nrow=3,ncol=3,data=c(2,4,6,3,6,9,5,10,15))  # Creating a m
se)
> y = matrix(nrow=3,ncol=3,data=c(2,4,6,3,6,9,5,10,15),byrow=TRUE)  # C
d row wise)
> t = matrix(nrow=2,ncol=3,data=100) ## Creating a matrix of single dat
> d = diag(1,nrow=2,ncol=2)          ## Creating a diagonal matrix
> x[2,3] # Accessing the matrix elements
[1] 10
>
> ## Properties of a matrix ##
> print(dim(x))
[1] 3 3
> print(attributes(x))
$dim
[1] 3 3
```

```
> print(nrow(x))
[1] 3
> print(ncol(x))
[1] 3
> print(mode(x))  # types of storage
[1] "numeric"
> print(t(x))     # diagonal of a matrix
     [,1] [,2] [,3]
[1,]    2    4    6
[2,]    3    6    9
[3,]    5   10   15
```

```
> print(x*4)   # multiplying a matrix with a constant term
     [,1] [,2] [,3]
[1,]    8   12   20
[2,]   16   24   40
[3,]   24   36   60
> print(y%*%y) # Matrix multiplication
     [,1] [,2] [,3]
[1,]   46   92  138
[2,]   69  138  207
[3,]  115  230  345
> print(y*y)   # Normal multiplication
     [,1] [,2] [,3]
[1,]    4   16   36
[2,]    9   36   81
[3,]   25  100  225
> print(crossprod(x)) # t(x)%*%x [transpose(martix) (matrix multiplication) matrix]
     [,1] [,2] [,3]
[1,]   56   84  140
[2,]   84  126  210
[3,]  140  210  350
```

```
> print(x+6*x) # addition of a matrix
     [,1] [,2] [,3]
[1,]   14   21   35
[2,]   28   42   70
[3,]   42   63  105
> print(6*x-x) # subtraction of a matrix
     [,1] [,2] [,3]
[1,]   10   15   25
[2,]   20   30   50
[3,]   30   45   75
>
> print(x[2,])  # second row of a matrix
[1]  4  6 10
> print(x[,2])  # second column of a matrix
[1] 3 6 9
```

(25th July 2020)

```r
empid  = c(1,2,3,4,5,6,7,8,9,10)           # vector empid
age    = c(30,37,45,32,50,31,53,21,54,42)  # vector age
gender = c(0,1,1,0,1,0,0,1,0,1)
status = c(1,2,1,1,2,2,1,2,1,2)

# data.frame
empinfo = data.frame(empid,age,gender,status)

# factor() --> will convert the numbers provided into certain specific labels
empinfo$gender = factor(empinfo$gender,labels=c("male","female"))   # 0-->male and 1-->female
empinfo$status = factor(empinfo$status,labels=c("staff","faculty")) # 1-->staff and 2-->faculty


# subset() --> will take a subset of data from the main frame with specific condition
genderm = subset(empinfo,empinfo$gender=='male')     # Taking the rows whose gender is male
genderf = subset(empinfo,empinfo$gender=='female')   # Taking the rows whose gender is female

# summary() --> summary statiscs of empinfo
summary(empinfo)
summary(genderm) # Summary statiscs of male employ
summary(genderf) # Summary statiscs of female employ
summary(empinfo$age) # Summary statiscs of age

# table()
# One way table
table1 = table(empinfo$gender)  # In gender, how many male and female are present???
table2 = table(empinfo$status)  # In status, how many staff's and faculty's are present???
# Two way table
table3 = table(empinfo$gender,empinfo$status) # gender --> row  and status --> column

## Graphical Representations in R
plot(empinfo$age,type="l",main="Age of employ",      # main --> heading
     xlab="emp id",ylab="age in years",col="red")     # type = 'l' --> type of the line
                                                      # xlab --> X-label  and ylab --> Y-label
                                                      # col  --> color of the line
# Pie-chart for the table
table4 = table(empinfo$gender)
pie(table4)

### barplot
barplot(table5,beside=T,xlim=c(1,15),ylim=c(0,5),
        col=c("blue","red"),legend=rownames(table5))

### boxplot
boxplot(empinfo$age~empinfo$status,col=c("red","blue"))

```

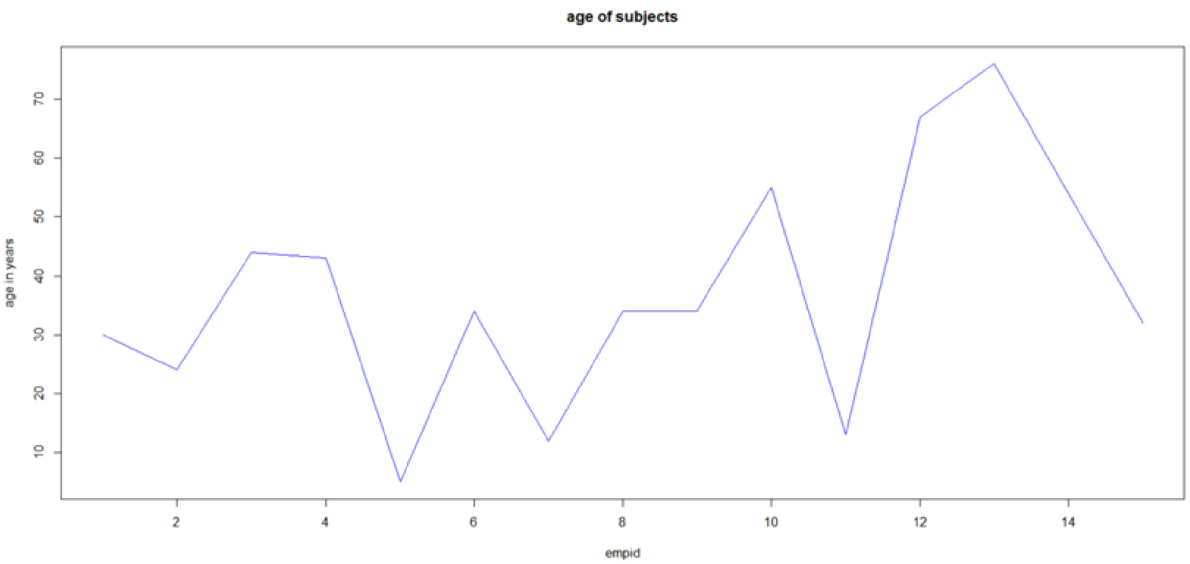
```
> empinfo
   empid age gender  status
1      1  30   male   staff
2      2  37 female faculty
3      3  45 female   staff
4      4  32   male   staff
5      5  50 female faculty
6      6  31   male faculty
7      7  53   male   staff
8      8  21 female faculty
9      9  54   male   staff
10    10  42 female faculty
> empinfo$gender
 [1] male   female female male   female male   male   female male   female
Levels: male female
> empinfo$status
 [1] staff   faculty staff   staff   faculty faculty staff   faculty staff   faculty
Levels: staff faculty
```

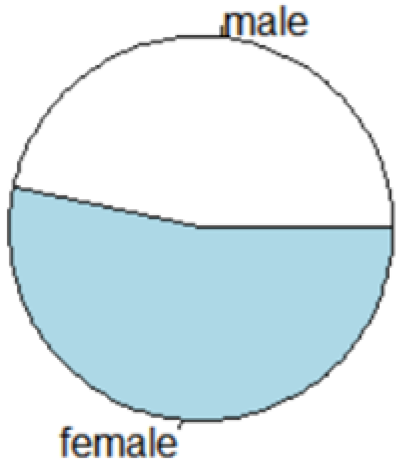
```
> genderm
  empid age gender  status
1     1  30   male   staff
4     4  32   male   staff
6     6  31   male faculty
7     7  53   male   staff
9     9  54   male   staff
> genderf
   empid age gender  status
2      2  37 female faculty
3      3  45 female   staff
5      5  50 female faculty
8      8  21 female faculty
10    10  42 female faculty
> |
```

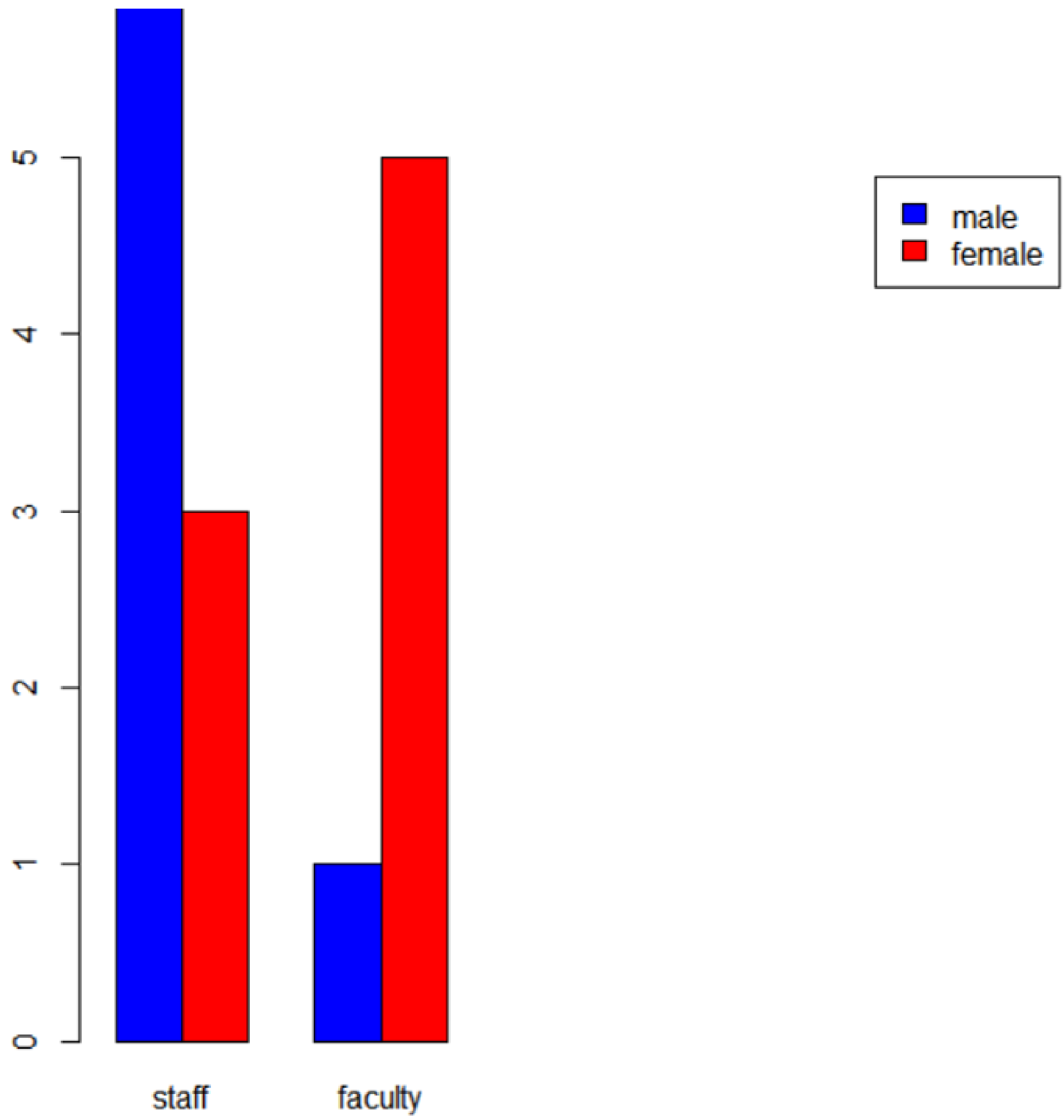
```
> summary(empinfo)
     empid            age           gender      status
 Min.   : 1.00   Min.   :21.00   male  :5   staff  :5
 1st Qu.: 3.25   1st Qu.:31.25   female:5   faculty:5
 Median : 5.50   Median :39.50
 Mean   : 5.50   Mean   :39.50
 3rd Qu.: 7.75   3rd Qu.:48.75
 Max.   :10.00   Max.   :54.00
> summary(genderm) # Summary statiscs of male employ
     empid          age         gender      status
 Min.   :1.0   Min.   :30   male  :5   staff  :4
 1st Qu.:4.0   1st Qu.:31   female:0   faculty:1
 Median :6.0   Median :32
 Mean   :5.4   Mean   :40
 3rd Qu.:7.0   3rd Qu.:53
 Max.   :9.0   Max.   :54
```

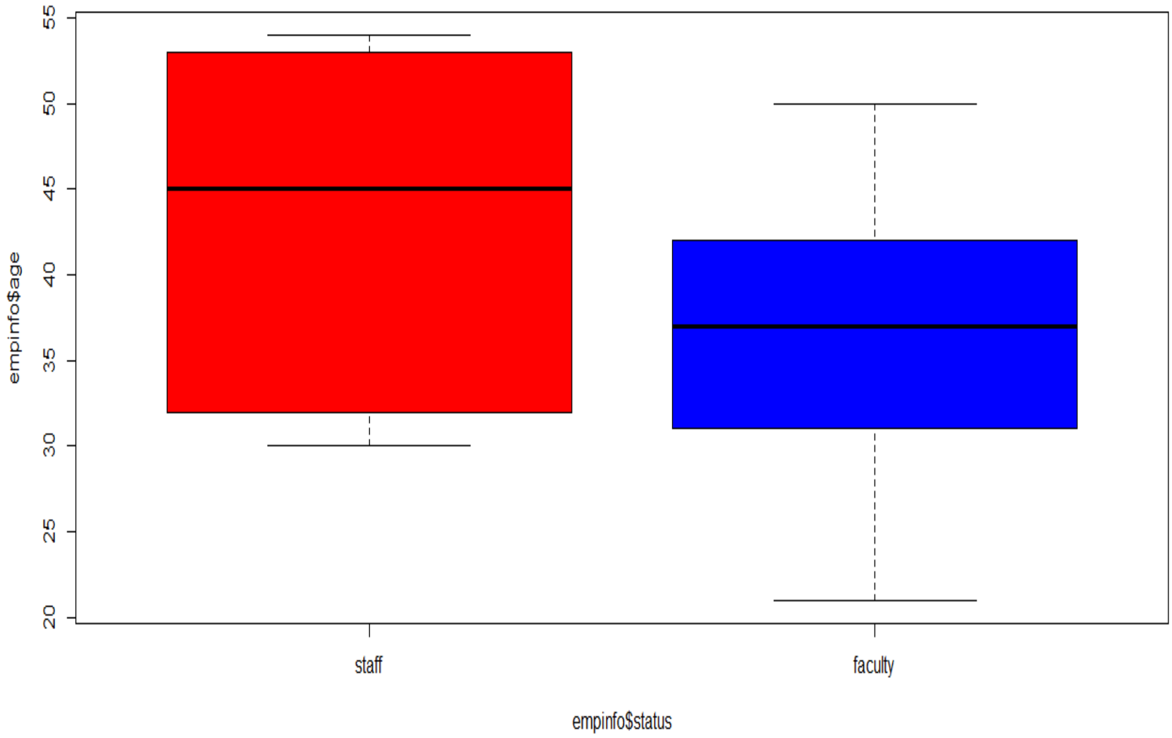
```
> summary(genderf) # Summary statiscs of female employ
     empid            age           gender      status
 Min.   : 2.0   Min.   :21   male  :0   staff  :1
 1st Qu.: 3.0   1st Qu.:37   female:5   faculty:4
 Median : 5.0   Median :42
 Mean   : 5.6   Mean   :39
 3rd Qu.: 8.0   3rd Qu.:45
 Max.   :10.0   Max.   :50
> summary(empinfo$age) # Summary statiscs of age
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  21.00   31.25   39.50   39.50   48.75   54.00
```

```
> table1

male female
   5      5
> table2

staff faculty
    5       5
> table3

        staff faculty
male        4       1
female      1       4
```




male
female


0
1
2
3
4
5
staff
faculty
male
female



empinfo$age
55
50
45
40
35
30
25
20
staff
faculty
empinfo$status

(18th August 2020)

Problem1:Twenty students , graduates and undergraduates, were enrolled in a statistics course. Their ages were 18,19,19,19,19,20,20,20,20,20,21,21,21,21,22,23,24,27,30,36.

a) Find Mean and Median of all students

b) Find median age of all students under 25 years.

c) Find modal age of all student

```
x = c(18,19,19,19,19,20,20,20,20,20,21,21,21,21,22,23,24,27,30,36)
cat('The mean of all the students   : ',mean(x))
cat('The median of all the students : ',median(x))

under_25 = x[x<25]
cat('The median age of all students under 25 : ',median(under_25))

xr = table(x)
print(xr)
mode = which(xr==max(xr))
cat('The modal age of all the students : ',mode)
```

```
> cat('The mean of all the students   : ',mean(x))
The mean of all the students   :  22
> cat('The median of all the students : ',median(x))
The median of all the students :  20.5
> under_25 = x[x<25]
> cat('The median age of all students under 25 : ',median(under_25))
The median age of all students under 25 :  20
> xr = table(x)
> print(xr)
x
18 19 20 21 22 23 24 27 30 36
 1  4  5  4  1  1  1  1  1  1
> mode = which(xr==max(xr))
> cat('The modal age of all the students : ',mode)
The modal age of all the students :  3
> mode
20
 3
```

Problem 2 : A survey of 25 faculty members is taken in a college to study their vocational mobility.They were asked the question “In addition to your present position ,at how many educational instistutes have served on the faculty?.Following is the frequency distribution of their responses .

X	0	1	2	3
f	8	11	5	1

Find mean and median of the distribution

```
## 2nd problem
x = c(0,1,2,3)
f = c(8,11,5,1)
y = rep(x,f)      # rep(value,number_of_times)
print(y)
mean   = (sum(y)/length(y))
median = median(y)
cat('The mean of the distribution   : ',mean)
cat('The median of the distribution : ',median)
```

```
> x = c(0,1,2,3)
> f = c(8,11,5,1)
> y = rep(x,f)      # rep(value,number_of_times)
> print(y)
 [1] 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 3
> mean   = (sum(y)/length(y))
> median = median(y)
> cat('The mean of the distribution   : ',mean)
The mean of the distribution   :  0.96
> cat('The median of the distribution : ',median)
The median of the distribution :  1
```

Problem 3 : Compute mean ,median , 1st Quartile, 3rd Quartile and mode of for the following frequency Distribution:

Height in Cm	***145-150***	***150-155***	***155-160***	***160-165***	***165-170***	***170-175***	***175-180***	***180-185***
No. of Adult men	***4***	***6***	***28***	***58***	***64***	***30***	***5***	***5***

```r
## 3rd problem  (Continuous frequency distribution)
# Mean
x = seq(147.5,182.5,5)
freq = c(4,6,28,58,64,30,5,5)
mean = sum(x*f)/sum(f)
cat('The mean is : ',mean)

# Median
cf = cumsum(freq)
N  = sum(freq)              # total frequency
median_class = min(which(cf>N/2))
c      = 5                  # width of the median class
m      = cf[median_class-1]     # cummulative frequency of the preceding median class
f      = freq[median_class]     # frequency of the median class
L      = x[median_class]-c/2    # lower limit of the median class
median = L+(((N/2)-m)/f)*c
cat('The median is : ',median)

# 1st quartile
quartile_class = min(which(cf>3*N/4))
c    = 5
m    = cf[quartile_class-1]
f    = freq[quartile_class]
L    = x[quartile_class]-c/2
q11 = L+(((N/4)-m)/f)*c
cat('1st quartile : ',q11)

# 3rd quartile
quartile_class = min(which(cf>3*N/4))
c    = 5
m    = cf[quartile_class-1]
f    = freq[quartile_class]
L    = x[quartile_class]-c/2
q13 = L+(((3*N/4)-m)/f)*c
cat('3rd quartile : ',q13)

# Mode
modal_class = which(freq==max(freq))
f0    = freq[modal_class]     # frequency of the modal class
f1    = freq[modal_class-1]   # frequency of the preceding modal class
f2    = freq[modal_class+1]   # frequency of the succeeding modal class
L     = x[modal_class]-c/2    # lower limit of the modal class
mode = L+((f0-f1)/(2*f0-f1-f2))*c
cat('The mode is : ',mode)
```

```
> x = seq(147.5,182.5,5)
> freq = c(4,6,28,58,64,30,5,5)
> mean = sum(x*f)/sum(f)
> cat('The mean is : ',mean)
The mean is :  1320
> cf = cumsum(freq)
> N  = sum(freq)             # total frequency
> median_class = min(which(cf>N/2))
> c      = 5                 # width of the median class
> m      = cf[median_class-1]     # cummulative frequency of the preceding median class
> f      = freq[median_class]     # frequency of the median class
> L      = x[median_class]-c/2    # lower limit of the median class
> median = L+(((N/2)-m)/f)*c
> cat('The median is : ',median)
The median is :  165.3125
> quartile_class = min(which(cf>3*N/4))
> c   = 5
> m   = cf[quartile_class-1]
> f   = freq[quartile_class]
> L   = x[quartile_class]-c/2
> ql1 = L+(((N/4)-m)/f)*c
> cat('1st quartile : ',ql1)
1st quartile :  161.4062
```

```
> quartile_class = min(which(cf>3*N/4))
> c   = 5
> m   = cf[quartile_class-1]
> f   = freq[quartile_class]
> L   = x[quartile_class]-c/2
> ql3 = L+(((3*N/4)-m)/f)*c
> cat('3rd quartile : ',ql3)
3rd quartile :  169.2188
> modal_class = which(freq==max(freq))
> f0    = freq[modal_class]     # frequency of the modal class
> f1    = freq[modal_class-1]   # frequency of the preceding modal class
> f2    = freq[modal_class+1]   # frequency of the succeeding modal class
> L     = x[modal_class]-c/2    # lower limit of the modal class
> mode  = L+((f0-f1)/(2*f0-f1-f2))*c
> cat('The mode is : ',mode)
The mode is :  165.75
```

```
########### Range #################
a=c(1,2,3,4,5,5,5,6,7,8)
cat('Range of a ',range(a))
cat('Difference between maximum and minimum : ',diff(range(a)))

## Using max() and min()
cat('Range : ',max(a)-min(a))
```

```
> a=c(1,2,3,4,5,5,5,6,7,8)
> cat('Range of a ',range(a))
Range of a  1 8> cat('Difference between maximum and minimum : ',diff(range(a)))
Difference between maximum and minimum :  7>
> ## Using max() and min()
> cat('Range : ',max(a)-min(a))
Range :  7
```

```r
a = c(1,2,3,4,5,5,5,6,7,8)
cat('Interquantile range : ')
print(IQR(a))

cat('Quantile : ')
print(quantile(a))

cat('1st quantile')
print(quantile(a,0.25))

cat('2nd quantile')
print(quantile(a,0.75))
```

```
> a = c(1,2,3,4,5,5,5,6,7,8)
> cat('Interquantile range : ')
Interquantile range : > print(IQR(a))
[1] 2.5
> cat('Quantile : ')
Quantile : > print(quantile(a))
  0%  25%  50%  75% 100% 
1.00 3.25 5.00 5.75 8.00 
> cat('1st quantile')
1st quantile> print(quantile(a,0.25))
 25% 
3.25 
> cat('2nd quantile')
2nd quantile> print(quantile(a,0.75))
 75% 
5.75 
```

Example:

An entomologist studying morphological variation in species of mosquito recorded the following data on body length: 1.2,1.4,1.3,1.6,1.0,1.5,1.7,1.1,1.2,1.3.Compute all the measures of disersion.

```
############ Measures of Dispersion #############
x=c(1.2,1.4,1.3,1.6,1.0,1.5,1.7,1.1,1.2,1.3)
cat('Range : ',range(x))
cat('Difference of Range : ',diff(range(x)))
cat('Variance : ',var(x))
cat('Standard Deviation : ',sd(x))
cat('Quantile ')
print(quantile(x))
cat('Inter quantile Range : ',IQR(x))

### Raw data #############
y = abs(x-mean(x))
cat('Mean Deviation about Mean : ',sum(y)/length(y))

y = abs(x-median(x))
cat('Mean Deviation about Median : ',sum(y)/length(y))
```

```
> x=c(1.2,1.4,1.3,1.6,1.0,1.5,1.7,1.1,1.2,1.3)
> cat('Range : ',range(x))
Range :  1 1.7
> cat('Difference of Range : ',diff(range(x)))
Difference of Range :  0.7
> cat('Variance : ',var(x))
Variance :  0.049
> cat('Standard Deviation : ',sd(x))
Standard Deviation :  0.2213594
> cat('Quantile ')
Quantile > print(quantile(x))
   0%   25%   50%   75%  100% 
1.000 1.200 1.300 1.475 1.700 
> cat('Inter quantile Range : ',IQR(x))
Inter quantile Range :  0.275
> y = abs(x-mean(x))
> cat('Mean Deviation about Mean : ',sum(y)/length(y))
Mean Deviation about Mean :  0.176
> y = abs(x-median(x))
> cat('Mean Deviation about Median : ',sum(y)/length(y))
Mean Deviation about Median :  0.17
> y = abs(x-mode(x))
```

1st September 2020

```r
## Karl Pearson(Correlation coefficient)
x=c(78,36,98,25,75,82,90,62,65,39)
y=c(84,51,91,60,68,62,86,58,53,47)

rxy = var(x,y)/(sqrt(var(x))*sqrt(var(y)))
cat('Karl Pearson coefficient : ',rxy)

## Direction function
cat('Karl Pearson coefficient : ',cor(x,y))
print(cor.test(x,y))
print(cor.test(x,y,method='pearson'))

```

```
> ## Karl Pearson(Correlation coefficient)
> x=c(78,36,98,25,75,82,90,62,65,39)
> y=c(84,51,91,60,68,62,86,58,53,47)
>
> rxy = var(x,y)/(sqrt(var(x))*sqrt(var(y)))
> cat('Karl Pearson coefficient : ',rxy)
Karl Pearson coefficient :  0.7804101
> ## Direction function
> cat('Karl Pearson coefficient : ',cor(x,y))
Karl Pearson coefficient :  0.7804101
> print(cor.test(x,y))

        Pearson's product-moment correlation

data:  x and y
t = 3.5302, df = 8, p-value = 0.00773
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2964488 0.9454659
sample estimates:
      cor
0.7804101
```

```
> print(cor.test(x,y,method='pearson'))

        Pearson's product-moment correlation

data:  x and y
t = 3.5302, df = 8, p-value = 0.00773
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2964488 0.9454659
sample estimates:
      cor
0.7804101
```

```
################# Problem-2 ##########################
x1=c(10,14,18,22,26,30)
y1=c(18,12,24,6,30,36)
print(cor(x1,y1))

################# Problem-3 ###########################
x2=c(10,15,12,17,13,16,24,14,22)
y2=c(30,42,45,46,33,34,40,35,39)
print(cor(x2,y2))
print(cor.test(x2,y2,method='pearson'))   # KarlPearson's solution
print(cor.test(x2,y2,method="spearman"))  # Spearan's solution

```

```
> x1=c(10,14,18,22,26,30)
> y1=c(18,12,24,6,30,36)
> print(cor(x1,y1))
[1] 0.6
> x2=c(10,15,12,17,13,16,24,14,22)
> y2=c(30,42,45,46,33,34,40,35,39)
> print(cor(x2,y2))
[1] 0.3307218
> print(cor.test(x2,y2,method='pearson'))   # KarlPearson's solution

        Pearson's product-moment correlation

data:  x2 and y2
t = 0.92718, df = 7, p-value = 0.3847
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.4272383  0.8156865
sample estimates:
      cor
0.3307218


> print(cor.test(x2,y2,method="spearman"))  # Spearan's solution

        Spearman's rank correlation rho

data:  x2 and y2
S = 72, p-value = 0.2912
alternative hypothesis: true rho is not equal to 0
sample estimates:
rho
0.4
```

```r
################ Problem-4 ##########################
x3=c(68,64,75,50,64,80,75,40,55,64)
y3=c(62,58,68,45,81,60,68,48,50,70)
print(cor(x3,y3))
print(cor.test(x3,y3,method="pearson"))
print(cor.test(x3,y3,method="spearman",exact=F)) # without rectification

```

```
> x3=c(68,64,75,50,64,80,75,40,55,64)
> y3=c(62,58,68,45,81,60,68,48,50,70)
> print(cor(x3,y3))
[1] 0.6240869
> print(cor.test(x3,y3,method="pearson"))

        Pearson's product-moment correlation

data:  x3 and y3
t = 2.2591, df = 8, p-value = 0.0538
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.009124948  0.900047265
sample estimates:
      cor
0.6240869

> print(cor.test(x3,y3,method="spearman",exact=F)) # without rectification

        Spearman's rank correlation rho

data:  x3 and y3
S = 73.326, p-value = 0.09542
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.5555979
```

```r
################ Problem-5 ############################
## Linear Regression ####
a=1:9    # a=seq(1,8)
b=c(9,8,10,12,11,13,14,16,15)
re1 = lm(a~b)   # a on b
print(re1)

re2 = lm(b~a) # b on a
print(re2)

print(summary(re1))
print(summary(re2))

```

```
> ################# Problem-5 ############################
> ## Linear Regression ####
> a=1:9   # a=seq(1,8)
> b=c(9,8,10,12,11,13,14,16,15)
> re1 = lm(a~b)  # a on b
> print(re1)

Call:
lm(formula = a ~ b)

Coefficients:
(Intercept)            b
      -6.40         0.95

> re2 = lm(b~a) # b on a
> print(re2)

Call:
lm(formula = b ~ a)

Coefficients:
(Intercept)            a
       7.25         0.95

>
> print(summary(re1))

Call:
lm(formula = a ~ b)

Residuals:
   Min     1Q Median     3Q    Max 
 -1.15  -0.80   0.05   0.80   1.15 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   -6.400      1.449  -4.418  0.00309 ** 
b              0.950      0.118   8.050 8.76e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 0.9142 on 7 degrees of freedom
Multiple R-squared:  0.9025,	Adjusted R-squared:  0.8886 
F-statistic: 64.79 on 1 and 7 DF,  p-value: 8.763e-05

> print(summary(re2))

Call:
lm(formula = b ~ a)

Residuals:
   Min     1Q Median     3Q    Max 
 -1.15  -0.80   0.05   0.80   1.15 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   7.2500     0.6641   10.92 1.20e-05 ***
a             0.9500     0.1180    8.05 8.76e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 0.9142 on 7 degrees of freedom
Multiple R-squared:  0.9025,	Adjusted R-squared:  0.8886 
F-statistic: 64.79 on 1 and 7 DF,  p-value: 8.763e-05
```

```r
###### Simple Linear regression
e=c(65,66,67,67,68,69,70,72)
f=c(67,68,65,68,72,72,69,71)
print(lm(e~f)) # e on f
print(lm(f~e)) # f on e

```

```
> e=c(65,66,67,67,68,69,70,72)
> f=c(67,68,65,68,72,72,69,71)
> print(lm(e~f)) # e on f

Call:
lm(formula = e ~ f)

Coefficients:
(Intercept)            f  
    30.3636       0.5455  

> print(lm(f~e)) # f on e

Call:
lm(formula = f ~ e)

Coefficients:
(Intercept)            e  
    23.6667       0.6667  
```

```
###### Multiple Regression
f=c(20,23,25,27,21,29,22,24,27,35)
g=c(8,13,8,18,23,16,10,12,14,20)
h=c(28,23,38,16,20,28,23,30,26,32)
print(lm(f~(g+h)))

```

```
> f=c(20,23,25,27,21,29,22,24,27,35)
> g=c(8,13,8,18,23,16,10,12,14,20)
> h=c(28,23,38,16,20,28,23,30,26,32)
> print(lm(f~(g+h)))

Call:
lm(formula = f ~ (g + h))

Coefficients:
(Intercept)            g            h
     5.1483       0.6190       0.4304
```

8th September 2020

Binominal and Poisson Distribution

› Problem 1:

1. Suppose X ~ Bin(10, 0.4), what is P(X = 7)?

```
> ## Binomial Distribution
> #dbinom(x,size,probability)
> # This function gives the probability density function at each point
> dbinom(7,size=10,prob=0.4)
[1] 0.04246733
```

› Problem 2:

2. Suppose Y ~ Bin(8, 0.15), what is P(Y < 3)?

```
> # pbinom(x,size,probability)
> # This function gives the cumulative probability of an event.
ability.
> pbinom(3,8,0.15)
[1] 0.9786475
```

› Problem 3:

3. Suppose W ~ Bin(50, 0.12), what is P(W > 2)?

```
> #P(X>2)=1-P(X<=2)
> cat('P(X>2) : ',1-pbinom(2,size=50,prob=0.12))
P(X>2) :  0.9487358
> 
```

Problem 4:

- In a box of switches it is known 10% of the switches are faulty. A technician is wiring 30 circuits, each of which needs one switch. What is the probability that (a) all 30 work, (b) at most 2 of the circuits do not work?

c)at-most 2 of the circuit works : P(X<=2)

Probability of failure(i.e switches are not working) → 10% (0.1)

Probability of success(i.e switches are working) → 90% (0.9)

```
## Problem-4
cat('All 30 works : ',dbinom(30,size=30,prob=0.9))
cat('At most 2 circuit do not work P(X>=27) : ',1-pbinom(27,size=30,prob=0.9))
cat('At most 2 circuit do not work P(X>=27) : ',pbinom(2,size=30,prob=0.1))
cat('At most 2 circuit works P(X<=2) : ',pbinom(2,size=30,prob=0.9))
```

```
> ## Problem-4
> cat('All 30 works : ',dbinom(30,size=30,prob=0.9))
All 30 works :  0.04239116
> cat('At most 2 circuit do not work P(X>=27) : ',1-pbinom(27,size=30,prob=0.9))
At most 2 circuit do not work P(X>=27) :  0.4113512
> cat('At most 2 circuit do not work P(X<=2)  : ',pbinom(2,size=30,prob=0.1))
At most 2 circuit do not work P(X<=2)  :  0.4113512
> cat('At most 2 circuit works P(X<=2) : ',pbinom(2,size=30,prob=0.9))
At most 2 circuit works P(X<=2) :  3.5506e-26
```

Problem 5:

- If 10% of the Screws produced by an automatic machine are defective, find the probability that out of 20 screws selected at random, there are
- (i) Exactly 2 defective (ii) At least 2 defectives
- (iii) Between 1 and 3 defectives (inclusive)

Probability of success (i.e defective screws) → 0.1

```
> ## Problem-5
> cat('Exactly 2 defective P(X=2)     : ',dbinom(2,size=20,prob=0.1))
Exactly 2 defective P(X=2)     :  0.2851798
> cat('Atleast 2 defectives P(X>=2)  : ',1-dbinom(1,size=20,prob=0.1))
Atleast 2 defectives P(X>=2)  :  0.7298297
> cat('Between 1 and 3 are defective P(1>=X<=3) : ',sum(dbinom(1:3,size=20,prob=0.1)))
Between 1 and 3 are defective P(1>=X<=3) :  0.74547
```

1) In a large consignment of electric bulbs 10% are defective. A random sample of 20 is taken for inspection. Find the probability that

(i) All are good bulbs,

(ii) At most there are 3 defective bulbs,

(iii) Exactly there are three defective bulbs.

(Solution: (i) 0.1216 (ii) 0.8666 (iii) 0.19)

Probability of defective → 10% (i.e 0.1)

Probability of good bulbs → 90% (i.e 0.9)

```
> ## Problem-6
> cat('All are good bulbs P(X=20) : ',dbinom(20,size=20,prob=0.9))
All are good bulbs P(X=20) :  0.1215767
> cat('Atmost 3 bulbs are defective P(X<=3) : ',pbinom(3,size=20,prob=0.1))
Atmost 3 bulbs are defective P(X<=3) :  0.8670467
> cat('Exactly 3 bulbs are defective P(X=3) : ',dbinom(3,size=20,prob=0.1))
Exactly 3 bulbs are defective P(X=3) :  0.1901199
```

For a random variable X with a binomial (20,1/2) distribution, find the following probabilities.

(i). Find Pr(X < 8)

(ii). Find Pr(X > 12)

(iii). Find Pr(8 <= X <= 1

```
> ### Problem-7
> cat('P(X<8) : ',pbinom(7,size=20,prob=0.5))
P(X<8) :  0.131588
> cat('P(X>12) = 1-P(X<=12) : ',1-pbinom(12,size=20,prob=0.5))
P(X>12) = 1-P(X<=12) :  0.131588
> cat('P(8<=X<=1) : ',sum(dbinom(1:8,size=20,prob=0.5)))
P(8<=X<=1) :  0.2517214
```

Poisson Distribution

Consider a computer system with Poisson job-arrival stream at an ` average of 2 per minute. Determine the probability that in any one-minute interval there will be

i. 0 jobs;

ii. exactly 2 jobs

iii. at most 3 arrivals.

```
> cat('0-jobs P(X=0) : ',dpois(0,lambda = 2))
0-jobs P(X=0) :  0.1353353
> cat('Exaclty 2 jobs P(X=2) ',dpois(2,lambda=2))
Exaclty 2 jobs P(X=2)  0.2706706
> cat('At-most 3 arrivals P(X<=3) ',ppois(3,lambda=2))
At-most 3 arrivals P(X<=3)  0.8571235
```

Let X be the number of heads in 10 tosses of a fair coin.

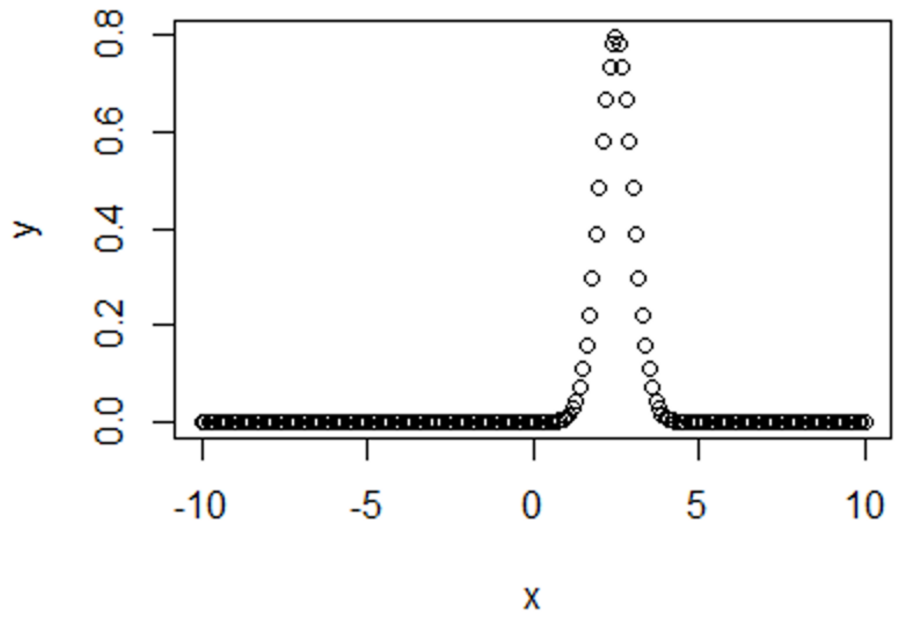
1. Find the probability of getting at least 5 heads (that is, 5 or more).
2. Find the probability of getting exactly 5 heads.
3. Find the probability of getting between 4 and 6 heads, inclusive

```
> ### Problem-2
> cat('Probability of getting at-least 5 heads P(X>=5)=1-P(X<5) : ',1-ppois(4,lambda=0.5))
Probability of getting at-least 5 heads P(X>=5)=1-P(X<5) :  0.0001721156
> cat('Probability of getting exactly 5 heads P(X=5) : ',dpois(5,lambda=0.5))
Probability of getting exactly 5 heads P(X=5) :  0.0001579507
> cat('Probability of getting between 4 and 6 heads  : ',sum(dpois(4:6,lambda=0.5)))
Probability of getting between 4 and 6 heads  :  0.00175062
```

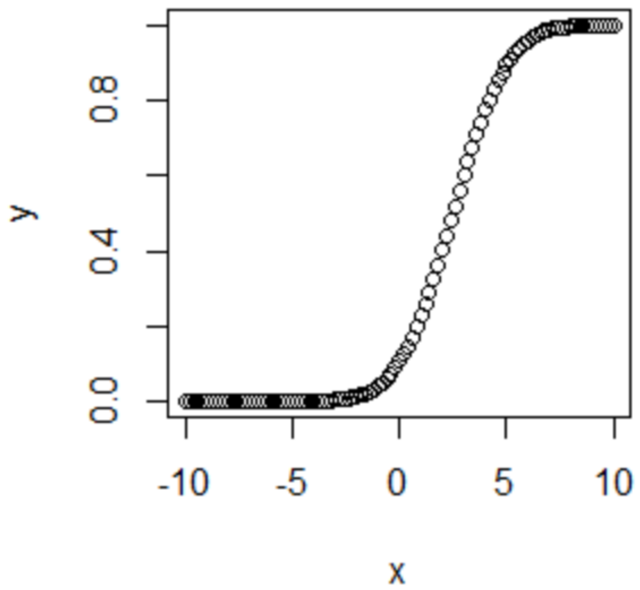
22th September

Normal Distribution

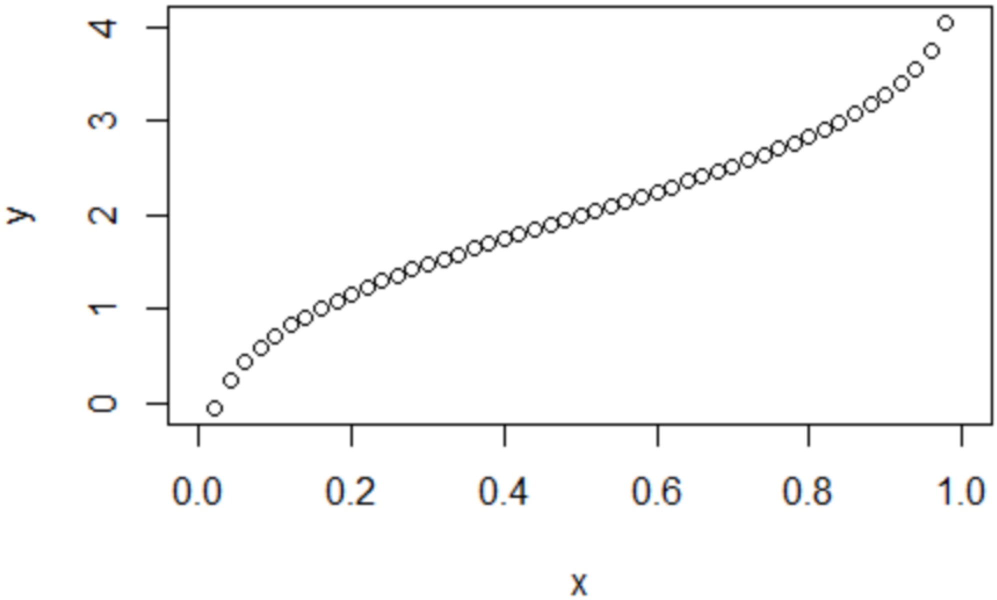
```r
# Sequence of numbers between -10 and 10 incrementing by 0.1
x=seq(-10,10,by=.1)
y=dnorm(x,mean=2.5,sd=0.5) # mean=2.5 and standard deviation=0.5
plot(x,y)
```



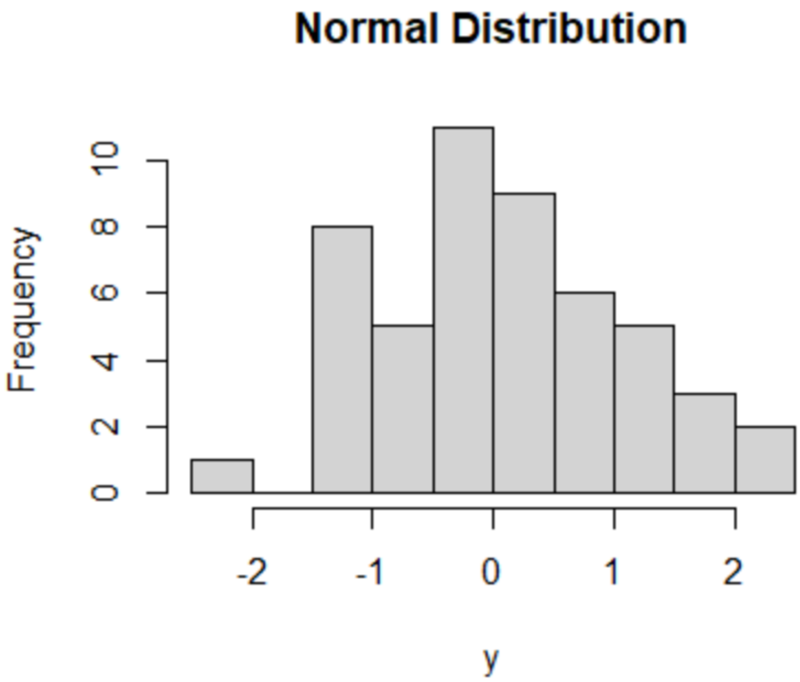

```r
x=seq(-10,10,by=0.2)
y=pnorm(x,mean=2.5,sd=2)
plot(x,y) # pnorm graph
```



```
x=seq(0,1,by=0.02)
y=qnorm(x,mean=2,sd=1)
plot(x,y) # qnorm graph
```



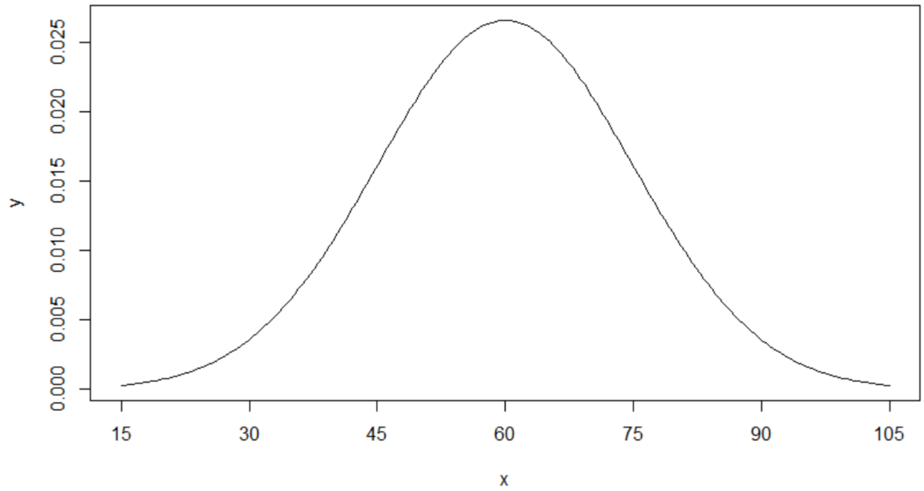

```
y=rnorm(50)
hist(y,main='Normal Distribution')
```



Program-1

Draw a normal distribution with the mean=60 and standard deviation=15.

```r
## Program-1 ##
x = seq(15,105,length=200)
y = dnorm(x,mean=60,sd=15)
plot(x,y,type="l",xaxt="n")
axis(1,at=c(15,30,45,60,75,90,105))
```



1)

Suppose X is normal with mean 527 and standard deviation 105. Compute P(X >=310)

```
> ## Problem-1
> cat('P(X>=310) = 1-P(X<310) : ',1-pnorm(309,mean=527,sd=105))
P(X>=310) = 1-P(X<310) :  0.9810618
```

2)

Find P(80 pts< x<95 pts.)

```
> ## Problem-2
> cat('P(80<x<95) : ',sum(dnorm(81:94,mean=527,sd=105)))
P(80<x<95) :  8.446068e-06
```

3)

The weekly wages of 1000 workmen are normally distributed around a mean of Rs. 70 with S.D of Rs 5.Estimate the number of workers whose weekly wages will be

(i) Between Rs 69 and Rs 72

(ii) Less than Rs 69

(iii) More than Rs 72

```
> ## Problem-3
> cat('P(69<=X<=72) = ',(pnorm(72,mean=70,sd=5)-pnorm(69,mean=70,sd=5))*1000)
P(69<=X<=72) =  234.6815
> cat('P(x<=69)      = ',(pnorm(69,mean=70,sd=5))*1000)
P(x<=69)      =  420.7403
> cat('P(X>72)=1-P(X<=72) =',(1-pnorm(72,mean=70,sd=5))*1000)
P(X>72)=1-P(X<=72) = 344.5783
```

4)

In a test on 2000 Electric bulbs ,it was found that the life of particular make, was normally distributed with an average life of 2040 hours and S.D of 60 hours. Estimate the number of bulbs likely to burn for

(i) More than 2150 hours

(ii) Less than 1950 hours

(iii) More than 1920 hours but less than 2160 hours

(iv) More than 2150 hours

```
> ## Problem-4
> cat('P(X>2150)=1-P(X<=2150) : ',(1-pnorm(2150,mean=2040,sd=60))*2000)
P(X>2150)=1-P(X<=2150) :  66.75302
> cat('P(X<=1950)         : ',pnorm(1950,mean=2040,sd=60)*2000)
P(X<=1950)         :  133.6144
> cat('P(1920<=X<=2160) : ',(pnorm(2160,mean=2040,sd=60)-pnorm(1920,mean=2040,sd=60))*2000)
P(1920<=X<=2160) :  1908.999
```

6th October

```r
### Sum-1
xbar=9900
mu=10000
sig=120
n=30
z=(xbar-mu)/(sig/sqrt(n))
print(z)

z_alpha1 = qnorm(0.05)  # left tailed
print(z_alpha1)

z_alpha = qnorm(1-0.05,lower.tail = F)
print(z_alpha)
print(pnorm(z))

z_alphar = qnorm(1-0.05)          # right tailed
print(z_alphar)
print(pnorm(z,lower.tail = F))
#######################################

```

```
> xbar=9900
> mu=10000
> sig=120
> n=30
> z=(xbar-mu)/(sig/sqrt(n))
> print(z)
[1] -4.564355
> z_alpha1 = qnorm(0.05)  # left tailed
> print(z_alpha1)
[1] -1.644854
> z_alpha = qnorm(1-0.05,lower.tail = F)
> print(z_alpha)
[1] -1.644854
> print(pnorm(z))
[1] 2.505166e-06
> z_alphar = qnorm(1-0.05)          # right tailed
> print(z_alphar)
[1] 1.644854
> print(pnorm(z,lower.tail = F))
[1] 0.9999975
> 
```

```r
### Sum-2
xbar=14.6
mu=15.4
sig=2.5
n=35
z=(xbar-mu)/(sig/sqrt(n))
print("Z : ")
print(z)

print("Z alpha : ")
print(qnorm(1-0.05/2))    # /2 two tailed test
print(2*pnorm(z))

```

```
> xbar=14.6
> mu=15.4
> sig=2.5
> n=35
> print("Z alpha : ")
[1] "Z alpha : "
> print(qnorm(1-0.05/2))    # /2 two tailed test
[1] 1.959964
> print(2*pnorm(z))
[1] 5.010332e-06
```

```r
###########################
# Sum-3
P=0.12
Q=1-P
p=30/214
n=214
z=(p-P)/(sqrt((P*Q/n)))
print(qnorm(1-0.05))             # right tailed
print(pnorm(z,lower.tail=F))
print(prop.test(30,214,0.12,alt = "greater",correct=FALSE))
#################
```

```
> P=0.12
> Q=1-P
> p=30/214
> n=214
> z=(p-P)/(sqrt((P*Q/n)))
> print(qnorm(1-0.05))
[1] 1.644854
> print(pnorm(z,lower.tail=F))
[1] 0.1817408
> print(prop.test(30,214,0.12,alt = "greater",correct=FALSE))

        1-sample proportions test without continuity correction

data:  30 out of 214, null probability 0.12
X-squared = 0.82583, df = 1, p-value = 0.1817
alternative hypothesis: true p is greater than 0.12
95 percent confidence interval:
 0.1056274 1.0000000
sample estimates:
        p
0.1401869
```

13th October

```
> x=c(5,6,8,1,12,4,3,9,6,10)
> y=c(2,3,6,8,10,1,2,8)
> t.test(x,y)  # t-test

        Welch Two Sample t-test

data:  x and y
t = 0.8679, df = 15.041, p-value = 0.3991
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.0374  4.8374
sample estimates:
mean of x mean of y 
      6.4       5.0 

> t.test(x,y,alt="greater") # right tail

        Welch Two Sample t-test

data:  x and y
t = 0.8679, df = 15.041, p-value = 0.1995
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 -1.42732       Inf
sample estimates:
mean of x mean of y 
      6.4       5.0 

> t.test(x,y,alt="less")    # left tail

        Welch Two Sample t-test

data:  x and y
t = 0.8679, df = 15.041, p-value = 0.8005
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
    -Inf 4.22732
sample estimates:
mean of x mean of y 
      6.4       5.0 

> x = c(12.9,13.5,12.8,15.6,17.2,19.2,12.6,15.3,14.4,11.3)
> y = c(12.0,12.2,11.2,13.0,15.0,15.8,12.2,13.4,12.9,11.0)
> t.test(x,y)

        Welch Two Sample t-test

data:  x and y
t = 1.7889, df = 15.367, p-value = 0.09335
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.3042834  3.5242834
sample estimates:
mean of x mean of y 
    14.48     12.87 
```

```
> t.test(x,y,alt='less')

        Welch Two Sample t-test

data:  x and y
t = 1.7889, df = 15.367, p-value = 0.9533
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
     -Inf 3.185243
sample estimates:
mean of x mean of y
    14.48     12.87
```

```
> plants = c(20,10,7,4)
> chisq.test(plants,p=c(9/16,3/16,3/16,1/16))

        Chi-squared test for given probabilities

data:  plants
X-squared = 1.9702, df = 3, p-value = 0.5786

Warning message:
In chisq.test(plants, p = c(9/16, 3/16, 3/16, 1/16)) :
  Chi-squared approximation may be incorrect
```

```
> # Good ness of fit (chi-square)
> x = c(5,4,3,2,1,0)
> n=5
> N=320
> P=0.5
> obf=c(14,56,110,88,40,12)
> exf=dbinom(x,n,P)*320    # expected frequency
> sum(obf)
[1] 320
> sum(exf)
[1] 320
> chisq=sum((obf-exf)*2/exf)
> qchisq(0.95,5)
[1] 11.0705
```

```
> x = c(0,1,2,3,4,5,6)
> f = c(275,72,30,7,5,2,1)
> Efx = sum(x*obf)
Warning message:
In x * obf :
  longer object length is not a multiple of shorter object length
> Ef = sum(obf)
> lambda = Efx/Ef
> expois = dpois(x,lambda)*sum(f)
> f1 = round(expois)
> sum(f)
[1] 392
> sum(f1)
[1] 386
> f-f1
[1] 247  -2 -68 -79 -52 -28 -12
> E = c(242,117,28,5)  # Omitted unwanted data and added
> O = c(275,72,30,15)
> chisq = sum((O-E)^2/E)
> df=2
> qchisq(0.95,df)
[1] 5.991465
> print('H0 is accepted')
[1] "H0 is accepted"
```

```
> x = c(0,1,2,3,4,5,6)
> f = c(109,53,131,12,35,21,12)
> Ef  = sum(f)
> Efx = sum(f*x)
> lambda = Efx/Ef
> expois = dpois(x,lambda)*sum(f)
> f1      = round(expois)
> print(sum(f))
[1] 373
> print(sum(f1))
[1] 373
> print(f-f1)
[1]  47 -58  31 -48   8  11   9
> O=c(109,53,131,12,35,33)
> E=c(62,111,100,60,27,13)
> cat('chi^square : ',chisq = sum((O-E)^2/E))
chi^square :  147.0849> df=5
> cat('chi^square_alpha : ',qchisq(0.95,df))
chi^square_alpha :  11.0705> print('H0 is rejected')
[1] "H0 is rejected"
```

20th October

```r
### 1-way classification
df=read.csv("F:\\2) Second Year 2020-2021\\Fall semester 2020-2021
r=c(t(as.matrix(df)))
f=c("Tech-A","Tech-B","Tech-C","Tech-D")
k=4  # number of columns
n=5  # number of rows
te = gl(k,1,k*n,factor(f))
cr = aov(r~te)
summary(cr)
```

```
> ### 1-way classification
> df=read.csv("F:\\2) Second Year 2020-2021\\Fall semester 2020-2021\
 Practice\\anova.csv")
> r=c(t(as.matrix(df)))
> f=c("Tech-A","Tech-B","Tech-C","Tech-D")
> k=4  # number of columns
> n=5  # number of rows
> te = gl(k,1,k*n,factor(f))
> cr = aov(r~te)
> summary(cr)
            Df Sum Sq Mean Sq F value Pr(>F)
te           3  12.95   4.317    0.68  0.577
Residuals   16 101.60   6.350
>
```

```r
### 2-way classification
df1=read.csv("F:\\2) Second Year 2020-2021\\Fall semester 2020-2021\
r1=c(t(as.matrix(df1)))
f=c("A","B","C","D","E","F")
blocks = gl(4,6)
treatments = gl(6,1,24)
rbfit = aov(r1~blocks+treatments)
summary(rbfit)
```

```
> df1=read.csv("F:\\2) Second Year 2020-2021\\Fall semester 2020-2021\
 Practice\\anova_2way.csv")
> r1=c(t(as.matrix(df1)))
> f=c("A","B","C","D","E","F")
> blocks = gl(4,6)
> treatments = gl(6,1,24)
> rbfit = aov(r1~blocks+treatments)
> summary(rbfit)
            Df Sum Sq Mean Sq F value   Pr(>F)
blocks       3  219.4   73.14   4.778   0.0157 *
treatments   5  901.2  180.24  11.773 9.28e-05 ***
Residuals   15  229.6   15.31
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```