Lab Manual

For

Probability and Statistics for Engineers (BS110)

University School of Automation & Robotics (USAR)

GGSIP University, East Delhi Campus

Surajmal Vihar, Delhi-110092

Chapter 1

Introduction to R-Software

Mr. Prashant Shah, Associate Professor and Head, Department of Statistics,
K. J. Somaiya College of Science and Commerce, Vidyavihar, Mumbai.

1.1 R as a programming language

While R is perhaps best known as a statistical tool for analyzing data or for making graphs, it is also really useful as a simple programming language and compiler. In R, a program is just any group of commands that you wish to run as a set, to achieve some output.

1.1.1 Using Text Editors and ".R" Files in R

By using a text editor, we can write whole groups of commands and have the computer run them separately or all together. Further, text editors allow you to save your program for later use.

There are three different types of windows that are used by R: console, graphics, and text editor windows. The window where you enter line commands is the R Console. When you used the "plot" command, it opened a new window, which is the graphics window. Text editor windows are just simple text editors that are smart enough to interact with R.

On a PC, go to "File" and open "New script". To execute commands, either highlights the command(s) or put the cursor anywhere on that line and push the button in upper corner of the main R window for "Run line or selection."

Creating a new document (or script) opens a simple text editor in R. You can then enter multiple lines of commands that are not executed until you are ready. And, instead of executing commands one by one, you can execute them all at once or any set of them together. You can also save the file (usually as a "____.R" file) and rerun these commands at a later time.

1.2 What is Statistics?

The subject of statistics deals with

- Collection of data.
- Presentation or organization of data.
- Analysis of data.
- Interpretation of, results of, analysis of data.

1.3 What is Data?

A set of numerical or other measured values

For Eg.
1. Salaries of employees.
2. Export (Rs. in crore) of a company during 2010 to 2015.
3. Daily credit/debit transactions in bank.
4. Carbon dioxide content in the air, in different regions during different seasons.
5. Patient's disease history in hospitals.

To analyse voluminous data, a number of statistical software are available such as

- R-software
- SAS **(Statistical Analysis System)**
- SPSS **(Statistical Package for the Social Sciences)**
- Minitab

R is the most comprehensive statistical analysis package available. It incorporates all of the standard statistical tests, models, and analyses, as well as providing a comprehensive language for managing and manipulating data. R is free and open source software, allowing anyone to use and, importantly, to modify it.

1.4 R commands, case sensitivity

Technically R is an expression language with a very simple syntax. **It is case sensitive**, so A and a are different symbols and would refer to different variables.

Elementary commands consist of either expressions or assignments. If an expression is given as a command, it is evaluated, printed (unless specifically made invisible), and the value is lost.

An assignment also evaluates an expression and passes the value to a variable but the result is not automatically printed.

Commands are separated either by a semi-colon (';'), or by a newline. Elementary commands can be grouped together into one compound expression by braces ('{' and '}').

Comments can be put almost anywhere, starting with a hashmark ('#'), everything to the end of the line is a comment.

If a command is not complete at the end of a line, R will give a different prompt, by default + on second and subsequent lines and continue to read input until the command is syntactically complete.

1.5 R-Commands to input data

a) Assignment Statement

- = or <-

b) Creating vectors

- c()
- scan()

c) Generating sequences

- :
- seq()
- seq(from = a, to = b, by = c)
- seq(length=d, from = a, by = c)

d) Replicating objects or elements

- rep()

1.6 Simple manipulations; numbers and vectors

1.6.1 Assignment:

An assignment means naming a value, so that it can be used later. Assignment has general form

Variable = expression or value (= is an assignment operator)

```
> x = 2 + 3   # x is assigned value 5
> x
[1] 5
> x + 2
[1] 7
> x = x * 3
> x
[1] 15
> x = 2 + 3; y = -4; z = x * y  # Commands are separated by a semi-colon (‘;’)
> x; y; z
[1] 5
[1] -4
[1] -20
> x = 2 + 3; y = -4; z = x * y; x; y; z    # Commands are separated by a semi-
                                           colon (‘;’)
[1] 5
[1] -4
[1] -20
```

1.6.2 Vectors

R operates on named data structures. The simplest such structure is the **numeric vector**, which is a single entity consisting of an ordered collection of numbers. To set up a vector named x, say, consisting of five numbers, namely 10.4, 5.6, 3.1, 6.4 and 21.7, use the R command

```
> x <- c(10.4, 5.6, 3.1, 6.4, 21.7)
```

This is an assignment statement using the function c() which in this context can take an arbitrary number of vector arguments (c stands for “combine.”). The idea is that a list of numbers is stored under a given name, and the name is used to refer to the data. The numbers within the c command are separated by commas. A list is specified with the c command, and assignment is specified with the “<-” symbols. Notice that the assignment operator (‘<-’), which consists of the two characters ‘<’ (“less than”) and ‘-’ (“minus”) occurring strictly side-by-side and it ‘points’ to the object receiving the value of the expression. A number occurring by itself in an expression is taken as a vector of length one.

If an expression is used as a complete command, the value is printed and lost. So now if we were to use the command

```
> 1/x
```

the reciprocals of the five values would be printed at the terminal (and the value of x, of course, unchanged).

The further assignment

```
> b <- c(x, 0, x)
```

would create a vector b with 11 entries consisting of two copies of x with a zero in the middle place.

To see what numbers is included in x type “x” and press the enter key:

```
> x <- c(10.4, 5.6, 3.1, 6.4, 21.7)
> x
[1] 10.4   5.6   3.1   6.4   21.7
> typeof(x)
[1] "double"
```

1.6.3 Accessing vectors:

Individual elements of a vector can be accessed by using indices.

```
> x <- c(10.4, 5.6, 3.1, 6.4, 21.7)
> x[3]            # third element of vector x is accessed.
[1] 3.1
> x[1]            # first element of vector x is accessed.
[1] 10.4
> x[2 : 4]        # elements from second to fourth of vector x are accessed.
[1] 5.6   3.1   6.4
> x[c(2,5)]       # elements having indices 2 and 5 are accessed.
[1] 5.6   21.7
> length(x)       # displays number of elements in vector x.
[1] 5
> x[3 : length(x)]            # elements having indices 3 to 5 of vector x
are accessed.
[1] 3.1   6.4   21.7
```

```
> x[4 : 2]    # elements from fourth to second reversely of vector x are
              accessed.
[1] 6.4   3.1   5.6
> x[0]
numeric(0)
> x[6]
[1] NA
> x[x > 6]        # elements of vector x having value > 6 are accessed.
[1] 10.4   6.4   21.7
> x[x < 6]        # elements of vector x having value < 6 are accessed.
[1] 5.6   3.1
```

Subset command can also be used with vectors.

```
> q = subset(x, x > 6)
> q
[1] 10.4  6.4 21.7
> p = subset(x, x < 6)
> p
[1] 5.6 3.1
> which (x < 6)  # displays index of elements of vector x whose value is < 6.
[1] 2 3
> x[-1]           # elements except first are accessed.
[1]  5.6  3.1  6.4 21.7
> x[c(-2,-5)]    # elements except second and fifth are accessed or x[-c(2,5)]
[1] 10.4  3.1  6.4
> x[-2 : -4]      # elements except second to fourth are accessed.
[1] 10.4 21.7
> x < 6
[1] FALSE  TRUE  TRUE FALSE FALSE
```

Notice that the first entry is referred to as the number 1 entry and the zero entry can be used to indicate how the computer will treat the data.

```
> 1/x
[1] 0.09615385 0.17857143 0.32258065 0.15625000 0.04608295
> x
[1] 10.4  5.6  3.1  6.4 21.7
> b <- c(x, 0, x)
> b
 [1] 10.4  5.6  3.1  6.4 21.7  0.0 10.4  5.6  3.1  6.4 21.7
```

You can store strings using both single and double quotes.

```
> t <- c("somaiya", "mumbai", 'new delhi')
> t
[1] "somaiya"   "mumbai"    "new delhi"
> typeof(t)
[1] "character"
```

1.6.4 Alternative way to create data vectors

Vectors can be created and data can be entered alternatively by using scan function.

```
> x = scan()
1: 3 -5 7
4: 9 0 6.7
```

```
7: -2
8:
Read 7 items
> x
[1]  3.0 -5.0  7.0  9.0  0.0  6.7 -2.0
> y = scan()
1: 2 5 8 4 -2
6: 9 5
8:
Read 7 items
> y
[1]  2  5  8  4 -2  9  5
```

scan() function has many other arguments such as ***what, nmax*** etc

- ***what***: This argument indicates types of data to be accepted, by default it is numeric. For character data type set what = “character”
- ***nmax***: This argument indicates maximum number of elements to be accepted.

```
> t = scan(what = "character")
1: "somaiya" "vidyavihar"
3:
Read 2 items
> t
[1] "somaiya"    "vidyavihar"
> x = scan(nmax = 4)
1: 5 -8 3 9 2 -11 6
Read 4 items
> x
[1]  5 -8  3  9
```

1.6.4 Vector arithmetic

Vectors can be used in arithmetic expressions, in which case the operations are performed element by element.

The elementary arithmetic operators are the usual +, -, *, / and ^ for raising to a power. In addition, several mathematical and statistical functions are also available in R for arithmetic operations. For eg.: log, log10, sort, min, max, range, length, exp, sin, cos, tan, sqrt, and so on, all have their usual meaning.

Vectors are mathematical objects. Standard arithmetic functions and operators apply to vectors on element wise basis.

While applying simple arithmetic functions and operators to vectors proper care should be taken. If the operands are of different lengths then shorter of the two is extended by repetition. However, if the length of the longer is not multiple of length of shorter then warning message is displayed.

```
> c(1,5,2,3) + c(1,3)
[1] 2 8 3 6
```

```
> c(1,5,2) + c(1,3)
[1] 2 8 3
```

Warning message:

```
In c(1, 5, 2) + c(1, 3) :
longer object length is not a multiple of shorter object length
```

1.7 Generating regular sequences

R has a number of facilities for generating commonly used sequences of numbers. For example 12:20 is the vector c(12, 13, ..., 20). The colon operator has high priority within an expression, so, for example 2*12:20 is the vector c(24, 26, ..., 40). Put n <- 8 and compare the sequences 1:n-1 and 1:(n-1).

```
> 12:20
[1] 12 13 14 15 16 17 18 19 20
> p <- 12:20
> p
[1] 12 13 14 15 16 17 18 19 20
> q <- 3*12:20
> q
[1] 36 39 42 45 48 51 54 57 60
> n = 8
> t <- 5:(n-1)
> t
[1] 5 6 7
> w <- 5:n - 1
> w
[1] 4 5 6 7
```

The construction 20:12 may be used to generate a sequence backwards.

```
> 20:12
[1] 20 19 18 17 16 15 14 13 12
```

The function seq() is a more general facility for generating sequences. It has five arguments, only some of which may be specified in any one call. The first two arguments, if given, specify the beginning and end of the sequence, and if these are the only two arguments given the result is the same as the colon operator. That is **seq(12,20)** is the same vector as **12:20**.

Parameters to seq(), and to many other R functions, can also be given in **named form**, in which case the order in which they appear is irrelevant. The first two parameters may be named **from=**value and **to=**value; thus seq(12,20), seq(from=12, to=20) and seq(to=20, from=12) are all the same as 12:20. The next two parameters to seq() may be named **by=**value and **length=**value, which specify a step size and a length for the sequence respectively. If neither of these is given, the default by=1 is assumed.

For example

```
> seq(-5, 5, by=.2) -> s3
> s3
```

```
[1] -5.0 -4.8 -4.6 -4.4 -4.2 -4.0 -3.8 -3.6 -3.4 -3.2 -3.0 -2.8 -2.6 -2.4 -
2.2
[16] -2.0 -1.8 -1.6 -1.4 -1.2 -1.0 -0.8 -0.6 -0.4 -0.2  0.0  0.2  0.4  0.6
0.8
[31] 1.0  1.2  1.4  1.6  1.8  2.0  2.2  2.4  2.6  2.8  3.0  3.2  3.4  3.6
3.8
[46] 4.0  4.2  4.4  4.6  4.8  5.0
```

Similarly following command generates a sequence of 18 elements

```
> s4 <- seq(length=18, from=-5, by=.2)
> s4
[1] -5.0 -4.8 -4.6 -4.4 -4.2 -4.0 -3.8 -3.6 -3.4 -3.2 -3.0 -2.8 -2.6 -2.4 -
2.2
[16] -2.0 -1.8 -1.6
```

rep() which can be used for replicating an object in various complicated ways. The simplest form is **s5 <- rep(x, times=5)** which will put five copies of x end-to-end in s5.

```
> x
[1] 305  16 122  68
> s5 <- rep(x, times=5)
> s5
[1] 305  16 122  68 305  16 122  68 305  16 122  68 305  16 122  68 305  16
122
[20] 68
```

Another useful version is **s6 <- rep(x, each=5)** which repeats each element of x five times before moving on to the next.

```
> s6 <- rep(x, each=5)
> s6
[1] 305 305 305 305 305  16  16  16  16  16 122 122 122 122 122  68  68  68
68
[20]  68
> s7 <- rep(1:4,c(2,1,2,1))
> s7
[1] 1 1 2 3 3 4
```

1.8 Matrix Operation

To form a matrix you can use following syntax.
matrix(data =, nrow =, ncol= ,byrow="FALSE").

data : Actual data may be written in any of the variable or values by using function c().

nrow : Number of rows of a matrix

ncol : Number of columns of a matrix

byrow : It specifies whether matrix values are filled row wise or column wise. FALSE is by default i.e. column wise. If you want row wise then use TRUE.

For example,

```
> a <- c(1,2,3,4,5,6,7,8,9,10,11,12)
> A <- matrix(data=a, nrow=3, ncol=4, byrow="TRUE")
> a
[1] 1 2 3 4 5 6 7 8 9 10 11 12
> A
      [,1]  [,2]  [,3]  [,4]
[1,] 1      2     3     4
[2,] 5      6     7     8
[3,] 9      10    11    12
```

Specific values in a vector or in a matrix are referenced using square brackets ([ ]). For example,

```
> x <- c(5,8,9,7,6)
> x
[1] 5 8 9 7 6
> x[2]
[1] 8
> A[2,4]
[1] 8
> A[3,]
[1] 9  10  11  12
> A[c(2,3),1]     #display 2nd and 3rd element of the first column of matrix A
[1] 5 9
> A[c(2,3),2]
[1]  6 10   #display 2nd and 3rd element of the second column of matrix A
```

Matrix operators are provided in the Table

Table 2: Matrix Operations

Operation or Function	Description
A * B	Element-wise multiplication
A %*% B	Matrix multiplication
t(A)	Transpose
diag(x)	Creates diagonal matrix with elements of x in the principal diagonal
diag(A)	Returns a vector containing the elements of the principal diagonal
diag(k)	If k is a scalar, this creates a k x k identity matrix.
solve(A,b)	Returns vector x in the equation b = Ax
solve(A)	Inverse of A where A is a square matrix.
ginv(A)	Moore-Penrose Generalized Inverse of A. it requires loading the MASS package.
y ←qr(A)$rank	rank is the rank of A.
cbind(A,B,...)	Combine matrices(vectors) horizontally. Returns a matrix.
rbind(A,B,...)	Combine matrices(vectors) vertically. Returns a matrix.

1.9 Some commonly used Built-in functions

```
> x <- c(-6,9,0,-3,8,2,-5,4)
> x
[1] -6  9  0 -3  8  2 -5  4
> length(x)       #Displays the number of elements of vector x
[1] 8
> max(x)          #displays the maximum element of vector x
[1] 9
> min(x)          #displays the minimum element of vector x
[1] -6
> range(x)        #displays the range of the values of vector x
[1] -6  9
> sum(x)         # displays sum of the values of vector x
[1] 9
> cumsum(x)   # displays the cumulative sum of the values of vector x
[1] -6  3  3  0  8 10  5  9
> mean(x)        # displays the mean of the values of vector x
[1] 1.125
> median(x)      # displays the median of the values of vector x
[1] 1
> sort(x)   # Sort the values of vector x in the increasing order
[1] -6 -5 -3  0  2  4  8  9
> sort(x, decreasing = T)    # Sort the values of vector x in the decreasing
                             order
[1]  9  8  4  2  0 -3 -5 -6
> var(x)                # Sample variance with denominator (n-1)
[1] 32.125
> which(x == 4)   # displays index of the required element of vector x
[1] 8
> y <- c(3,4,-5)
> prod(y)         # displays product of the values of vector y
[1] -60
```

round( ): Syntax for the function is round(object, digits)
This function rounds object upto digits decimals. For example,

```
> round(3.2156,3)
[1] 3.216
```

1.10 Data frames

Data frames can be created by using data.frame. A data frame may be regarded as a matrix. It may be displayed in matrix form, and its rows and columns extracted using matrix indexing conventions. It is a list of vectors of the same length. (If the vectors included in the data frame are not of the same length then vector having less elements is recycled a whole number of times)

```
> x <- c(-5,7,-3,8); y = 8:11; z = rep(-5,4); p = seq(1,12,3)
> q = c(1,5)
> r = 5:7
> x;y;z;p;q;r
[1] -5  7 -3  8
```

```
[1]  8  9 10 11
[1] -5 -5 -5 -5
[1]  1  4  7 10
[1] 1 5
[1] 5 6 7
> d1 = data.frame(x,y)
> d1
          x      y
   1     -5      8
   2      7      9
   3     -3     10
   4      8     11
```

First column indicates row numbers.

```
> d2 = data.frame(q,p)
> d2
          q      p
   1      1      1
   2      5      4
   3      1      7
   4      5     10
```

In this data frame d2 vector having fewer elements (i.e. vector q) is recycled a whole number of times (2 times, so that its length becomes as that of length of other vector p)

Different columns in data frame are vectors. Names can be given to these columns while creating data frames.

```
> d4 = data.frame("maths" = x, "stats" = y)
> d4
          maths       stats
   1         -5           8
   2          7           9
   3         -3          10
   4          8          11
```

Rows in data frames can be given names using **row.names** which is a vector of character strings indicating names of rows.

```
> d5 = data.frame("maths" = x, "stats" = y, row.names = c("Amit", "Vidya",
"Ganesh", "Tina"))
> d5
            maths      stats
 Amit          -5          8
 Vidya          7          9
 Ganesh        -3         10
 Tina           8         11
```

1.11 Accessing data from data frames

Data from data frame can be accessed using $ notation

```
> d5 $ maths
[1] -5  7 -3  8
```

```
> d5 $ maths[3]
[1] -3
> d5[4,2]
[1] 11
```

1.12 Inbuilt data sets or Resident data sets

The data sets that come with R or one of the packages are known as Inbuilt data sets. To view all Inbuilt data sets names from package 'datasets' use following command.

```
> data()
```

For accessing existing data sets, command is as follows

```
> data(data set name)
> data(co2)
> co2             # displays data set co2
Note: Data frame can also be created using in-built data editor edit similar
to MS-Excel.
> stud <- edit(data.frame()) #this command displays in-built spread sheet.
> stud
           var1     var2
1         fybsc       45
2         fybsc       50
3         sybsc       55
4           msc       60
> names(stud) <- c("Standard","Marks")
> stud
       Standard    Marks
1         fybsc       45
2         fybsc       50
3         sybsc       55
4           msc       60
```

1.13 Importing Data from Excel

The function read.table() is the easiest way to import data into R. The preferred raw data format is either a tab delimited or a comma-separate file (CSV).
Working directory can be checked using getwd().
Store the excel file in csv format in this working directory.

```
> d1 <- read.table("temp1.csv",header=TRUE, sep=",")
# This creates dataframe d1
> d1
        Roll.No       Name      Marks
1            21      fgdgf         45
2            22       wqeq         78
3            23      zxcvz         60
4            25      jkljl         47
> dm = as.matrix(exp)
> dm
```

```
       Item                  Ramesh        Ganesh
[1,]   "Food"                "1600"        "1200"
[2,]   "Rent"                "1500"        "2000"
[3,]   "Electricity"         "1000"        "1500"
[4,]   "Misc."                "900"        "3000"
```

Chapter 2

Graphs and Diagram

Mr. Prashant Shah, Associate Professor and Head, Department of Statistics,
K. J. Somaiya College of Science and Commerce, Vidyavihar, Mumbai.

2.1 Introdution

Statistical data can be represented in the form of diagrams such as

- Simple bar diagram
- Multiple bar diagram
- Subdivided bar diagram
- Pie diagram or pie chart

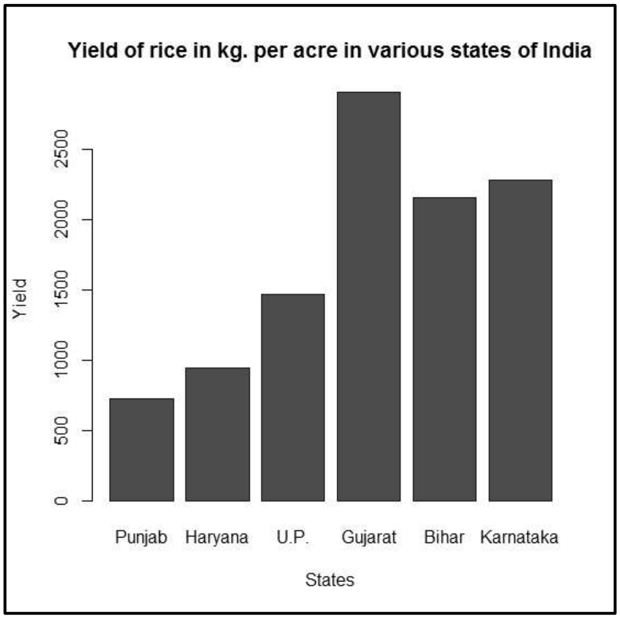
2.2 Bar Diagrams

```
> barplot(height, beside = T, names.arg = NULL, col = NULL, border =
par("fg"), main = NULL, xlab = NULL, ylab = NULL,  xlim = NULL, ylim =
NULL,...)
```

height: Either a vector or matrix of values describing the bars which make up the plot. If height is a vector, the plot consists of a sequence of rectangular bars with heights given by the values in the vector. If height is a matrix and beside is FALSE then each bar of the plot corresponds to a column of height, with the values in the column giving the heights of stacked sub-bars making up the bar. If height is a matrix and beside is TRUE, then the values in each column are juxtaposed rather than stacked.
names.arg: A vector of names to be plotted below each bar or group of bars. If this argument is omitted, then the names are taken from the names attribute of height if this is a vector, or the column names if it is a matrix.
main: Overall title for the plot.
beside: A logical value. If FALSE, the columns of height are portrayed as stacked bars, and if TRUE the columns are portrayed as juxtaposed bars (adjoining or contiguous bars)
Example: The following table gives the average approximate yield of rice in kg. per acre in various states of India in 2003-04. Represent it by **Simple Bar diagram**.

State :	Punjab	Haryana	U.P.	Gujarat	Bihar	Karnataka
Yield :	728	943	1469	2903	2153	2276

```
> x <- c("Punjab", "Haryana", "U.P.", "Gujarat", "Bihar", "Karnataka")
> y <- c(728, 943, 1469, 2903, 2153, 2276)
> x
[1] "Punjab"    "Haryana"   "U.P."      "Gujarat"   "Bihar"     "Karnataka"
> y
[1]  728  943 1469 2903 2153 2276
> barplot(y, names.arg = x, col = "red", border = "blue", main = "Yield of rice
in kg. per acre in various states of India", xlab = "States", ylab = "Yield")
```

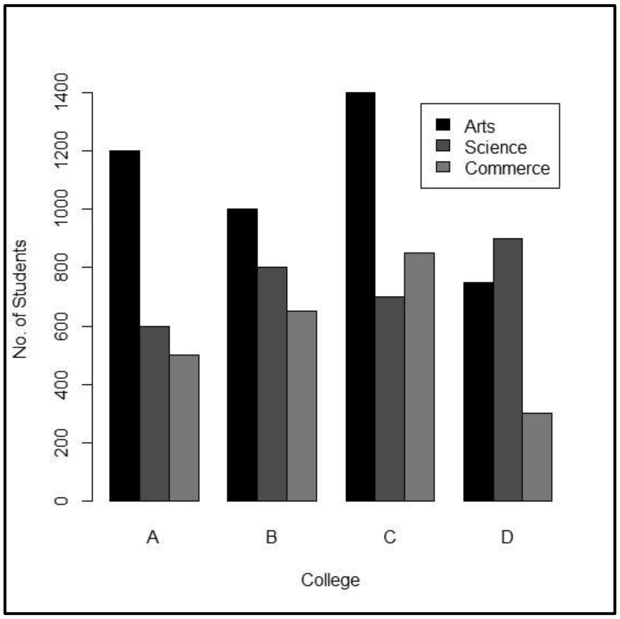


Example: Represent the following data on faculty-wise distribution of students, by **multiple bar diagram**.

College	Arts	Science	Commerce
A	1200	600	500
B	1000	800	650
C	1400	700	850
D	750	900	300

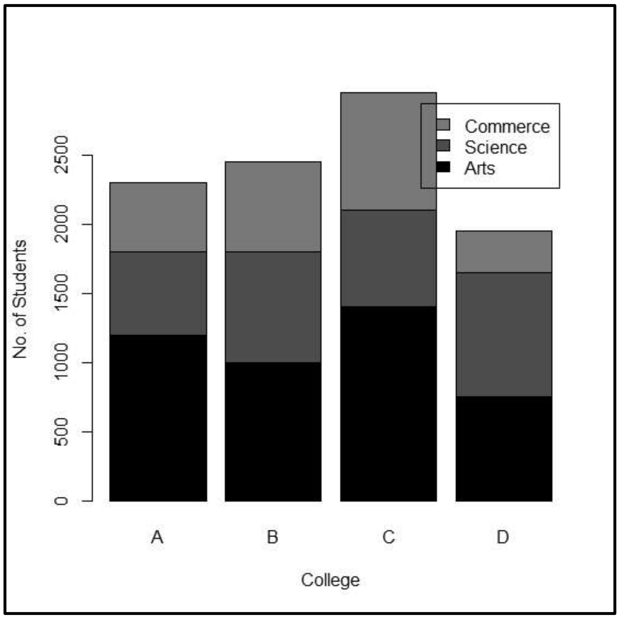
```
> clg <- c("A", "B", "C", "D")
> clgA <- c(1200, 600, 500)
> clgB <- c(1000, 800, 650)
> clgC <- c(1400, 700, 850)
> clgD <- c(750, 900, 300)
> d = data.frame(clgA, clgB, clgC, clgD)
> d
          clgA     clgB    clgC    clgD
1         1200     1000    1400     750
2          600      800     700     900
3          500      650     850     300
> d1 = as.matrix(d)
> d1
          clgA     clgB    clgC    clgD
[1,]      1200     1000    1400     750
```

```
[2,]       600      800      700      900
[3,]       500      650      850      300
> barplot(d1, beside = T, names.arg = clg, col = 1:2:3, legend = c("Arts",
"Science", "Commerce"),xlab = "College", ylab = "No. of Students")
```

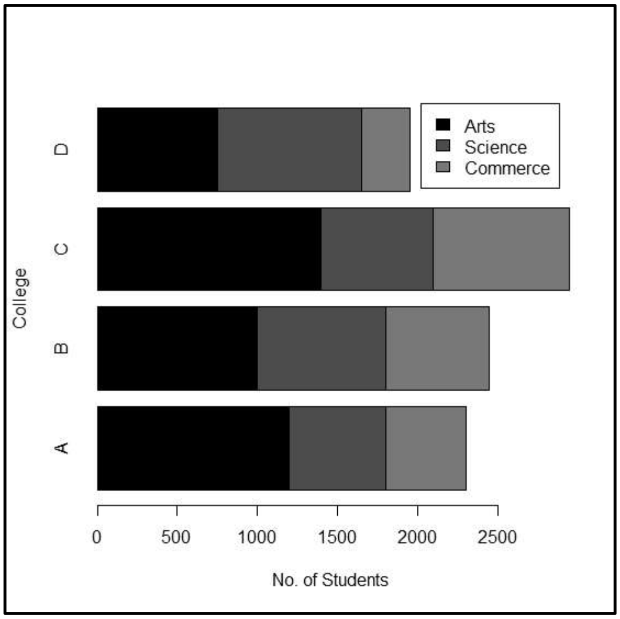


For the above example draw **subdivided bar diagram.**

```
> barplot(d1, beside = F, names.arg = clg, col = 1:2:3:4, legend = c("A",
"B", "C", "D"),xlab = "College", ylab = "No. of Students")
```



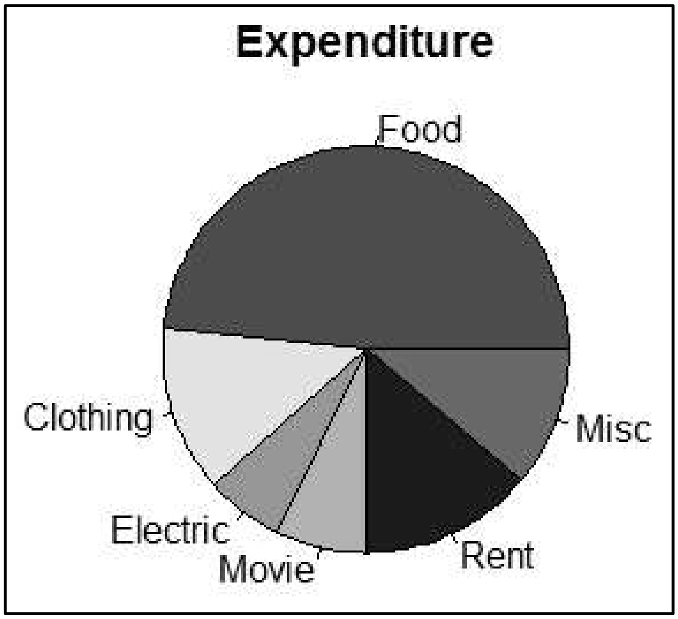
```
barplot(d1, beside = F, horiz = T, names.arg = clg, col = 1:2:3, legend =
c("Arts", "Science", "Commerce"),ylab = "College", xlab = "No. of Students")
```



Example: Represent the following data by a **pie diagram**:

Item :	Food	Clothing	Recreation	Indian	Rent	Miscellaneous
Expenditure (in Rs.)	87	24	11	13	25	20

```
> itm <- c("Food", "Clothing", "Electric", "Movie", "Rent", "Misc")
> exp ,- c(87, 24, 11, 13, 25, 20)
> pie(exp, main = "Expenditure", labels = itm, radius = 1,
col=rainbow(length(exp)))
```



2.3 Graphical Representation of data

Statistical data can be represented in the form of graphs such as

- Histogram
- Frequency polygon
- Ogive curve

R supports commands hist, plot, lines, points etc for drawing above graphs.

2.3.1 Histogram

```
> hist(x, breaks = classlimits, freq/probability = False/True, density =
NULL, col = NULL, border = NULL, main = paste("Histogram of", xname), xlim =
range(breaks), ylim = NULL, xlab = xname, ylab=yname,  axes = TRUE . . . .)
```

x: A vector of values for which the histogram is desired.

breaks: A vector giving breakpoints (class limits) for histogram. This can be done using c() or seq(). For eg: **breaks=c(100, 300, 500, 700)** Compute a histogram for the raw data values and set the bins (bars) such that they run from 100 to 300, 300 to 500 and 500 to 700. However, the c() function can make your code very messy sometimes. That is why you can instead use **breaks=**seq(x, y, z). The values of x, y and z are determined by yourself and represent, in order of appearance, the begin number of the x-axis, the end number of the x-axis and the interval in which these numbers appear.

```
> brk <- seq(148,178,5)
> hist(x, breaks = brk)
```

This command creates histogram with class limits 148 to 153, 153 to 158, 158 to 163, 163 to 168, 168 to 173, 173 to 178.

Note that you can also combine the two functions:

```
> hist(x, breaks=c(100, seq(200,700, 150)))
```

Make a histogram for the vector x, start at 100 on the x-axis, and from values 200 to 700, make the bins 150 wide

freq/probability: logical; if TRUE, the histogram graphic is a representation of frequencies; if FALSE, probability densities, are plotted (so that the histogram has a total area of one). Defaults to TRUE *if and only if* breaks are equidistant (and probability is not specified).

density: the density of shading lines, in lines per inch. The default value of NULL means that no shading lines are drawn.

col: a colour to be used to fill the bars. The default of NULL yields unfilled bars.

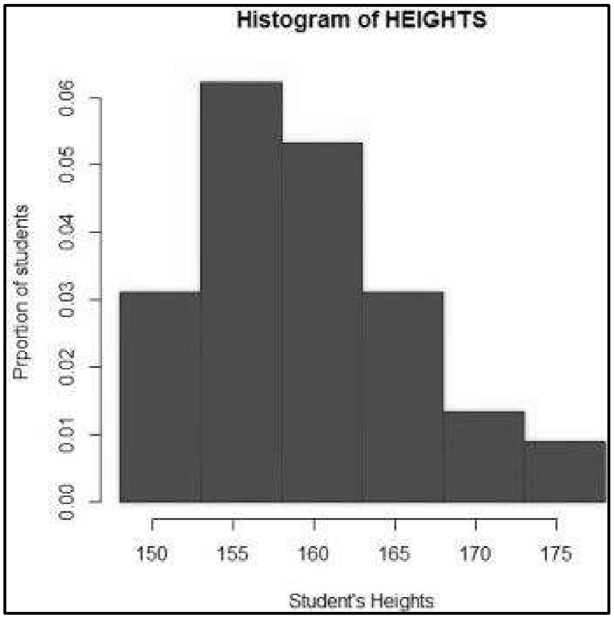
border: the color of the border around the bars. The default is to use the standard foreground color.

main: Overall title for the plot.

```
> brk <- seq(148,178,5)
> xnme = “Heights”
> hist(x, breaks = brk, freq = FALSE, main = paste("Histogram of" , xnme))

> x <- scan()
1: 170 151 154 160 158 154 171 156 160 157 148 165 158
14: 160 157 159 155 151 152 161 156 164 156 163 174 153 170 149 166 154
```

```
31: 166 160 160 161 154 163 164 160 148 162 167 165 158 158 176
46:
Read 45 items
> hist(x)
> hist(x, breaks = brk, freq = FALSE, col = "red", border = "blue", main =
paste("Histogram of" , xnme), xlab = "Student's Heights", ylab="Prportion of
students")
```



Histogram for ungrouped frequency data

x:	150	155	160	165	170	175
f:	6	11	14	9	3	2

```
> x <- seq(150,175,5)
> f <- c(6,11,14,9,3,2)
> y <- rep(x,f)
> hist(y)
> t = seq(147.5,177.5,5)
> hist(y, breaks = t)
```

Histogram for grouped frequency data

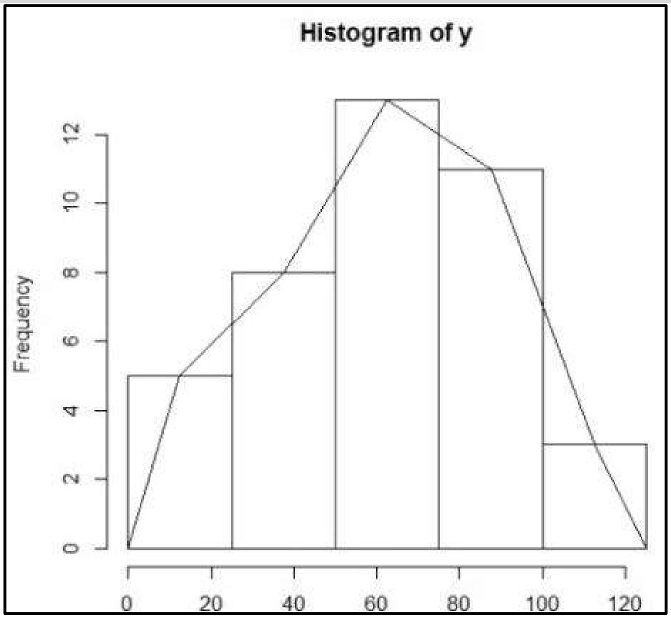
C.I.	0-25	25-50	50-75	75-100	100-125
f:	5	8	13	11	3

```
> midx <- seq(12.5,112.5,25)
> f <- c(5,8,13,11,3)
> cls_limit <- seq(0,125,25)
> y <- rep(midx,f)
> hist(y)
> hist(y, breaks=cls_limit)
```

2.3.2 Frequency polygon

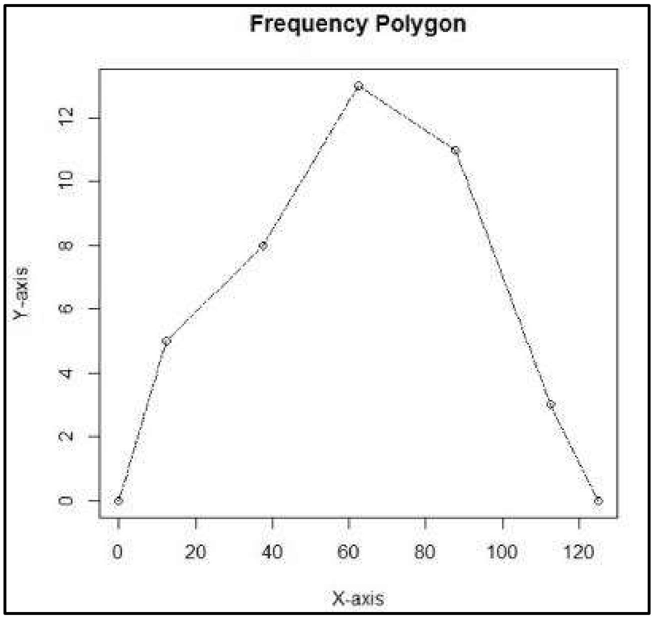
It is obtained by joining the points (x_i, f_i) where x_i is the midpoint of the i^{th} class interval and f_i is the corresponding frequency.

```
> lb <- seq(0,100,25)
> ub <- seq(25, 125, 25)
> midx <- (lb+ub)/2
> f <- c(5,8,13,11,3)
> x0 <- c(0, midx, 125)
> f0 <- c(0,f,0)
> y <- rep(midx,f)
> bks <- seq(0,125,25)
> hist(y,breaks=bks)
> lines(x0, f0)
```



OR

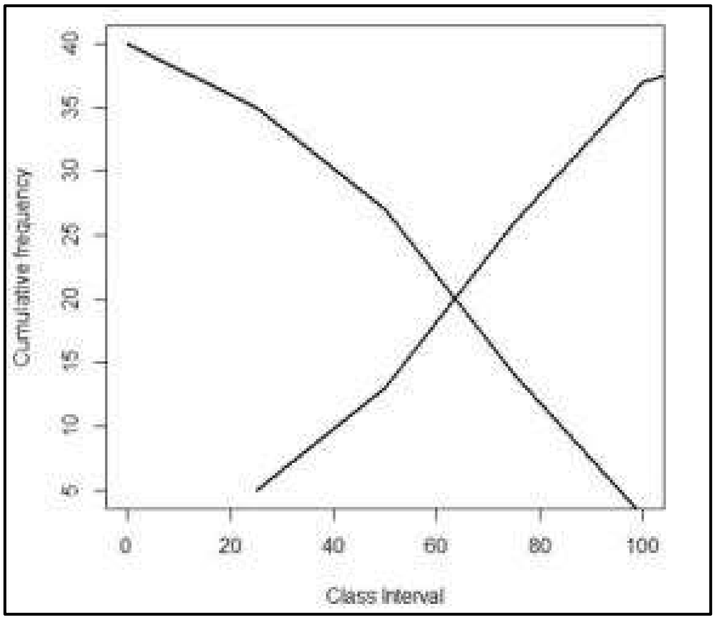
```
> plot(x0,f0, main = "Frequency Polygon", xlab ="X-axis", ylab = "Y-axis",
type = "o", lty =6,  xlim = range(min(x0),max(x0)))
```



2.3.3 Ogives

C.I.	0-25	25-50	50-75	75-100	100-125
f:	5	8	13	11	3

```
> f <- c(5,8,13,11,3)
> f
[1]  5  8 13 11  3
> lc <- cumsum(f)
> lc
[1]  5 13 26 37 40
> uc <- 1:5
> uc
[1] 1 2 3 4 5
> for(i in 5:1)
+ {uc[i] <- sum(f[5:i])}
> uc
[1] 40 35 27 14  3
> lbx <- seq(0,100,25)
> lbx
[1]   0  25  50  75 100
> ubx <- seq(25,125,25)
> ubx
[1]  25  50  75 100 125
> plot(ubx,lc,type = "l",xlim = c(0,100),xlab = "Class Interval", ylab =
"Cumulative frequency",lwd =2)
> lines(lbx,uc,type = "l",xlim = c(0,100),xlab = "Class Interval", ylab =
"Cumulative frequency",lwd =2)
```



Chapter 3

Measures of Central Tendency

Mrs. Pratiksha M. Kadam, Assistant Professor, Department of Statistics,
K. C. College, Churchgate, Mumbai – 400 020.

3.1 Introduction

According to Prof. Bowley, “Measures of central tendency (averages) are statistical constants which enable us to comprehend in a single effort the significance of the whole.” In this chapter we discuss the functions in R to calculate various measures of central tendency.

There are different types of averages.

1. Mathematical Averages:
   a. Arithmetic mean
   b. Geometric mean
   c. Harmonic mean
2. Positional Averages:
   a. Partition Values
      - Medians
      - Quartiles
      - Deciles
      - Percentiles
   b. Mode

3.2 Mathematical Averages

3.2.1 Arithmetic mean

For raw data:

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

Where n = the number of terms
x_i = i^{th} observation

For ungrouped frequency distribution:

$$\bar{x} = \frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}$$

Where n= total number of observations
x_i = ith observation; f_i = frequency of ith observation

For grouped frequency distribution:

$$\bar{x} = \frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}$$

Where x_i = mid-point of ith class interval
f_i = frequency of ith class

3.2.2 Geometric Mean

For raw data:

$$\bar{x} = \left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{n}}$$

Where n = the number of terms
x_i = ith observation

For ungrouped frequency distribution:

$$\bar{x} = \left(\prod_{i=1}^{n} {x_i}^{f_i}\right)^{\frac{1}{N}}$$

Where N=$\sum_{i=1}^{n} f_i$
n= total number of observations)
x_i = ith observation; f_i = frequency of ith observation

For grouped frequency distribution:

$$\bar{x} = \left(\prod_{i=1}^{n} {x_i}^{f_i}\right)^{\frac{1}{N}}$$

Where N=$\sum_{i=1}^{n} f_i$
x_i =mid-point of ith class interval
f_i = frequency of ith class

3.2.3 Harmonic Mean

For raw data:

$$\bar{x} = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}}$$

Where n = the number of terms
x_i = ith observation

For ungrouped frequency distribution:

$$\bar{x} = \frac{N}{\sum_{i=1}^{n} \frac{f_i}{f_i x_i}}$$

Where N=$\sum_{i=1}^{n} f_i$
n= total number of observations)
x_i = i^{th} observation
f_i = frequency of i^{th} observation
For grouped frequency distribution:

$$\bar{x} = \frac{N}{\sum_{i=1}^{n} \frac{f_i}{f_i x_i}}$$

Where N=$\sum_{i=1}^{n} f_i$
x_i =mid-point of i^{th} class interval
f_i = frequency of i^{th} class

3.3 Positional Averages

3.3.1 Partition Values

a) Median

Median is the value that divides the data into two equal parts, when the data is arranged in numerical order. It is the middle value when data size N is odd. It is the mean of the middle two values, when data size N is even.

For ungrouped frequency distribution:

Find the cumulative frequencies for the data. The value of the variable corresponding to which a cumulative frequency is greater than (N+1)/2 for the first time.(Where f_i = frequency of i^{th} observation, N=$\sum_{i=1}^{n} f_i$)

For grouped frequency distribution:

First obtain the cumulative frequencies for the data. Then mark the class corresponding to which a cumulative frequency is greater than N/2 for the first time. Find the cumulative frequencies for the data. The value of the variable corresponding to which a cumulative frequency is greater than (N+1)/2 for the first time.(Where f_i = frequency of i^{th} observation, N=$\sum_{i=1}^{n} f_i$.) Then that class is median class. Then median is evaluated by the following formula:

$$median = l_1 + \ (l_2 - l_1)\left(\frac{\frac{N}{2} - cf}{f_m}\right)$$

Where $N=\sum_{i=1}^{n} f_i$
f_i = frequency of i^{th} class; l_1= lower limit of the median class;
l_2= upper limit of the median class; f_m= frequency of the median class.
cf = cumulative frequency of the class proceeding to the median class.

b) Quartiles

The data can be divided in to four equal parts by three points. These three points are known as quartiles. The quartiles are denoted by Qi, i = 1,2,3. Qi is the value corresponding to (iN/4)th observation after arranging the data in the increasing order.

For grouped frequency distribution:
First we obtain the cumulative frequencies for the data. Then mark the class corresponding to which a cumulative frequency is greater than (iN)/4 for the first time. (Where f_i = frequency of i^{th} observation, $N=\sum_{i=1}^{n} f_i$). Then that class is Qi class. Then Qi is evaluated by formula:
i= 1, 2, 3

$$Q_i = l_1 + (l_2 - l_1)\left(\frac{\frac{iN}{4} - cf}{f_q}\right)$$

Where l_1= lower limit of the Qi class
l_2= upper limit of the Qi class
cf = cumulative frequency of the class proceeding to the Qi class.
f_q= frequency of the Qi class.

c) Deciles

Deciles are nine points which divided the data in to ten equal parts. Di is the value corresponding to (iN/10)th observation after arranging the data in the increasing order.

For grouped frequency distribution:
First obtain the cumulative frequencies for the data. Then mark the class corresponding to which a cumulative frequency is greater than (iN)/10 for the first time. (Where f_i = frequency of i^{th} observation, $N=\sum_{i=1}^{n} f_i$). Then that class is Di class. Then Di is evaluated by the following formula:

$$D_i = l_1 + (l_2 - l_1)\left(\frac{\frac{iN}{10} - cf}{f_d}\right)$$

i= 1, 2, …………10.
Where l_1= lower limit of the Di class
l_2= upper limit of the Di class; f_d= frequency of the Di class.
cf = cumulative frequency of the class proceeding to the Di class.

d) Percentile

Percentiles are ninety-nine points which divided the data in to hundred equal parts. Pi is the value corresponding to (iN)/100th observation after arranging the data in the increasing order.

For grouped frequency distribution:
First obtain the cumulative frequencies for the data. Then mark the class corresponding to which a cumulative frequency is greater than (iN)/100 for the first time. (Where f_i = frequency of ith observation, $N=\sum_{i=1}^{n} f_i$) Then that class is Pi class. Then Pi is evaluated by the following formula:

$$P_i = l_1 + (l_2 - l_1)\left(\frac{\frac{iN}{100} - cf}{f_p}\right)$$

Where i=1, 2, … , 100
l_1= lower limit of the Pi class; l_2= upper limit of the Pi class; f_p= frequency of the Pi class.
cf = cumulative frequency of the class proceeding to the Pi class;

3.3.2 Mode

The mode is the most frequent data value. Mode is the value of the variable which is predominant in the given data series. Thus in case of discrete frequency distribution, mode is the value corresponding to maximum frequency. Sometimes there may be no single mode if no one value appears more than any other. There may also be two modes (bimodal), three modes (trimodal), or more than three modes (multi-modal).

For grouped frequency distributions:
The modal class is the class with the largest frequency. After identifying modal class mode is evaluated by using interpolated formula. This formula is applicable when classes are of equal width.

$$Mode = l_1 + (l_2 - l_1)\left(\frac{d_1}{d_1 + d_2}\right)$$

Where l_1= lower limit of the modal class
l_2= upper limit of the modal class
$d_1 = f_m - f_0$ and $d_2 = f_m - f_1$
f_m= frequency of the modal class
f_0 = frequency of the class preceding to the modal class,
f_1= frequency of the class succeeding to the modal class.

3.4 Calculations of Measures of Central Tendency using R

Note: In R code red coloured text denotes the code for the calculation and blue coloured text denotes the output of the code written before that statement.

For measures of central tendency, we need to install package “psych” from CRAN. Before we start executing these functions we must load package “psych”.

To install “psych” Package in R:

In R Gui, Click on Packages menu and select the option “Install package(s)”, Select 0-cloud [https] from the country options and click on OK. Then a list of functions will be displayed. From that list select function “psych” and click on Install.

To load “psych” package in R:

In R Gui, Click on Packages menu and select the option “Load package”. List of installed packages will be shown. From that list select “psych” and click on OK.

Examples solved using R

1. Given the following data about average rainfall in every month in the year of 2017.

Month	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sept	Oct	Nov	Dec
Rainfall (in mm)	10	10	10	10	10	560	640	520	320	90	20	10

Calculate Arithmetic, Geometric, Harmonic mean, Median and Mode, First quartile, 56th percentile and 3rd decile for the above data.

R code:

```
> #ungrouped data
> rainfall = c(10, 10, 10, 10, 10, 560, 640, 520, 320, 90, 20, 10)
> mean(rainfall)
[1] 184.1667
> geometric.mean(rainfall)
[1] 46.69096
> harmonic.mean(rainfall)
[1] 17.92363
> median(rainfall)
[1] 15
# we define a function mode as follows:
> mode <- function(x) {
+    uniqx <- unique(x)
+    uniqx[which.max(tabulate(match(x, uniqx)))]
+ }
> mode(rainfall)
[1] 10
> quantile(rainfall, .25)
 25%
10
> quantile(rainfall, .56)
 56%
31.2
> quantile(rainfall,.3)
30%
10
```

2. The information about days and number of working hours for a week is given in the following table. Saturday and Sunday are holidays so working hours are not counted.

Day	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
Working Hours	NA	8	6	5.5	7	4.5	NA

Calculate arithmetic, geometric, harmonic mean, median, mode, third quartile, 32nd percentile value and 8th decile of the above data.

R code:

```
> #ungrouped data with NA values
> x=c(NA, 8, 6, 5.5, 7, 4.5, NA)
> mean(x)
[1] NA
> # as NA is included mean is not calculated. We need to exclude NA values to
calculate the mean of the given data.
> mean(x, na.rm=TRUE) #na.rm represents remove NA values.
[1] 6.2
> geometric.mean(x, na.rm=TRUE)
[1] 6.081111
> harmonic.mean(x, na.rm=TRUE)
[1] 5.962573
> median(x, na.rm = TRUE)
[1] 6
> mode <- function(x) {
+    uniqx <- unique(x)
+    uniqx[which.max(tabulate(match(x, uniqx)))]
+ }
> y=na.omit(x)#to remove NA from the dataset.
> mode(y)
[1] 8
> x=c(NA, 8, 6, 5.5, 7, 4.5, NA)
> y=na.omit(x)
> quantile(y,.75)
75%
  7
> quantile(y,.32)
 32%
5.64
> quantile(y,.8)
80%
7.2
```

3. The table shows the scores obtained by a group of players in a test. Find the arithmetic, geometric, harmonic mean, median, mode and first quartile, 21st percentile and 6th decile of the scores.

Scores	0	1	2	3	4	5	6
Frequency	3	5	4	6	4	5	3

R code:

```
> x=c(0, 1, 2, 3, 4, 5, 6)
> f=c(3, 5, 4, 6, 4, 5, 3)
> n=sum(f)
> y=rep(x,f)
> local({pkg <- select.list(sort(.packages(all.available =
TRUE)),graphics=TRUE)
+ if(nchar(pkg)) library(pkg, character.only=TRUE)})
> mean(y)
[1] 3
> geometric.mean(y)
[1] 0
> harmonic.mean(y)
[1] 0
> median(y)
[1] 3
> mode <- function(x) {
+    uniqx <- unique(x)
+    uniqx[which.max(tabulate(match(x, uniqx)))]
+ }
> mode(y)
[1] 3
> quantile(y,.25)
 25%
1.25
> quantile(y,.21)
21%
  1
> quantile(y,.6)
60%
```

4. The following data represents the distribution of monthly electricity bills of the families in a society. Find Arithmetic, geometric, harmonic mean, median and mode, Q_1, Q_3, D_7 and P_{68}.

Bill in (Rs.)	0-200	200-400	400-600	600-800	800-1000	1000-1200	1200-1400
Frequency	1	3	11	14	9	4	2

R code:

```
> ub=c(200, 400, 600, 800, 1000, 1200, 1400)
> lb=c(0,200, 400, 600, 800, 1000, 1200)
> h=200
> x=(lb+ub)/2
> f=c(1, 3, 11, 14, 9, 4, 2)
> n=sum(f)
> am =sum(x*f)/n
> am
[1] 713.6364
> gm=10^(sum(f*log10(x))/n)
> gm
[1] 655.632
> hm=n/sum(f/x)
> hm
[1] 570.1341
> lcf=cumsum(f)
> medc=min(which(lcf>n/2))
> med=lb[medc]+(n/2-lcf[medc-1])*h/f[medc]
> med
[1] 700
> modc=which(f==max(f))
> mode=lb[modc]+h*((f[modc]-f[modc-1])/(2*f[modc]-f[modc-1]-f[modc+1] ))
> mode
[1] 675
> q1c=min(which(lcf>n/4))
> q1=lb[q1c]+(n/4-lcf[q1c-1])*h/f[q1c]
> q1
[1] 527.2727
> q3c=min(which(lcf>3*n/4))
> q3=lb[q3c]+(3*n/4-lcf[q3c-1])*h/f[q3c]
> q3
[1] 888.8889
> d7c=min(which(lcf>7*n/10))
> d7=lb[d7c]+(7*n/10-lcf[d7c-1])*h/f[d7c]
> d7
[1] 840
> p68c=min(which(lcf>68*n/100))
> p68=lb[p68c]+(68*n/100-lcf[p68c-1])*h/f[p68c]
> p68
[1] 820.4444
```

3.5 References:

1. R for Beginners, Emmanuel Paradis
2. Descriptive Statistics, Vipul Publications, Mrs. M. J. Golba.

Chapter 4

Measure of Dispersion

Dr. Bhagat Gayval, Assistant Professor, Department of Statistics,
K. C. College, Churchgate, Mumbai – 400 020.

4.1 Range

It is difference between the smallest and largest values of the data. The range is the size of the smallest interval which contains all the data and provides an indication of Statistical dispersion. It is measured in the same units as the data. Since it only depends on two of the observations, it is most useful in representing the dispersion of small data sets.
Symbolically, Range=Max-Min

$$\text{Coefficient of Range} = \frac{\text{Max}-\text{Min}}{\text{Max}+\text{Min}}$$

4.2 Quartile Deviation

It is also measure of dispersion and it has cover 50% of data from all values. Quartile deviation (Q.D.) is given by formula:

$$\text{Q.D.} = \frac{1}{2}(Q_3 - Q_1)$$

Coefficient of Q.D. $= \frac{Q_3 - Q_1}{Q_3 + Q_1}$
Where Q_1 is the first quartile and Q_3 is the third quartile of the distribution.

4.3 Mean Deviation about ‘a’

Mean deviation is useful for finding the dispersion since it’s based upon all the observation and it is defined as the arithmetic mean of absolute deviations taken from any average or any value.
It is defined as follows:

$$\text{Mean Deviation about a} = \frac{1}{n}\sum_{1}^{n}|x_i - a|$$

Where ‘a’ can be mean or median or mode or any specified value.
In case of ungrouped/grouped frequency distribution

$$\text{Mean Deviation about a} = \frac{1}{N}\sum_{1}^{n} f_i|x_i - a|$$

Coefficient of mean deviation:

$$\text{Coefficient of } \textbf{mean deviation} = \frac{\text{Mean Deviation about a}}{\text{a}}$$

4.4 Variance

Variance is measures how far a data set is spread out and it is defined as the arithmetic mean of squares of deviations of the given values taken from arithmetic mean.

It is defined as

$$Var = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

Where $\bar{x}$ the mean, n is is the no. of observations of the data.

4.5 Standard Deviation

It is a measure that is used to quantify the amount of variation or dispersion of a set of data values. A low standard deviation indicates that the data points tend to be close to the expected value of the set, while a high standard deviation indicates that the data points are spread out over a wider range of values.

It is defined as

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$Coefficient\ of\ Variation\ (CV) = \frac{\sigma}{\bar{x}} \times 100$$

4.6 Examples

4.6.1 Section A-Raw Data-R coding and Example

Example – Find the range, Quartile Deviation, Mean deviation about median, Variance, Standard Deviation and their coefficients for the following data-
25,29,30,17,19,30,18,28,31,33,26,28

Range and Coefficient of range (Crange)

```
> x<-c(25,29,30,17,19,30,18,28,31,33,26,28)
> r<-range(x)
> r
[1] 17 33
> diff(r)
[1] 16
> Crange =(max(x)-min(x))/(max(x)+min(x))
> Crange
[1] 0.32
#Quartile Deviation (QD) & Coefficient of QD
> QD=(quantile(x,0.75)-quantile(x,0.25))/2
> QD
 > 3.25
> CoeffQD=(quantile(x,0.75)-
quantile(x,0.25))/(quantile(x,0.75)+quantile(x,0.25))
> CoeffQD
0.1214953
 # Mean Deviation from median
# Library (‘psych’)
# mad function calculates Mean Deviation from median
> mad(x)
[1] 3.7065
> cmd=(mad(x))/(median(x))        #calculated coefficient of mean deviation
about median
> cmd
[1] 0.132375
# Variance & CV
> variance<-var(x)   #sample variance
> variance
[1] 28.87879
> CV=(sd(x)*100)/mean(x)
> CV
[1] 20.53719
# Standard Deviation(SD) & Standard Error (SE)
> SD<-sd(x)  #sample standard deviation
> psd=(SD*sqrt(length(x)-1))/ sqrt(length(x))
> psd     #population standard deviation
[1] 5.145116
> cv=(psd/mean(x))*100
> cv
[1] 19.66287
```

4.6.2 Section B -ungrouped data set

Example – Find the range, Quartile Deviation, Mean deviation, Variance, Standard Deviation and their coefficients for the following data-

X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Frequency	5	14	21	23	60	80	86	125	112	93	56	43	32	24	22	16

# Range and Coefficient of range

```
> grp=seq(0,15,by=1)
> f=c(5,14,21,23,60,80,86,125,112,93,56,43,32,24,22,16)
> data=rep(grp,f)
> r<-range(data)
> diff(r)
[1] 15
> coeffrange=(max(data)-min(data))/(max(data)+min(data))
> coeffrange
[1] 1
#Quartile Deviation (QD) & Coefficient of QD
QD=(quantile(data,0.75)-quantile(data,0.25))/2
> QD
1.625
> CoeffQD=(quantile(data,0.75)-
quantile(data,0.25))/(quantile(data,0.75)+quantile(data,0.25))
> CoeffQD
0.220339
```

# Mean Deviation from median

Library ('psych')

mad function calculates Mean Deviation from median

```
> mad(data)
[1] 2.9652
> cmd=(mad(data))/(median(data))  #calculated coefficient of mean deviation
about median
> cmd
[1] 0.4236
```

# Variance & Coefficient of Variance (CV)

```
> variance<-var(data)
> variance
[1] 9.548566
```

# Standard Deviation(SD) & Standard Error (SE)

```
> SD<-sd(data)  #sample standard deviation
> psd=(SD*sqrt(length(data)-1))/ sqrt(length(data))
> psd     #population standard deviation
[1] 3.088172
> cv=(psd/mean(data))*100
> cv
[1] 40.66811
```

4.6.3 Section C -Grouped data set

Example – Find the range, Quartile Deviation, Mean absolute deviation, Variance, Standard Deviation for the following data-

Age	20-30	30-40	40-50	50-60	60-70
No. of person	25	42	28	15	10

```
> grp=seq(0,15,by=1)
> f=c(5,14,21,23,60,80,86,125,112,93,56,43,32,24,22,16)
> data=rep(grp,f)
> cmd=(mad(data))/(median(data)) #calculated coefficient of mean deviation
about median
> cmd
[1] 0.4236
> SD<-sd(data)  #sample standard deviation
> psd=(SD*sqrt(length(data)-1))/ sqrt(length(data))
> psd     #population standard deviation
[1] 3.088172
> cv=(psd/mean(data))*100
> cv
[1] 40.66811
> lb = seq(20,60,10)
> lb
[1] 20 30 40 50 60
> ub = seq(30,70,10)
> ub
[1] 30 40 50 60 70
> midx = (lb+ub)/2
> midx
[1] 25 35 45 55 65
> f = c(25,42,28,15,10)
> y = rep(midx,f)
> range = ub[length(ub)] - lb[1] #calculates range
> range
[1] 50
> cf = cumsum(f) #calculates cumulative frequency of greter than type
> cf
[1]  25  67  95 110 120
> q1_mincf = min(which(cf >= sum(f)/4))
> q1_mincf
[1] 2
> q1_l1 = lb[q1_mincf]; q1_l2 = ub[q1_mincf]
> q1_l1;q1_l2
[1] 30
[1] 40
> h = (q1_l2-q1_l1)
> h
[1] 10
> first_quart = q1_l1 + (h*(sum(f)/4-cf[q1_mincf-1])/f[q1_mincf])
> first_quart
[1] 31.19048
> x_bar = sum(f*midx)/sum(f)
> x_bar
[1] 40.25
> dev_mean = f * (midx - x_bar)^2
> dev_mean
[1] 5814.062 1157.625  631.750 3263.438 6125.625
> variance = sum(dev_mean)/sum(f)
> variance
[1] 141.6042
```

4.7 Skewness and Kurtosis

4.7.1 Skewness

Lack of symmetry in distribution is called as Skewness. We know that the Skewness can be positive or negative or zero. If the relation of mean>median>mode then it will be positive and curved as right tail. If the relation of mean<median<mode then it will get negative and curve as left tail. If the values of mean=median=mode then there is no Skewness.

Mathematically measures of Skewness have studied as follows:

(A) Absolute Skewness measures:

I) Karl Person's measure of Skewness=Mean-Mode=3(Mean-Median)

II) Bowley's measure of Skewness=(Q_3-Q_2)-(Q_2-Q_1)

(B) Relative or coefficient of Skewness measures:

I) Karl Person's coefficient of Skewness

$$SKp = \frac{Mean - Mode}{S.D.} = \frac{3(Mean - Median)}{S.D.}$$

If SK_P>0 the curve is positively skewed, if SK_P=0 then the curve is symmetric andz if SK_P<0 then the curve is said to be negatively skewed curve.

II) Bowley's coefficient of Skewness

$$SK_B = \frac{(Q_3 + Q_1 - 2Q_2)}{(Q_3 - Q_1)}$$

If SK_B>0 the curve is positively skewed, if SK_B=0 then the curve is symmetric and if SK_B<0 then the curve is said to be negatively skewed curve.

III) Measures based on moments

$$\beta_1 = \frac{\mu_3^2}{\mu_2^3}$$

Relative measure of Skewness

$$\gamma_1 = \pm\sqrt{\beta_1}$$

If $\gamma_1 > 0$ then the curve is positively skewed, if $\gamma_1 = 0$ then the curve is symmetric and if $\gamma_1 < 0$ then the curve is negatively skewed curve.

4.7.2 Kurtosis:

Kurtosis enables us to have an idea about the flatness or peakedness of the frequency curve. Kurtosis is measuredly compared with normal distribution. Mainly Kurtosis will be defined by three types such as Leptokurtic, Mesokurtic and Platykurtic distribution.

Mesokurtic distribution is as likely as normal distribution. In Leptokurtic distribution, the Kurtosis greater than Mesokurtic distribution and in Platykurtic distribution the Kurtosis is less than Mesokurtic distribution.

It is defined as follows:

$$\beta_2 = \frac{\mu_4}{\mu_2^2} \quad , \quad \gamma_2 = \beta_2 - 3$$

Where Platykurtic curve is defined as $\beta_2 < 3$ or $\gamma_2 < 0$,
Leptokurtic curve is defined as $\beta_2 > 3$ or $\gamma_2 > 0$,
And Mesokurtic curve is defined as $\beta_2 = 3$ or $\gamma_2 = 0$.

4.7.3 Examples

Raw Data-R coding and Example

Example – Find the Skewness and Kurtosis and for the following data-
25,29,30,17,19,30,18,28,31,33,26,28

```
> # Karl Person’s coefficient of Skewness
> x<-c(25,29,30,17,19,30,18,28,31,33,26,28)
> psd=(SD*sqrt(length(x)-1))/ sqrt(length(x))
> skp=(3*(mean(x)-median(x)))/ psd
> skp
[1] -1.859036
> #Bowley’s coefficient of Skewness
> a=quantile(x,0.75);  b=quantile(x,0.25);  c=2*quantile(x,0.5)
> num=a+b-c;  denom=a-b
> skb=num/denom;  skb
-0.3846154
> # Measure based on Moments
> library(moments)
> skw=skewness(x); skw
[1] -0.6760079
> cs=sqrt(abs(skw))
> coefficient=-(cs)
> coefficient
[1] -0.822197
> # Kurtosis -based on Moments
> library(moments)
> kur=kurtosis(x)
> kur
[1] 2.068371
> coefK=kur-3
> coefK
[1] -0.9316292
```

Chapter 7

Probability and Probability Distributions

Dr. Asha A. Jindal, Associate Professor and Head, Department of Statistics,
K. C. College, Churchgate, Mumbai – 400 020.

7.1 Probability

In real life, experiments are classified into two categories.

- Deterministics experiments
- Probabilistics experiments

In probability theory we are concerned with random experiments. The set of all possible outcomes of a random experiment is called as a sample space.

In computing probabilities of different events using R software we use function choose (n,r) which gives the value of number of combination of n objects taken r at a time(order is not important) whereas function factorial (n)/factorial (n-r) gives the value of number of n objects taken r at a time(order is important).

If a random experiment results in 'n' equally likely, mutually exclusive and exhaustive cases and if 'm' of them are favourable to the event A then the probability of event A is the ratio of m to n.

$$\mathrm{P(A)} = \frac{m}{n} = \frac{Total\ No.of\ cases\ favourable\ to\ event\ A}{Total\ no.of\ cases}$$

1) calculate a) $^{10}C_3$ b)$^{8}C_4$ c) $^{9}P_3$ d) $^{5}P_2$.

Solution:

```
> al=choose (10,3)
> al
[1] 120
> a2=choose (8,4)
> a2
[1] 70
> a3=factorial (9)/factorial (9-3)
> a3
[1] 504
> a4=factorial (5)/factorial (5-2)
> a4
[1] 20
```

2) In a group of 6 boys and 4 girls, four children are to be selected. In how many different ways can they be selected such that at least one boy should be there?

Solution:

```
> q= (choose (6,1) *choose (4,3) +choose (6,2) *choose (4,2) +choose (6,3)
*choose (4,1) +choose(6,4) *choose (4,0))
> q
[1] 209
```

3) From a group of 7 men and 6 women, five persons are to be selected to form a committee so that at least 3 men are there in the committee. In how many ways can it be done?

Solution:

```
> r =(choose (7,3) *choose (6,2) +choose (7,4) *choose (6,1) +choose (7,5)
*choose (6,0))
> r
[1] 756
```

4) In how many different ways can the letters of the word 'CORPORATION' be arranged so that the vowels always come together?

Solution:

```
> s= (factorial (7)/factorial (2) *factorial (5)/factorial (3))
> s
[1] 50400
```

5) How many 3-letter words with or without meaning, can be formed out of the letters of the word, 'LOGARITHMS', if repetition of letters is not allowed?

Solution:

```
> t=factorial (10)/factorial (10-3)
> t
[1] 720
```

6) In how many different ways can the letters of the word, 'LEADING', be arranged such that the vowels should always come together?

Solution:

```
> u=factorial (5) *factorial (3)
> u
[1] 720
```

7) How many arrangements can be made out of the letters of the word, 'ENGINEERING'?

Solution:

```
> v=factorial (11)/ (factorial (2) ^factorial (3) *factorial (3) *factorial
(2))
> v
[1] 277200
```

8) How many 6-digit telephone numbers can be formed if each number starts with 35 and no digit appears more than once?

Solution:

```
> w=factorial (8)/factorial (8-4)
> w
[1]1680
```

9) A box contains 4 red,3 white and 2 blue balls. Three balls are drawn at random. Find out the number of ways of selecting the balls of different colours?

Solution:

```
> X= (choose (4,1) *choose (3,1) *choose (2,1))
> x
[1] 24
```

10) What is the probability of drawing two Ace cards from well shuffled pack of 52 playing cards?

Solution:

```
>y= (choose (4,2)/choose (52,2))
>y
[1] 0.004524887
```

11) A box contains 5 red and 7 blue marbles.A sample of 4 is drawn at random what is probability of selecting at least two blue marbles?

Solution:

```
>z=(choose (5,2) *choose (7,2) +choose (5,1) *choose (7,3) +choose (5,0)
*choose (7,4))/ (choose(12,4))
> z
[1] 0.8484848
```

7.2 Probability Distributions

Binomial Distribution

R supports following functions related to binomial distribution with specified parameters.

dbinom(x,n,p)	It gives individual binomial probability at X=x.
pbinom(x,n,p)	It gives cumulative binomial probability function. P(X≤ x).
qbinom(x,n,p)	It gives quantile fuction.
rbinom(m,n,p)	It generates a random sample of size m from binomial distribution.

Similar functions starting with letter d, p, q and r are used in connection with different distributions.
Following are some commonly used distributions with their R names.

Distributions	R name	Additional Arguments
Binomial	binom	Size, probability
Poisson	Pois	Parameter lambda
Hypergeometric	hyper	M, N-M, n
Geometric	geom	probability
Negative Binomial	nbinom	Size, probability
Uniform	unif	min, max
Exponential	exp	rate
Normal	norm	mean, sd
Log--normal	lnorm	meanlog, sdlog
Cauchy	cauchy	location, scale
Gamma	gamma	shape, scale
Beta	beta	shape1, shape2, ncp
Student's t	t	df, ncp
F	f	df1, df2, ncp
Chi-square	chisq	df, ncp
Logistic	logis	location, scale
Weibull	weibull	shape, scale
Wilcoxon	wilcox	m. n

1) If X~Bino (10,0.6). Find a) P(X=0) b) P(X=2) c) P(X≤3) d) P(X>5)

Solution:

Given : X~Bin (n=10, p=0.6)

```
> a1=dbinom (0,10,0.6)
> a1
[1] 0.0001048576
 b] P(X=2)
> b1=dbinom (2,10,0.6)
> b1
[1] 0.01061683
 c] P(X<=3)
> c1=pbinom (3,10,0.6)
> c1
[1] 0.05476188
 d] P(X>5)
> d1=1-pbinom (5,10,0.6)
> d1
[1] 0.6331033
```

2) If X~P (3.2). Find a) P(X=0) b) P(X=3) c) P(X=5) d) P(X<=1) e) P(X>3) f) P(X≥5) .

Solution:

```
> X~P(m=3.2)
> a1=dpois (0,3.2)
> a1
[1] 0.0407622
> b1=dpois (3,3.2)
> b1
[1] 0.222616
> c1=dpois (5,3.2)
> c1
[1] 0.1139794
> d1=ppois (10,3.2)
> d1
[1] 0.9995028
> e1=1-ppois (3,3.2)
> e1
[1] 0.3974803
> f1=1-ppois (5,3.2)
> f1
[1] 0.1054081
```

3) If X~HyperGeo (N=25, M=5, n=3).
Find a) P(X=0) b) P(X=2) c) P(X=5) d) P(X≤1) e) P(X>3) f) P(X≥2).

Solution:

Given :X ~ HyperGeo (N = 25, M = 5, n = 3)

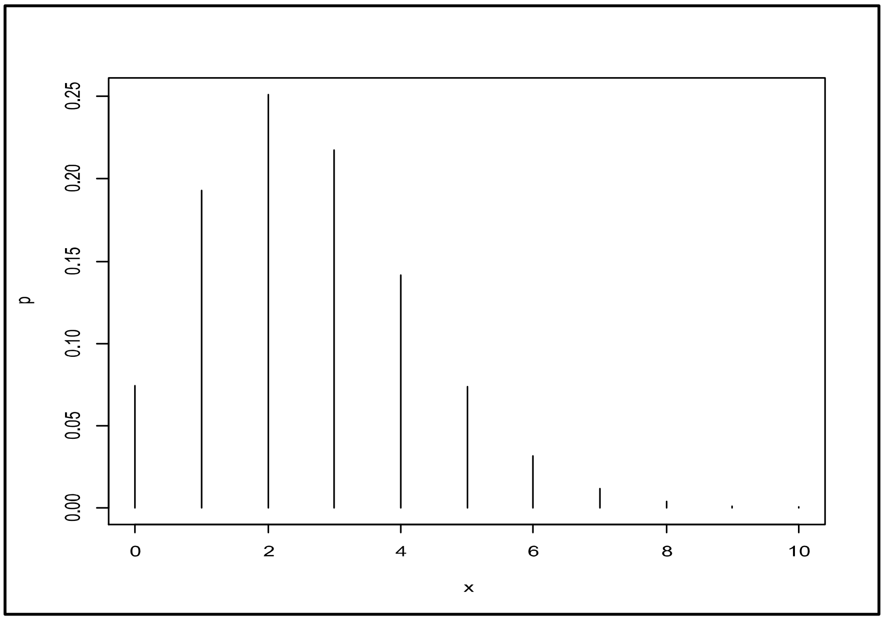
```
> a1=dhyper (0,5,20,3)
> a1
[1] 0.4956522
> b1=dhyper (2,5,20,3)
> b1
[1] 0.08695652
> c1=dhyper (5,5,20,3)
> c1
[1] 0
> d1=phyper (1,5,20,3)
> d1
[1] 0.9086957
> e1=1-phyper (3,5,20,3)
> e1
[1] 0
> f1=1-phyper (2,5,20,3)
> f1
[1] 0.004347826
```

4) Plot probability mass function (pmf)and distribution function for the following ramdom variables a)X~P (2.6)   b) X~Bino (8,0.65) c) X~ HyperGeo (N=50, M=10, n=7)

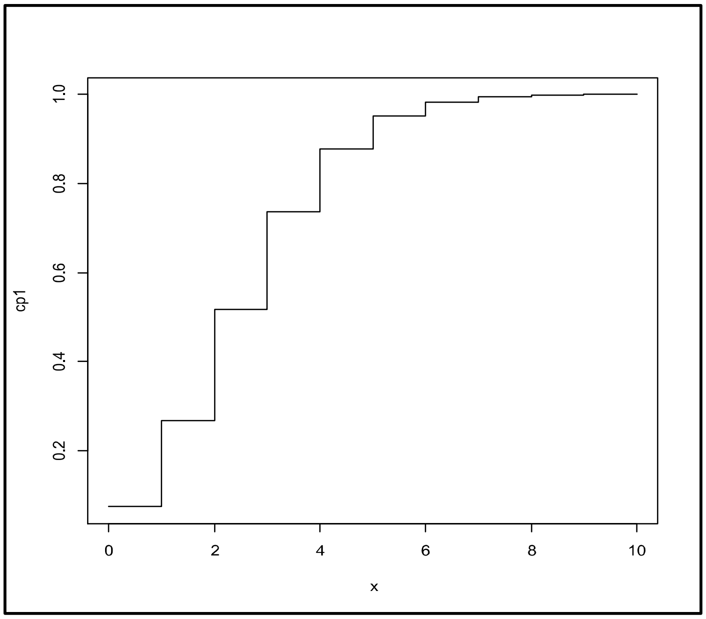
Solution:

a) X~P (2.6)

```
> m=2.6
> x=0:10
> p=dpois (x, m)
> d=data.frame (x, p)
> d
               x          p
    1.         0      0.0742735782
    2.         1      0.1931113034
    3.         2      0.2510446944
    4.         3      0.2175720684
    5.         4      0.1414218445
    6.         5      0.0735393591
    7.         6      0.0318670556
    8.         7      0.0118363349
    9.         8      0.0038468089
   10.         9      0.0011113003
   11.        10      0.0002889381
> plot (x, p,"h")
```

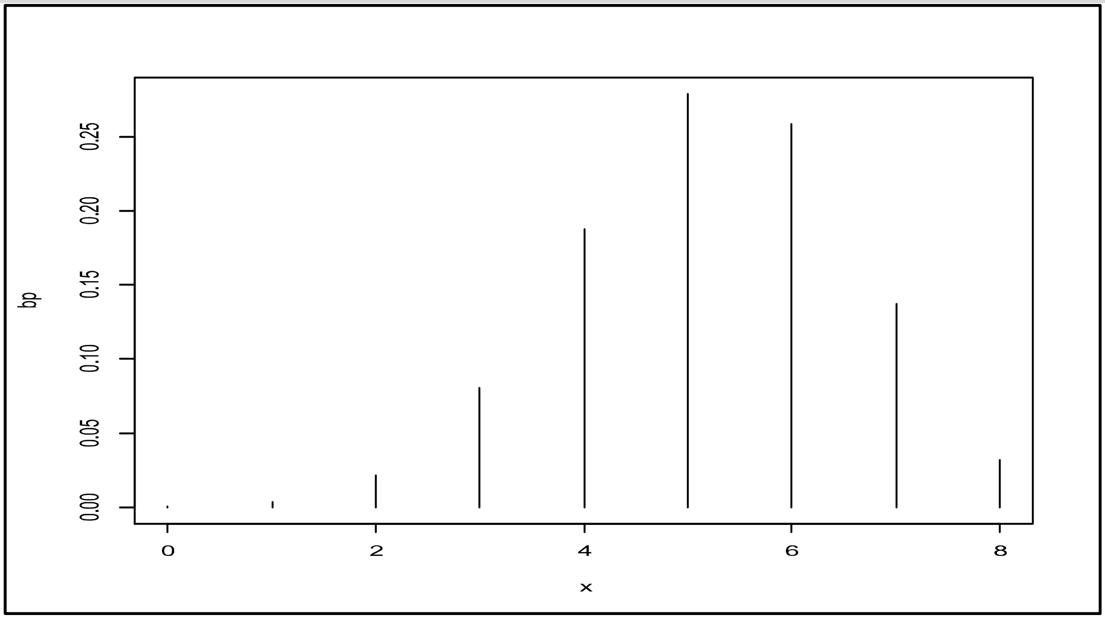


```
> cp=ppois(x, m)
> cp1=round(cp,4)
> d1=data.frame(x, cp1)
> plot (x, cp1,"s")
```

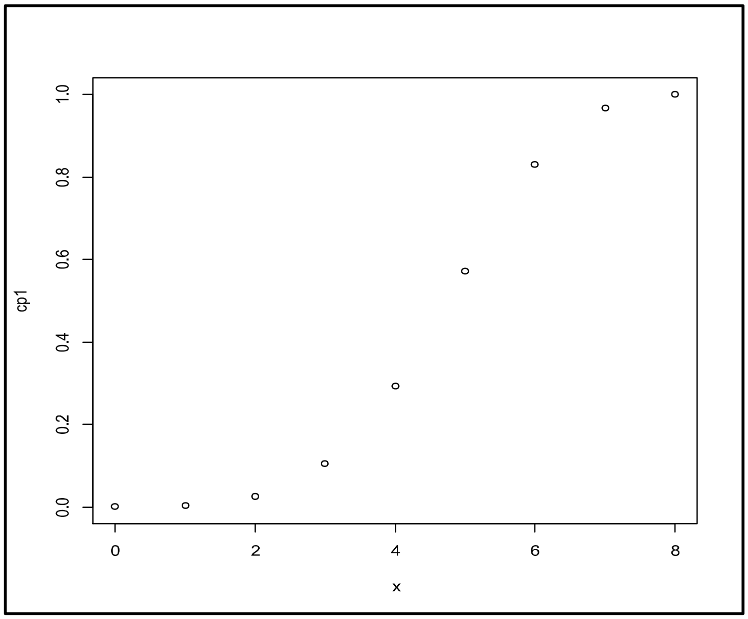


b) X~ Bino (8,0.65)

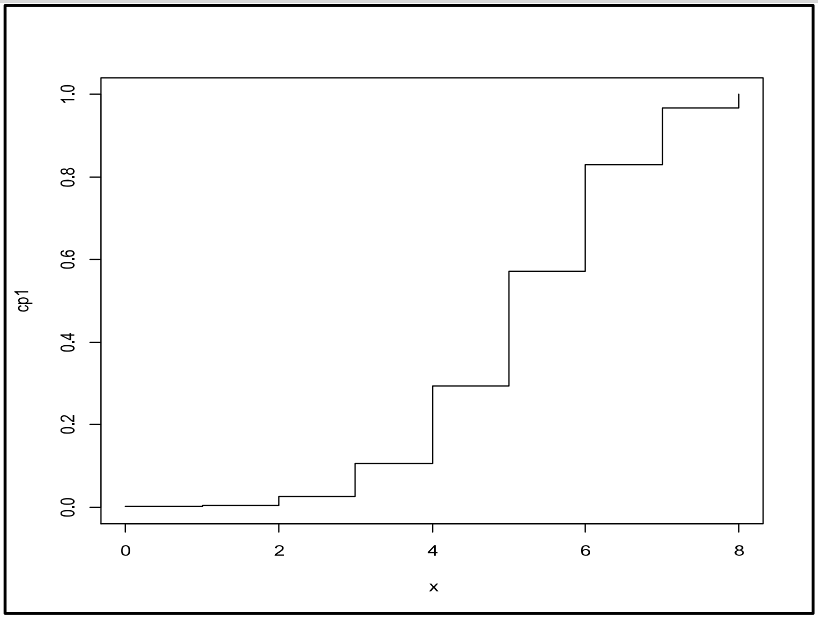
```
> n=8; p=0.65
> x=0: n
> bp=dbinom(x, n, p)
> d=data.frame("x-values"=x,"probabilities"=bp)
> d
     x.values    probabilities
 1          0     0.0002251875
 2          1     0.0033456434
 3          2     0.0217466823
 4          3     0.0807733916
 5          4     0.1875096590
 6          5     0.2785857791
 7          6     0.2586867948
 8          7     0.1372623809
 9          8     0.0318644813
>plot (x, bp,"h")
```



```
> cp=pbinom (x, n, p)
> cp1=round (cp,4)
> d1=data.frame(x, cp1)
> plot (x, cp1)
```

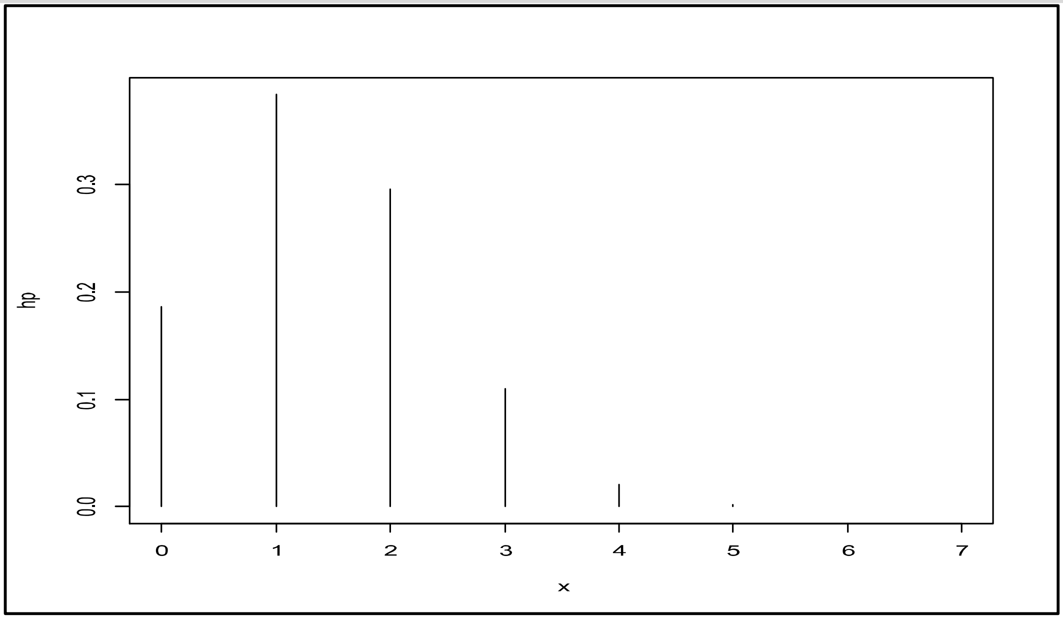


```
> plot (x, cp1,"s")
```

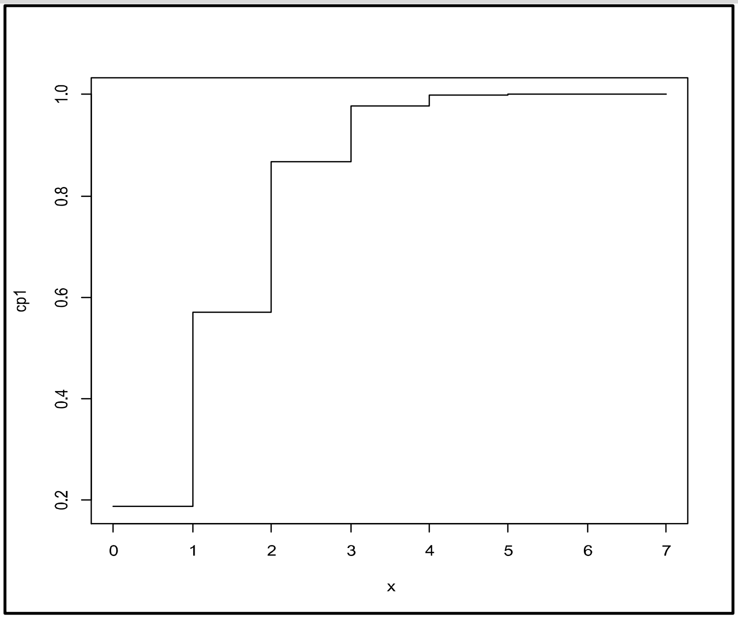


c) X ~ HyperGeo (N=50, M=10, n=7)

```
> N=50; M=10; n=7
> x=0: n
> hp=dhyper (x, M, N-M, n)
> d=data.frame(x, hp)
> d
        x        hp
 1      0   1.866514e-01
 2      1   3.842822e-01
 3      2   2.964463e-01
 4      3   1.097949e-01
 5     4   2.077201e-02
 6     5   1.967875e-03
 7     6   8.409722e-05
 8     7   1.201389e-06
>plot (x, hp,"h")
```



```
> cp=phyper (x, M, N-M, n)
> cp1=round(cp,4)
> di=data.frame(x, cp1)
> di
        x      cp1
 1      0     0.1867
 2      1     0.5709
 3      2     0.8674
 4      3     0.9772
 5      4     0.9979
 6      5     0.9999
 7      6     1.0000
 8      7     1.0000
>plot (x, cp1,"s")
```



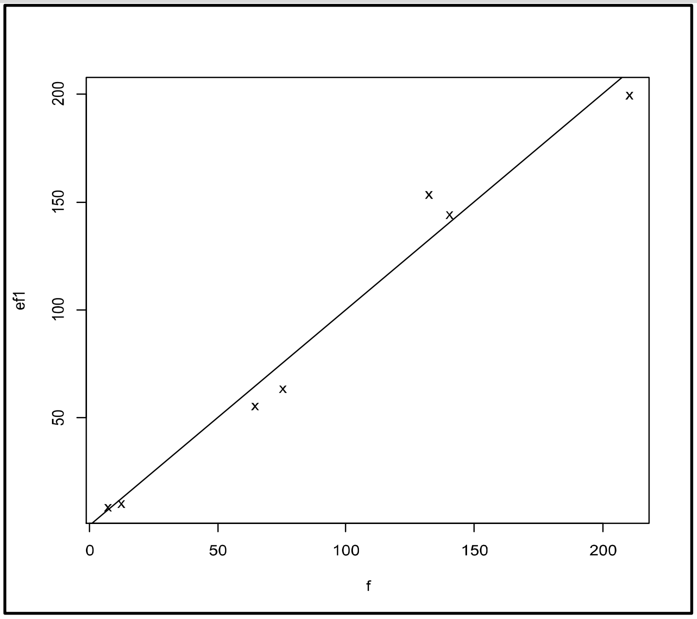
5) A set of similar fair coins are tossed 640 times with the following result – no. of

Heads:	0	1	2	3	4	5	6
Frequency:	7	64	140	210	132	75	12

Fit the binomial distribution to the data.

Solution:

```
> x=0:6
> f=c (7,64,140,210,132,75,12)
> m=sum(x*f)/sum(f)
> n=max(x)
> p=m/n; q=1-p
> px=dbinom (x, n, p)
> px1=round(px,4)
> ef=sum(f)*px1
> ef1=round(ef,0)
> d=data.frame(x, f,"expected frequency"=ef1)
> d
      x     f   expected.frequency
  1   0     7                    9
  2   1    64                   56
  3   2   140                  145
  4   3   210                  200
  5   4   132                  154
  6   5    75                   64
  7   6    12                   11
>plot (f, ef1, pch="x"); abline (0,1)
```

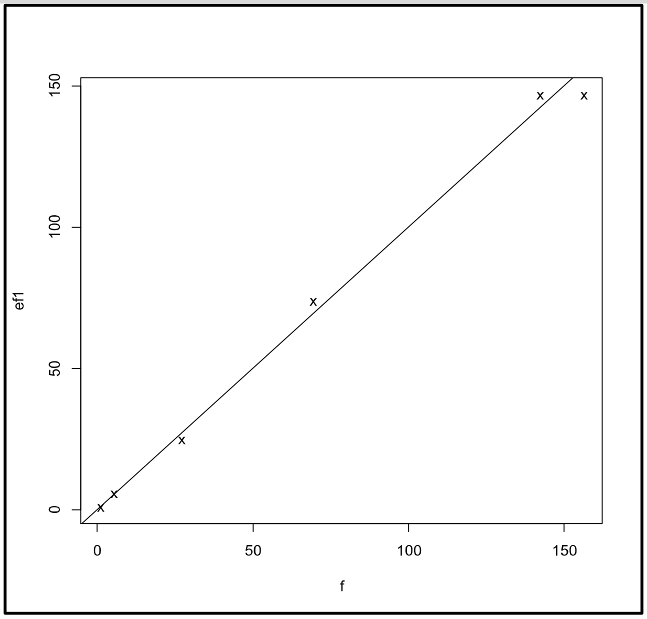


6) Fit the Poisson distribution to the following data with respect to the

Number of red blood corpuscles (x) per cell – x:	0	1	2	3	4	5
no. of cells	142	156	69	27	5	1

Solution:

```
> x=0:5
> f=c (142,156,69,27,5,1)
> m=sum(x*f)/sum(f)
> px=dpois (x, m)
> px=round(px,4)
> ef=sum(f)*px
> ef1=round(ef,0)
>d=data.frame(x, f,"expected frequency"=ef)
> d
      x     f    expected.frequency
  1   0   142                147.16
  2   1   156                147.16
  3   2    69                 73.56
  4   3    27                 24.52
  5   4     5                  6.12
  6   5     1                  1.24
>plot (f, ef1, pch="x"); abline (0,1)
```



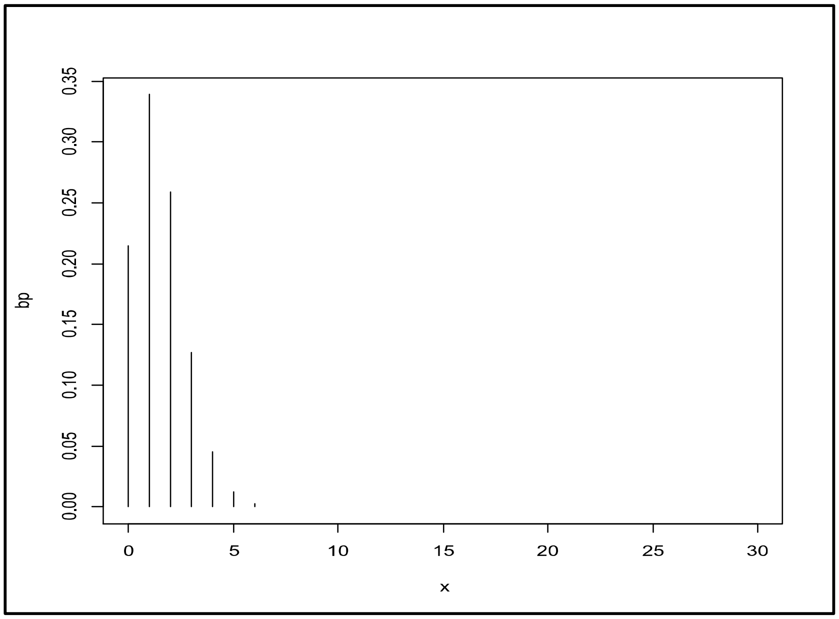
7) Plot the pmf of a] X~Bino (30,0.05) b]X~P (1.5) and comment on graph

Solution:

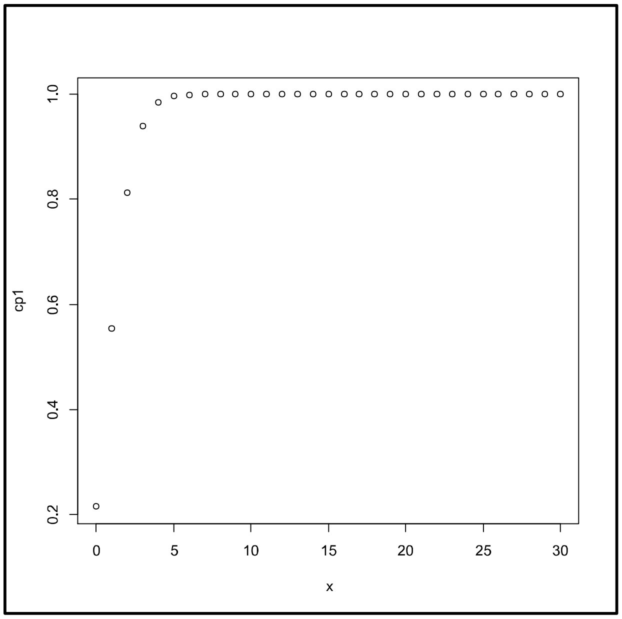
Given: X~Bino (30,0.05)

```
> n=30; p=0.05
> x=0: n
> bp=dbinom (x, n, p)
> d=data.frame("x-values"=x,"probabilities"=bp)
> d
      x.values   probabilities
  1          0    2.146388e-01
  2          1    3.389033e-01
  3          2    2.586367e-01
  4          3    1.270496e-01
  5          4    4.513605e-02
  6          5    1.235302e-02
  7          6    2.708997e-03
  8          7    4.888415e-04
  9          8    7.396944e-05
```

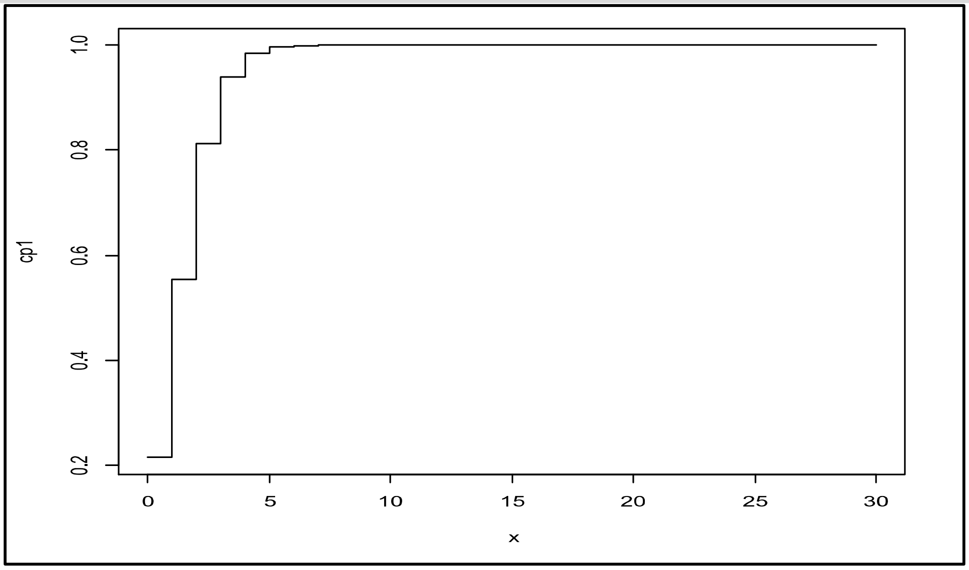
```
 10          9    9.516536e-06
 11         10    1.051828e-06
 12         11    1.006534e-07
 13         12    8.387780e-09
 14         13    6.112552e-10
 15         14    3.906518e-11
 16         15    2.193133e-12
 17         16    1.082138e-13
 18         17    4.690382e-15
 19         18    1.782894e-16
 20         19    5.926516e-18
 21         20    1.715570e-19
 22         21    4.299675e-21
 23         22    9.257674e-23
 24         23    1.694769e-24
 25         24    2.601619e-26
 26         25    3.286255e-28
 27         26    3.326169e-30
 28         27    2.593504e-32
 29         28    1.462502e-34
 30         29    5.308539e-37
 31         30    9.313226e-40
> plot (x, bp,"h")
```



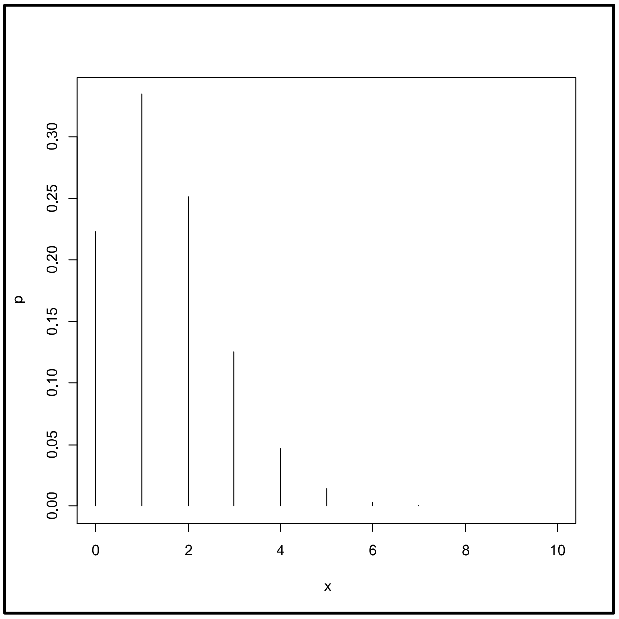
```
> cp=pbinom (x, n, p)
> cp1=round (cp,4)
> d1=data.frame(x, cp1)
> plot (x, cp1)
```



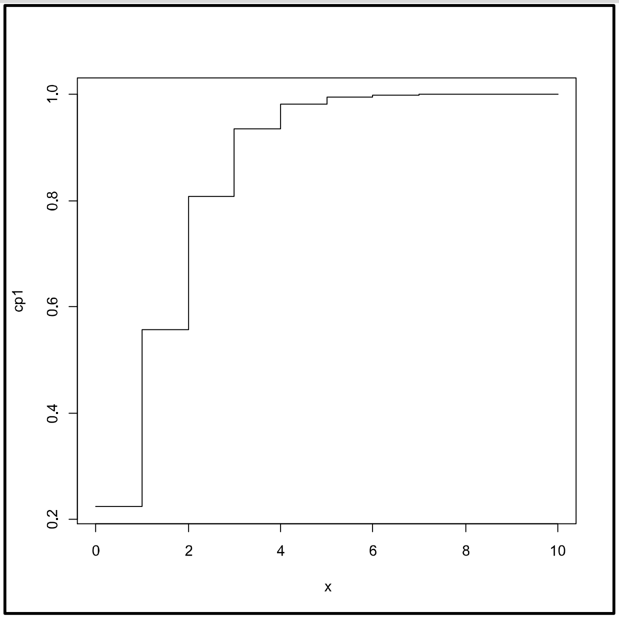
```
> plot (x, cp1,"s")
```



```
> # b]X~P (1.5) #
> m=1.5
> x=0:10
> p=dpois (x, m)
> d=data.frame(x, p)
> d
        x              p
  1     0      2.231302e-01
  2     1      3.346952e-01
  3     2      2.510214e-01
  4     3      1.255107e-01
  5     4      4.706652e-02
  6     5      1.411996e-02
  7     6      3.529989e-03
  8     7      7.564262e-04
  9     8      1.418299e-04
 10     9      2.363832e-05
 11    10      3.545748e-06
> plot (x, p,"h")
```



```
> cp=ppois (x, m)
> cp1=round(cp,4)
> d1=data.frame(x, cp1)
> plot (x, cp1,"s")
```



8) X ~ Negative Bin (r=2, P= 0.05) then compute

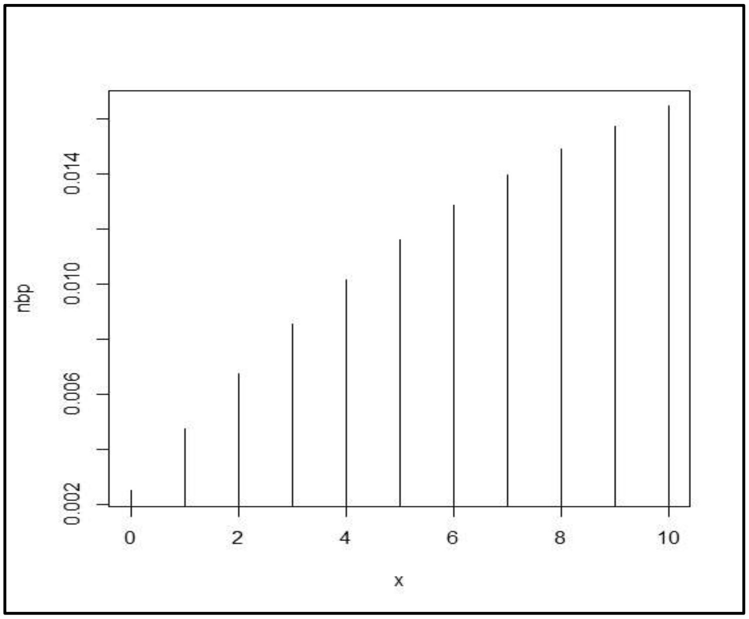
i. **P(X=0), P(X=1),P(X≤1), P(X≥2)**

ii. **Evaluate Nbinomial probabilities and plot the graph of p.m.f and c.d.f.**

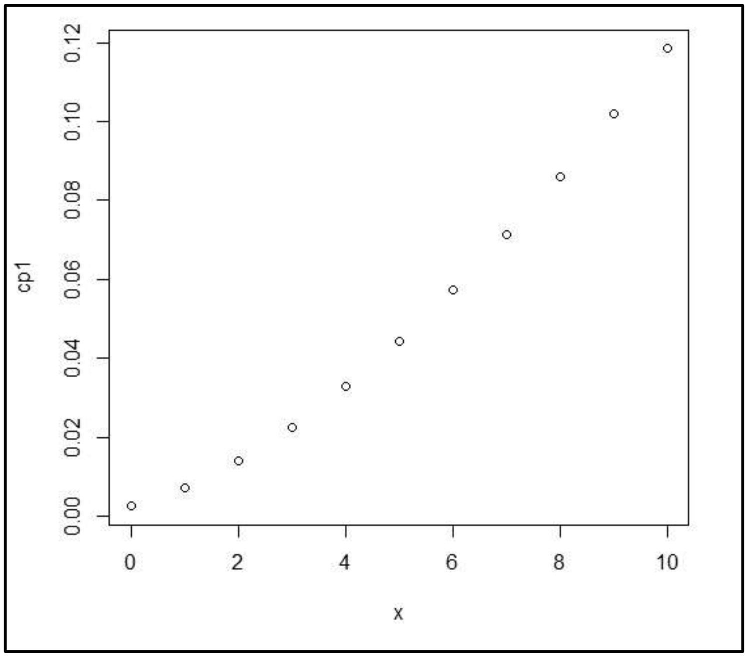
Solution:i.

```
> dnbinom(0,2,0.05)
[1] 0.0025
> dnbinom(1,2,0.05)
[1] 0.00475
> pnbinom(1,2,0.05)
[1] 0.00725
> 1-pnbinom(1,2,0.05)
[1] 0.99275
ii)
```

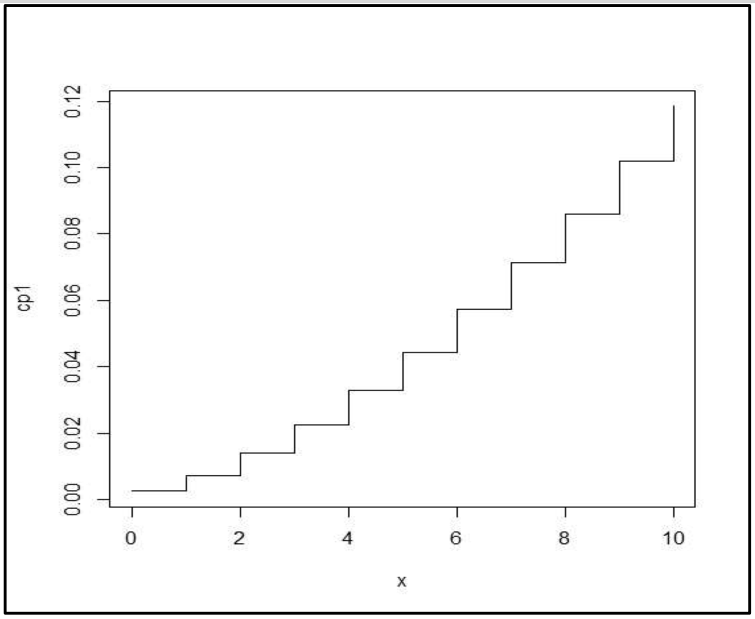
```
> p=0.05;r=2
> x=0:10
> nbp=dnbinom(x,r,p)
> d=data.frame("X-Value"=x,"Probability"=nbp)
>  d
           X.X.Value      Probability
      1            0       0.00250000
      2            1       0.00475000
      3            2       0.00676875
      4            3       0.00857375
      5            4       0.01018133
      6            5       0.01160671
      7            6       0.01286411
      8            7       0.01396675
      9            8       0.01492696
     10            9       0.01575624
     11           10       0.01646527
> plot(x,nbp,"h")
```



```
> cp1=round(cp,4)#round function round off cp values upto 4 decimal
> d1= data.frame(x,cp1)
> d1
          x       cp1
   1      0    0.0025
   2      1    0.0073
   3      2    0.0140
   4      3    0.0226
   5      4    0.0328
   6      5    0.0444
   7      6    0.0572
   8      7    0.0712
   9      8    0.0861
  10      9    0.1019
  11     10    0.1184
> plot(x,cp1) #Just points are plotted
```



```
> plot(x,cp1,"s") #It gives step function
```

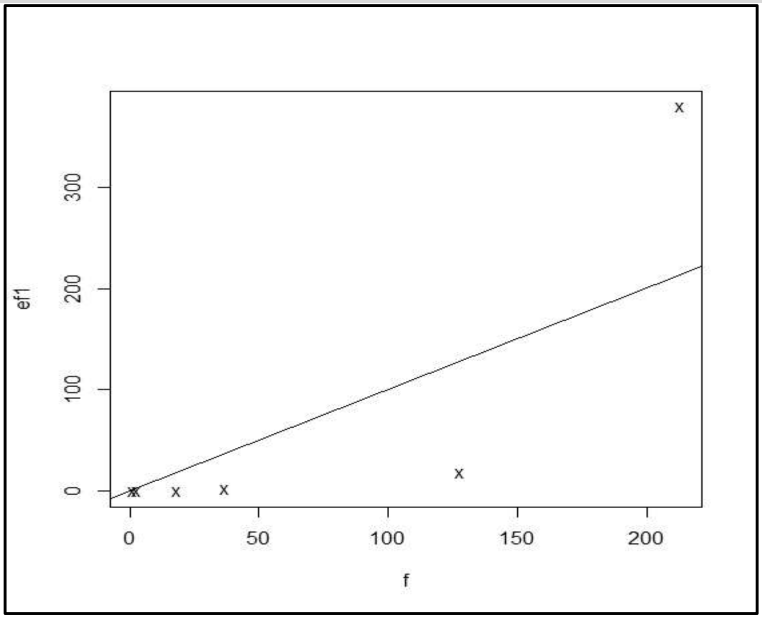


9) Fit the Negative Binomial Distribution to following data:

X:0	**1**	**2**	**3**	**4**	**5**
f: 213	**128**	**37**	**18**	**4**	**5**

Solution:

```
> x=0:5;f=c(213,128,37,18,3,1)
> m=sum(f*x)/sum(f)
> var=(sum(f*x*x)/sum(f))-m*m
> p=m/var;q=1--p;r=m*p/q
> px=dnbinom(x,r,p)
> px1=round(px,5)
> ef=sum(f)*px1
> ef1=round(ef,0)
> d=data.frame(x,f,"exp.freq."=ef1)
> d
      x     f    exp.freq
 1    0   213         379
 2    1   128          19
 3    2    37           2
 4    3    18           0
 5    4     3           0
 6    5     1           0
> plot(f,ef1,pch="x");abline(0,1) #pch gives the point markers
```



10) Let X~ N (50,40). Find P (X≤60), P(X≥100) , P(10≤X≤20) and P(X≤k)=0.293.

Solution:

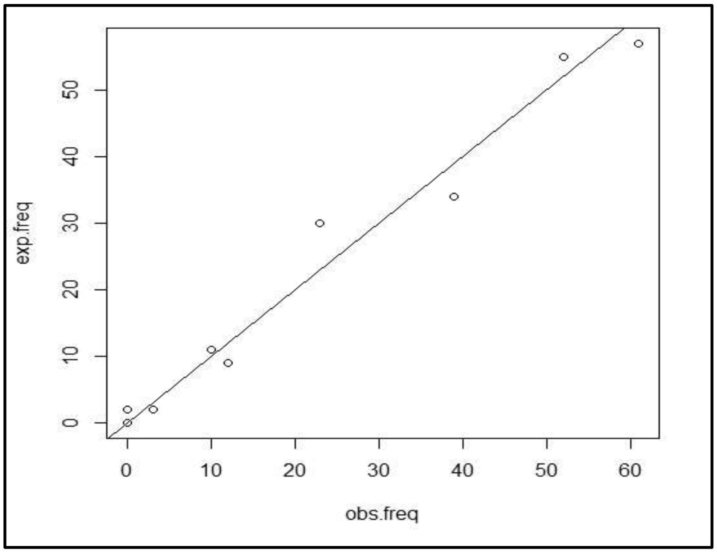
```
> mu=50; sd=sqrt(40)
> p1=pnorm(60,mu,sd)
> p1
[1] 0.9430769
> p2=1-pnorm(100,mu,sd)
> p2
[1] 1.332268e-15
> p3=pnorm(20,mu,sd)-pnorm(10,mu,sd)
> p3
[1] 1.050591e-06
> p4=qnorm(0.293,mu,sd)
> p4
[1] 46.55538
```

11) Fit a normal distribution to the following data of height (in cms) of 200 Indian adult males

Height in cms	**144-150**	**150-156**	**156-162**	**162-168**	**168-174**	**174-180**	**180-186**
No of Adults	**3**	**12**	**23**	**52**	**61**	**39**	**10**

Solution:

```
> l1=seq(144,180,6)
> u1=seq(150,186,6)
> f=c(3,12,23,52,61,39,10)
> x=(l1+u1)/2
> n=sum(f)
> k=length(f)
> m=sum(f*x)/n;v=sum(f*(x-m)^2)/n;sd=sqrt(v)
> l1=c(-9999,l1,186)
> cp=pnorm(l1,m,sd)
> p=diff(cp)
> p=c(p,1-cp[k+2])
> u1=c(144,u1,9999);f=c(0,f,0)
> ef=round(n*p,0)
>  d=data.frame("Lower  Limit"=l1,"Upper  Limit"=u1,"Obs.freq"=f,"prob"=p,"cum
prob"=cp,"expfreq"=ef)
> d
    Lower.Limit  Upper.Limit  Obs.freq  prob          cum.prob      expfreq
 1        -9999          144         0  0.0009277682  0.0000000000        0
 2          144          150         3  0.0085408285  0.0009277682        2
 3          150          156        12  0.0474590553  0.0094685967        9
 4          156          162        23  0.1504843558  0.0569276520       30
 5          162          168        52  0.2727415211  0.2074120077       55
 6          168          174        61  0.2828190953  0.4801535289       57
 7          174          180        39  0.1677990586  0.7629726242       34
 8          180          186        10  0.0569156032  0.9307716828       11
 9          186         9999         0  0.0123127140  0.9876872860        2
> plot(f,ef,xlab="obs.freq",ylab="exp.freq","p")
> abline(0,1)
```

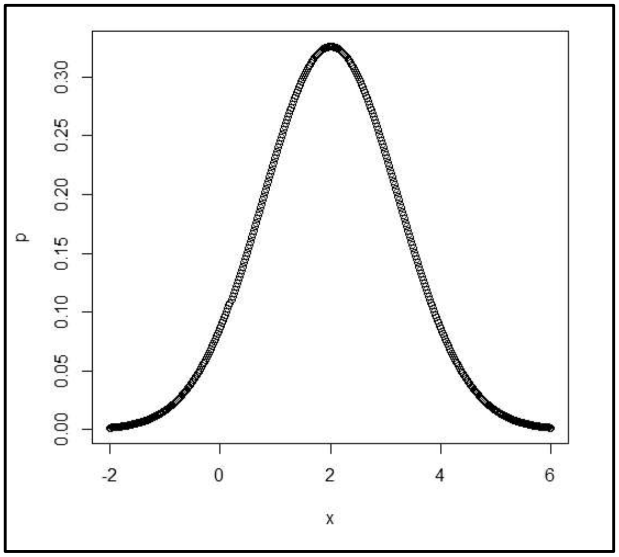


12) Find a) P(X ≤ 0.8) b) P (X > 0.5)

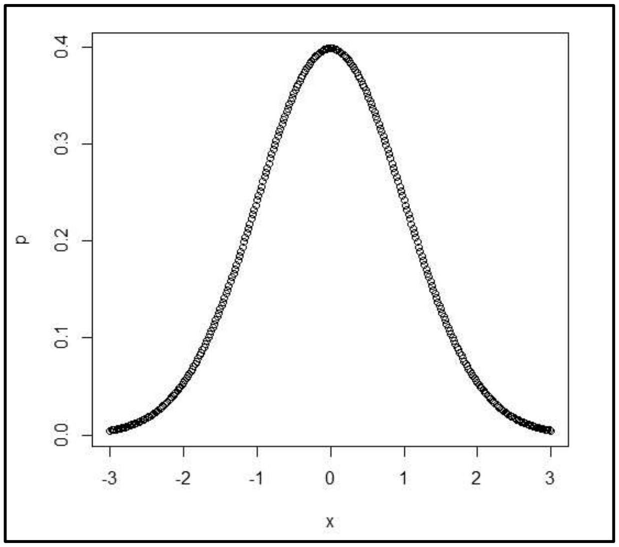
If, i. $X\sim\text{Normal}(2,1.5)$ ii. $X\sim\text{Normal}(0,1)$ iii. $X\sim\text{Exp}(1.5)$ iv. $X\sim\text{beta}(2,1.5)$
v. $X\sim\text{Gamma}(2,1.5)$ vi. $X\sim\text{ChiSq}(10)$ vii. $X\sim t(8)$ viii. $X\sim F(10,10)$
ix. $X\sim U(0,5)$

Solution:

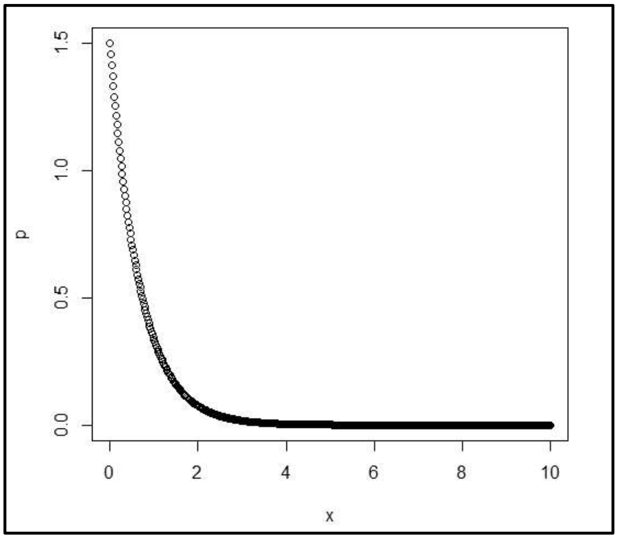
```
> a=pnorm(0.8,2,sqrt(1.5),lower.tail=1)
> a
[1] 0.1635934
> b=pnorm(0.5,2,sqrt(1.5),lower.tail=0)
> b
[1] 0.8896643
> x=seq(-2,6,by=0.02)
> p=dnorm(x,2,sqrt(1.5))
> plot(x,p)
```



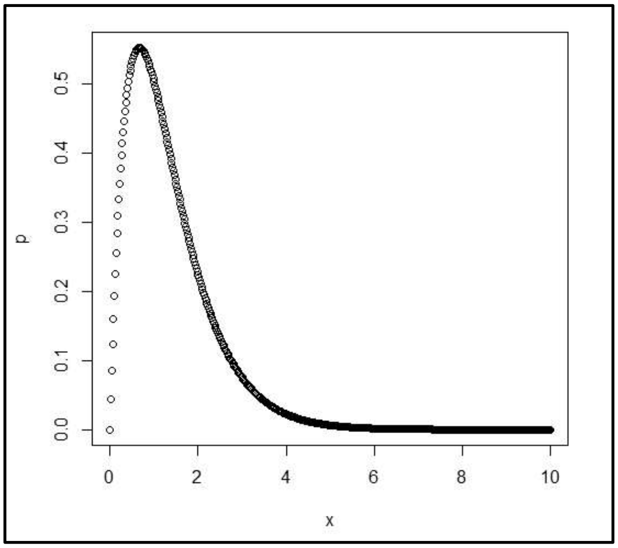
```
> a=pnorm(0.8,0,sqrt(1),lower.tail=1)
> a
[1] 0.7881446
> b=pnorm(0.5,0,sqrt(1),lower.tail=0)
> b
[1] 0.3085375
> x=seq(-3,3,by=0.02)
> p=dnorm(x,0,sqrt(1))
> plot(x,p)
```



```
> a=pexp(0.8,1.5,lower.tail=1)
> a
[1] 0.6988058
> b=pexp(0.5,1.5,lower.tail=0)
> b
[1] 0.4723666
> x=seq(0,10,by=0.02)
> p=dexp(x,1.5)
> plot(x,p)
```

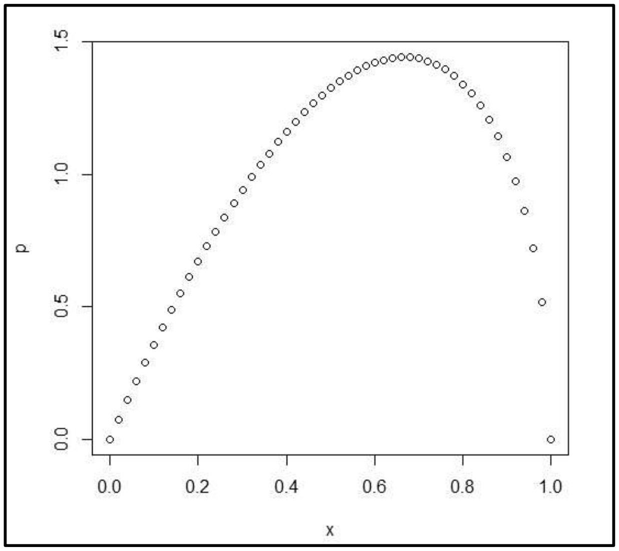


```
> a=pgamma(0.8,2,1.5)
> a
[1] 0.3373727
> b=pgamma(0.5,2,1.5,lower.tail=0)
> b
[1] 0.8266415
> x=seq(0,10,by=0.02)
> p=dgamma(x,2,1.5)
> plot(x,p)
```

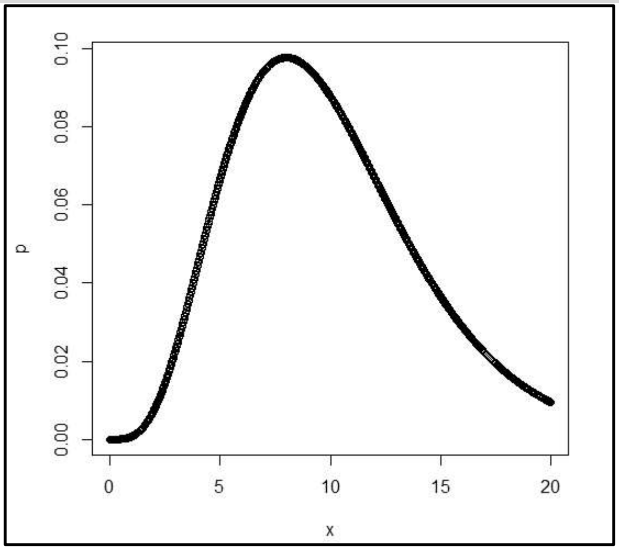


```
> a=pbeta(0.8,2,1.5)
> a
[1] 0.803226
```

```
> b=pbeta(0.5,2,1.5,lower.tail=0)
> b
[1] 0.6187184
> x=seq(0,1,by=0.02)
> p=dbeta(x,2,1.5)
> plot(x,p)
```

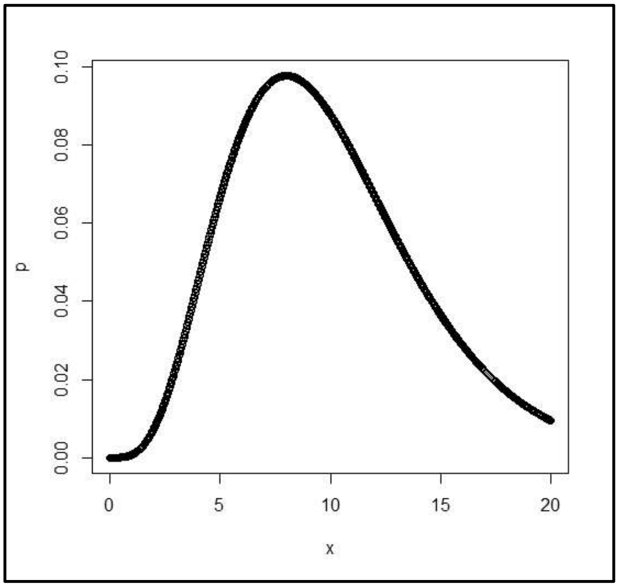


```
> a=pchisq(0.8,10)
> a
[1] 6.124333e-05
> b=pchisq(0.5,10,lower.tail=0)
> b
[1] 0.9999934
> x=seq(0,20,by=0.02)
> p=dchisq(x,10)
> plot(x,p)
```

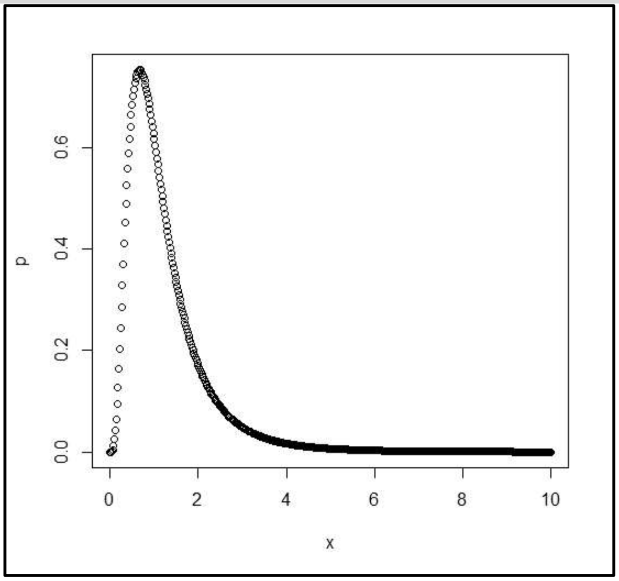


```
> a=pt(0.8,8)
> a
```

```
[1] 0.7765933
> b=pt(0.5,8,lower.tail=0)
> b
[1] 0.315268
> x=seq(-10,10,by=0.02)
> p=dt(x,8)
```

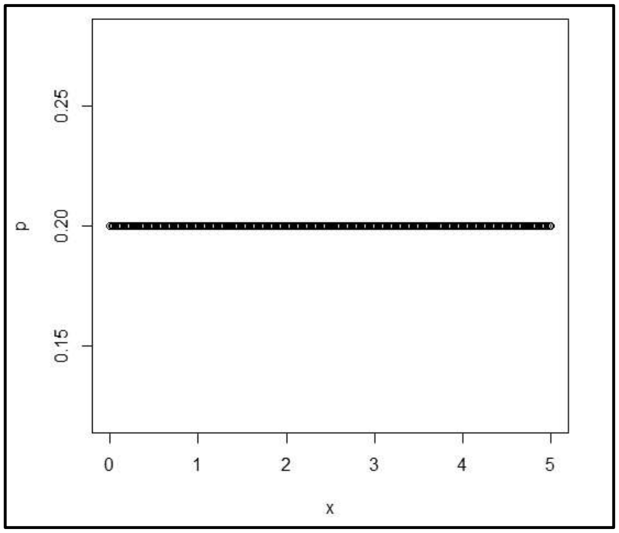


```
> a=pf(0.8,10,10)
> a
[1] 0.3655069
> b=pf(0.5,10,10,lower.tail=0)
> b
[1] 0.8551542
> x=seq(0,10,by=0.02)
> p=df(x,10,10)
> plot(x,p)
```



```
> a=punif(0.8,0,5)
> a
[1] 0.16
```

```
> b=punif(0.5,0,5,lower.tail=0)
> b
[1] 0.9
> x=seq(0,5,by=0.02)
> p=dunif(x,0,5)
> plot(x,p)
```



Chapter 8

Sampling Distribution and Central Limit Theorem using R

Dr. Rajendra Nana Chavhan, Assistant Professor Department of Statistics,
K. C. College, Churchgate, Mumbai – 400 020.

8.1 Introduction

In this chapter, I have demonstrated the sampling distribution of some well-known statistics as sample mean, sample variance and sample median. I used Poisson, Normal and Exponential distributions. I have also demonstrated the central limit theorem using sampling distributions.

8.2 Sampling Distribution

The sampling distribution of statistic is the distribution of statistic, considered as a random variable, when derived from random sample of size n. It may be considered as distribution of the statistic for all possible random samples from the same population of a given size. I have demonstrated sampling distribution of

1. Sample mean of discrete random variable with probability function
2. Sample mean of $X \sim Exp(1.2)$
3. Sample variance of $X \sim N(5,9)$
4. Sample median where $X \sim Poisson(3.1)$

One can extend the study of sampling distributions with other sample statistic and distributions. This sampling distributions can be used for determining empirical probabilities.

Procedure for studying the sampling distribution
I used the simulation technique for studying the sampling distribution of different statistic using well known discrete as well as continuous probability distributions. I used sample size $n = 5, 10, 25$ and 50, and 1000 repetitions. I used following steps

Step 1. Drawing of random sample from considered population.

Step 2. Calculation of sample statistic for different sample size ($n = 5, 15, 25$ and 50)

Step 3. Comparison of population value with expected value of sample statistic for different sample size ($n = 5, 15, 25$ and 50) i.e. comparison of mean.

Step 4. Comparison of variation of sample statistic for different sample size ($n = 5, 15, 25$ and 50) by studying variance.

Step 5. Drawing of histogram for overall comparison.

8.2.1 Sampling distribution of sample mean of discrete random variable with probability function

Consider the following probability distribution

X	:	0	1	2	3
$P(X = x)$	:	0.1	0.4	0.3	0.2

Here $E(X) = 1.6$ and $Var(X) = 0.84$, we study the sampling distribution of sample mean. We now that $E(\bar{X}) = 1.6$ and $Var(\bar{X}) = \frac{0.84}{n}$. I have written R-Program 1 for studying the sampling distribution of sample mean for above discrete probability distribution.

R-Program 1: R code for studying Sampling distribution of sample mean of discrete random variable

```
set.seed(1)    #for producing the same sequence of random variable every time
n=50;                           #sample size
rep=1000;                       #repetitions
xv=c(0,1,2,3)                   #X values
prob=c(0.1,0.4,0.3,0.2)         #Probability Values
#random sample from Discrete Distribution
x1=sample(xv,n*rep,replace = TRUE,prob=prob);
x=matrix(x1,rep,n)              #arrangement of random numbers in matrix
s.mean5=rowMeans(x[,1:5])       #sample mean n=5
s.mean10=rowMeans(x[,1:10])     #sample mean n=10
s.mean25=rowMeans(x[,1:25])     #sample mean n=25
s.mean50=rowMeans(x[,1:50])     #sample mean n=50
s.mean=data.frame(s.mean5,s.mean10,s.mean25,s.mean50)  #bind all means
apply(s.mean,2,mean);apply(s.mean,2,var)   #Calculation of mean and variance
par(mfrow=c(2,2));
hist(s.mean5,xlab = "(a)",main="n=5");
hist(s.mean10,xlab = "(b)",main="n=10");
hist(s.mean25,xlab = "(c)",main="n=25");
hist(s.mean50,xlab = "(d)",main="n=50")
```

We put the numerical output of R-Program 1, i.e. five point summary, mean and variance of sample mean of sizes $n = 5, 15, 25$ and 50 in the Table 1.

Table 1: Descriptive statistics of sample mean of discrete distribution

Sample size(n)	Minimum	Q1	Q2	Mean	Q3	Maximum	Variance
5	0.40	1.20	1.60	1.572	1.80	2.80	0.1719
10	0.80	1.40	1.60	1.587	1.80	2.50	0.0898

25	1.12	1.44	1.60	1.587	1.72	2.20	0.0358
50	1.14	1.52	1.60	1.600	1.68	2.04	0.0173

One can see that as sample increases mean of sample mean approaches to population mean and variances approaches to $\frac{0.84}{n}$. We can also see the shape of the sample means for considered sample sizes from Figure 1.

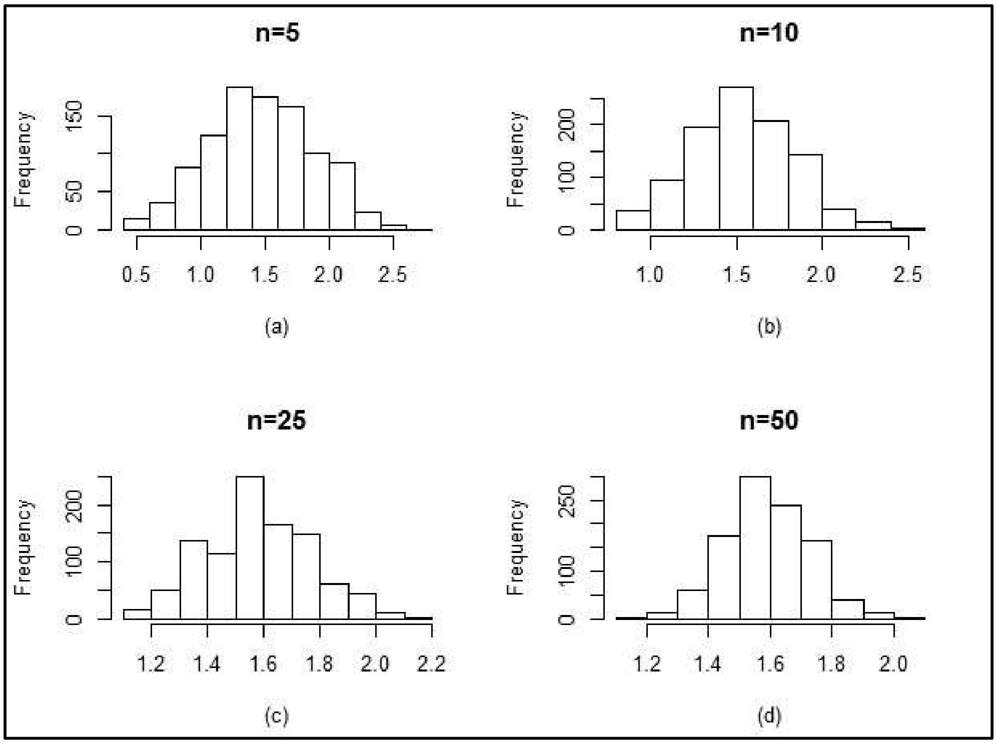


Figure 1: Sampling distribution of sample mean of discrete probability distribution for sample size (a) $\mathbf{n = 5}$ (b) $\mathbf{n = 10}$ (c) $\mathbf{n = 25}$ and (d) $\mathbf{n = 50}$.

One can observe the overall shape, changing pattern of shape, variation, outliers, Skewness, outliers etc. of sample mean from Figure 1. We can conclude that mean of sample mean is concentrating towards the population mean $E(X) = 1.6$ whereas variation decreases.

8.2.2 Sampling distribution of sample mean where $X \sim N(10, 4)$

Here I studied the sampling distribution of sample mean where parent population is normal with mean 10 and variance 4. I have written the following R-Program 2 for studying Sampling distribution of sample mean where $X \sim N(10,4)$.

R-Program 2: R code for studying Sampling distribution of sample mean of $X \sim N(10, 4)$

```
set.seed(25)    #for producing the same sequence of random variable everytime
n=50;                          #sample size
rep=1000;                      #repetitions
x1=rnorm(rep*n,10,2);          #random sample from Population N(10,4)
x=matrix(x1,rep,n)             #arrangement of random numbers in matrix
```

```
s.mean5=rowMeans(x[,1:5])       #sample mean n=5
s.mean10=rowMeans(x[,1:10])     #sample mean n=10
s.mean25=rowMeans(x[,1:25])     #sample mean n=25
s.mean50=rowMeans(x[,1:50])     #sample mean n=50
s.mean=data.frame(s.mean5,s.mean10,s.mean25,s.mean50)  #bind all means
summary(s.mean) #gives six point summary(min,Q1,Q2,mean,Q3 and max)
apply(s.mean,2,var) #Calculation of Variance
par(mfrow=c(2,2));
hist(s.mean5,xlab = "(a)",main="n=5");
hist(s.mean10,xlab = "(b)",main="n=10");
hist(s.mean25,xlab = "(c)",main="n=25");
hist(s.mean50,xlab = "(d)",main="n=50")
```

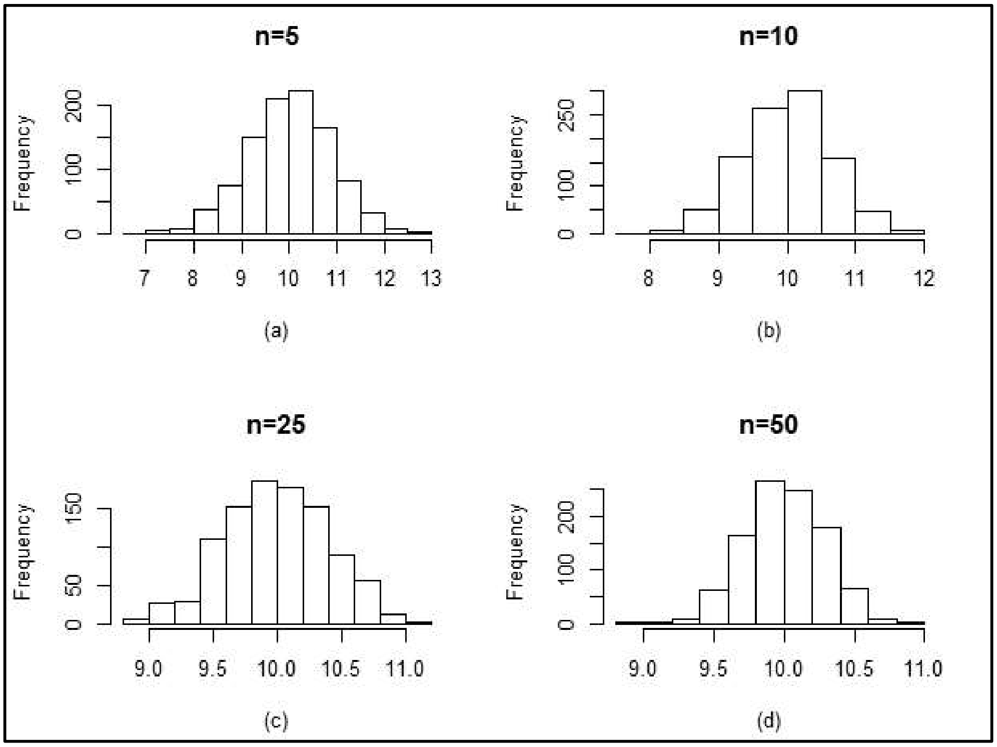


Figure 2: Sampling distribution of sample mean of $X \sim N(10,4)$ for sample of sizes (a) n = 5 (b) n = 10 (c) n = 25 and (d) n = 50.

Figure 2 shows the histogram for sample mean of sizes (a) n = 5 (b) n = 10 (c) n = 25 and (d) n = 50 where parent population is $N(10,4)$. One can observe the frequency distribution, overall shape of sample mean of normal distribution having mean 10 and variance 4. As sample size increases, sample mean gets closer to population mean with decrement in variances and spread. This can be confirmed from descriptive statistics given in Table 2. Numerical output of R-Program 2, i.e. five point summary, mean and variance of sample mean of sizes $n = 5, 15, 25$ and 50 is given in the Table 2.

Table 2: Descriptive statistics of sample mean of $N(10,4)$

Sample size(n)	Minimum	Q1	Q2	Mean	Q3	Maximum	Variance
5	6.630	9.388	10.030	10.005	10.598	12.763	0.817
10	7.625	9.549	10.016	9.994	10.432	11.813	0.407

25	8.827	9.716	9.993	9.989	10.278	11.050	0.156
50	8.961	9.812	10.000	10.002	10.204	10.971	0.076

8.2.3 Sampling distribution of sample variance where $X \sim Exp(1.2)$

Here I studied the sampling distribution of sample variance where sample is drawn from exponential distribution with parameter 1.2. I have written the following R-Program 3 for studying sampling distribution of sample variances where $X \sim Exp(1.2)$

R-Program 3: R code for studying Sampling distribution of sample variance of $X \sim Exp(1.2)$

```
set.seed(25)   #for producing the same sequence of random variable every time
n=50;                  #sample size
rep=1000;              #repetitions
x1=rexp(rep*n,1.2);#random sample from Population Exponential with mean=1/1.2
x=matrix(x1,rep,n);              #arrangement of random numbers in matrix
s.var5=apply(x[,1:5],1,var);         #sample variance n=5
s.var10=apply(x[,1:10],1,var);       #sample variance n=10
s.var25=apply(x[,1:25],1,var);       #sample variance n=25
s.var50=apply(x[,1:50],1,var);       #sample variance n=50
s.var=data.frame(s.var5,s.var10,s.var25,s.var50)  #bind all variances
summary(s.var)  #gives six point summary(min,Q1,Q2,mean,Q3 and max)
apply(s.var,2,var) #Calculation of Variance
par(mfrow=c(2,2));
hist(s.var5,xlab = "(a)",main="n=5");
hist(s.var10,xlab = "(b)",main="n=10");
hist(s.var25,xlab = "(c)",main="n=25");
hist(s.var50,xlab = "(d)",main="n=50")
```

Numerical output of R-Program 3, i.e. five point summary, mean and variance of sample variance of $Exp(1.2)$ of sizes $n = 5, 15, 25$ and 50 is given in the Table 3.

Table 3: Descriptive statistics of sample variance of $Exp(1.2)$

Sample size(n)	Minimum	Q1	Q2	Mean	Q3	Maximum	Variance
5	0.004	0.182	0.398	0.692	0.829	14.281	0.914
10	0.029	0.301	0.500	0.678	0.852	7.894	0.405
25	0.128	0.416	0.613	0.697	0.844	3.696	0.171
50	0.194	0.494	0.649	0.695	0.831	2.267	0.080

Figure 3 shows the histogram of sample variance of sizes (a) $\mathrm{n} = 5$ (b) $\mathrm{n} = 10$ (c) $\mathrm{n} = 25$ and (d) $\mathrm{n} = 50$ where parent population is exponential with parameter 1.2. From Figure 3, one can see that distribution of sample variance is positively skewed.

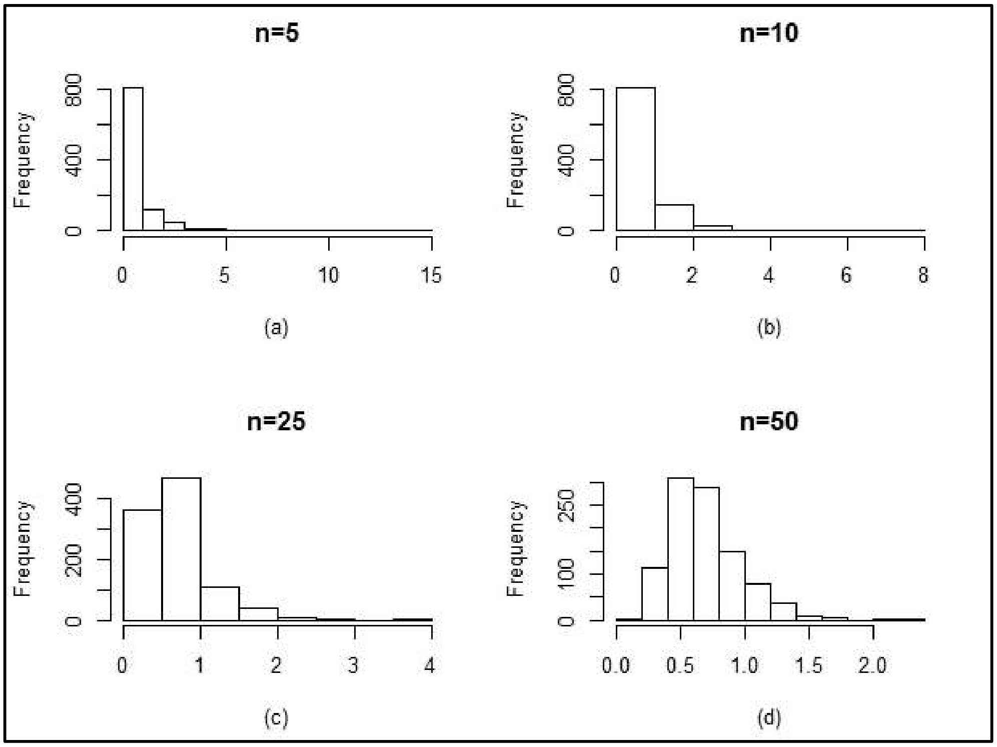


Figure 3: Sampling distribution of sample variance of $X \sim Exp(1.2)$ for sample of sizes (a) n = 5 (b) n = 10 (c) n = 25 and (d) n = 50.

8.2.4 Sampling distribution of sample median where $X \sim Pois(3.1)$:

Here I studied the sampling distribution of sample median where sample is drawn from Poisson distribution with mean 3.1. I have written the following R-Program 4 for studying sampling distribution of sample median where $X \sim Pois(3.1)$.

R-Program 4: R code for studying sampling distribution of sample median of $X \sim Pois\ (3.1)$

```
set.seed(25)   #for producing the same sequence of random variable every time
n=50;                  #sample size
rep=1000;              #repetitions
x1=rpois(rep*n,3.1);    #random sample from Population Poisson with mean=3.1
x=matrix(x1,rep,n);              #arrangement of random numbers in matrix
s.med5=apply(x[,1:5],1,median);       #sample median n=5
s.med10=apply(x[,1:10],1,median);       #sample median n=10
s.med25=apply(x[,1:25],1,median);       #sample median n=25
s.med50=apply(x[,1:50],1,median);       #sample median n=50
s.med=data.frame(s.med5,s.med10,s.med25,s.med50)  #bind all Medians
summary(s.med)  #gives six point summary(min,Q1,Q2,mean,Q3 and max)
apply(s.med,2,var) #Calculation of Variance
par(mfrow=c(2,2));
hist(s.med5,xlab = "(a)",main="n=5");
hist(s.med10,xlab = "(b)",main="n=10");
hist(s.med25,xlab = "(c)",main="n=25");
hist(s.med50,xlab = "(d)",main="n=50")
```

Table 3 contains the descriptive statistics of sample median for different sample sizes obtained from numerical output of R-Program 4.

Table 3: Descriptive statistics of sample median of $\boldsymbol{Poiss}(\mathbf{3.1})$

Sample size(n)	Minimum	Q1	Q2	Mean	Q3	Maximum	Variance
5	1	2	3	2.976	4	6	0.9624
10	1	2.5	3	2.944	3.5	5.5	0.4783
25	2	3	3	2.924	3	5	0.2605
50	2	3	3	2.943	3	4	0.1082

Figure 4 shows histogram of sample median of Poisson with mean 3.1 which shows frequency distribution of sample median.

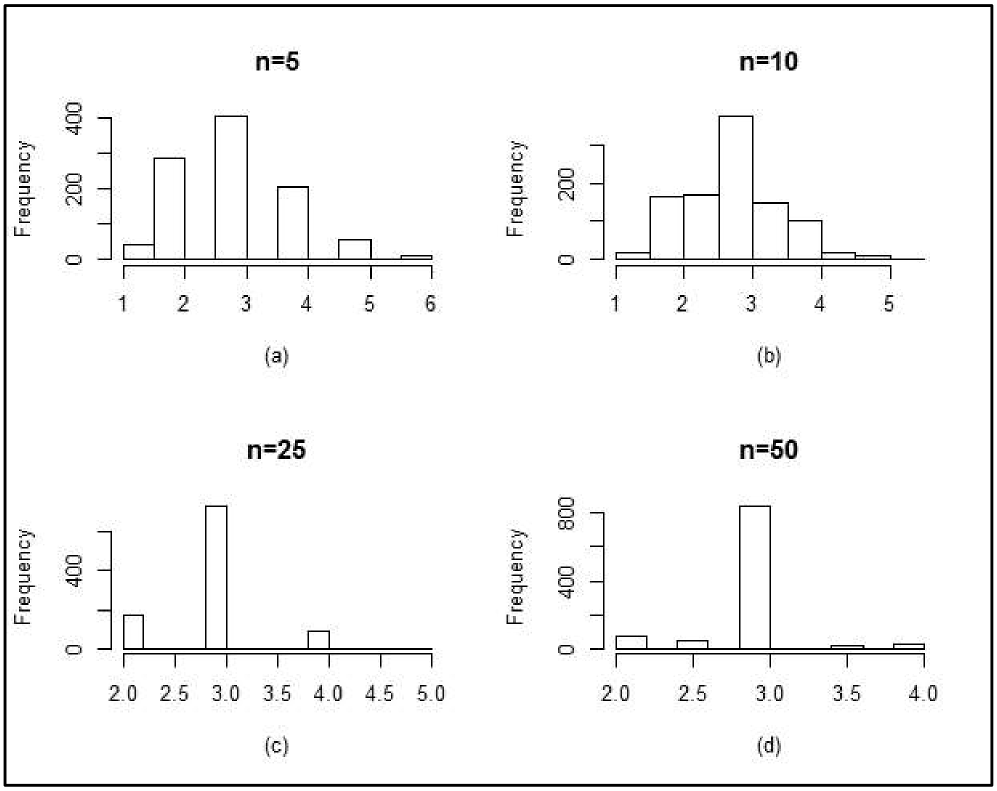


Figure 4: Sampling distribution of sample median of $X \sim \boldsymbol{Pois}(\mathbf{3.1})$ for sample of sizes (a) n = 5 (b) n = 10 (c) n = 25 and (d) n = 50.

8.3 Central Limit Theorem (CLT)

If $X_1, X_2, \ldots\ldots X_n$ is a random sample of size n (large) from any probability distribution (either discrete or continuous) with finite mean μ and variance σ^2then sample mean $\bar{X} = \frac{\sum_{i=1}^{n} x_i}{n}$ will tends to normal distribution with mean μ and variance $\frac{\sigma^2}{n}$. Here I demonstrated the CLT for the following probability distributions

1. Negative Binomial Distribution
2. Continuous Uniform Distribution

I used $n = 10, 50$, 100 and 250 for demonstration. Shapiro test is used to test normality. I have also plot histogram along with normal curve to asses the normality.

8.3.1 Negative Binomial Distribution

Consider $X_1, X_2 \ldots X_n$ is random sample from negative binomial with $k = 5$ and $p = 0.7$. Here X represents the number of failure before k sucusses. I have written the following R-Program 5 for studying sampling distribution of sample mean and to demonstrate the CLT where $X \sim NB(5{,}0.7)$.

R-Program 5: R code for demonstration of CLT of $\boldsymbol{X \sim NB(5, 0.7)}$

```
set.seed(5)    #for producing the same sequence of random variable every time
n=250;                  #sample size
rep=1000;              #repetitions
x1=rnbinom(rep*n,5,0.7);   #random sample from Negative Binomial k=5, p=0.7
x=matrix(x1,rep,n);               #arrangement of random numbers in matrix
s.mean10=apply(x[,1:10],1,mean);       #sample mean n=10
s.mean50=apply(x[,1:50],1,mean);       #sample mean n=50
s.mean100=apply(x[,1:100],1,mean);       #sample mean n=100
s.mean250=apply(x[,1:250],1,mean);       #sample mean n=250
nt10=shapiro.test(s.mean10);      #Normality test of sample mean n=10
nt50=shapiro.test(s.mean50);      #Normality test of sample mean n=50
nt100=shapiro.test(s.mean100);      #Normality test of sample mean n=100
nt250=shapiro.test(s.mean250);      #Normality test of sample mean n=250
#P-value of the normality test
print(c(nt10$p.value,nt50$p.value,nt100$p.value,nt250$p.value))
#Function from plotting Histogram with Normal curve
hist_curve<-function(x){
  N=length(x);H=hist(x,breaks=50,xlab="",main="");dx=(H$breaks[2]-
H$breaks[1]);
  x0=H$breaks;x1=c(x0[1]-dx/2,x0+dx/2);
  lines(x1,N*dnorm(x1,mean(x),sd(x))*dx,col="blue")
}
par(mfrow=c(2,2));
hist_curve(s.mean10);title(main="n=10",xlab="(a)");
hist_curve(s.mean50);title(main="n=50",xlab="(b)");
hist_curve(s.mean100);title(main="n=100",xlab="(c)");
hist_curve(s.mean250);title(main="n=250",xlab="(d)");
```

Table 5 shows the P-value of Shapiro test of normality.

Table 5: P-value for Shapiro test of normality

Sample size(n)	10	50	100	250
P-value	0.0000	0.1241	0.3139	0.7999

CLT hold for $n = 50, 100, 250$ which can be confirmed from P-value given in Table 5. In Figure 5, I used to draw histogram with normal curve. One can see the normal curve fits well for (b) n=50, (c) n=100 and (d) n=250. As sample size increases normal curve fits well.

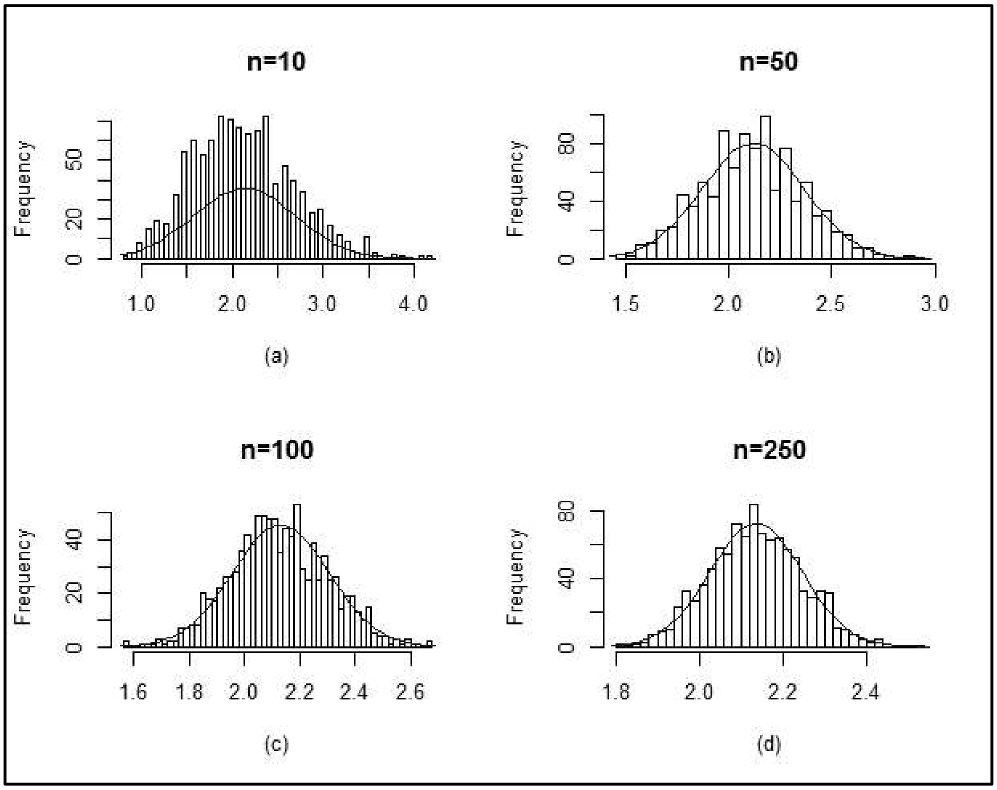


Figure 5: Sampling distribution of sample mean with normal curve of $X\sim NB(5, 0.7)$ for sample of sizes (a) $\mathbf{n = 10}$ (b) $\mathbf{n = 50}$ (c) $\mathbf{n = 100}$ and (d) $\mathbf{n = 250}$.

8.3.2 Continuous uniform distribution

Consider $X_1, X_2 \ldots\ X_n$ is random sample from continuous uniform distribution in the interval $(0, 10)$. I have written the following R-Program 6 for studying sampling distribution of sample mean and to demonstrate the CLT where $X\sim U(0, 10)$.

R-Program 6: R code for demonstration of CLT of $X\sim U(0\ , 10)$

```
set.seed(50)   #for producing the same sequence of random variable every time
n=250;                  #sample size
rep=1000;              #repetation
x1=runif(rep*n,0,10);   #random sample from Negative Binomial k=5, p=0.7
x=matrix(x1,rep,n);               #arrangment of random numbers in matrix
s.mean10=apply(x[,1:10],1,mean);        #sample mean n=10
s.mean50=apply(x[,1:50],1,mean);        #sample mean n=50
s.mean100=apply(x[,1:100],1,mean);        #sample mean n=100
s.mean250=apply(x[,1:250],1,mean);        #sample mean n=250
nt10=shapiro.test(s.mean10);      #Normality test of sample mean n=10
nt50=shapiro.test(s.mean50);      #Normality test of sample mean n=50
nt100=shapiro.test(s.mean100);      #Normality test of sample mean n=100
nt250=shapiro.test(s.mean250);      #Normality test of sample mean n=250
```

```
p.value=c(nt10$p.value,nt50$p.value,nt100$p.value,nt250$p.value)  #P-value of
the normality test
#Function from plotting Histogram with Normal curve
hist_curve<-function(x){
  N=length(x);H=hist(x,breaks=50,xlab="",main="");dx=(H$breaks[2]-
H$breaks[1]);
  x0=H$breaks;x1=c(x0[1]-dx/2,x0+dx/2);
  lines(x1,N*dnorm(x1,mean(x),sd(x))*dx,col="blue")
}
par(mfrow=c(2,2));
hist_curve(s.mean10);title(main="n=10",xlab="(a)");
hist_curve(s.mean50);title(main="n=50",xlab="(b)");
hist_curve(s.mean100);title(main="n=100",xlab="(c)");
hist_curve(s.mean250);title(main="n=250",xlab="(d)")
```

Table 6 shows the P-value of Shapiro test of normality.

Table 6: P-value for Shapiro test of normality

Sample size(n)	10	50	100	250
P-value	0.0348	0.2414	0.2984	0.3321

CLT hold for $n = 50, 100, 250$ which can be confirmed from P-value given in Table 6. In Figure 6, I used to draw histogram with normal curve. One can see the normal curve fits well for (b) n=50, (c) n=100 and (d) n=250. As sample size increases normal curve fits well.

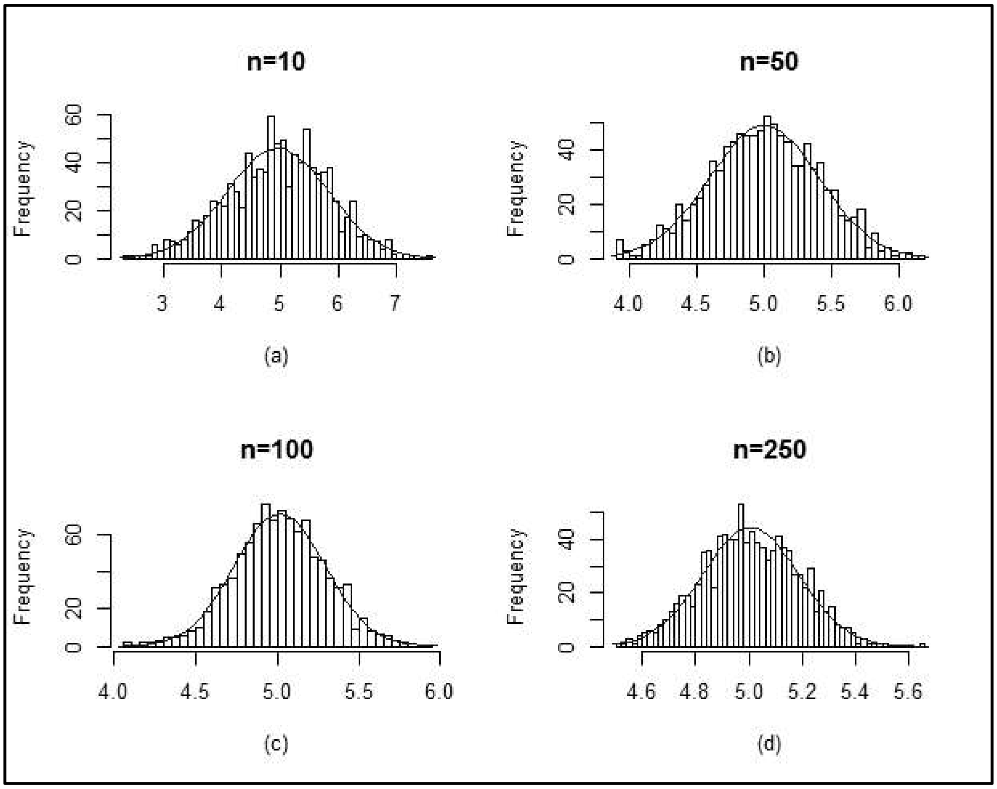


Figure 6: Sampling distribution of sample mean with normal curve of $X \sim U(0, 10)$ for sample of sizes (a) $\mathrm{n} = 10$ (b) $\mathrm{n} = 50$ (c) $\mathrm{n} = 100$ and (d) $\mathrm{n} = 250$.

8.4 Some important notes

- One can extend the study of sampling distributions with other sample statistic and distributions.
- This sampling distributions can be used for determining empirical probabilities.
- One can verify the other results like CLT.
- Sampling distributions of complicated statistic can be studied.

8.5 References

- Verzani, J. (2014). *Using R for introductory statistics*. CRC Press.

Chapter 9

Statistical Tests Using R

Dr. Rajendra Nana Chavhan, Assistant Professor, Department of Statistics,
K. C. College, Churchgate, Mumbai – 400 020.

9.1 Introduction

In this chapter, I have demonstrated the one sample t-test, two sample t-test, paired t-test, chi-square test for variance, F-test for equality of two variances with example in R programming. This article is useful for students, teachers and researchers in applied sciences.

9.2 t-test

9.2.1 One sample t-test

One sample t-test is used to investigate whether population mean (μ) is regarded as some specified value μ_o, based on a random sample. That is, to test the significance of the difference between the sample mean ($\bar{X}$) and the assumed population mean μ_o. We assume population from which, the sample of size n drawn is Normal distribution whose population mean is unknown. We test one of the following null hypothesis (H_0) and alternative hypothesis (H_1) at α level of significance.

a) H_0 : There is no significant difference between the sample mean $\bar{X}$ and the assumed population mean μ. i.e., $H_0 : \mu = \mu_0$ vs $H_1 : \mu \neq \mu_0$
b) $H_0 : \mu \leq \mu_0$ vs $H_1 : \mu > \mu_0$
c) $H_0 : \mu \geq \mu_0$ vs $H_1 : \mu < \mu_0$

The test statistic for testing the above hypothesis is

$$t = \frac{\bar{X} - \mu}{\hat{\sigma}/\sqrt{n}}$$

Where $\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$ and $\hat{\sigma}^2 = S^2 = \frac{\sum_{i}^{n}(X_i - \bar{X})^2}{n-1}$

Under H_0, the test statistic follows t distribution with $(n-1)$ degrees of freedom. We take the decision whether to reject the null hypothesis or not based on P-value. If P-value $< \alpha$ then we rejects the null hypothesis and if P-value $\geq \alpha$ then we does not enough evidence to reject the null hypothesis. The P-value is calculated as

For a) $H_1 : \mu \neq \mu_0$, P-value=$2 \times P(T > |t|)$
b) $H_1 : \mu > \mu_0$, P-value=$P(T > t)$
c) $H_1 : \mu < \mu_0$, P-value=$P(T < t)$

where T follows t distribution with $(n-1)$ degrees of freedom.

9.2.2 Two sample t-test

Two sample t-test is used to investigate the null hypothesis of the difference between mean of the two populations is some constant value, based on two random samples. We assume that the populations from which, the two samples drawn, are Normal distributions which have unknown and same variance. A random sample of size m observations $X_1, X_2, \dots, X_m$ be drawn from population with unknown mean μ_1 and a random sample of size n observations $Y_1, Y_2, \dots, Y_n$ be drawn from population with unknown mean μ_2. We assume that both the populations have equal variances. We test one of the following null hypothesis (H_0) and alternative hypothesis (H_1) at α level of significance.

a) H_0 : The difference between two population mean is some constant value c. i.e. H_0: $\mu_1 - \mu_2 = c$ vs $H_1 : \mu_1 - \mu_2 \neq c$
b) H_0: $\mu_1 - \mu_2 \leq c$ vs $H_1 : \mu_1 - \mu_2 > c$
c) H_0: $\mu_1 - \mu_2 \geq c$ vs $H_1 : \mu_1 - \mu_2 < c$

The test statistic for testing the above hypothesis is

$$t = \frac{(\bar{X} - \bar{Y}) - (\mu_1 - \mu_2)}{S \times \sqrt{\left(\frac{1}{m} + \frac{1}{n}\right)}}$$

where $\bar{X} = \frac{\sum_{i=1}^{m} X_i}{m}, \bar{Y} = \frac{\sum_{i=1}^{n} Y_i}{n}$ and $S^2 = \frac{\sum_{i=1}^{m}(X_i - \bar{X})^2 + \sum_{i=1}^{n}(Y_i - \bar{Y})^2}{m+n-2}$

Under H_0, the test statistic follows t distribution with $(m+n-2)$ degrees of freedom. The P-value is calculated as

For a) $H_1 : \mu_1 - \mu_2 \neq c$, P-value=$2 \times P(T > |t|)$
b) $H_1 : \mu_1 - \mu_2 > c$, P-value=$P(T > t)$
c) $H_1 : \mu_1 - \mu_2 < c$, P-value=$P(T < t)$

where T follows t distribution with $(m+n-2)$ degrees of freedom.

If the assumption of equality of variance of two samples does not hold then the test statistics for testing the null hypothesis is

$$t = \frac{(\bar{X} - \bar{Y}) - (\mu_1 - \mu_2)}{\sqrt{\left(\frac{S_1}{m} + \frac{S_2}{n}\right)}}$$

where $\bar{X} = \frac{\sum_{i=1}^{m} X_i}{m}$, $\bar{Y} = \frac{\sum_{i=1}^{n} Y_i}{n}$, $S_1^2 = \frac{\sum_{i=1}^{m}(X_i - \bar{X})^2}{m-1}$ and $S_2^2 = \frac{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}{n-1}$

Under $H_0 : \mu_1 - \mu_2 = c$, the test statistic follows t distribution with v degrees of freedom where $v = \frac{\left(\frac{S_1^2}{m} + \frac{S_2^2}{n}\right)}{\frac{S_1^4}{m^2(m-1)} + \frac{S_2^4}{n^2(n-1)}}$. This t-test commonly known as **Welch Two Sample t-test**.
Method of calculation of P-value is same as per two sample t-test.

9.2.3 Paired t-test

Paired t-test is used to investigate the significance of the difference between before and after the treatment in the sample. Let $X_1, X_2, \dots X_n$ be the observations made initially from n individuals as a random sample of size n. A treatment is applied to the above individuals and observations are made after the treatment and are denoted by $Y_1, Y_2, \dots, Y_n$. That is, (X_i, Y_i) denotes the pair of observations obtained from the i^{th} individual, before and after the treatment applied. Let μ_X is unknown population mean before the treatment and μ_Y is the unknown population mean after the treatment. We assume that the populations from which, the two samples drawn, are Normal distribution and observations are collected in a pair. We test one of the following null hypothesis (H_0) and alternative hypothesis (H_1) at α level of significance.

a) H_0 : There is no significant difference between before and after the treatment applied. i.e. treatment applied, is ineffective. i.e., $H_0 : \mu_d = \mu_X - \mu_Y = c$ vs $H_1 : \mu_d \neq c$
b) $H_0 : \mu_d \leq 0$ vs $H_1 : \mu_d > c$
c) $H_0 : \mu_d \geq 0$ vs $H_1 : \mu_d < c$

The test statistic for testing the above hypothesis is

$$t = \frac{\bar{d} - \mu_d}{S_d/\sqrt{n}}$$

where $\bar{d} = \frac{\sum_{i=1}^{n} d_i}{n}$, $d_i = X_i - Y_i$ and $S_d^2 = \frac{\sum_{i}^{n}(d_i - \bar{d})^2}{n-1}$

Under H_0, the test statistic follows t distribution with $(n-1)$ degrees of freedom. The P-value is calculated as

For	a) $H_1 : \mu_d \neq c$,	P-value=$2 \times P(T > \lvert t \rvert)$
	b) $H_1 : \mu_d > c$,	P-value=$P(T > t)$
	c) $H_1 : \mu_d < c$	P-value=$P(T < t)$

where T follows t distribution with $(n-1)$ degrees of freedom.
In R programming, the **t.test( )** function produces the variety of t-tests. We will discuss the different t-tests by following Example 1, 2 and 3.

Example 1 (One Sample t-test): A sample of 13 students from a government school has the following scores in a test.

89 88 78 76 78 78 86 83 82 76 72 77 92.

Do this data support that i) the mean mark of the school students is 80? Test at 5% level.

ii) the mean mark of the school students is more than 75? Test at 1% level.

iii) the mean mark of the school students is less than 85? Test at 10% level.

Solution:

i) Here we test, $H_0 : \mu = 80$ against $H_1 : \mu \neq 80$.

```
x=c(89,88,78,76,78,78,86,83,82,76,72,77,92)   #data
t.test(x,mu=80) #by default alternative is two sided and level is 5%
```

Output

```
One Sample t-test
data:  x
t = 0.68885, df = 12, p-value = 0.504
alternative hypothesis: true mean is not equal to 80
95 percent confidence interval:
77.50427  84.80342
sample estimates:
mean of x
81.15385
```

R Output gives the test statistic t, degrees of freedom and P-value.

Here P-value is 0.504>0.05, hence we do not have enough evidence to reject H_0 (i.e. Accept H_0). Output also gives additional information about the confidence interval with sample estimate of μ. Here 95% confidence interval is (77.50427, 84.80342) which also support the decision taken from P-value as 80 is included in the confidence interval.

ii) Here we test, $H_0 : \mu \leq 75$ against $H_1 : \mu > 75$.

```
x=c(89,88,78,76,78,78,86,83,82,76,72,77,92) #data
t.test(x,mu=75,alternative = "greater",cof.level=0.99)
```

Output

```
      One Sample t-test

data:  x
t = 3.6739, df = 12, p-value = 0.001592
alternative hypothesis: true mean is greater than 75
95 percent confidence interval:
 78.16846      Inf
sample estimates:
mean of x
 81.15385
```

Here P-value is 0.001592<0.01, hence we reject H_0 (i.e. Accept H_1). Output also gives one sided confidence interval with sample estimate of μ which support the decision taken from P-value.

iii) Here we test, $H_0 : \mu \geq 85$ against $H_1 : \mu < 85$.

```
x=c(89,88,78,76,78,78,86,83,82,76,72,77,92)
t.test(x,mu=85,alternative = "less",cof.level=0.9)
```

Output:

```
      One Sample t-test

data:  x
t = -2.2962, df = 12, p-value = 0.02024
alternative hypothesis: true mean is less than 85
95 percent confidence interval:
     -Inf 84.13923
sample estimates:
mean of x
 81.15385
```

Here P-value is 0.02024<0.1, hence we reject H_0 (i.e. Accept H_1). Output also gives one sided confidence interval with sample estimate of μ which support the decision taken from P-value.

Example 2 (Two Sample t-test): The yield of two varieties of mango (in tons) on two independent sample of 10 and 12 plants are given below.

Variety-A: 22 24 26 23 26 30 32 34
Variety-B: 28 25 26 30 32 30 33 28 30 35

i) Test whether the yield of Variety-A is not equal to Variety-B at 2% level of significance.

ii) Test whether the difference between yield of Variety-A is less than Variety-B by 2 tones at 5% level of significance.

iii) Test whether the difference between yield of Variety-A is more than Variety-B by 0.5 tones at 10% level of significance.

iv) Test whether the yield of Variety-A is not equal to Variety-B at 5% level of significance assume unequal variances of both samples.

Solution:

i) Here we test, $H_0: \mu_1 - \mu_2 = 0$ against $H_1: \mu_1 - \mu_2 \neq 0$

```
x=c(22,24,26,23,26,30,32,34)            #first sample data
y=c(28,25,26,30,32,30,33,28,30,35)      #second sample data
t.test(x,y,var.equal = TRUE, conf.level = 0.98)
#by default c=0 and alternative
#hypothesis is two sided
```

Output:

```
Two Sample t-test
data:  x and y
t = -1.4607, df = 16, p-value = 0.1634
alternative hypothesis: true difference in means is not equal to 0
98 percent confidence interval:
 -7.129169  1.979169
sample estimates:
mean of x mean of y
   27.125    29.700
```

Here P-value is 0.1634>0.02, hence we do not have enough evidence to reject H_0 (i.e. Accept H_0). Output also give confidence interval of difference of means with sample estimates of μ_1 and μ_2 which support the decision taken from P-value.

ii) Here we test, $H_0: \mu_1 - \mu_2 \geq 2$ against $H_1: \mu_1 - \mu_2 < 2$

```
x=c(22,24,26,23,26,30,32,34)              #first sample data
y=c(28,25,26,30,32,30,33,28,30,35)        #second sample data
t.test(x,y,var.equal = TRUE, mu=2,alternative = "less", conf.level = 0.95)
```

Output:

```
      Two Sample t-test

data:  x and y
t = -2.5953, df = 16, p-value = 0.009763
alternative hypothesis: true difference in means is less than 2
95 percent confidence interval:
      -Inf 0.5026423
sample estimates:
mean of x mean of y 
   27.125    29.700 
```

Here P-value is 0.009763<0.05, hence we reject H_0 (i.e. Accept H_1). Output also gives one sided confidence interval of difference of means with sample estimates of μ_1 and μ_2 which support the decision taken from P-value.

iii) Here we test, $H_0: \mu_1 - \mu_2 \leq 0.5$ against $H_1: \mu_1 - \mu_2 > 0.5$

```
x=c(22,24,26,23,26,30,32,34)              #first sample data
y=c(28,25,26,30,32,30,33,28,30,35)        #second sample data
t.test(x,y,var.equal = TRUE, mu=0.5,alternative = "greater", conf.level = 0.9)
```

Output:

```
Two Sample t-test
data:  x and y
t = -1.7444, df = 16, p-value = 0.9499
alternative hypothesis: true difference in means is greater than 0.5
90 percent confidence interval:
 -4.931434       Inf
sample estimates:
mean of x mean of y 
   27.125    29.700 
```

Here P-value is 0.9499>0.1, hence we do not have enough evidence to reject H_0 (i.e. Accept H_0). Output also give confidence interval of difference of means with sample estimates of μ_1 and μ_2 which support the decision taken from P-value.

iv) Here we test, $H_0: \mu_1 - \mu_2 = 0$ against $H_1: \mu_1 - \mu_2 \neq 0$ where assumption of equality of variance of two sample does not hold.

```
x=c(22,24,26,23,26,30,32,34)              #first sample data
y=c(28,25,26,30,32,30,33,28,30,35)        #second sample data
t.test(x,y) #by default c=0, alternative hypothesis is two sided and los=5%
            #by default variances are not equal
```

Output:

```
            Welch Two Sample t-test

data:  x and y
t = -1.4037, df = 12.172, p-value = 0.1854
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -6.565645  1.415645
sample estimates:
mean of x mean of y
   27.125    29.700
```

Here P-value is 0.1854>0.05, hence we do not have evidence to reject H_0 (i.e. Accept H_0). Output also give confidence interval of difference of means with sample estimates of μ_1 and μ_2 which support the decision taken from P-value.

Example 3 (Paired t-test): A new variety of health drink in the market for weight of infants. A sample of 10 babies was selected and was given the above diet for a month and the weights were observed before (X) and after (Y) the diet given.

X :	6.6	6.85	6.75	7.2	6.75	6.65	6.7	7.3	6.9	6.6
Y :	6.9	7.3	7	7.6	6.85	7.3	6.7	7.45	7.3	6.5

i) Examine whether there is significant difference between before and after the healthy drink diet at 5% level of significance.

ii) Examine whether the weight gain after the healthy drink diet is more than 0.2 kg at 1% level of significance.

iii) Examine whether the weight loss after the healthy drink diet is less than 0.5 kg at 10% level of significance.

Solution:

i) Here we test, $H_0: \mu_d = \mu_X - \mu_Y = 0$ against $H_1: \mu_d \neq 0$

```
x=c(6.6,6.85,6.75,7.2,6.75,6.65,6.7,7.3,6.9,6.6) #Before Treatment Data
y=c(6.9,7.3,7,7.6,6.85,7.3,6.7,7.45,7.3,6.5)    #After Treatment Data
t.test(x,y,paired = TRUE) #by default c=0, alternative is two sided and los=5%
```

Output:

```
Paired t-test
data:  x and y
t = -3.6211, df = 9, p-value = 0.005563
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.42242786 -0.09757214
sample estimates:
mean of the differences
                  -0.26
```

Here P-value is 0.005563<0.05, hence we reject H_0 (i.e. Accept H_1). Output also gives confidence interval and sample estimate of μ_d which also support the decision taken from P-value.

ii) Here we test, $H_0\colon \mu_d = \mu_X - \mu_Y \leq 0.2$ against $H_1\colon \mu_d > 0.2$

```
x=c(6.6,6.85,6.75,7.2,6.75,6.65,6.7,7.3,6.9,6.6) #Before Treatment Data
y=c(6.9,7.3,7,7.6,6.85,7.3,6.7,7.45,7.3,6.5)    #After Treatment Data
t.test(x,y,paired = TRUE,mu=0.2,conf.level = 0.99,alternative = "greater")
```

Output:

```
      Paired t-test

data:  x and y
t = -6.4065, df = 9, p-value = 0.9999
alternative hypothesis: true difference in means is greater than 0.2
99 percent confidence interval:
 -0.4625854        Inf
sample estimates:
mean of the differences
                  -0.26
```

Here P-value is 0.9999>0.01, hence we do not have evidence to reject H_0 (i.e. Accept H_0). Output also gives confidence interval and sample estimate of μ_d which also support the decision taken from P-value.

iii) Here we test, $H_0\colon \mu_d = \mu_X - \mu_Y \geq 0.5$ against $H_1\colon \mu_d < 0.5$

```
x=c(6.6,6.85,6.75,7.2,6.75,6.65,6.7,7.3,6.9,6.6) #Before Treatment Data
y=c(6.9,7.3,7,7.6,6.85,7.3,6.7,7.45,7.3,6.5)    #After Treatment Data
t.test(x,y,paired = TRUE,mu=0.5,conf.level = 0.9,alternative = "less")
```

Output:

```
Paired t-test

data:  x and y
t = -10.585, df = 9, p-value = 1.113e-06
alternative hypothesis: true difference in means is less than 0.5
90 percent confidence interval:
       -Inf -0.1606955
sample estimates:
mean of the differences
                  -0.26
```

Here P-value is <0.1, hence we reject H_0 (i.e. Accept H_1). Output also gives confidence interval and sample estimate of μ_d which also support the decision taken from P-value.

9.3 Chi-square Test for Variance:

Chi-square test for variance is used to test the population variance σ^2 regarded as σ_0^2 based on a random sample of size n which is drawn from normal population with mean μ and variance σ_0^2(both μ and σ^2 are unknown) . We investigate the significance of the difference between the assumed population variance σ_0^2 and the sample variance. We test one of the following null hypothesis (H_0) and alternative hypothesis (H_1) at α level of significance.

a) H_0 : There is no significant difference between the sample variance S^2 and the assumed population variance σ_0^2. i.e., $H_0 : \sigma^2 = \sigma_0^2$ vs $H_1 : \sigma^2 \neq \sigma_0^2$

b) $H_0 : \sigma^2 \leq \sigma_0^2$ vs $H_1 : \sigma^2 > \sigma_0^2$

c) $H_0 : \sigma^2 \geq \sigma_0^2$ vs $H_1 : \sigma^2 < \sigma_0^2$

The test statistic for testing the above hypothesis is

$$\chi^2 = \frac{(n-1)S^2}{\sigma_0^2}$$

Where $\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$ and $S^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n-1}$

Under H_0, the test statistic follows χ^2 distribution with $(n-1)$ degrees of freedom. We take the decision whether to reject the null hypothesis or not based on P-value. If P-value $< \alpha$ then we reject the null hypothesis and if P-value $\geq \alpha$ then we do not enough evidence to reject the null hypothesis. The P-value is calculated as

For a) $H_1 : \sigma^2 \neq \sigma_0^2$, P-value$= 2 \times (1 - P(\chi_{n-1}^2 < \chi^2))$

b) $H_1 : \sigma^2 > \sigma_0^2$, P-value$=P(\chi_{n-1}^2 > \chi^2)$

c) $H_1 : \sigma^2 < \sigma_0^2$, P-value$=P(\chi_{n-1}^2 < \chi^2)$

Where χ^2 follows χ^2 distribution with $(n-1)$ degrees of freedom (i.e.. χ_{n-1}^2).

If μ is known then test statistic is $\chi^2 = \frac{\sum_{i=1}^{n}(X_i - \mu)^2}{\sigma_0^2}$ and is follows χ^2 distribution with n degrees of freedom (i.e.. χ_n^2).

In R programming, there is no inbuilt function for chi-square test for variance testing. Here we write the code in R, as per discussed procedure. We discuss the code with the following example 4 and 5.

Example 4: A lifetime of a certain brand of bulb (in hours) produced by his company is as follows

3360	3720	3300	3420	3240	3420	3450	3540	3750	3780

i) Test whether the variance is 30000 or not at 5% level.

ii) Test whether the variance is more than 20000 at 10% level.

iii) Test whether the variance is less than 33000 at 2% level.

Solution:

i) Here we test, $H_0 : \sigma^2 = \sigma_0^2 = 30000$ against $H_0 : \sigma^2 \neq \sigma_0^2 = 30000$

```
x=c(3360,3720,3300,3420,3240,3420,3450,3540,3750,3780)  #data
s.2=33000;                                     #assumed population variance
n=length(x)                                           #size of data
chisqare.stat=(n-1)*var(x)/s.2;                       #test statistic
#Calculation of p-value here alternative is two sided
if (qchisq(alp/2,n-1)<chisqare.stat)
{p.value=pchisq(chisqare.stat,n-1)}else
{p.value=pchisq(chisqare.stat,n-1)}
# Output
cat("\t \t Chi-square Test for Variance\n",
    "alternative hypothesis: true variance is not equal to" , s.2,"\n",
    "test statistic=",chisqare.stat, "\t", "df=",n-1,"\t","p-value=",p.value);
```

Output:

```
            Chi-square Test for Variance
alternative hypothesis: true variance is not equal to 33000
test statistic= 10.10182      df= 9       p-value= 0.6846111
```

Here P-value is 0.6846111>0.05, hence we do not have enough evidence to reject H_0 (i.e. Accept H_0).

ii) Here we test, $H_0 : \sigma^2 \le \sigma_0^2 = 20000$ against $H_0 : \sigma^2 > \sigma_0^2 = 20000$

```
x=c(3360,3720,3300,3420,3240,3420,3450,3540,3750,3780)  #data
s.2=20000;                                       #assumed population variance
n=length(x)                                             #size of data
chisqare.stat=(n-1)*var(x)/s.2;                         #test statistic
#Calculation of p-value here alternative is greater than type
p.value=1-pchisq(chisqare.stat,n-1);
# Output
cat("\t \t Chi-square Test for Variance\n",
    "alternative hypothesis: true variance greater than" , s.2,"\n",
    "test statistic=",chisqare.stat, "\t", "df=",n-1,"\t","p-value=",p.value);
```

Output:

```
                  Chi-square Test for Variance
alternative hypothesis: true variance greater than 20000
test statistic= 16.668        df= 9       p-value= 0.05417611
```

Here P-value is 0.05417611<0.1, hence we reject H_0 (i.e. Accept H_1).

iii) Here we test, $H_0 : \sigma^2 \ge \sigma_0^2 = 35000$ against $H_0 : \sigma^2 < \sigma_0^2 = 35000$

```
x=c(3360,3720,3300,3420,3240,3420,3450,3540,3750,3780)  #data
s.2=40000;                                                #assumed population
variance
n=length(x)                                             #size of data
chisqare.stat=(n-1)*var(x)/s.2;                         #test statistic
#Calculation of p-value here alterrnative is less than type
p.value=pchisq(chisqare.stat,n-1);
# Output
cat("\t \t Chi-square Test for Variance\n",
    "alternative hypothesis: true variance less than" , s.2,"\n",
    "test statistic=",chisqare.stat, "\t", "df=",n-1,"\t","p-value=",p.value);
```

Output:

```
            Chi-square Test for Variance
alternative hypothesis: true variance less than 40000
test statistic= 8.334   df= 9       p-value= 0.4991312
```

Here P-value is 0.4991312>0.02, hence we do not have enough evidence to reject H_0 (i.e. Accept H_0).

Example 5: A average yield of mango is 650 per mango tree and random sample of 10 mango trees has the following yield in a year:

760 650 640 560 580 540 620 680 760 780

i) Test whether variance is 6500 or not at 1% level of significance.

ii) Test whether variance is more than 7500 at 5% level of significance.
iii) Test whether variance is less than 4500 at 10% level of significance.

Solution:

i) Here μ is known and we test, $H_0 : \sigma^2 = \sigma_0^2 = 6500$ against $H_0 : \sigma^2 \neq \sigma_0^2 = 6500$

```
x=c(760,650,640,560,580,540,620,680,760,780)  #data
mu=650;                                        #population mean
s.2=6500;                                      #assumed population variance
n=length(x)                                    #size of data
chisqare.stat=sum((x-mu)^2)/s.2;               #test statistic
#Calculation of p-value here alternative is two sided
p.value=2*(1-pchisq(chisqare.stat,n))
# Output
cat("\t \t Chi-square Test for Variance\n",
    "alternative hypothesis: true variance is not equal to" , s.2,"\n",
    "test statistic=",chisqare.stat, "\t", "df=",n,"\t","p-value=",p.value);
```

Output:

```
                   Chi-square Test for Variance
 alternative hypothesis: true variance is not equal to 6500
 test statistic= 10.47692      df= 10      p-value= 0.7993858
```

Here P-value is 0.7993858>0.01, hence we do not have evidence to reject H_0 (i.e. Accept H_0).

ii) Here μ is known and we test, $H_0 : \sigma^2 \leq \sigma_0^2 = 7500$ against $H_0 : \sigma^2 = \sigma_0^2 > 7500$

```
x=c(760,650,640,560,580,540,620,680,760,780)  #data
mu=650;                                        #population mean
s.2=7500;                                      #assumed population variance
n=length(x)                                    #size of data
chisqare.stat=sum((x-mu)^2)/s.2;               #test statistic
#Calculation of p-value here alternative is greater than type
p.value=1-pchisq(chisqare.stat,n);
# Output
cat("\t \t Chi-square Test for Variance\n",
    "alternative hypothesis: true variance is greater than" , s.2,"\n",
    "test statistic=",chisqare.stat, "\t", "df=",n,"\t","p-value=",p.value);
```

Output:

```
             Chi-square Test for Variance
 alternative hypothesis: true variance is greater than 7500
 test statistic= 9.08    df= 10      p-value= 0.5245285
```

Here P-value is 0.5245285>0.05, hence we do not have evidence to reject H_0 (i.e. Accept H_0).

iii) Here μ is known and we test, $H_0 : \sigma^2 \geq \sigma_0^2 = 4500$ against $H_0 : \sigma^2 = \sigma_0^2 < 4500$

```
x=c(760,650,640,560,580,540,620,680,760,780)  #data
mu=650;                                        #population mean
s.2=4500;                                      #assumed population variance
n=length(x)                                    #size of data
chisqare.stat=sum((x-mu)^2)/s.2;               #test statistic
#Calculation of p-value here alternative is less than type
p.value=pchisq(chisqare.stat,n);
# Output
cat("\t \t Chi-square Test for Variance\n",
    "alternative hypothesis: true variance is less than" , s.2,"\n",
    "test statistic=",chisqare.stat, "\t", "df=",n,"\t","p-value=",p.value);
```

Output:

```
                          Chi-square Test for Variance
alternative hypothesis: true variance is less than 4500
test statistic= 15.13333       df= 10      p-value= 0.8727241
```

Here P-value is 0.8727241>0.1, hence we do not have evidence to reject H_0 (i.e. Accept H_0).

9.4 F-test for equality of two variances:

F-test is used to test the variances of the two populations are equal, based on two random samples. We assume that the populations from which, the two samples drawn, are Normal distributions. A random sample of size m observations $X_1, X_2, \ldots, X_m$ be drawn from population with unknown variance σ_1^2 and a random sample of size n observations $Y_1, Y_2, \ldots, Y_n$ be drawn from population with unknown variance σ_2^2. We test one of the following null hypothesis (H_0) and alternative hypothesis (H_1) at α level of significance.

a) H_0 : There is no difference between two population variance i.e. H_0: $\sigma_1^2 = \sigma_2^2$ vs $H_1 : \sigma_1^2 \neq \sigma_2^2$

b) H_0: $\sigma_1^2 \leq \sigma_2^2$ vs $H_1 : \sigma_1^2 > \sigma_2^2$

c) H_0: $\sigma_1^2 \geq \sigma_2^2$ vs $H_1 : \sigma_1^2 < \sigma_2^2$

The test statistic for testing the above hypothesis is $F = \frac{S_1^2}{S_2^2}$

Where $S_1^2 = \frac{\sum_{i=1}^{m}(X_i - \bar{X})^2}{m-1}, S_2^2 = \frac{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}{n-1} \bar{X} = \frac{\sum_{i=1}^{m} X_i}{m}$, and $\bar{Y} = \frac{\sum_{i=1}^{n} Y_i}{n}$,

Under $H_0 : \sigma_1^2 = \sigma_2^2$, the test statistic F follows F distribution with $(m-1, n-1)$ degrees of freedom. We take the decision whether to reject the null hypothesis or not based on P-value. If P-value $< \alpha$ then we reject the null hypothesis and if P-value $\geq \alpha$ then we do not enough evidence to reject the null hypothesis. The P-value is calculated as

For a) $H_1 : \sigma_1^2 \neq \sigma_2^2$, P-value= $2 \times (1 - P(F_{(m-1,n-1)} < F))$
b) $H_1 : \sigma_1^2 > \sigma_2^2$, P-value=$P(F_{(m-1,n-1)} > F)$
c) $H_1 : \sigma_1^2 < \sigma_2^2$, P-value=$P(F_{(m-1,n-1)} < F)$

Where F follows F distribution with $(m-1, n-1)$ degrees of freedom.
In R programming, there is inbuilt function **var.test()** for F test for testing equality of two variances. We will demonstrate the var.test() function by **Example 6**.

Example 6: The yield of two varieties of mango (in tons) on two independent sample of 10 and 12 plants are given below.

Variety-A:	22	24	26	23	26	30	32	34		
Variety-B:	28	25	26	30	32	30	33	28	30	35

i) Test whether the variance of variety-A is not equal to Variety-B at 5% level of significance.

ii) Test whether the variance of variety-A is greater than Variety-B at 10% level of significance.

iii) Test whether the variance of variety-A is less than Variety-B at 1% level of significance.

Solution:

i) Here we test $H_0 : \sigma_1^2 = \sigma_2^2$ against $H_1 : \sigma_1^2 \neq \sigma_2^2$

```
X=c(22,24,26,23,26,30,32,34)                      #first sample data
y=c(28,25,26,30,32,30,33,28,30,35)         #second sample data
var.test(x,y)          #by default alternative is two sided and los=5%
```

Output:

```
      F test to compare two variances

data:  x and y
F = 2.0141, num df = 7, denom df = 9, p-value = 0.3238
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.4798759 9.7142569
sample estimates:
ratio of variances
          2.014062
```

Here P-value is 0.3238>0.05, Hence we do not have enough evidence to reject H_0.(i.e. Accept H_0). Output also gives 95% confidence interval for ratio of variance with their sample estimates which also support the decision taken from P-value.

ii) Here we test $H_0 : \sigma_1^2 \leq \sigma_2^2$ against $H_1 : \sigma_1^2 > \sigma_2^2$

```
x=c(22,24,26,23,26,30,32,34)                      #first sample data
y=c(28,25,26,30,32,30,33,28,30,35)         #second sample data
var.test(x,y,alternative = "greater",conf.level = 0.9)
```

Output:

```
      F test to compare two variances

data:  x and y
F = 2.0141, num df = 7, denom df = 9, p-value = 0.1619
alternative hypothesis: true ratio of variances is greater than 1
90 percent confidence interval:
 0.8039161       Inf
sample estimates:
ratio of variances
          2.014062
```

Here P-value is 0.1639>0.10, Hence we do not have enough evidence to reject H_0.(i.e. Accept H_0).

iii) Here we test $H_0 : \sigma_1^2 \geq \sigma_2^2$ against $H_1 : \sigma_1^2 < \sigma_2^2$

```
x=c(22,24,26,23,26,30,32,34)                      #first sample data
y=c(28,25,26,30,32,30,33,28,30,35)               #second sample data
var.test(x,y,alternative = "less",conf.level = 0.99)
```

Output:

```
      F test to compare two variances

data:  x and y
F = 2.0141, num df = 7, denom df = 9, p-value = 0.8381
alternative hypothesis: true ratio of variances is less than 1
99 percent confidence interval:
  0.00000 13.53198
sample estimates:
ratio of variances
          2.014062
```

Here P-value is 0.8381>0.01, Hence we do not have enough evidence to reject H_0.(i.e. Accept H_0).

9.5 References:

- Verzani, J. (2014). *Using R for introductory statistics*. CRC Press.
- Rajagopalan V. (2006). Selected Statistical Tests. New Age International (P) limited, Publishers

Chapter 12

Analysis of Varince (ANOVA) using R

Dr. Kalpana Dilip Phal, Associate Professor and Head,
B.N.Bandodkar College of Science, Thane, Chendani Thane (West) 400601.

12.1 Introduction

In this chapter a very popular statistical tool namely Analysis of variance(ANOVA) has been explained . Statistical analysis of i)One way classified data or ii)Two way classified data is explained and with the help of R code the execution is shown, together with interpretations of R output.

12.2 ANOVA

Test of significance for the difference between two population means can be carried out using t-test, under certain set of assumptions .But in many situations like biological or agricultural experiments we come upon a problem of comparing more than 2 population means. For example effect of different conditions on seed germination is same or does it differ significantly? Different types of feed on animals do have same gain in weight? etc. We are also interested in knowing what is the effect of various independent factors on the response or dependent variable. For example How yield of paddy crop responses towards different fertilizers used such as vermi compost, bio compost or chemical fertilizers. Analysis of variance is a powerful tool for both of these purposes.

Variations in observations of a data set is inherited. According to father of Dr. R. A. Fisher the causes of these variations may be broadly classified as assignable and chance causes. In anova the estimate of total variations are split up into variations due to various independent factors Some of which are assignable and remaining variation is due to chance factor. The variation is due to chance factor are experimental error

In anova following assumptions are made
i) Model applied is linear ii) Various effects influencing response variable are additive
iii) Observations are independent and iv) Errors are normally distributed IID r.v.s

According to the number of factors variations those influence response variable experiment yields are considered as i)One way classified data or ii)Two way classified data etc.

12.2.1 One way ANOVA

Here Y the response variable is influenced by one factor,ususlly called as treatments
Model : y_{ij} is the response of j^{th} experimental unit receiving i^{th} treatment
$y_{ij}= \mu + \alpha_i + \varepsilon_{ij}$ where i=1 to p and j=1 to r_i, $n=\sum r_i$
Assumptions 1)Model is additive 2), μ is general mean
3) ε_i follows IN(0 ,σ^2) 4) $\varepsilon_{i\,i}$ are independent 5) α_i effect of i^{th} treatment is fixed effect.
The hypothesis we want to test regarding homogeneity of various treatment means in population which reduces to
H_0: $\alpha_1 = \alpha_2 =$………. $\alpha_p = 0$ against H_1: They differ significantly.

ANOVA table

Source	d.f	S.S	MSS	F ratio
Between/treatment	p-1	SStreatment=$\sum_{i=1}^{p}\frac{y_{i.}^2}{r_i} - \frac{y_{..}^2}{N}$	$\frac{\text{SStreatment}}{p-1}$	$\frac{\text{MSStreatment}}{\text{MSSerror}}$
Within/error	n-p	+		
Total	n-1	$\sum_i\sum_j y_{ij}^2 - \frac{y_{..}^2}{n}$		

If calculated F ratio > $F_{\alpha,p-1,n-p}$,then H_0 is rejected. We conclude that treatments differ significantly at confidence level α % (usually α =5% or 1%). MSSerror is treated as an unbiased estimate of σ^2 .The test of significance of all treatments simultaneously may exhibit significant differences in the means of treatment , but multiple comparison test for pairs of treatments guarantees which treatment means differ significantly.
I)Critical difference C.D:

$t_{(n-p),\frac{\alpha}{2}}$. is two tailed α % value of t distribution with n-p d.f .C.D= . $t_{(n-p),\frac{\alpha}{2}}$. $\sqrt{\text{MSSerror}}$

$|\bar{y}_i - \bar{y}_j|$ > C.D The n we conclude i^{th} treatment shows significant difference from j^{th} treatment

II)Tukeys'Honest significant difference test : $|\bar{y}_i - \bar{y}_j| > q_{\alpha,p,,n-p}\sqrt{\frac{\text{MSSerror}}{n}}$ Where $q_{\alpha,p,,n-p}$ is studentised range for which tables are available.

12.2.2 Two way ANOVA (r observations per cell)

Here there are two factors A an B say, influencing Y variable .The case with r observation per cell is discussed here.

Model : y_{ijk} is the response of k^{th} experimental unit receiving i^{th} level of factor A and j^{th} level of factor B
$y_{ijk}= \mu + \alpha_i + \beta_j + \Upsilon_{ij} + \varepsilon_{ijk}$ where i=1 to p j=1 to q, k=1 to r

Assumptions 1)Model is additive 2) μ is general mean

3) ε_{ijk} follows IN(0 ,σ^2) 4) $\varepsilon_{i\ ik}$ are independent 5) α_i effect of i^th level of factor A and β_j is effect of j^th level of factor B . Υ_{ij} is interaction effect between i^th level of factor A and j^th level of and are fixed effects.

SStotal=SSA+SSB+SSAB+SSerror
ANOVA

Source	d.f	S.S	MSS	F ratio
Factor A	p-1	SSA=$\sum_{i=1}^{p}\frac{y_{i..}^2}{qr}-\frac{y_{...}^2}{pqr}$	$\frac{\text{SSA}}{p-1}$	F_A
Factor B	q-1	SSB=$\sum_{j=1}^{q}\frac{y_{\cdot j..}^2}{qr}-\frac{y_{...}^2}{pqr}$	$\frac{\text{SSB}}{q-1}$	F_B
Factor AB	(p-1)(q-1)	SSAB=$\sum\sum_{j=1}^{q}\frac{y_{ij..}^2}{r}-\sum_{i=1}^{p}\frac{y_{i..}^2}{qr}-\sum_{j=1}^{q}\frac{y_{\cdot j..}^2}{qr}+\frac{y_{...}^2}{pqr}$	$\frac{\text{SSAB}}{(p-1)(q-1)}$	F_{AB}
Residual	pq(r-1)	$\sum_{k}\sum_{i}\sum_{j}y_{ijk}^2-\sum_{i}\sum_{j=1}^{q}\frac{y_{ij..}^2}{r}$	MSresidual	
Total	pqr-1	$\sum_{k}\sum_{i}\sum_{j}y_{ijk}^2-\frac{y_{..}^2}{n}$		

First test 1) H_0 : Υ_{ij} =0 for all i,j
F_{AB} =MSAB/MSerror, F_{AB} > F α, (p-1)(q-1) ,pq(r-1) ,then conclude that there is interaction between two factors.It makes no sense in carrying out following test. Rather we must held one level of factor A constant and test H_{0B}using one way ANOVA. And we must held one level of factor B constant and test H_{0A} using one way ANOVA .
2)H_{0A}: α_1 = α_2 =.......... α_p = 0 against H_{1A}: They differ significantly . F_A =MSA/MSressidual
3)H_{0B}: β_1 = β_2 =.......... β_q= 0 against H_{1B}: They differ significantly . F_B =MSB/ MSressidual

12.2.3 Two way ANOVA (one observations per cell)

Model : y_{ij} is the response unit receiving i^th level of factor A and j^th level of factor B
y_{ij}= μ +α_i + β_j +ε_{ijk} where i=1 to p and j=1 to q, n=pq
SStotal=SSA+SSB +SSerror

ANOVA

Source	d.f	S.S	MSS	F ratio
Factor A	p-1	SSA=$\sum_{i=1}^{p}\frac{y_{i..}^2}{q}-\frac{y_{...}^2}{pq}$	$\frac{\text{SSA}}{p-1}$	$\frac{\text{MSSA}}{MSSerror}$
Factor B	q-1	SSB=$\sum_{j=1}^{q}\frac{y_{\cdot j..}^2}{q}-\frac{y_{...}^2}{pq}$	$\frac{\text{SSB}}{q-1}$	$\frac{\text{MSSB}}{MSSerror}$

Error	(p-1)(q-1)	$\sum_{k}\sum_{i}\sum_{j} y_{ijk}^2 - \sum_{i} \sum_{j=1}^{q} \frac{y_{ij..}^2}{r}$	MSerror= $\frac{\text{SSerror}}{(p-1)(q-1)}$	
Total	pq-1	$\sum_{k}\sum_{i}\sum_{j} y_{ijk}^2 - \frac{y_{..}^2}{n}$		

The hypothesis we want to test regarding homogeneity of various means of i)factor A and ii)factor B in population which reduces to

i)H_{0A}: $\alpha_1 = \alpha_2 = \ldots\ldots\ldots \alpha_p = 0$ against H_{1A}: They differ significantly .
ii)If calculated F ratio > $F_{\alpha,p-1,n-1}$,then H_{0A} is rejected. We conclude that means of levels of factor A differ significantly at α %.

i)H_{0B}: $\beta_1 = \beta_2 = \ldots\ldots\ldots \beta_q = 0$ against H_{1B}: They differ significantly .
ii)If calculated F ratio > $F_{\alpha,q-1,n-1}$,then H_{0B} is rejected. We conclude that means of levels of factor B differ significantly at α %.

R code for ANOVA

Examples 1: Th grade point average (GPA-4 point scale) of students participating in college sports program are compared .The data are as under.

Football	Tennis	Hockey
3.2	3.8	2.6
2.6	3.1	1.9
2.4	2.6	1.7
2.4	3.9	2.5
1.8	3.2	1.9

Do different sports have significant effect on GPA? .Apply Tuckey's multiple comparison test.

Solution . Here we apply ANOVA on way as GPA are classified according to one factor = sports

```
#data should be read treatment wise #To read treatments
>GPA=c(3.2,2.6,2.4,2.4,1.8,3.8,3.1,2.6,3.9,3.3,2.6,1.9,1.7,2.5,1.9)
>Sport=c(rep("Football",5),rep("Tennis",5),rep("Hockey",5))
>d=data.frame(Sport,GPA)
# anova oneway
>av1=aov(GPA~Sport,data=d)
>summary(av1)
```

Output:

```
                 Df    Sum Sq    Mean Sq     F value      Pr(>F)
 Sport            2     3.929     1.9647       8.456   0.00511**
 Residuals       12     2.788     0.2323
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Interpretation: As F calculated is highly significant(**)Treatments differ significantly sports person's GPA differ according sport. We apply Tuckey's test for comparing sports pairwise.

```
>TukeyHSD(av1,"Sport",ordered=F,conf.level=0.95)
# One can also use  plot(TukeyHSD(av1,"Sport"))
Output:
Tukey multiple comparisons of means    95% family-wise confidence level
Fit: aov(formula = GPA ~ Sport, data = d)$

 Sport                   diff          lwr         upr            p  adj
 Hockey-Football        -0.36   -1.17329741    0.4532974    0.4860718
 Tennis-Football         0.86    0.04670259    1.6732974    0.0381404
 Tennis-Hockey           1.22    0.40670259    2.0332974    0.0046180
```

Interpretation: No sport shows significant difference in GPA means

Example2 : Four varieties of wheat are planted at 3 different locations and their yields (units per plot)are recorded as below.:

Variety↓ Location→	Location 1	Location 2	Location 3
Variety1	14.3	7.6	19.2
Variety2	13.4	3.9	12.6
Variety3	18.4	13.4	15.1

Carry out analysis to check whether different locations or different varieties have significant effect on yield of wheat?..

Solution:

```
#data should be read variety wise
>yield=c(14.3,13.4,18.4,7.6,3.9,13.4,19.2,12.6,15.1)
>loc=c(rep("L1",3),rep("L2",3),rep("L3",3))
>variety=c("V1","V2","V3","V1","V2","V3","V1","V2","V3")
>result=aov(yield~ loc+variety)
>summary(result)
```

Output:

```
              Df     Sum Sq    Mean Sq    F value      Pr(>F)
 loc           2     103.79      51.89      6.389      0.0568
 variety       2      49.79      24.89        065      0.1559
 Residuals     4      32.49       8.12
```

Interpretation: The Calculated F ratio are not significant, as p value is > .05 The yield does not change significantly as location changes. Even the differences in varieties do not have significant influence on yield. Varieties do not differ significantly.

Example 3: An engineer suspects that surface finish of a metal part is influenced by type of paint used and drying time.Drying times are selected by him are 20,25,30 minutes. and he randomly choses paint I, II.Conducted experiment yielded following data analyse it. Is there any interaction present between paint and drying time?

paint↓	Drying Times(minutes)		
	20	25	30
I	74,64,50	73,61,44	78,85,92
II	92,86,68	98,73,88	66,45,85

Solution:

```
> DT=c(74,64,50,92,86,68,73,61,44,98,73,88,78,85,92,66,45,85)
> paint=c(rep("I",3),rep("II",3))
> DRT1=c(paint)
> DRT2=c(paint)
> DRT3=c(paint)
> DRT=c("DRT1","DRT2","DRT3")
> d=data.frame(DT,paint,DRT)
> fit=aov(DT~paint*DRT,data=d)
fit=aov(DT~paint*DRT,data=d)
> summary(fit)
```

Output:

```
              Df    Sum Sq   Mean Sq    F value    Pr(>F)
paint          1       356     355.6      1.250     0.285
DRT            2       421     210.4      0.740     0.498
paint:DRT      2       315     157.4      0.553     0.589
Residuals     12      3413     284.4
```

Interpretation: Interaction between drying time and paint is not significant. we can perform test for equality of paint means or for drying time means. Using error or error +interaction S.S.

i) H_{0A}: $\alpha_1 = \alpha_2 = \dots\dots\dots \alpha_p = 0$ against H_{1A}: paints differ significantly .ii)Since calculated F ratio < $F_{\alpha,p-1,n-1}$, so H_{0A} is not rejected. We conclude that means of paints do not differ significantly at confidence level 5 %.

ii) H_{0B}: $\beta_1 = \beta_2 = \dots\dots\dots \beta_q = 0$ against H_{1B}: Drying times differ significantly .

ii) Here calculated F ratio < $F_{\alpha,q-1,n-1}$,so H_{0B} is not rejected. We conclude that means of Dryng timesdo not differ significantly at 5 %.