```
plot(c(1,2,3,4,5),c(1,4,9,16,25))
x=seq(-pi,pi,0.1)
y=sin(x)
plot(x,y)
plot(x,y,pch=c(4,5,6),col=c('red','blue','violet','green'))
#Set a plotting window with one row and two columns.
x=seq(-pi,pi,0.1)
y=sin(x)
par(mfrow=c(1,2))
plot(x,y,type='l')
plot(x,y,pch=c(4,5,6),col=c('red','blue','violet','green'))
#Set space for 2 rows and 3 columns.

#Plot out the graphs using various options.
par(mfrow=c(2,3))
plot(x,cos(x),col=c('blue','orange'),type='o',pch=19,lwd=2,cex=1.5)

plot(x,x*2,col='red',type='l')

plot(x,x^2/3+4.2, col='violet',type='o',lwd=2,lty=1)

plot(c(1,3,5,7,9,11),c(2,7,5,10,8,10),type='o',lty=3,col='pink',lwd=4)
###labels
labelset <-c('one','three','five','seven','nine','eleven')
```

```
x1<- c(1,3,5,7,9,11)

y1 <- c(2,7,5,10,8,10)

plot(x1,y1,type='o',lty=3,col='pink',lwd=4,main="This is a
graph",col.main='blue',xlab="Time",ylab="Performance")

text(x1+0.5,y1,labelset,col='red')
```

Experiment-1

Introduction: Understanding Data types; importing/exporting data

Aim: The purpose of this experiment is to learn the input data types, various arithmetic operations of dataset and importing/exporting data in R

Procedure:
Step by step procedure to conduct the required experiment –
1. Input and creation of dataset using R

2. Perform various arithmetic operations on the dataset using R

3. Explore various types of data import using R

Introduction to R

R is an open-source programming language that is widely used as a statistical software and data analysis tool. R generally comes with the Command-line interface. R is available across widely used platforms like Windows, Linux, and macOS.

Statistical Features of R

- **Basic Statistics:** The most common basic statistics terms are the mean, mode, and median. These are all known as "Measures of Central Tendency." So using the R language we can measure central tendency very easily.
- **Static graphics:** R is rich with facilities for creating and developing interesting static graphics. R contains functionality for many plot types including graphic maps, mosaic plots, biplots, and the list goes on.
- **Probability distributions:** Probability distributions play a vital role in statistics and by using R we can easily handle various types of probability distribution such as Binomial Distribution, Normal Distribution, Chi-squared Distribution and many more.
- **Data analysis:** It provides a large, coherent and integrated collection of tools for data analysis.
- **R Packages:** One of the major features of R is it has a wide availability of libraries. R has CRAN(Comprehensive R Archive Network), which is a repository holding more than 100000 packages.

Programming in R

Since R is much similar to other widely used languages syntactically, it is easier to code and learn in R. Programs can be written in R in any of the widely used IDE like **R Studio, Rattle, Tinn-R**, etc., After writing the program save the file with the extension **.r**. To run the program use the following command on the command line:
R file_name.r

To install R on Windows OS

- Go to the CRAN website. https://cran.r-project.org/
- Click on "Download R for Windows".
- Click on "install R for the first time" link to download the R executable (.exe) file.
- Run the R executable file to start installation, and allow the app to make changes to your device.
- Select the installation language.

Install RStudio

- If you want to work with R in your local machine, installing R is not enough. R does not come with a GUI-based platform. Most users install a separate IDE which allows them to interact with R. It gives them additional functionality such as help, preview, etc.
- The most popular IDE for the R programming language is **RStudio**. You can follow these steps to install RStudio on your Windows machine.
- Visit https://www.rstudio.com/products/rstudio/download/#download to download the free version of RStudio for any platform you want.
- Once the download is completed, you need to open the executable file to start the installation process.
- An installation wizard will appear on the screen. Click on the next button.
- On the next prompt, it will ask you to select the start menu folder for shortcut creation. Click on the install button. Once the installation is completed, click on Finish.
- You have now successfully installed RStudio in your local machine.

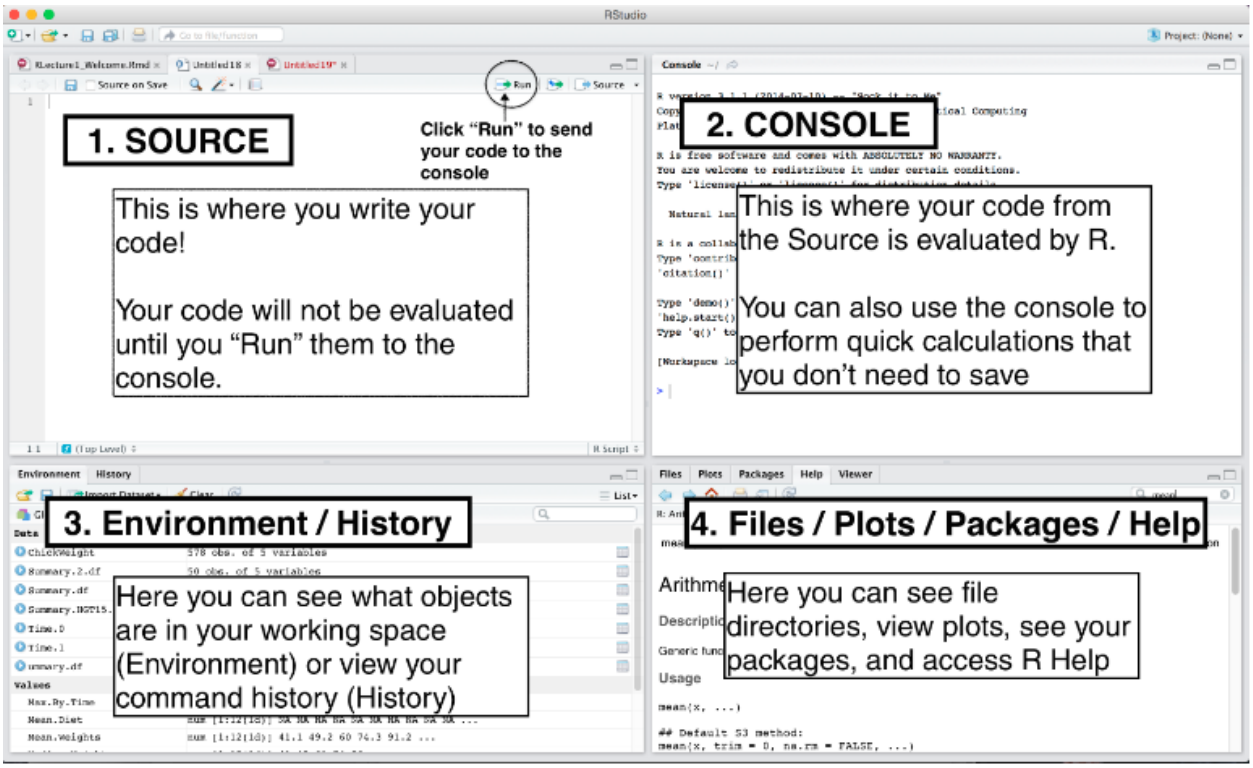
R Online Compilers

Another way to run R programs is to simply use an online environment. You don't have to go through the hassles of installing R and RStudio in this case. There are lots of competitive R compilers that you can find in a single Google search.

The most commonly used online R compilers are:

- JDoodle online R Editor
- Paiza.io online R Compiler
- IdeaOne R Compiler

The four RStudio Windows



Codes and Results

```
# Generate data
1:10

##  [1]  1  2  3  4  5  6  7  8  9 10

# Assign variable name to the value
X=10;  X<-10; 10->X;
# To combine numeric values into a vector
c(1,2,5)

## [1] 1 2 5

#Arithmetic operations of vectors are performed member wise.
a = c(1, 3, 5, 7)
b = c(2, 4, 6, 8)
#addition
a+b

## [1]  3  7 11 15

#subtraction
a-b
```

```
## [1] -1 -1 -1 -1

#constant multiplication
5*a

## [1]  5 15 25 35

#product
a*b

## [1]  2 12 30 56

#division
a/b

## [1] 0.5000000 0.7500000 0.8333333 0.8750000

# character object is used to represent string values in R
X=as.character(5.2)
X

## [1] "5.2"

#Concatenation of strings
paste("Baa", "Baa", "Black", "Sheep")

## [1] "Baa Baa Black Sheep"
```

Installing an R Package

- R packages provide a powerful mechanism for extending the functionality of R
- R packages can be obtained from CRAN or other repositories
- The install.packages() can be used to install packages at the R console

  Eg. install.packages("moments")
- This command downloads the moments package from CRAN and installs it on your computer
- Any packages on which this package depends will also be downloaded and installed
- Multiple R packages can be installed at once with a single call to install.packages()

  Eg. install.packages(c("moments", "ggplot2", "devtools"))

Loading R Packages

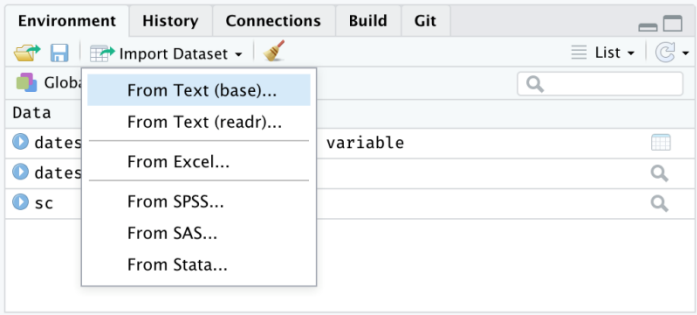
Installing a package does not make it immediately available to you in R; it must load the package.

The library() function loads packages that have been installed so that you may access the functionality in the package

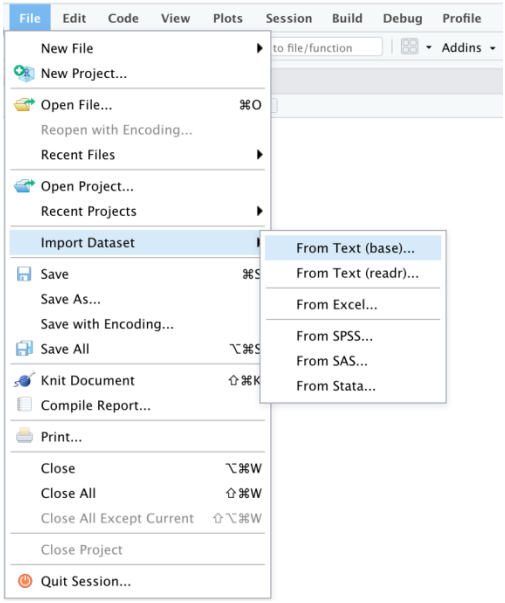
Importing Data

Importing data into R is a necessary step that, at times, can become time intensive. To ease this task, the RStudio includes new features to import data from: csv, xls, xlsx, sav, dta, por, sas and stata files.

The data import features can be accessed from the environment pane or from the tools menu. The importers are grouped into 3 categories: Text data, Excel data and statistical data. To access this feature, use the "Import Dataset" dropdown from the "Environment" pane:



Or through the "File" menu, followed by the "Import Dataset" submenu:

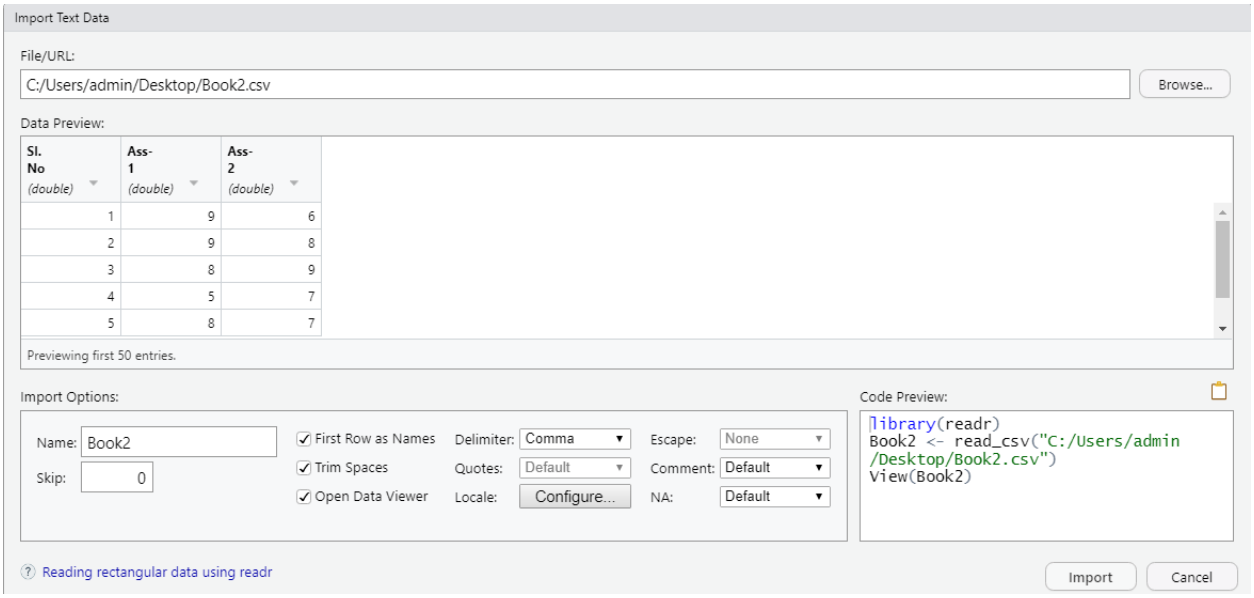


Importing data from Text and CSV files

Importing "From Text (readr)" files allows you to import CSV files and in general, character delimited files using the readr package. This Text importer provides support to:

- Import from the file system or a url

- Change column data types
- Skip or include-only columns
- Rename the data set
- Skip the first N rows
- Use the header row for column names
- Trim spaces in names
- Change the column delimiter
- Encoding selection
- Select quote, escape, comment and NA identifiers

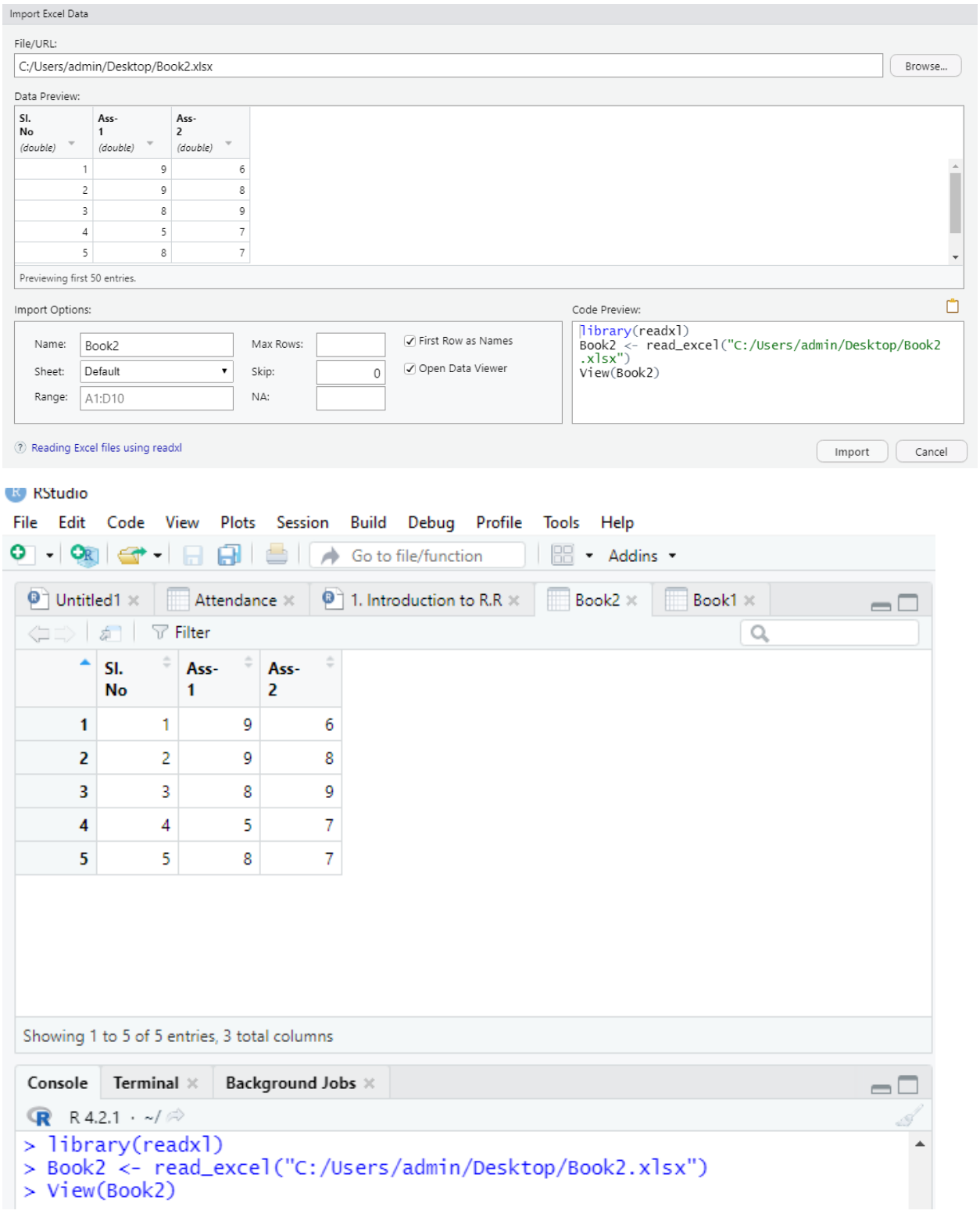


Or read.csv(file.choose()) can be used through R-console.

Importing data from Excel files

The Excel importer provides support to:

- Import from the file system or a url
- Change column data types
- Skip columns
- Rename the data set
- Select an specific Excel sheet
- Skip the first N rows
- Select NA identifiers



Conclusion:

Installation, input, output, import and various arithmetic operations have been explored in R

Experiment-2

Computing Summary Statistics /plotting and visualizing data using Tabulation and Graphical Representations

Aim:
The purpose of this experiment is to learn the different alignment of data set and various graphical representations in R

Procedure:
Step by step procedure to conduct the required experiment –
1. Arrangement of data using various R functions
2. Visualize the data set using various R functions

Code and Results:

```
#creating a vector empid
empid=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)
empid

##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15

# creating a vector age
age=c(30,37,45,32,50,60,35,32,34,43,32,30,43,50,60)
age

##  [1] 30 37 45 32 50 60 35 32 34 43 32 30 43 50 60

# creating a vector gender
gender=c(0,1,0,1,1,1,0,0,1,0,0,1,1,0,0)
gender

##  [1] 0 1 0 1 1 1 0 0 1 0 0 1 1 0 0

# creating a vector status
status=c(1,1,2,2,1,1,1,2,2,1,2,1,2,1,2)
status

##  [1] 1 1 2 2 1 1 1 2 2 1 2 1 2 1 2

# reating a data frame (Combining vectors)
empinfo=data.frame(empid,age,gender,status)
empinfo

##    empid age gender status
## 1      1  30      0      1
## 2      2  37      1      1
```

```
## 3      3  45      0      2
## 4      4  32      1      2
## 5      5  50      1      1
## 6      6  60      1      1
## 7      7  35      0      1
## 8      8  32      0      2
## 9      9  34      1      2
## 10    10  43      0      1
## 11    11  32      0      2
## 12    12  30      1      1
## 13    13  43      1      2
## 14    14  50      0      1
## 15    15  60      0      2
```

```
# labeling character to numeric
empinfo$gender=factor(empinfo$gender,labels=c("male","female"))
empinfo$gender
```

```
##  [1] male   female male   female female female male   male   female male
## [11] male   female female male   male
## Levels: male female
```

```
empinfo$status=factor(empinfo$status,labels=c("staff","faculty"))
empinfo$status
```

```
##  [1] staff   staff   faculty faculty staff   staff   staff   faculty
faculty
## [10] staff   faculty staff   faculty staff   faculty
## Levels: staff faculty
```

```
empinfo
```

```
##    empid age gender  status
## 1      1  30   male   staff
## 2      2  37 female   staff
## 3      3  45   male faculty
## 4      4  32 female faculty
## 5      5  50 female   staff
## 6      6  60 female   staff
## 7      7  35   male   staff
## 8      8  32   male faculty
## 9      9  34 female faculty
## 10    10  43   male   staff
## 11    11  32   male faculty
## 12    12  30 female   staff
## 13    13  43 female faculty
## 14    14  50   male   staff
## 15    15  60   male faculty
```

```r
# Extract male data
male=subset(empinfo,empinfo$gender=="male")
male
```

```
##    empid age gender  status
## 1      1  30   male   staff
## 3      3  45   male faculty
## 7      7  35   male   staff
## 8      8  32   male faculty
## 10    10  43   male   staff
## 11    11  32   male faculty
## 14    14  50   male   staff
## 15    15  60   male faculty
```

```r
# Extract female data
female=subset(empinfo, empinfo$gender=='female')
female
```

```
##    empid age gender  status
## 2      2  37 female   staff
## 4      4  32 female faculty
## 5      5  50 female   staff
## 6      6  60 female   staff
## 9      9  34 female faculty
## 12    12  30 female   staff
## 13    13  43 female faculty
```

```r
# summary statistics for empinfo data
summary(empinfo)
```

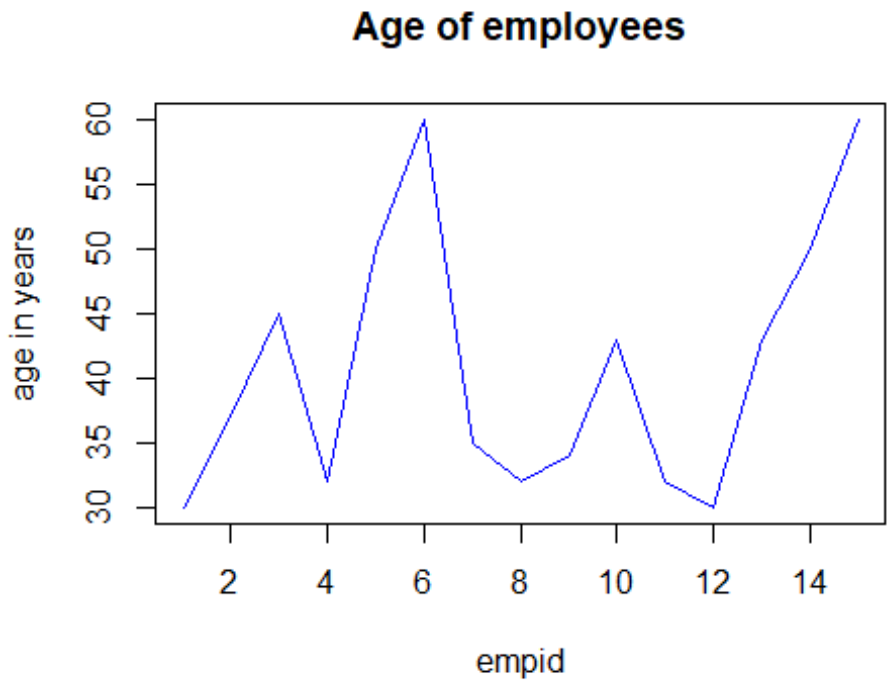
```
##      empid           age           gender      status
##  Min.   : 1.0   Min.   :30.00   male  :8   staff  :8
##  1st Qu.: 4.5   1st Qu.:32.00   female:7   faculty:7
##  Median : 8.0   Median :37.00
##  Mean   : 8.0   Mean   :40.87
##  3rd Qu.:11.5   3rd Qu.:47.50
##  Max.   :15.0   Max.   :60.00
```

```r
# summary statistics of male,female and age
summary(male)
```

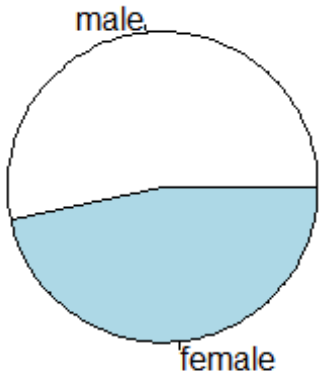
```
##      empid             age           gender      status
##  Min.   : 1.000   Min.   :30.00   male  :8   staff  :4
##  1st Qu.: 6.000   1st Qu.:32.00   female:0   faculty:4
##  Median : 9.000   Median :39.00
##  Mean   : 8.625   Mean   :40.88
##  3rd Qu.:11.750   3rd Qu.:46.25
##  Max.   :15.000   Max.   :60.00
```

```r
summary(female)
```

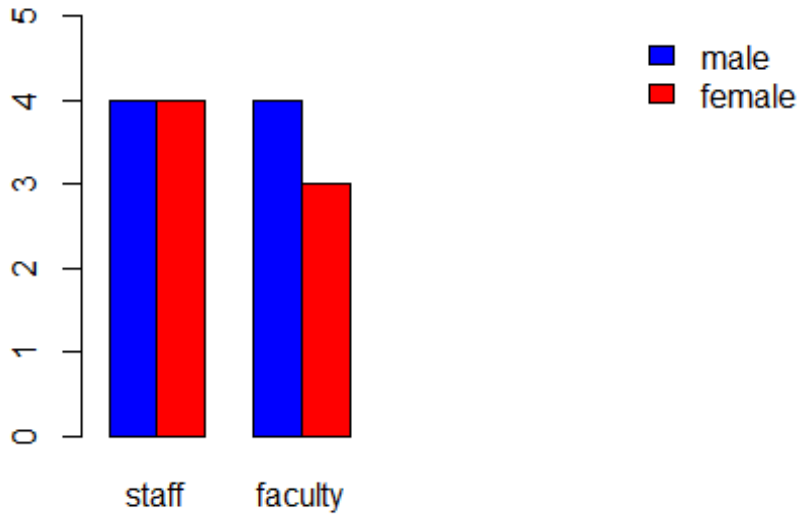
```
##      empid             age           gender      status
##  Min.   : 2.000   Min.   :30.00   male  :0   staff  :4
##  1st Qu.: 4.500   1st Qu.:33.00   female:7   faculty:3
##  Median : 6.000   Median :37.00
##  Mean   : 7.286   Mean   :40.86
##  3rd Qu.:10.500   3rd Qu.:46.50
##  Max.   :13.000   Max.   :60.00

summary(age)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   30.00   32.00   37.00   40.87   47.50   60.00

# creating table (one-way)
table1=table(empinfo$gender)
table1

##
##   male female
##      8      7

table2=table(empinfo$status)
table2

##
##   staff faculty
##       8       7

# creating table (two-way)
table3=table(empinfo$gender, empinfo$status)
table3

##
##          staff faculty
##   male       4       4
##   female     4       3

# Graphical representation (scatterplot)
plot(empinfo$age,type="l",main="Age of employees",xlab="empid",ylab="age in
years",col="blue")
```



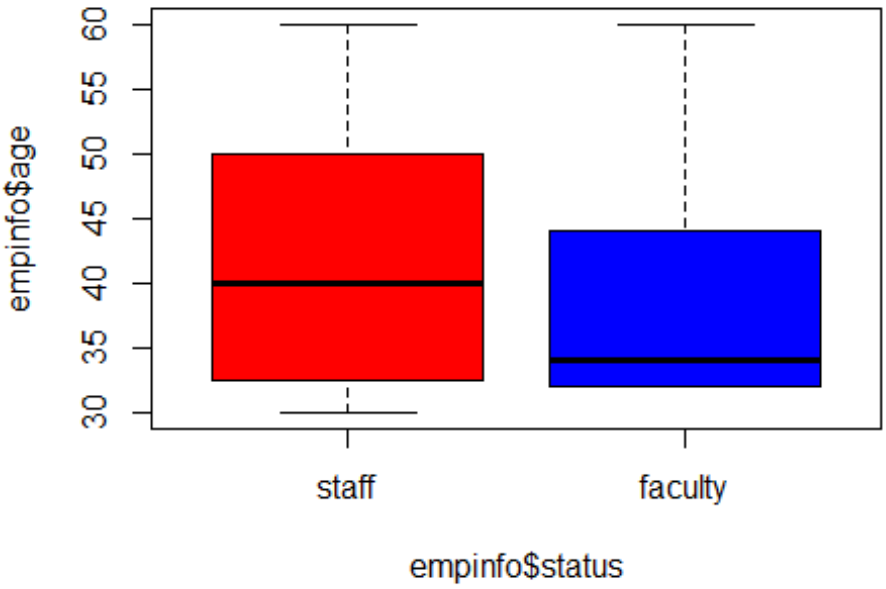

```
# Graphical representation (Pie chart)
pie(table1)
```



```r
# Graphical representation (Bar plot)
barplot(table3,beside=T,xlim=c(1,15),ylim=c(0,5),col=c("blue", "red"))
legend("topright",legend=rownames(table3),fill=c('blue','red'),bty="n")
```




```r
# Graphical representation (Box plot)
boxplot(empinfo$age~empinfo$status,col=c('red','blue'))
```



Conclusion:

Different alignment of data set and various graphical representations in R have been explored and executed.

Experiment-3

Applying correlation and simple linear regression model to real data set; computing and interpreting the coefficient of determination

Aim: To understand the simple correlation and linear regression with computation and interpretation

Introduction

The most commonly used techniques for investigating the relationship between two quantitative variables are correlation and linear regression. Correlation quantifies the strength of the linear relationship between a pair of variables, whereas regression expresses the relationship in the form of an equation.

Correlation:

A correlation coefficient is a statistical measure of the degree to which changes to the value of one variable predict change to the value of another. When the fluctuation of one variable reliably predicts a similar fluctuation in another variable, there's often a tendency to think that means that the change in one causes the change in the other.

Regression:

Regression analysis is a statistical tool to study the nature and extent of functional relationship between two or more variables and to estimate (or predict) the unknown values of dependent variable from the known values of independent variable.

Simple Linear Regression:

Simple linear regression model we have the following two regression lines:

1. Regression line of Y on X: This line gives the probable value of Y (Dependent variable) for any given value of X (Independent variable). Regression line of Y on X : $Y - \dot{Y} = byx\ (X - \dot{X})$ OR : $Y = a + bX$

2. Regression line of X on Y: This line gives the probable value of X (Dependent variable) for any given value of Y (Independent variable). Regression line of X on Y : $X - \dot{X} = bxy\ (Y - \dot{Y})$ OR : $X = a + bY$

In the above two regression lines or regression equations, there are two regression parameters, which are "a" and "b". Here "a" is unknown constant and "b" which is also denoted as "byx" or "bxy", is also another unknown constant popularly called as regression coefficient. Hence, these "a" and "b" are two unknown constants (fixed numerical values) which determine the position of the line completely.

Procedure:

- Input/Import the data set
- Determine the correlation and regression line using R functions
- Visualize the regression line using R functions

Code and Result:

```
# Problem-1
# Import the inbuilt data set "cars"
data=cars
data
```

```
##    speed dist
## 1      4    2
## 2      4   10
```

```
## 3      7    4
## 4      7   22
## 5      8   16
## 6      9   10
## 7     10   18
## 8     10   26
## 9     10   34
## 10    11   17
## 11    11   28
## 12    12   14
## 13    12   20
## 14    12   24
## 15    12   28
## 16    13   26
## 17    13   34
## 18    13   34
## 19    13   46
## 20    14   26
## 21    14   36
## 22    14   60
## 23    14   80
## 24    15   20
## 25    15   26
## 26    15   54
## 27    16   32
## 28    16   40
## 29    17   32
## 30    17   40
## 31    17   50
## 32    18   42
## 33    18   56
## 34    18   76
## 35    18   84
## 36    19   36
## 37    19   46
## 38    19   68
## 39    20   32
## 40    20   48
## 41    20   52
## 42    20   56
## 43    20   64
## 44    22   66
## 45    23   54
## 46    24   70
## 47    24   92
## 48    24   93
## 49    24  120
## 50    25   85
```

```r
# Summary of the data set
summary(data)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

```r
# Variance of "speed"
v1=var(data$speed)
v1
```

```
## [1] 27.95918
```

```r
# Variance of "dist"
v2=var(data$dist)
v2
```

```
## [1] 664.0608
```

```r
# Covariance between "speed" and "dist"
covariance=cov(data$speed,data$dist)
covariance
```

```
## [1] 109.9469
```

```r
#or
covariance=var(data$speed,data$dist)
covariance
```

```
## [1] 109.9469
```

```r
# correlation coefficient using Pearson's formula
corr=covariance/(sd(data$speed)*sd(data$dist))
corr
```

```
## [1] 0.8068949
```

```r
# or
corr=cor(data$speed,data$dist)
corr
```

```
## [1] 0.8068949
```

```r
# Test for association between paired samples
cor.test(data$speed,data$dist)
```

```
##
##  Pearson's product-moment correlation
##
```

```
## data:  data$speed and data$dist
## t = 9.464, df = 48, p-value = 1.49e-12
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6816422 0.8862036
## sample estimates:
##       cor
## 0.8068949
```

```
cor.test(data$speed,data$dist,method="pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  data$speed and data$dist
## t = 9.464, df = 48, p-value = 1.49e-12
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6816422 0.8862036
## sample estimates:
##       cor
## 0.8068949
```

```
cor.test(data$speed,data$dist,method="spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  data$speed and data$dist
## S = 3532.8, p-value = 8.825e-14
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.8303568
```

```
# Visualize the samples
plot(data$speed,data$dist)
# Linear Regression model of "speed" with respect to "dist"
regression1=lm(data$speed~data$dist)
regression1
```

```
##
## Call:
## lm(formula = data$speed ~ data$dist)
##
## Coefficients:
## (Intercept)    data$dist
##      8.2839       0.1656
```

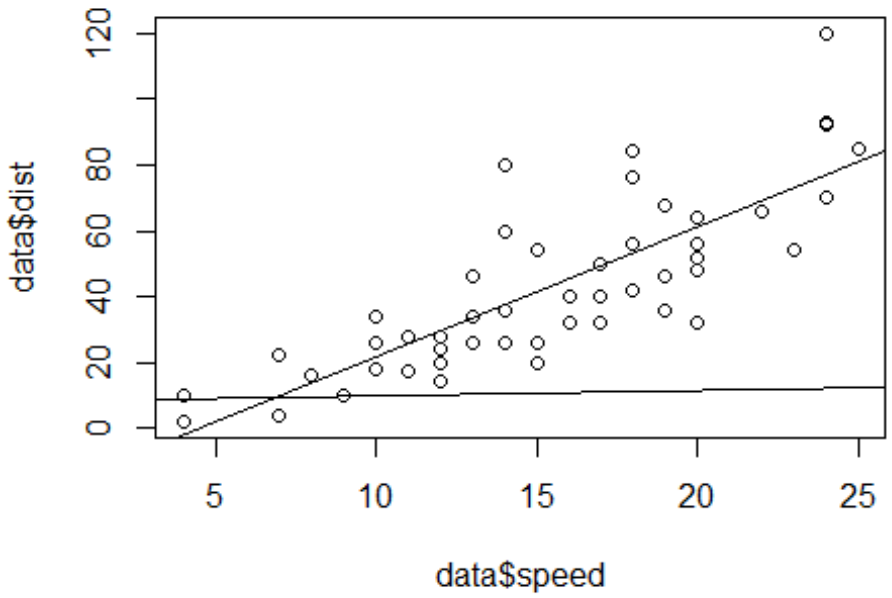
```r
# Visualize linear regression line
abline(regression1)
summary(regression1)
```

```
##
## Call:
## lm(formula = data$speed ~ data$dist)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.5293 -2.1550  0.3615  2.4377  6.4179
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.28391    0.87438   9.474 1.44e-12 ***
## data$dist    0.16557    0.01749   9.464 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.156 on 48 degrees of freedom
## Multiple R-squared:  0.6511, Adjusted R-squared:  0.6438
## F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

```r
# Linear Regression model of "dist" with respect to "speed"
regression2=lm(data$dist~data$speed)
regression2
```

```
##
## Call:
## lm(formula = data$dist ~ data$speed)
##
## Coefficients:
## (Intercept)   data$speed
##     -17.579        3.932
```

```r
abline(regression2)
```




```
summary(regression2)

##
## Call:
## lm(formula = data$dist ~ data$speed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.069  -9.525  -2.272   9.215  43.201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.5791     6.7584  -2.601   0.0123 *
## data$speed    3.9324     0.4155   9.464 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.38 on 48 degrees of freedom
## Multiple R-squared:  0.6511, Adjusted R-squared:  0.6438
## F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

Problem:-

The body weight and the BMI of 12 school going children are given in the following table

Weight	***15***	***26***	***27***	***25***	***25.5***	***27***	***32***	***18***	***22***	***20***	***26***	***24***
BMI	***13.35***	***16.12***	***16.74***	***16.00***	***13.59***	***15.73***	***15.65***	***13.85***	***16.07***	***12.8***	***13.65***	***14.42***

Let us fit a simple regression model BMI on weight and examine the results.

```
#Problem-2
weight=c(15,26,27,2,25.5,27,32,18,22,20,26,24)
weight
```

```
##  [1] 15.0 26.0 27.0  2.0 25.5 27.0 32.0 18.0 22.0 20.0 26.0 24.0
```

```
bmi=c(133.35,16.1,16.74,16,13.59,15.73,15.65,13.85,16.07,12.8,13.65,14.42)
bmi
```

```
##  [1] 133.35  16.10  16.74  16.00  13.59  15.73  15.65  13.85  16.07  12.80
## [11]  13.65  14.42
```

```
cor(weight,bmi)
```

```
## [1] -0.2841834
```

```
mode1<-lm(bmi~weight)
summary.lm(mode1)
```

```
##
## Call:
## lm(formula = bmi ~ weight)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.838 -10.253  -6.582  -2.659  99.734
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   52.334     30.978   1.689    0.122
## weight        -1.248      1.331  -0.937    0.371
##
## Residual standard error: 34.39 on 10 degrees of freedom
## Multiple R-squared:  0.08076,    Adjusted R-squared:  -0.01116
## F-statistic: 0.8786 on 1 and 10 DF,  p-value: 0.3707
```

Interpretation :

Correlation r=0.5790,which is the correlation coefficient between the body 'weight' and BMI.There is a positive correlation between these two variables.The Value of R^2 is 0.3353,which means that about 33.53% variation in BMI can be explained by 'weight'through this linear model.This is apparently low because more than 67% of variation remains unexplained.There could be several reasons for this and one of them is that there might be some other influencing variables that have not been included in the present model.

The F value shown in the above output gives the statistics for the variance ratio test of the regression model.The significance of F, which is given as 0.0485,is the p value of the F-test carried out in ANOVA. If this value is less than 0.05 we say that the regression is statistical significant at 5% level of significance .Here regression is significant which means that the relationship is not an occurrence by chance

In the above output we find b_0 is the intercept which value of 10.73487 and b_1 is the regression coefficient due to weight with a value of 0.1710.The regression coefficient is positive ,which shows that the BMI is positively related to weight,

The regression output can be written as mathematical equation

$$BMI = 10.7349 + 0.1710 * weight$$

Suppose body weight of one student is known as 25 kg.Using the above equation,the estimated BMI is 15.01.since this is only an estimate we have to intercept it as the average BMI corresponding to the given weight assuming that other parameters are unchanged.

Conclusion: The simple correlation and linear regression equation have been computed and interpreted.

Experiment - 4

Applying multiple linear regression model to real dataset; computing and interpreting the multiple coefficients of determination

Aim: To understand the multiple linear regression model with computation and interpretation using R

Introduction

Multiple linear regression attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. Every value of the independent variable x is associated with a value of the dependent variable y.
Multiple linear regression models are defined by the equation

$$Y=\beta 0+\beta 1X1+\beta 2X2+\ldots+\beta pXp+\epsilon$$

Procedure:

- Import the data set
- Determine the multiple linear regression using R functions
- Visualize the multiple linear regression using R functions

Note: Please make sure that the following package is already installed.

"Scatterplot 3d"

Codes and Results:

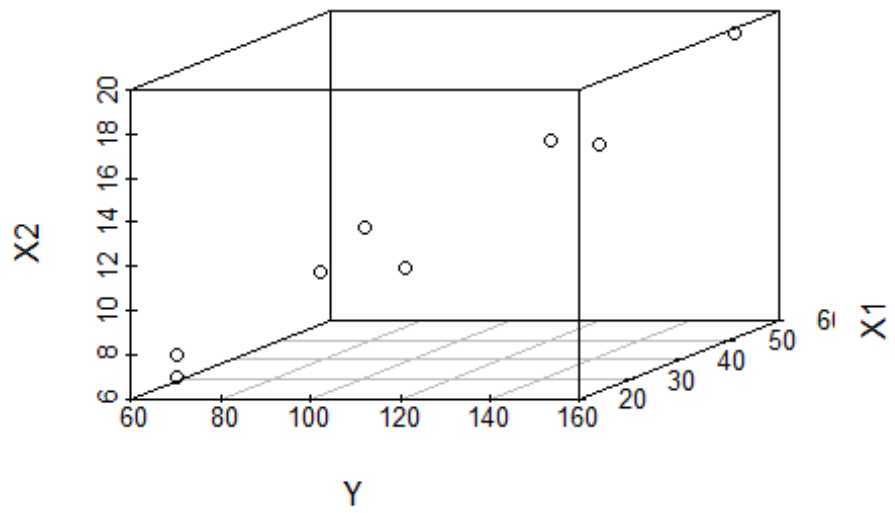
Problem 1: *The sale of a Product in lakhs of rupees(Y) is expected to be influenced by two variables namely the advertising expenditure X1 (in'OOORs) and the number of sales persons(X2) in a region. Sample data on 8 Regions of a state has given the following results*

Area	Y	X1	X2
1	110	30	11
2	80	40	10
3	70	20	7
4	120	50	15
5	150	60	19
6	90	40	12
7	70	20	8
8	120	60	14

```
# Input the variables
Y=c(110,80,70,120,150,90,70,120)
Y

## [1] 110  80  70 120 150  90  70 120

X1=c(30,40,20,50,60,40,20,60)
X1

## [1] 30 40 20 50 60 40 20 60

X2=c(11,10,7,15,19,12,8,14)
X2

## [1] 11 10  7 15 19 12  8 14

# Linear regression model of Y on X1 and X2
RegModel=lm(Y~X1+X2)
RegModel

##
## Call:
## lm(formula = Y ~ X1 + X2)
##
## Coefficients:
## (Intercept)           X1           X2
##     16.8314      -0.2442       7.8488

# Summary of the data
summary(RegModel)

##
## Call:
## lm(formula = Y ~ X1 + X2)
```

```
##
## Residuals:
##       1       2       3       4       5       6       7       8
##  14.157  -5.552   3.110  -2.355  -1.308 -11.250  -4.738   7.936
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  16.8314    11.8290   1.423   0.2140
## X1           -0.2442     0.5375  -0.454   0.6687
## X2            7.8488     2.1945   3.577   0.0159 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.593 on 5 degrees of freedom
## Multiple R-squared:  0.9191, Adjusted R-squared:  0.8867
## F-statistic:  28.4 on 2 and 5 DF,  p-value: 0.001862
```

```
# install.packages("scatterplot3d")
library(scatterplot3d)
# Plot the data set
scatterplot3d(Y,X1,X2)
```



Interpretation :

Now the regression the regression model is

$$Y = 16.834 - 0.2442 * X1 + 7.8488 * X2$$

Since R^2 is 0.9593 and the ANOVA shows that the F-ratio is significant, this model can be taken as good-fit in explaining the sales in terms of the other two variables

```
#Problem 2
data=mtcars
data
```

```
##                      mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4           21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag       21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710          22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive      21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout   18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
## Valiant             18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
## Duster 360          14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
## Merc 240D           24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
## Merc 230            22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
## Merc 280            19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## Merc 280C           17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
## Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
## Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
## Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
## Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
## Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
## Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
## Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
## Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
## Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
## Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
## AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
## Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
## Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
## Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
## Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
## Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
## Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
## Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
## Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
## Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

```
X=mtcars$mpg
X
```

```
##  [1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 17.8 16.4 17.3 15.2
10.4
```

```
## [16] 10.4 14.7 32.4 30.4 33.9 21.5 15.5 15.2 13.3 19.2 27.3 26.0 30.4 15.8
19.7
## [31] 15.0 21.4
```

```
Y=mtcars$disp
Y
```

```
##  [1] 160.0 160.0 108.0 258.0 360.0 225.0 360.0 146.7 140.8 167.6 167.6
275.8
## [13] 275.8 275.8 472.0 460.0 440.0  78.7  75.7  71.1 120.1 318.0 304.0
350.0
## [25] 400.0  79.0 120.3  95.1 351.0 145.0 301.0 121.0
```

```
Z=mtcars$hp
Z
```

```
##  [1] 110 110  93 110 175 105 245  62  95 123 123 180 180 180 205 215 230
66  52
## [20]  65  97 150 150 245 175  66  91 113 264 175 335 109
```

```
RegModel<- lm(Z~X+Y)
RegModel
```

```
##
## Call:
## lm(formula = Z ~ X + Y)
##
## Coefficients:
## (Intercept)            X            Y
##    172.2204      -4.2732       0.2614
```

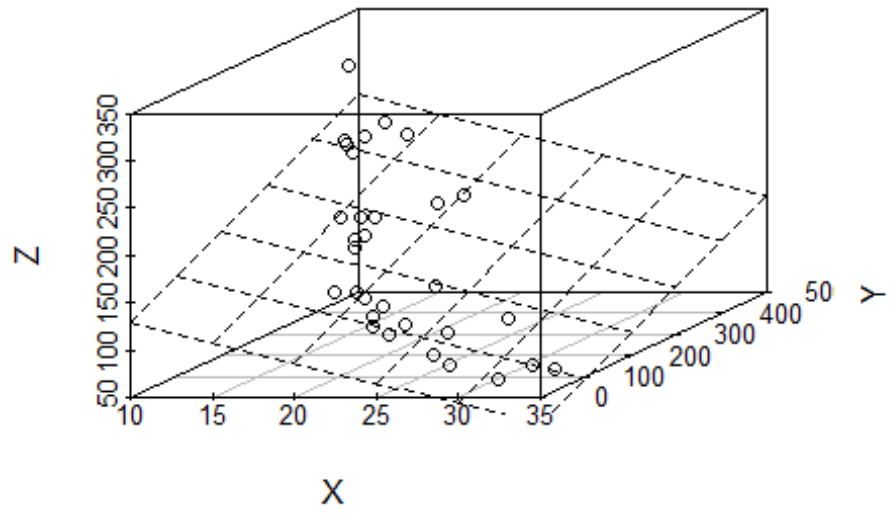
```
summary(RegModel)
```

```
##
## Call:
## lm(formula = Z ~ X + Y)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -48.70 -17.67 -10.16  10.12 148.19
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 172.2204    69.9014   2.464   0.0199 *
## X            -4.2732     2.3027  -1.856   0.0737 .
## Y             0.2614     0.1120   2.335   0.0267 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 41.01 on 29 degrees of freedom
## Multiple R-squared:  0.6653, Adjusted R-squared:  0.6423
## F-statistic: 28.83 on 2 and 29 DF,  p-value: 1.279e-07
```

```
library(scatterplot3d)
graph=scatterplot3d(X,Y,Z)
# Visualize the plane
graph$plane3d(RegModel)
```

Conclusion:

Multiple linear regression model has been explored and visualized.



Experiment - 5

Fitting the probability distributions: Binomial distribution

Aim: To understand discrete probability distribution using R

Introduction:
A discrete distribution is one in which the data can only take on certain values, for example integers. For a discrete distribution, probabilities can be assigned to the values in the distribution. These distributions model the probabilities of random variables that can have discrete values as outcomes. Example: Binomial distribution, Poisson distribution

A binomial distribution is a discrete probability distribution that gives the success probability in n Bernoulli trials. The probability of getting a success is given by p. It is represented as X ~ Binomial(n, p). The pmf is given as follows:

$$P(X = x) = nC_x p^x (1-p)^{n-x}, \qquad x = 0,1,2,\dots,n$$

Procedure:

- Input/Import the data set
- Determine the probabilities of the random variable using Binomial distribution in R
- Visualize the probability distribution using R functions

Problem:
Four coins are tossed simultaneously. What is the probability of getting (i) 2 heads (ii) atleast 2 heads (iii) atmost 2 heads (iv) Expectation of x (v) Variance of x (vi) Visualize the probability distribution

Code and Results:

```
# number of trials
n=4
n
```

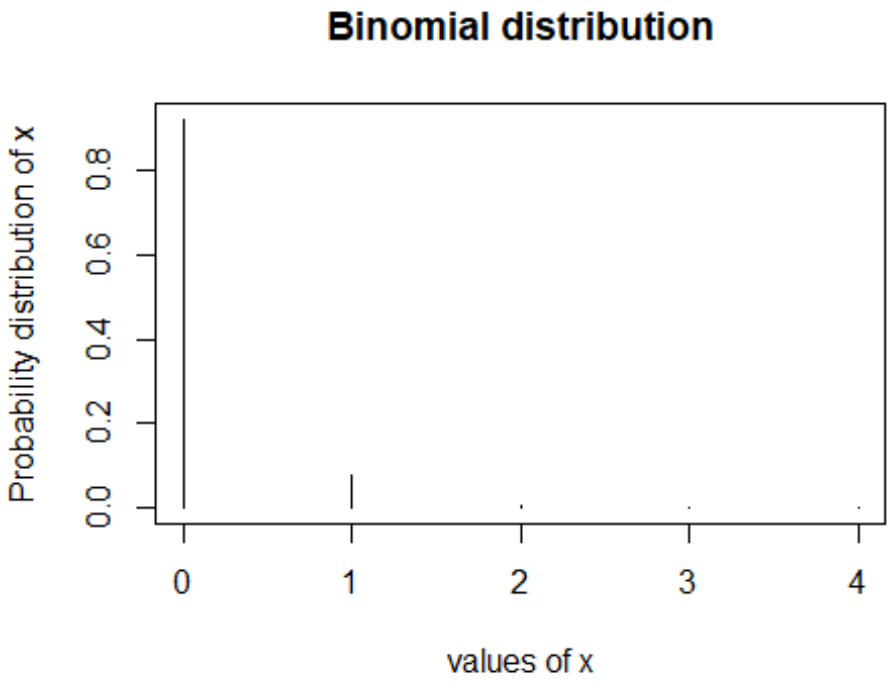
```
## [1] 4
```

```
#probability of success
p=0.02
p
```

```
## [1] 0.02
```

```
# (i) probability of getting exactly 2 heads
dbinom(2,n,p)
```

```
## [1] 0.00230496

# (ii) probability of getting atleast 2 heads
sum(dbinom(2:4,n,p))

## [1] 0.00233648

#or
1-pbinom(1,n,p)

## [1] 0.00233648

# (iii) probability of getting atmost 2 heads
sum(dbinom(0:2,n,p))

## [1] 0.9999685

# or
pbinom(2,n,p)

## [1] 0.9999685

#(iv) Expectation of x
x=0:n
px=dbinom(x,n,p)
Ex=weighted.mean(x,px)
Ex

## [1] 0.08

# (v) variance of x
Varx=weighted.mean(x*x,px)-(weighted.mean(x ,px))^2
Varx

## [1] 0.0784

# (vi) Visualize probability distribution
plot(x,px,type="h",xlab="values of x",ylab="Probability distribution of
x",main="Binomial distribution")
```



Conclusion: Suitable R functions have been explored to understand Binomial distribution.

Experiment – 6

Normal distribution, Poisson distribution

Aim: To understand Poisson distribution and Normal distribution using R functions

Introduction:
A discrete distribution is one in which the data can only take on certain values, for example integers. For a discrete distribution, probabilities can be assigned to the values in the distribution. These distributions model the probabilities of random variables that can have discrete values as outcomes. Example: Binomial distribution, Poisson distribution

Poisson distribution is a discrete probability distribution that is widely used in the field of finance. It gives the probability that a given number of events will take place within a fixed time period. The notation is written as X ~ Pois(λ), where λ>0. The pmf is given by the following formula:

$$P(X = x) = \frac{e^{-\lambda}\lambda^{x}}{x!}, \qquad x = 0,1,2, \dots$$

Procedure:

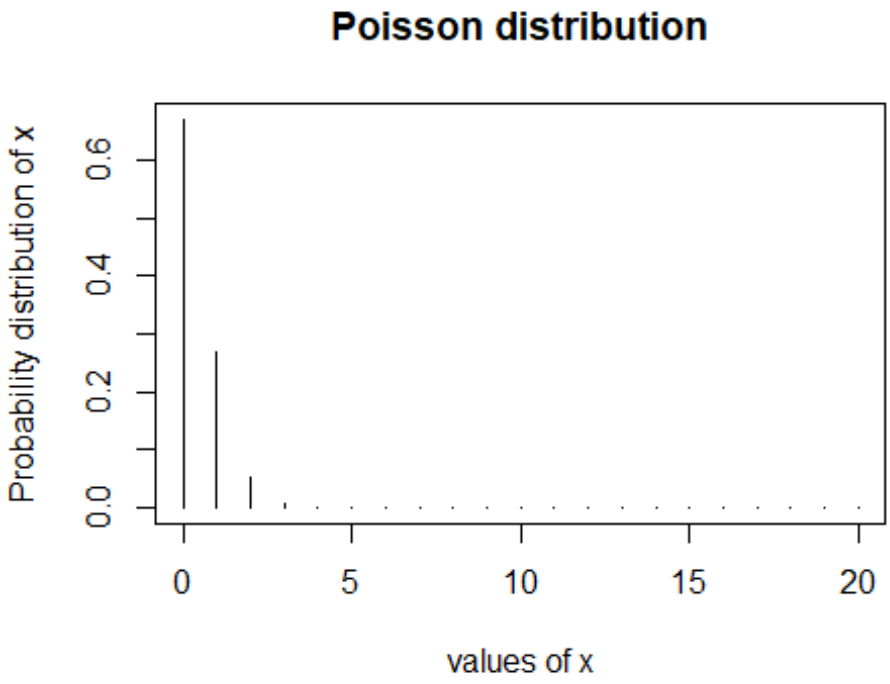
- Import the data set
- Determine the probabilities of the random variable using Poisson distribution in R
- Visualize the probability distribution using R functions

Problem:
A manufacturer of pins knows that 2% of his products are defective. If he sells pins in boxes of 20 and find the number of boxes containing (i) at least 2 defective (ii) exactly 2 defective (iii) at most 2 defective pins in a consignment of 1000 boxes (iv) plot the distribution (v) E(x) (vi) Variance of X?

Codes and Results:

```
#Poisson Distribution
# number of trails
m=20
m

## [1] 20

# probability of success
ps=0.02
# poisson parameter
lambda=m*ps
lambda
```

```
## [1] 0.4

#at least 2 defectives
p1=sum(dpois(2:m,lambda))
p1

## [1] 0.06155194

# (i) number of boxes containing at least 2 defectives
round(1000*p1)

## [1] 62

#exactly 2 defectives
p2=dpois(2,lambda)
p2

## [1] 0.0536256

# (ii) number of boxes containing exactly 2 defectives
round(1000*p2)

## [1] 54

#at most 2 defectives
p3=sum(dpois(0:2, lambda))
p3

## [1] 0.9920737

# (iii) number of boxes containing at most 2 defectives
round(1000*p3)

## [1] 992

# (iv) plot the distribution
x1=0:m
px1=dpois(x1,lambda)
plot(x1,px1,type="h",xlab="values of x",ylab="Probability distribution of
x",main="Poisson distribution")
```




```
#(v) E(x)
Ex1=weighted.mean(x1,px1)
Ex1

## [1] 0.4

# (vi) variance of x
Varx1=weighted.mean(x1*x1,px1)-(weighted.mean(x1 ,px1))^2
Varx1

## [1] 0.4
```

Normal Distribution

The Normal Distribution is defined by the probability density function for a continuous random variable in a system. Let us say, f(x) is the probability density function and X is the random variable.
$f(x) \geq 0$ for all $x \in (-\infty, \infty)$ and $\int_{-\infty}^{\infty} f(x)dx = 1$

The probability density function of normal or Gaussian distribution is given by;

$$f(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

Where,
x is the variable
μ is the mean
σ is the standard deviation

Procedure:

- Generating the data set
- Determine the probabilities of the random variable using Normal distribution in R
- Visualize the probability distribution using R functions

Problem:
A company finds that the time taken by one of its engineers to complete or repair job has a normal distribution with mean 20 minutes and S.D 5 minutes. State what proportion of jobs take:

i. Less than 15 minutes
ii. More than 25 minutes
iii. Between 15 and 25 minutes
iv. Plot the distribution
v. Table the distribution

Code and Results:

```
# Generating the data x
x=seq(0,40)
x
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24
## [26] 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
```

```
# find the density function of x
y=dnorm(x,mean=20,sd=5)
y
```

```
##  [1] 2.676605e-05 5.838939e-05 1.223804e-04 2.464438e-04 4.768176e-04
##  [6] 8.863697e-04 1.583090e-03 2.716594e-03 4.478906e-03 7.094919e-03
## [11] 1.079819e-02 1.579003e-02 2.218417e-02 2.994549e-02 3.883721e-02
## [16] 4.839414e-02 5.793831e-02 6.664492e-02 7.365403e-02 7.820854e-02
## [21] 7.978846e-02 7.820854e-02 7.365403e-02 6.664492e-02 5.793831e-02
## [26] 4.839414e-02 3.883721e-02 2.994549e-02 2.218417e-02 1.579003e-02
## [31] 1.079819e-02 7.094919e-03 4.478906e-03 2.716594e-03 1.583090e-03
## [36] 8.863697e-04 4.768176e-04 2.464438e-04 1.223804e-04 5.838939e-05
## [41] 2.676605e-05
```

```
# plot the normal distribution curve
plot(x,y,type='l')
# Proportion of jobs take less than 15 minutes
p1=pnorm(15,mean=20,sd=5)
p1
```

```
## [1] 0.1586553
```

```
x2=seq(0,15)
x2
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
```

```
y2=dnorm(x2,mean=20,sd=5)
y2
```

```
##  [1] 2.676605e-05 5.838939e-05 1.223804e-04 2.464438e-04 4.768176e-04
##  [6] 8.863697e-04 1.583090e-03 2.716594e-03 4.478906e-03 7.094919e-03
## [11] 1.079819e-02 1.579003e-02 2.218417e-02 2.994549e-02 3.883721e-02
## [16] 4.839414e-02
```

```
polygon(c(0,x2,15),c(0,y2,0),col='yellow')
```

```
#Proportion of jobs take more than 25 minutes
p2=pnorm(40,mean=20,sd=5)-pnorm(25,mean=20,sd=5)
p2
```

```
## [1] 0.1586236
```

```
x1=seq(25,40)
x1
```

```
##  [1] 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
```

```
y1=dnorm(x1,mean=20,sd=5)
y1
```

```
##  [1] 4.839414e-02 3.883721e-02 2.994549e-02 2.218417e-02 1.579003e-02
##  [6] 1.079819e-02 7.094919e-03 4.478906e-03 2.716594e-03 1.583090e-03
## [11] 8.863697e-04 4.768176e-04 2.464438e-04 1.223804e-04 5.838939e-05
## [16] 2.676605e-05
```

```
polygon(c(25,x1,40),c(0,y1,0),col='red')
```

```
#Proportion of jobs take between 15 and 25 minutes
p3=pnorm(25,mean=20,sd=5)-pnorm(15,mean=20,sd=5)
p3
```

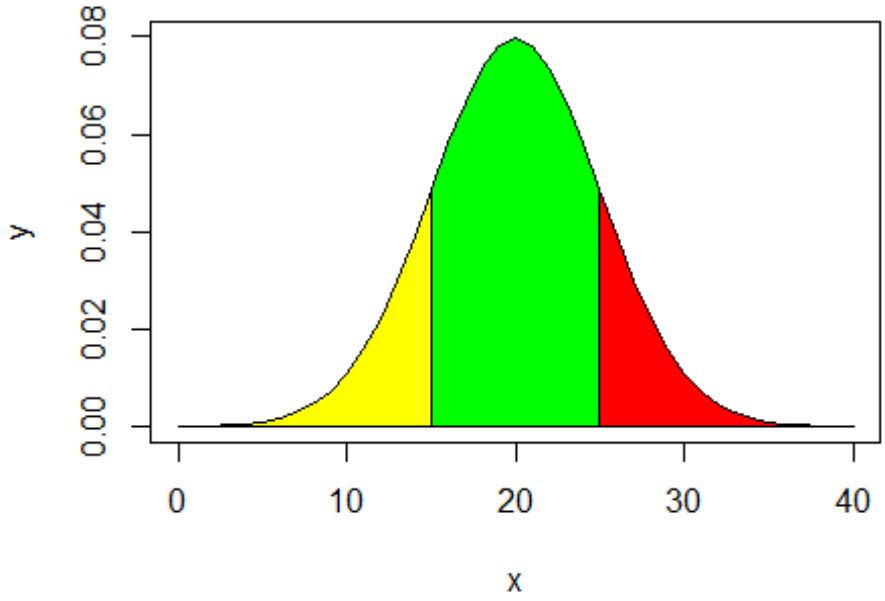
```
## [1] 0.6826895
```

```
x3=seq(15,25)
x3
```

```
##  [1] 15 16 17 18 19 20 21 22 23 24 25
```

```
y3=dnorm(x3,mean=20,sd=5)
y3
```

```
##  [1] 0.04839414 0.05793831 0.06664492 0.07365403 0.07820854 0.07978846
##  [7] 0.07820854 0.07365403 0.06664492 0.05793831 0.04839414

polygon(c(15,x3,25),c(0,y3,0),col='green')
```




```
# Probability distribution
data.frame(p1,p2,p3)

##          p1        p2        p3
## 1 0.1586553 0.1586236 0.6826895
```

Conclusion: Poisson distribution and Normal distribution have been explored using R.

Experiment-7

Testing of hypothesis for one sample mean and proportion from real time problems

Aim: To understand the testing of hypothesis for large sample tests using R functions

Testing of Hypothesis - Large Sample mean Test

Introduction

Hypothesis tests are used to make decisions or judgments about the value of a parameter, such as the population mean. There are two approaches for conducting a hypothesis test; the critical value approach and the P-value approach. Since a sample statistic is being used to make decisions or judgments about the value of a parameter it is possible that the decision reached is an error; there are two types of errors made when conducting a hypothesis test; Type I Error and Type II Error.

Test of significance of the difference between sample mean and population mean

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

Procedure:

- Import the data set
- Determine the critical value and sample statistic using R functions
- Conclude the problem using R functions

Problem

Suppose the mean weight of King Penguins found in an Antarctic colony last year was 15.4 kg. In a sample of 35 penguins same time this year in the same colony, the mean penguin weight is 14.6 kg. Assume the population standard deviation is 2.5 kg. At 0 .05 significance level, can we reject the null hypothesis that the mean penguin weight does not differ from last year?

Codes and Results:

```
# Input the sample mean
xbar=14.6
xbar

## [1] 14.6

# Input the population mean
mu0=15.4
mu0
```

```
## [1] 15.4
```

```
# Input the standard deviation
sigma=2.5
sigma
```

```
## [1] 2.5
```

```
# Input the sample size
n=35
n
```

```
## [1] 35
```

```
# Test Statistic
z=(xbar-mu0)/(sigma/sqrt(n))
z
```

```
## [1] -1.893146
```

```
# Level of significance
alpha=0.05
alpha
```

```
## [1] 0.05
```

```
# Two-tailed critical value
zhalfalpha=qnorm(1-(alpha/2))
zhalfalpha
```

```
## [1] 1.959964
```

```
c(-zhalfalpha,zhalfalpha)
```

```
## [1] -1.959964  1.959964
```

```
# To find p-value
pval=2*pnorm(z)
pval
```

```
## [1] 0.05833852
```

```
# conclusion
if(pval>alpha){print("Accept Null hypothesis")} else{print("Reject Null
hypothesis")}
```

```
## [1] "Accept Null hypothesis"
```

Testing of Hypothesis - Large Sample proportion Test

Test of significance of the difference between sample proportion and population proportion

$$z = \frac{p - P}{\sqrt{\frac{PQ}{n}}}$$

Procedure:

- Import the data set
- Determine the critical value and sample statistic using R functions
- Conclude the problem using R functions

Problem:

The fatality rate of typhoid patients is believed to be 17.26%. In a certain year 640 patients suffering from typhoid were treated in a metropolitan hospital and only 63 patients died. Can you consider the hospital efficient?

Code and Results:

```
# Input the data
# Size of the sample
n=640
n
```

```
## [1] 640
```

```
# Sample proportion
Sprop=63/n
Sprop
```

```
## [1] 0.0984375
```

```
# Population proportion
Pprop=0.1726
Pprop
```

```
## [1] 0.1726
```

```
# probability of failure
q=1-Pprop
q
```

```
## [1] 0.8274
```

```
# test statistic
z=(Sprop-Pprop)/sqrt(Pprop*q/n)
z
```

```
## [1] -4.964736
```

```
#critical value
E=qnorm(.975)
```

```r
# critical region
c(-E,E)
```

```
## [1] -1.959964  1.959964
```

```r
# confidence interval
Sprop+c(-E,E)*sqrt(Pprop*(1-Pprop)/n)
```

```
## [1] 0.06915985 0.12771515
```

```r
# Conclusion
if(z>-E && z<E){print("Hospital is not efficient")} else{print("Hospital is
efficient")}
```

```
## [1] "Hospital is efficient"
```

Conclusion: Testing of hypothesis for large sample tests using R functions has been explored and concluded.

Experiment - 8

Testing of hypothesis for two sample means and proportion from real time problems

Aim: To understand the testing of hypothesis for large sample tests using R functions

Testing of Hypothesis - Two Sample mean Test

Test of significance of the difference between two sample means

$$z = \frac{\overline{X}_1 - \overline{X}_2}{\sqrt{\frac{{\sigma_1}^2}{n_1} + \frac{{\sigma_2}^2}{n_2}}}$$

Procedure:

- Import the data set
- Determine the critical value and sample statistic using R functions
- Conclude the problem using R functions

Problem

In a random sample of size 500, the mean is found to be 20. In another independent sample of size 400, the mean is 15. Could the samples have been drawn from the same population with S.D 4?

Codes and Results:

```
# Input the sample mean
xbar=20
xbar

## [1] 20

ybar=15
ybar

## [1] 15

# Input the standard deviation
sigma=4
sigma

## [1] 4
```

```r
# Input the sample size
n1=500
n1
```

```
## [1] 500
```

```r
n2=400
n2
```

```
## [1] 400
```

```r
# Test Statistic
z=(xbar-ybar)/(sigma*sqrt((1/n1)+(1/n2)))
z
```

```
## [1] 18.6339
```

```r
# Level of significance
alpha=0.05
alpha
```

```
## [1] 0.05
```

```r
# Two-tailed critical value
zalpha=qnorm(1-(alpha/2))
zalpha
```

```
## [1] 1.959964
```

```r
# conclusion
if(z<=zalpha){print("Accept Null hypothesis")} else{print("Reject Null
hypothesis")}
```

```
## [1] "Reject Null hypothesis"
```

Testing of Hypothesis - Two Sample proportion Test

Test of significance of the difference between two sample proportions

$$z = \frac{p_1 - p_2}{\sqrt{PQ\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

Procedure:

- Import the data set
- Determine the critical value and sample statistic using R functions
- Conclude the problem using R functions

Problem:

In a large city A, 20% of a random sample of 900 schools boys had a slight physical defect. In another large city B, 18.5% of a random sample of 1600 school boys had the same defect. Is the difference between the proportions significant?

Code and Results:

```
# Input the sample proportions
p1=0.20
p1
```

```
## [1] 0.2
```

```
p2=0.185
p2
```

```
## [1] 0.185
```

```
# Input the sample size
n1=900
n1
```

```
## [1] 900
```

```
n2=1600
n2
```

```
## [1] 1600
```

```
# To find approximate population proportion
P=(n1*p1+n2*p2)/(n1+n2)
P
```

```
## [1] 0.1904
```

```
Q=1-P
# Test Statistic
z=(p1-p2)/sqrt(P*Q*sqrt((1/n1)+(1/n2)))
z
```

```
## [1] 0.1871665
```

```
# Level of significance
alpha=0.05
alpha
```

```
## [1] 0.05
```

```r
# Two-tailed critical value
zalpha=qnorm(1-(alpha/2))
zalpha
```

```
## [1] 1.959964
```

```r
# conclusion
if(z<=zalpha){print("Accept Null hypothesis")} else{print("Reject Null
hypothesis")}
```

```
## [1] "Accept Null hypothesis"
```

Conclusion: Testing of hypothesis for large sample tests using R functions has been explored and concluded.

Experiment - 9

Applying the t-test for independent and dependent samples

Aim: To understand the Student's t-test for independent and dependent samples using R

Testing of Hypothesis – t-Test

Introduction

If the sample size is less than 30 i.e., n < 30, the sample may be regarded as small sample.

Student's t-test:

The student's t-test is mentioned as a method of testing the theory about the mean of a small sample drawn from a normally distributed population where the standard deviation of the given population is unknown.
The t distribution belonging under a family of curves in which the number of degrees of freedom specifies a particular curve. As the sample size (and the degrees of freedom) increases, the t distribution approaches the bell shape of the standard normal distribution. In common, for tests involving the mean of a sample of size greater than 30, then the normal distribution is applied.

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

Paired t- test:

The paired *t*-test is a method used to test whether the mean difference between pairs of measurements is zero or not. We can use the test when your data values are paired measurements. For example, you might have before-and-after measurements for a group of people. Also, the distribution of differences between the paired measurements should be normally distributed.

$$t = \frac{\mu_d}{\frac{s}{\sqrt{n}}}$$

Problem: 1 (Student's t-test)

Two independent samples of sizes 8 and 7 contained the following values:

Sample 1	19	17	15	21	16	18	16	14
Sample 2	15	14	15	19	15	18	16	20

Is the difference between the sample means significant?

Code and Results:

```
# Problem 1

# input the data
sample1=c(19,17,15,21,16,18,16,14)
sample1

## [1] 19 17 15 21 16 18 16 14

sample2=c(15,14,15,19,15,18,16,20)
sample2

## [1] 15 14 15 19 15 18 16 20

# output using t-distribution
t=t.test(sample1,sample2)
t

##
##  Welch Two Sample t-test
##
## data:  sample1 and sample2
## t = 0.44721, df = 13.989, p-value = 0.6616
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.898128  2.898128
## sample estimates:
## mean of x mean of y
##      17.0      16.5

# test-statistic
cv=t$statistic
cv

##         t
## 0.4472136

#critical value
tv=qt(0.975,14)
tv

## [1] 2.144787

#conclusion
if(cv <= tv){print("Accept Ho")} else{print("Reject Ho")}

## [1] "Accept Ho"
```

Problem: 2 (Paired t-test)

The following data relate to the marks obtained by 10 students in two test, one held at the beginning of a year and the other at the end of the year after intensive coaching. Do the data indicate that the students have got benefited by coaching?

Test 1	19	17	15	21	16	18	16	14	19	20
Test 2	15	14	15	19	15	18	16	20	22	19

```
# Problem 2

#Paired- t-test
# input the data
test1=c(19,17,15,21,16,18,16,14,19,20)
test1

##  [1] 19 17 15 21 16 18 16 14 19 20

test2=c(15,14,15,19,15,18,16,20,22,19)
test2

##  [1] 15 14 15 19 15 18 16 20 22 19

t=t.test(sample1,sample2,paired=TRUE)
t

##
##  Paired t-test
##
## data:  sample1 and sample2
## t = 0.46771, df = 7, p-value = 0.6542
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.02789  3.02789
## sample estimates:
## mean of the differences
##                     0.5

# level of significance
alpha=0.05
# p-value
tv=t$p.value
tv

## [1] 0.6542055

# conclusion
if(tv >alpha){print("Accept Ho")} else{print("Reject Ho")}

## [1] "Accept Ho
```

F-test

When pairs of samples are taken from a normal population, the ratios of the variances of the samples in each pair will always follow the same distribution, the F-distribution.

The F-statistic is simply:

$$F = \frac{\sigma_1^2}{\sigma_2^2}$$

$$\sigma_1^2 = \frac{n_1 s_1^2}{n_1 - 1}, \quad d.f = n_1 - 1$$

$$\sigma_2^2 = \frac{n_2 s_2^2}{n_2 - 1}, \quad d.f = n_2 - 1$$

Problem: 3 (F-test)

Two independent samples of sizes 8 and 7 contained the following values:

Sample 1	19	17	15	21	16	18	16	14
Sample 2	15	14	15	19	15	18	16	20

Is the difference between the sample means significant?

Procedure:

- Import the data set
- Determine the critical value and sample statistic using R functions
- Conclude the problem using R functions

Codes and Results:

```
# Problem 3

# Variance test or F-test
sample1=c(19,17,15,21,16,18,16,14)
sample1

## [1] 19 17 15 21 16 18 16 14

sample2=c(15,14,15,19,15,18,16,20)
sample2

## [1] 15 14 15 19 15 18 16 20

# output using t-distribution
f=var.test(sample1,sample2)
f

##
##  F test to compare two variances
```

```
##
## data:  sample1 and sample2
## F = 1.0588, num df = 7, denom df = 7, p-value = 0.9418
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.2119805 5.2887274
## sample estimates:
## ratio of variances
##           1.058824
```

```
# test-statistic
cv=f$statistic
cv
```

```
##        F
## 1.058824
```

```
#critical value
tv=qf(0.95,7,7)
tv
```

```
## [1] 3.787044
```

```
#conclusion
if(cv <= tv){print("Accept Ho")} else{print("Reject Ho")}
```

```
## [1] "Accept Ho"
```

Conclusion: Student’s t-test and F-test have been explored and executed.

Experiment – 10

Applying Chi-square test for goodness of fit test and Contingency test to real dataset

Aim: To understand Chi-square test for goodness of fit and independent of attributes using R

Introduction

A Pearson's chi-square test is a statistical test for categorical data. It is used to determine whether your data are significantly different from what you expected. There are two types of Pearson's chi-square tests:

- The **chi-square goodness of fit test** is used to test whether the frequency distribution of a categorical variable is different from your expectations.
- The **chi-square test of independence** is used to test whether two categorical variables are related to each other.

Both of Pearson's chi-square tests use the same formula to calculate the test statistic, chi-square (X^2):

$$X^2 = \sum \frac{(O - E)^2}{E}$$

Where:

- X^2 is the chi-square test statistic
- *O* is the observed frequency
- *E* is the expected frequency

Procedure:

- Import the data set
- Determine the critical value and sample statistic using R functions
- Conclude the problem using R functions

Problem: 1

Five coins are tossed 256 times. The number of heads observed by binomial distribution is given below. Examine if the coins are unbiased by employing chi-square goodness of fit.

No. of heads	0	1	2	3	4	5
Frequency	5	35	75	84	45	12

Codes and Results:

```
# Problem : 1

# Goodness of fit

# Number of coins
n=5
n

## [1] 5

# level of significance
alpha=0.05
alpha

## [1] 0.05

N=256 # Total number of tosses
N

## [1] 256

P = 0.5  # probability of getting head
P

## [1] 0.5

x = c(0:n);x

## [1] 0 1 2 3 4 5

obf = c(5,35,75,84,45,12)# observed frequencies
obf

## [1]  5 35 75 84 45 12

exf = (dbinom(x,n,P)*256) # expected frequencies
exf

## [1]  8 40 80 80 40  8

# check the condition if the observed and expected frequencies sum are equal
sum(obf)

## [1] 256

sum(exf)

## [1] 256

# output using Chisq-distribution
chisq<-sum((obf-exf)^2/exf)
cv = chisq;cv
```

```
## [1] 4.8875

# critical value using Chisq-distribution
tv = qchisq(1-alpha,n);tv

## [1] 11.0705

# Hypothesis conclusion
if(cv <= tv){print("Accept H0/Fit is good")} else{print("Reject H0/Fit is not
good")}

## [1] "Accept H0/Fit is good"
```

Problem: 2

From the following information state whether the condition of the child is associated with the condition of the house.

Condition of child	**Condition of house Clean**	**Condition of house dirty**
Clean	69	51
Fairly clean	81	20
dirty	35	44

Codes and Results:

```
# Problem : 2

# Independent of attributes
# Input the data
data<-matrix(c(69,51,81,20,35,44),ncol=2,byrow=T)
data

##      [,1] [,2]
## [1,]   69   51
## [2,]   81   20
## [3,]   35   44

# number of data
l=length(data);l

## [1] 6

# output by Chisq-distribution
cv=chisq.test(data)
cv

##
##  Pearson's Chi-squared test
```

```
##
## data:  data
## X-squared = 25.629, df = 2, p-value = 2.721e-06
```

```
# p-value
cv=cv$p.value
cv
```

```
## [1] 2.72114e-06
```

```
# Hypothesis conclusion
if(cv >alpha){print("Attributes are independent")} else{print("Attributes are
not independent")}
```

```
## [1] "Attributes are not independent"
```

Conclusion: Chi-squared test has been explored through R functions.

Experiment – 11

Performing ANOVA for real dataset for Completely Randomized Design, Randomized Block design, Latin square Design

Aim: To understand the one way, two way and three way analysis of variances using R

Completely Randomized Design

Introduction

Responses among experimental units vary due to many different causes, known and unknown. The process of the separation and comparison of sources of variation is called the Analysis of Variance (AOV).
The AOV can be used for this purpose. It involves:
1. The partitioning of the total sum of squares of the experiment into each specified source of variation.
2. The estimation of the variance per experimental unit from these sources of variation.
3. The comparison of these variances by F-tests, which will lead to conclusions concerning the equality of the means.
The completely randomized design (CRD) refers to the random assignment of experimental units to a set of treatments. It is essential to have more than one experimental unit per treatment to estimate the magnitude of experimental error and to make probability statements concerning treatment effects.

Analysis of variance of a CRD

Source	df	Sum of squares (SS)	Mean squares (MS)	Observed F
Total	kr - 1	TSS		
Between treatments	k - 1	SST	MST	MST/MSE
Within treatments (experimental error)	k(r - 1)	SSE	MSE	

where r is the replication number per treatment.

Procedure:

- Import the data set
- Determine the summary and ANOVA using R functions
- Visualize the problem using R functions

Problem:

A car rental agency, which uses 5 different brands of tyres in the process of deciding the brand of tyre to purchase as standard equipment for its fleet, finds that each of 5 tyres of each brand last the following number of kilometres (in thousands):

A	B	C	D	E
36	46	35	45	41
37	39	42	36	39
42	35	37	39	37
38	37	43	35	35
47	43	38	32	38

Test the hypothesis that the five brands have almost the same average life.

Code and Results:

```
#One-way ANOVA
# Types of tyres
A=c(36,37,42,38,47)
B=c(46,39,35,37,43)
C=c(35,42,37,43,38)
D=c(45,36,39,35,32)
E=c(41,39,37,35,38)
group<-data.frame(cbind(A,B,C,D,E))
group
```

```
##    A  B  C  D  E
## 1 36 46 35 45 41
## 2 37 39 42 36 39
## 3 42 35 37 39 37
## 4 38 37 43 35 35
## 5 47 43 38 32 38
```

```
summary(group)
```

```
##        A               B               C               D               E
##  Min.   :36   Min.   :35   Min.   :35   Min.   :32.0   Min.   :35
##  1st Qu.:37   1st Qu.:37   1st Qu.:37   1st Qu.:35.0   1st Qu.:37
##  Median :38   Median :39   Median :38   Median :36.0   Median :38
```

```
##  Mean   :40   Mean   :40   Mean   :39   Mean   :37.4   Mean   :38
##  3rd Qu.:42   3rd Qu.:43   3rd Qu.:42   3rd Qu.:39.0   3rd Qu.:39
##  Max.   :47   Max.   :46   Max.   :43   Max.   :45.0   Max.   :41
```

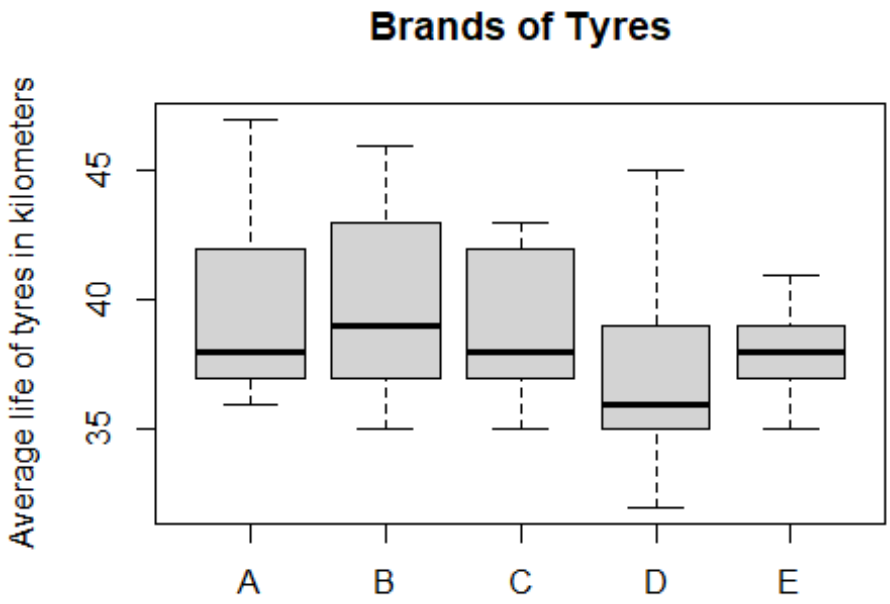
```
# stack vector from data frame
stgr<-stack(group);stgr
```

```
##    values ind
## 1      36   A
## 2      37   A
## 3      42   A
## 4      38   A
## 5      47   A
## 6      46   B
## 7      39   B
## 8      35   B
## 9      37   B
## 10     43   B
## 11     35   C
## 12     42   C
## 13     37   C
## 14     43   C
## 15     38   C
## 16     45   D
## 17     36   D
## 18     39   D
## 19     35   D
## 20     32   D
## 21     41   E
## 22     39   E
## 23     37   E
## 24     35   E
## 25     38   E
```

```
# completely randomized design
crd<-aov(values~ind,data=stgr)
# ANOVA table
summary(crd)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## ind          4   27.4    6.86   0.422  0.791
## Residuals   20  325.2   16.26
```

```
# Visualization of data
boxplot(group, ylab="Average life of tyres in kilometers",main="Brands of
Tyres")
```



Randomized Block Design

Introduction

A randomized block design is a type of experiment where participants who share certain characteristics are grouped together to form blocks, and then the treatment (or intervention) gets randomly assigned within each block. The objective of the randomized block design is to form groups where participants are similar, and therefore can be compared with each other.

ANOVA Table for a Randomized Block Design

Source	*df*	*SS*	*MS*	*F*
Treatments	$k-1$	SST	$MST = SST/(k-1)$	MST/MSE
Blocks	$b-1$	SSB	$MSB = SSB/(b-1)$	MSB/MSE
Error	$n-k-b+1$	SSE	$MSE = SSE/(n-k-b+1)$	
Total	$n-1$	$TotalSS$		

Procedure:

- Import the data set
- Determine the summary and ANOVA using R functions
- Visualize the problem using R functions

Problem:

The following table gives monthly sales (in thousand rupees) of a certain firm in the 3 states by its four salesmen.

States	**Salesmen**			
	I	II	III	IV
A	6	5	3	8
B	8	9	6	5
C	10	7	8	7

Setup the analysis of variance table and test whether there is any significant difference (i) between the salesmen (ii) between sales in the states.

Code and Results:

```
#Monthly sales of States
StateA=c(6,5,3,8)
StateA

## [1] 6 5 3 8

StateB=c(8,9,6,5)
StateB

## [1] 8 9 6 5

StateC=c(10,7,8,7)
StateC

## [1] 10  7  8  7

#frame the data set
Group<-data.frame(cbind(StateA,StateB,StateC))
Group

##   StateA StateB StateC
## 1      6      8     10
## 2      5      9      7
## 3      3      6      8
## 4      8      5      7

Sales=c(t(as.matrix(Group))); Sales

##  [1]  6  8 10  5  9  7  3  6  8  8  5  7

f=c("State A","State B","State C")
f

## [1] "State A" "State B" "State C"
```

```r
g=c("Salesman1","Salesman2","Salesman3","Salesman4")
g
```

```
## [1] "Salesman1" "Salesman2" "Salesman3" "Salesman4"
```

```r
# number of columns
k=ncol(Group)
k
```

```
## [1] 3
```

```r
# number of rows
n=nrow(Group)
n
```

```
## [1] 4
```

```r
# Generate factor levels of States
States=gl(k,1,n*k,factor(f))
States
```

```
##  [1] State A State B State C State A State B State C State A State B State
C
## [10] State A State B State C
## Levels: State A State B State C
```

```r
# Generate factor levels of Salesmen
Salesmen=gl(n,k,n*k,factor(g))
Salesmen
```

```
##  [1] Salesman1 Salesman1 Salesman1 Salesman2 Salesman2 Salesman2 Salesman3
##  [8] Salesman3 Salesman3 Salesman4 Salesman4 Salesman4
## Levels: Salesman1 Salesman2 Salesman3 Salesman4
```

```r
# ANOVA table
anova=aov(Sales ~ States + Salesmen)
summary(anova)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## States       2 12.667   6.333   1.839  0.238
## Salesmen     3  8.333   2.778   0.806  0.535
## Residuals    6 20.667   3.444
```

Latin square Design

The Latin square design applies when there are repeated exposures/treatments and two other factors. This design avoids the excessive numbers required for full three way ANOVA.

The analysis of variance table for LSD is as follows:

Sources of Variation	Degrees of Freedom	Sum of Squares	Mean Squares	F-ratio
Rows	t-1	RSS	RMS	RMS/EMS
Columns	t-1	CSS	CMS	CMS/EMS
Treatments	t-1	TrSS	TrMS	TrMS/EMS
Error	(t-1)(t-2)	ESS	EMS	
Total	t^2-1	TSS		

Problem:

Perform Latin Square Design for the following.

Consider analyzing the productivity of five kinds of manure, five kinds of cultivation, and five kinds of crops. As follows, the data are organized in a Latin Square format:

	cultP	cultQ	cultR	cultS	cultT
manure1	"P42"	"R47"	"Q55"	"S51"	"T44"
manure2	"T45"	"Q54"	"R52"	"P44"	"S50"
manure3	"R41"	"P46"	"DS7"	"T47"	"Q48"
manure4	"Q56"	"S52"	"T49"	"R50"	"P43"
manure5	"S47"	"T49"	"P45"	"Q54"	"R46"

The three factors are: manure (manure1:5), cultivation (cultP:T), crop(P:T).

Codes and Results:

```
#creating dataframes in R
manure=c(rep("manure1",1), rep("manure2",1), rep("manure3",1),
rep("manure4",1), rep("manure5",1))
cultivation=c(rep("cultP",5), rep("cultQ",5), rep("cultR",5), rep("cultS",5),
rep("cultT",5))
```

```
crop=c("P","T","R","Q","S", "R","Q","P","S","T", "Q","R","S","T","P",
"S","P","T","R","Q", "T","S","Q","P","R")
freq=c(42,45,41,56,47, 47,54,46,52,49, 55,52,57,49,45, 51,44,47,50,54,
44,50,48,43,46)
data=data.frame(cultivation,manure,crop,freq)
data
```

```
##    cultivation  manure crop freq
## 1        cultP manure1    P   42
## 2        cultP manure2    T   45
## 3        cultP manure3    R   41
## 4        cultP manure4    Q   56
## 5        cultP manure5    S   47
## 6        cultQ manure1    R   47
## 7        cultQ manure2    Q   54
## 8        cultQ manure3    P   46
## 9        cultQ manure4    S   52
## 10       cultQ manure5    T   49
## 11       cultR manure1    Q   55
## 12       cultR manure2    R   52
## 13       cultR manure3    S   57
## 14       cultR manure4    T   49
## 15       cultR manure5    P   45
## 16       cultS manure1    S   51
## 17       cultS manure2    P   44
## 18       cultS manure3    T   47
## 19       cultS manure4    R   50
## 20       cultS manure5    Q   54
## 21       cultT manure1    T   44
## 22       cultT manure2    S   50
## 23       cultT manure3    Q   48
## 24       cultT manure4    P   43
## 25       cultT manure5    R   46
```

```
#recreating the original table, using the matrix function
matrix(data$crop,5,5)
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,] "P"  "R"  "Q"  "S"  "T"
## [2,] "T"  "Q"  "R"  "P"  "S"
## [3,] "R"  "P"  "S"  "T"  "Q"
## [4,] "Q"  "S"  "T"  "R"  "P"
## [5,] "S"  "T"  "P"  "Q"  "R"
```

```
matrix(data$freq,5,5)
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]   42   47   55   51   44
## [2,]   45   54   52   44   50
## [3,]   41   46   57   47   48
```

```
## [4,]   56   52   49   50   43
## [5,]   47   49   45   54   46
```

```
#creating the anova table
fit=lm(freq~manure+cultivation+crop,data)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: freq
##             Df Sum Sq Mean Sq F value   Pr(>F)
## manure       4  17.76   4.440  0.7967 0.549839
## cultivation  4 109.36  27.340  4.9055 0.014105 *
## crop         4 286.16  71.540 12.8361 0.000271 ***
## Residuals   12  66.88   5.573
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Conclusion: The problems on ANOVA have been executed using R