LISA 2020

Laboratory for Interdisciplinary Statistical Analysis

Basic Statistical Methods for Physical Sciences

With Examples in R

Monday Osagie Adenomon, Ph.D, FRSS, FASI, CStat

Chartered Statistician (CStat, RSS-UK)

Facilitator

. . . from Background of Theories to the Realm of Realities . . .

Foundation of Laboratory for Econometrics & Applied Statistics of Nigeria (Aka FOUND-LEAS-IN-NIGERIA)

…we Found Leas in Nigeria, Come Invest in it…

Supported by

ISBN: 978-978-57688-1-7

Contacts

Facilitator

Monday Osagie Adenomon, Ph.D
Department of Statistics,
Nasarawa State University, Keffi, P.M.B 1022, Keffi,
Nasarawa State, Nigeria.
adenomonmo@nsuk.edu.ng ; +2347036990145

Eric Vance, Ph.D
Associate Professor,
Department of Applied Mathematics,
University of Colorado Boulder, USA.
Director of the Laboratory for Interdisciplinary Statistical
Analysis (LISA).

DEDICATION

This Workshop book is dedicated to all users of Statistics

PREFACE

This course and workshop book is intended to take statistics from the background of theories to the realm of realities. Also, this little book is to set the plat form towards interdisciplinary Statistics with interest to physical Sciences. This course and workshop is divided into six sections (Chapters): General introduction of Statistical computing and R software; Descriptive and Elementary Statistics (including Probability Distributions); Correlation and Regression Analyses (Ordinary and Robust) including diagnostic testing and comparison of estimates with real observation; and Analysis of Variance (ANOVA); Non parametric statistics (Mann Whitney, Kruskall Wallis, Friendman tests) and Bootstrapping with R. Without doubt this course and workshop book will meet the desires of users of Statistics.

TABLE OF CONTENT

Quotes

For God so loved the world, that he gave his only begotten Son, that whosoever believeth in him should not perish, but have everlasting life. **John 3:16**

And unto man he said, Behold, the fear of the Lord, that is wisdom; and to depart from evil is understanding. **Job 28:28**

Whatsoever thy hand findeth to do, do it with thy might; for there is no work, nor device, nor knowledge, nor wisdom, in the grave, whither thou goest- **Ecclesiastes 9:10**

Success is what comes after you stop making excuses- **Lius Galarza**

Statistician is someone that plays in everybody backyard

The teaching of Statistics has no value if it is not supported by Statistical computing based on the real data sets- **Arun Kumar Sinha (Professor of Statistics since 1970, India).**

If no one believes in your capability, you believe in your capability and go for the best and do the best- **Adenomon, M. O.**

KEY: Keep Educating Yourself

Do not try but Do it

CHAPTER ONE

1.0 General Introduction

Statistics has become a strong tool in the modern day research. Virtually all disciplines use statistical procedures to drive home their points in a concise form for reporting and decision-making. Its application can be in biology, chemistry, physics, ecology, geology, engineering, medicine, neuroscience, computer sciences and in all courses in physical Sciences.

What is Statistics?

Statistics in the plural form is often used to refer to numerical or non-numerical facts or numbers (Oyejola & Adebayo, 2004). Steel and Torrie (1980) defines statistics as the science, pure and applied, of creating, developing and analyzing techniques such that the uncertainty of inductive inferences may be evaluated. Oyejola and Adebayo (2004) itemized the mainly concern of statistics. They are:

i. Designing or planning of experimental investigations and sample surveys.
ii. Summarizing the numbers collected from such experiments and surveys.
iii. Inferring facts about the population utilizing information from the sample.

1.1 Introduction to Statistical Computing

Statistics has become a strong tool in the modern day research. Virtually all disciplines use statistical procedures to drive home their points in a concise form for reporting and decision-making.

In order to meet these vast demands of various professions, a numbers of statistical packages had been developed. The three best known packages are SAS (Statistical Analysis System), SPSS (Statistical Packages for Social Sciences), and BMDP (Biomedical Computer Programs) and R. Others are STATGRAPHICS, GENSTAT, STATA, MINITAB, MS EXCEL, EVIEWS etc. these general-purpose packages offer a wide range of statistical technique, they contain programs for analysis of variance, multiple regression analysis, Chi-square analysis, time series analysis and most other technique in statistics.

Since it is easy to do a manual calculation on small amount of data, the use of personal computer (PC) and statistical software package for large amount of data for decision making either in research, government or business cannot be overemphasize. Since it is not easy to do calculation on large amount of data, it becomes imperative to discuss how to use SPSS, MINITAB, EVIEWS and R to solve statistical problem.

1.2 Statistical Applications

Statistical applications come in three forms, thus:

Stand Alone Program: This is common with beginners who write simple programs to carry-out a statistical functions such as Mean, Variance, Standard deviation and matrix operations.

Integrated Application: Statistical procedures that come with suite application packages like word processing and electronic spreadsheet. They are embedded program modules that are referenced as functions in tables' or spreadsheets' cells. The limitation of this type of statistical programs is that they are unable to handle complex statistical analysis.

Specialized Statistical Packages: The packages are completely dedicated to statistical analysis. They contain a number of procedures that are integrated together solely for the purpose of statistical analysis. They can perform simple and complex analyses including statistical chartings. The windows-based statistical applications have spreadsheets to enter statistical data, to format and arrange them in ways suitable for the type of analysis in mind. Popular among the statistical packages are SAS, STATISTICA, SPSS, MINITAB, EVIEWS and R.

From the foregoing overview, we are going to use SPSS, MINITAB, EVIEWS and R to perform simple and complex statistical analysis that are among the specialized statistical packages. As mentioned earlier that using manual calculation on large amount of data will waste time, affecting decision making which may in-turn affects the overall performance for which the analysis is met for, but using a specialized statistical packages, it saves time, help in decision making and in-turn contribute to the overall performance for which it is met for.

1.3 Benefits of Statistical Analysis

Statistical analysis is a creative process that results in important contributions to many different understanding, namely:

(i). Increased profits in business, the reason is that statistical software help you understand information that is it saves time, optimize resources and increase productivity

(ii). Improve treatment for disease as in biostatistical analysis.

(iii). Insights into social phenomena.

Statistical procedures can be applied to various issues of life, ranging from simple counting to more complex ones like testing of hypothesis.

1.4 Steps in Hypothesis Testing

The following are steps to carry out hypothesis testing:

Step 1: State the null and alternative hypothesis and specify the level of significance.

Step 2: State the test statistic.

Step 3: State the decision rule.

Step 4: Perform calculations-Compute value of test statistic and obtain critical value from the table.

Step 5: Determine the statistic decision and draw reasonable conclusion.

1.5 Introduction to R

R is a system for statistical analyses and graphics created by Ihaka & Gentleman (1996). R is both a software and a language considered as a dialect of the S language created by the AT&T Bell laboratories.

R is a language and environment for statistical computing and graphics, provides a wide variety of statistical methods (time series analysis, linear and nonlinear modelling, classical statistical tests, and so on) and graphical techniques, and is highly extensible.

R is freely distributed under the terms of the GNU General Public License and the statistical and mathematical packages are downloaded and installed through the internet (that is online). Its development and distribution are carried out by several statisticians known as the R Development Core Team (2005).

R is now widely used in academic research, education and industry. It is constantly growing with new versions of the core software released regularly and more than 2,600 packages available (Eubank & Kupresanin, 2011). The R language is, arguably, the de facto standard for statistical research purposes. There are now many books that detail its use (along with that of add-on packages) for the solution of data analysis problems.

In a broader sense, R is a very powerful functional language that merely happens to have built-in (and add-on) tools that perform some of the standard (and not so standard) statistical calculations with data

According to Paradis, (2005), the major advantages of R are (i) R has many functions and packages for statistical analyses and graphics, (ii) R language allows the user to program loops to successively analyze several data sets, (iii) It is possible in R to combine in a single program different statistical functions to perform more complex analysis.

R is especially power for data manipulation, calculations and plots. Its features include:

i. An integrated and very well-conceived documentation system.

ii. Efficient procedures for data treatment and storage.

iii. A vast and coherent collection of statistical procedures for data analysis

iv. A suit of operators for calculations on tables, especially matrices

v. Advanced graphical capabilities.

vi. A simple and Efficient programming language, including conditioning, loops, recursion, and input-output possibilities.

1.6 Installing R on a Windows PC

R setup can be downloaded from https://cran.r-project.org As at today 16-09-2020 R. 4.0.2 for Windows and Mac while another news says R.4.0.3 has been release.

You can also download R studio from https://rstudio.com

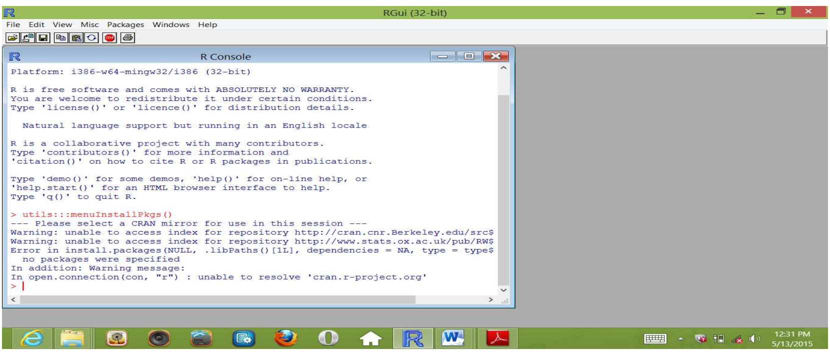
You can down Tinn-R editor from https://tinn-r.soft112.com

Packages are installed into R through online

To install R on the Windows computer, the following steps are to be followed:

1. Go to http://ftp.heanet.ie/mirrors/cran.r-project.org.
2. Under “Download and Install R”, click on the “Windows” link.
3. Under “Subdirectories”, click on the “base” link.
4. On the next page, there is a “Download R 3.2.0 for Windows” (or R.X.X.X, where X.X.X gives the version of R, eg. R 3.2.0). Click on this link.
5. You may be asked if you want to save or run a file “R-3.2.0-win32.exe”. Choose “Save” and save the file on the Desktop. Then double-click on the icon for the file to run it.
6. You choose language to install it in – e.g. English.
7. The R Setup Wizard will appear in a window. Click “Next” at the bottom of the R Setup wizard window.
8. The next page says “Information” at the top. Click “Next” again.
9. The next page says “Information” at the top. Click “Next” again.
10. The next page says “Select Destination Location” at the top. By default, it will suggest to install R in “C:\Program Files” on your computer.

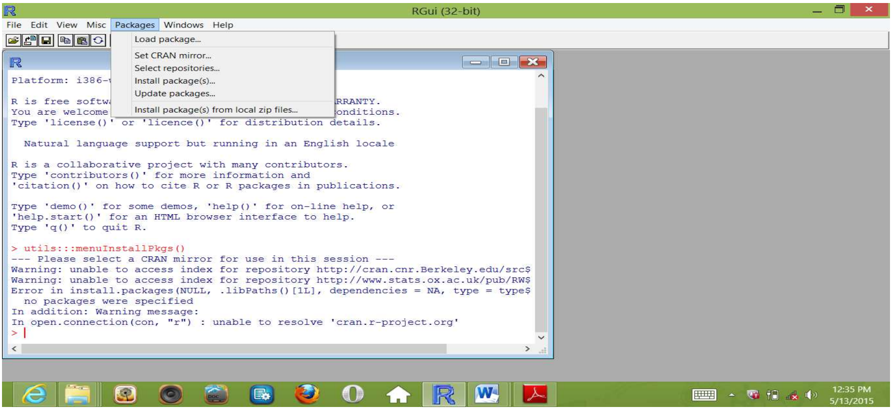
11. Click “Next” at the bottom of the R Setup wizard window.
12. The next page says “Select components” at the top. Click “Next” again.
13. The next page says “Start-up options” at the top. Click “Next” again.
14. The next page says “Select start menu folder” at the top. Click “Next” again.
15. The next page says “Select additional tasks” at the top. Click “Next” again.
16. R should now be installed. This will take about a minute. When R has finished, you will see “Completing the R for Windows Setup Wizard” appear. Click “Finish”.
17. To start R, follow step 18 or 19:
18. Check if there is an “R” icon on the desktop of the computer that you are using. If so, double-click on the “R” icon to start R. If the “R” icon did not appear, try step 19 instead.
19. Click on the “Start” button at the bottom left of the computer screen, and then choose “All programs”, and start R by selecting “R” (or R X.X.X, where X.X.X gives the version of R, eg. R 3.2.0) from the menu of programs.
20. The R console (a rectangle) should pop up a window similar to the following:



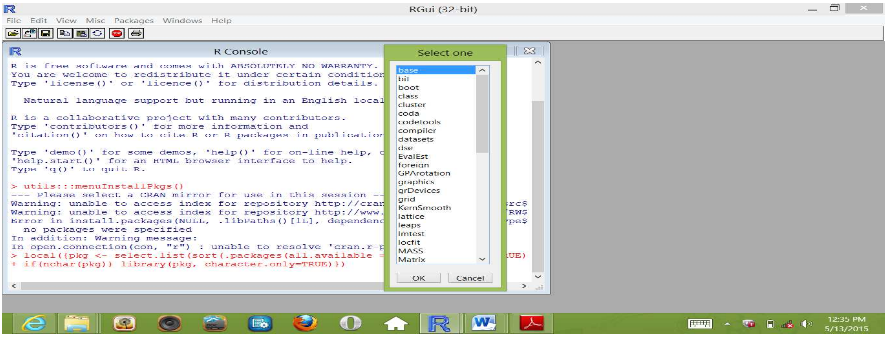
How to load or install packages

First to load package in R you do the following

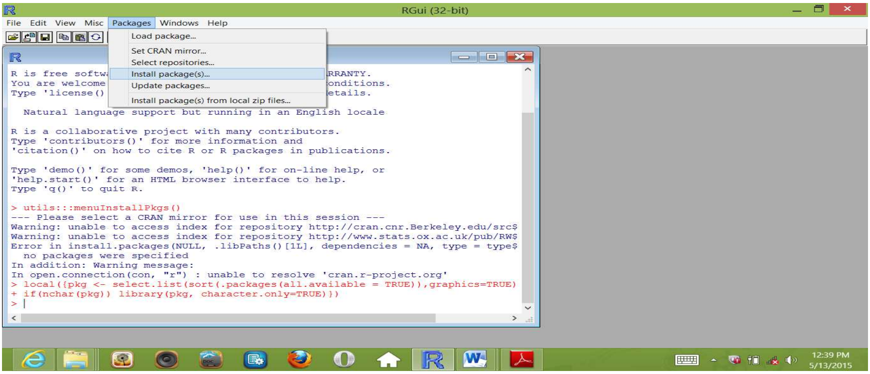
1. Click on packages and then click on load package



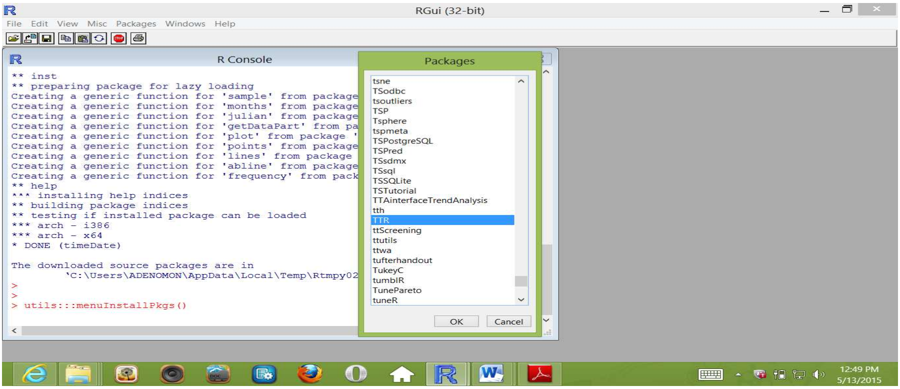
2. Select the appriopriate package of your choice by scrolling up or down



Then to install packages you must first be connected to the internet. Then click on packages and click on install packages as seen below



Then click the appropriate repository link and select the packages to install and click OK as seen below.



In this course and workshop, the following R packages will be employed

i. fBasics
ii. grDevices
iii. lattice
iv. lmtest
v. mctest
vi. car
vii. **multcomp**
viii. forecast
ix. zoo

x. agricolae
xi. boot

1.7 Definition of Terms

Computational Statistics: Computational statistics is the development and application of computational methods for problems in statistics.

Algorithms: Algorithm refers to as a step-by-step description of the calculations that must be undertaken to provide the desired solution.

Round-off error: This error arises from the fact that irrational numbers cannot be stored in their entirely in the finite amount of memory available in a computer.

Functions or subroutines: Is a procedural programming that decomposes a problem into component parts that can be solved using one or more subprograms.

Functional Languages: In functional languages one expresses the computational as evaluation of a function.

Programming Language: Is a formal computer language designed to communicate instructions to a machine, particularly the computer. Programming languages can be used to create programs to control the behavior of a machine or to express algorithms. The purpose of a computer language is to provide an avenue of ‘communication’ between a (typically human) user and a computer.

Goal Oriented Language: The programmer specifies definitions and rules and lets the system find a solution satisfying the definitions by using a built-in general loop.

namespace: Is a collection of definitions of variables, functions and other key components associated with a library or program that have been gathered together for various possible reasons.

Random Sample: Is a collection of random variables $X_1, . . . X_n$ is a random sample if they are all independent and have the same probability distribution.

Pseudo-random number generation (PRNG): A PRNG is an algorithm that, starting from an initial seed (or seeds) produces a sequence of numbers that behaves as if it were a random sample from a particular probability distribution when analyzed using statistical goodness-of-fit test.

A prime number: A prime number is a positive integer or natural number for which there are only two natural number divisors that produce another number as the quotient, that is, for which the division has a zero remainder.

Relative Prime: Two natural numbers are relatively prime if they have only 1 as a common divisor.

Measurement Error: Is the difference between an observed variable and the variable that belongs in a multiple regression equation. Also is the random or systematic error arising during data collection of variables.

Array: Is an ordered arrangement of numbers or other items of information, such as those in a list or table. In computing, an array has its own name, or identifier, and each number of the array is identified by a superscript used with the identifier. An array can be examined by a program and a particular item of information extracted by using this identifier and subscript.

CHAPTER TWO

2.0 Descriptive Statistics

In descriptive statistics, data are described using table, charts and graphs.

Illustrations

Oyejola & Adebayo (2004): Suppose that in the enumeration from farms, the yields of groundnut from 40 farms expressed in kg/ha are as given below:

699 662 599 545 613 627 681 595 522 701
595 627 599 746 590 708 533 763 631 577
636 636 686 640 663 672 636 695 623 698
686 681 636 636 586 722 681 636 717 654

Enter the data in R as follows

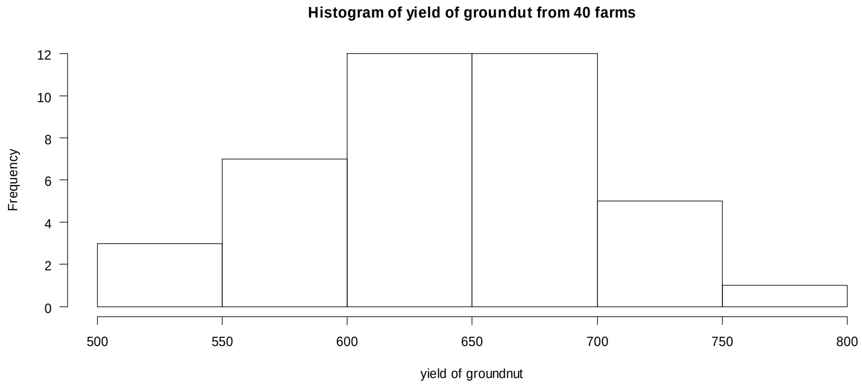
```
table1<-
c(699,662,599,545,613,627,681,595,522,701,595,627,599,746,590,7
08,533,763,631,577,636,636,686,640,663,672,636,695,623,698,686,
681,636,636,586,722,681,636,717,654)
```

To present the data in the form of table, that is, to show the values and their respective frequencies. We use the codes below

```
counts<-table(table1)
counts
table1
522 533 545 577 586 590 595 599 613 623 627 631 636 640 654 662 663 672 681 686
  1   1   1   1   1   1   2   2   1   1   2   1   6   1   1   1   1   1   3   2
695 698 699 701 708 717 722 746 763
  1   1   1   1   1   1   1   1   1
```

Now to present the data using histogram is as follows:

```
hist(table1,xlim=c(500,800),las=1,breaks=5,xlab="yield        of
groundnut",main="Histogram of yield of groundut from 40 farms")
```



To plot the data using Bar chart, we use the code below:

```
barplot(counts,xlab="yield  of  groundnut",main="Bar  Chart  of
yield of groundut from 40 farms")
```

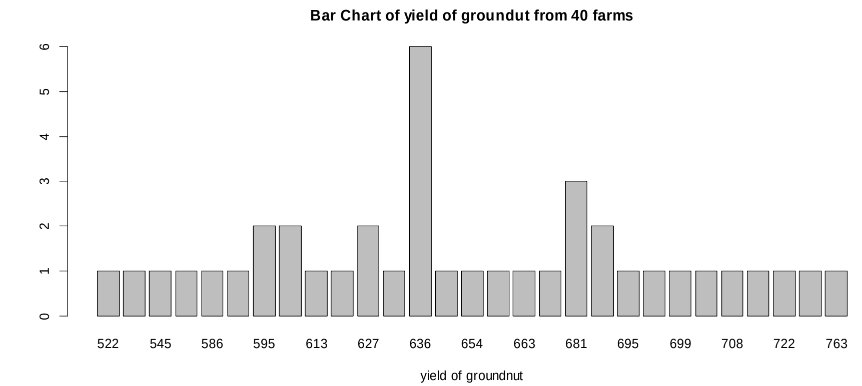
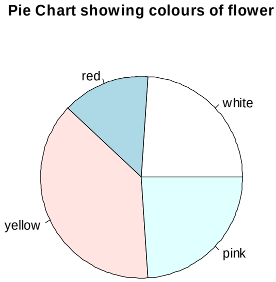


Table showing the colour of flower

Colour	Frequency
White	12
Red	7
Yellow	19
Pink	12
Total	50

We use the data in the table above to draw a pie chart. We use the code below:

```
library(grDevices)
pie.frequency<-c(12,7,19,12)
names(pie.frequency)<-c("white","red","yellow","pink")
pie(pie.frequency,main="Pie Chart showing colours of flower")
```



2.1 Elementary Statistics

We will compute some elementary statistics such as mean, median, variance and standard deviation. We consider the data below:

Enter the data in R as follows

```
table1<-
c(699,662,599,545,613,627,681,595,522,701,595,627,599,746,590,7
08,533,763,631,577,636,636,686,640,663,672,636,695,623,698,686,
681,636,636,586,722,681,636,717,654)
summary(table1)
  Min.   1st Qu.  Median    Mean   3rd Qu.    Max.
  522.0   609.5   636.0     645.8   686.0     763.0
mean(table1)
[1] 645.825
median(table1)
[1] 636
var(table1)
[1] 3079.789
sd(table1)
[1] 55.49585
```

2.2 Data Generation from Different Distribution

Normal Distribution

We use `rnorm` to generate data from normal distribution, that is `rnorm(n, mean, standard deviation)`

To generate 30 observations with mean 0 and standard deviation 1

```
rnorm(30)
 [1]  0.837987998  0.775147442 -1.985816491 -1.319997270  1.023823640
 [6]  0.391522313  0.801640981  0.196571622  1.652881154  1.293937555
[11]  0.230480858  0.069785426  0.668223341 -0.774438872 -0.684200723
[16]  0.008252105  0.747087007  0.076187935 -0.274694453  1.045344910
[21]  0.548741341 -0.381944626 -0.541872942  0.706822922  0.521940399
[26] -0.378021229  1.597398190  0.206939083 -0.108691362 -0.411680039
```

To generate 50 observations with mean 20 and standard deviation 2.5

```
rnorm(50,20,2.5)
 [1] 22.32062 21.55554 21.25586 19.56949 19.81879 19.15063 17.57346 20.33444
 [9] 20.47265 18.73295 22.85250 16.51198 23.83290 21.30112 22.06676 17.57912
[17] 21.64928 22.43863 18.60973 28.72518 20.43730 17.76599 14.48220 18.40880
[25] 20.39169 18.77115 21.86442 19.15307 16.53339 22.68265 20.33601 23.82345
[33] 25.41594 19.65359 21.66705 23.37735 18.11877 21.82349 18.91305 18.44989
[41] 22.99678 18.15751 16.82928 17.01331 22.63988 18.47914 21.51215 18.75217
[49] 22.26183 24.75929
```

Uniform distribution

The uniform assume minimum value as zero (1) and maximum value as 1 (one). The code for generating data from uniform distribution as given below:

`runif(n, min val., max val.)`

now to generate data from uniform distribution of size 30 and minimum value of zero (0) and maximum value of one (1). Is as follows

```
runif(30)
[1] 0.03528576 0.20409264 0.53286450 0.19661985 0.62269957 0.99284717
 [7] 0.99829896 0.19312412 0.75861656 0.94805228 0.52859996 0.19327873
[13] 0.27008494 0.58954891 0.80923770 0.41090877 0.29374921 0.95195846
[19] 0.59667636 0.84690380 0.02094311 0.88250099 0.31756397 0.80165433
```

```
[25] 0.17930265 0.20785230 0.80478940 0.71421147 0.97083390 0.48173434
```

Now to generate data from uniform distribution of size 30 and minimum value of 0.1 and maximum value of 0.9. Is as follows

```
runif(30,0.1,0.9)
 [1] 0.2931572 0.1336960 0.3569332 0.1487373 0.2096555 0.8060056 0.6811592
 [8] 0.4554822 0.8170134 0.7535014 0.4019798 0.7723267 0.4615288 0.6859749
[15] 0.8590194 0.5901704 0.6380250 0.5464811 0.3637731 0.3997953 0.2377473
[22] 0.1729549 0.4315148 0.3787067 0.5386235 0.2440551 0.4448157 0.8250459
[29] 0.6859744 0.1667474
```

Geometric distribution

This code `rgeom(n, prob)`is used to generate data from a geometric distribution where 0<prob<=1.

To generate data from geometric distribution of size 20 with prob=0.8. Is as follows

```
rgeom(20,0.8)
[1] 0 1 0 1 0 0 0 0 1 0 2 0 0 0 0 0 0 0 0 0
```

To generate data from geometric distribution of size 30 with prob=0.5. Is as follows

```
rgeom(30,0.5)
[1] 0 1 0 0 1 2 0 0 2 2 1 0 1 0 2 2 0 3 2 1 0 0 2 0 0 0 0 0 0 0
```

Poisson Distribution

This code `rpois(n, lambda)` is used to generate data from a Poisson distribution where lamda is equal to mean and variance.

To generate data from Poisson distribution of size 30 with lamda=2.5. Is as follows

```
rpois(30, 2.5)
[1] 4 2 3 3 3 1 4 1 4 0 1 3 4 4 1 1 5 2 4 3 2 3 0 2 2 2 1 5 3 0
```

To generate data from Poisson distribution of size 40 with lamda=4.5. Is as follows

```
rpois(40, 4.5)
[1] 5 6 4 7 8 2 2 5 5 4 5 0 5 4 2 4 6 6 7 5 3 3 2 1 4 2 5 4 5 8 4 4 2 3 5 3 4 3
[39] 4 7
```

Binomial Distribution

This code `rbinom(n, size, prob)` is used to generate data from a Binomial distribution where n is number of observations, size is the number of trial and 0<prob<=1.

To generate data from Binomial distribution of size 2 and n=5 with prob=0.5. Is as follows

```
rbinom(5, 2, 0.5)
[1] 2 0 1 2 0

rbinom(10, 2, 0.5)
[1] 0 2 1 1 0 2 1 0 0 0

rbinom(10, 3, 0.5)
[1] 2 2 2 1 1 3 2 1 1 2
```

2.3 Normality Testing

In statistics, normality tests are used to determine if a data set is well-modeled by a normal distribution and to compute how likely it is for a random variable underlying the data set to be normally distributed. Some time graphical methods such as the use of histogram. Tests of univariate normality include the following:

D’Agostino’s K-squared test; Jarque-Bera Test; Anderson-Darling test; Cramer-von Mises Criterion; Lilliefors test; Kolmogorov-Smirnov test; Shapiro-Wilk test; Pearson’s Chi-square test

The R functions for testing normality using fBasics are:

`ksnormTest`	Kolmogorov-Smirnov normality test,
`shapiroTest`	Shapiro-Wilk's test for normality,
`jarqueberaTest`	Jarque--Bera test for normality,

`dagoTest` D'Agostino normality test.

Illustrations

```
table1<-
c(699,662,599,545,613,627,681,595,522,701,595,627,599,746,590,7
08,533,763,631,577,636,636,686,640,663,672,636,695,623,698,686,
681,636,636,586,722,681,636,717,654)
```

To perform normality we the fBasics package with the following codes

```
library(fBasics)
ksnormTest(table1)
shapiroTest(table1)
jarqueberaTest(table1)
dagoTest(table1)

ksnormTest(table1)

Title:
 One-sample Kolmogorov-Smirnov test

Test Results:
  STATISTIC:
    D: 1
  P VALUE:
    Alternative Two-Sided: < 2.2e-16
    Alternative      Less: < 2.2e-16
    Alternative   Greater: 1

Description:
 Sat Jan 13 13:32:10 2018 by user: ADENOMON

shapiroTest(table1)

Title:
 Shapiro - Wilk Normality Test

Test Results:
  STATISTIC:
    W: 0.9836
  P VALUE:
```

```
    0.8176

Description:
 Sat Jan 13 13:32:10 2018 by user: ADENOMON

jarqueberaTest(table1)

Title:
 Jarque - Bera Normalality Test

Test Results:
  STATISTIC:
    X-squared: 0.3297
  P VALUE:
    Asymptotic p Value: 0.848

Description:
 Sat Jan 13 13:32:10 2018 by user: ADENOMON

dagoTest(table1)

Title:
 D'Agostino Normality Test

Test Results:
  STATISTIC:
    Chi2 | Omnibus: 0.2352
    Z3  | Skewness: -0.483
    Z4  | Kurtosis: -0.0438
  P VALUE:
    Omnibus  Test: 0.889
    Skewness Test: 0.6291
    Kurtosis Test: 0.965

Description:
 Sat Jan 13 13:32:12 2018 by user: ADENOMON
```

CHAPTER THREE

3.0 Correlation Analysis

Simple correlation measure the degree of relationship between two or more variables.

Illustration

The following are the dosage of a drug and reduction in blood sugar level from 7 patients.

Dosage(X): 0.38 0.51 0.19 0.53 0.39 0.38 0.66

Reduction in Blood Sugar (Y): 50 72 36 64 52 56 80

The R codes to run the following simple correlations are below

```
x<-c(0.38,0.51,0.19,0.53,0.39,0.38,0.66)
y<-c(50,72,36,64,52,56,80)
library(fBasics)
correlationTest(x, y, "pearson")
correlationTest(x, y, "kendall")
spearmanTest(x, y)

correlationTest(x, y, "pearson")

Title:
 Pearson's Correlation Test

Test Results:
  PARAMETER:
    Degrees of Freedom: 5
  SAMPLE ESTIMATES:
    Correlation: 0.9701
  STATISTIC:
    t: 8.9454
  P VALUE:
    Alternative Two-Sided: 0.000291
    Alternative      Less: 0.9999
    Alternative   Greater: 0.0001455
  CONFIDENCE INTERVAL:
    Two-Sided: 0.8058, 0.9957
         Less: -1, 0.9942
      Greater: 0.8544, 1

Description:
```

```
 Fri Jan 19 08:56:31 2018

> correlationTest(x, y, "kendall")

Title:
 Kendall's tau Correlation Test

Test Results:
  SAMPLE ESTIMATES:
    tau: 0.7807
  STATISTIC:
    z: 2.4306
    T | Exact: 2.4306
  P VALUE:
    Alternative         Two-Sided: 0.01507
    Alternative Two-Sided | Exact: 0.01507
    Alternative              Less: 0.9925
    Alternative      Less | Exact: 0.9925
    Alternative           Greater: 0.007537
    Alternative   Greater | Exact: 0.007537

Description:
 Fri Jan 19 08:56:31 2018

> spearmanTest(x, y)

Title:
 Spearman's rho Correlation Test

Test Results:
  SAMPLE ESTIMATES:
    rho: 0.9009
  STATISTIC:
    S: 5.5475
  P VALUE:
    Alternative Two-Sided: 0.005621
    Alternative      Less: 0.9972
    Alternative   Greater: 0.00281

Description:
 Fri Jan 19 08:56:31 2018
```

Assuming we have x1, x2 and y variables. To find the possible simple correlation coefficients: is as follows

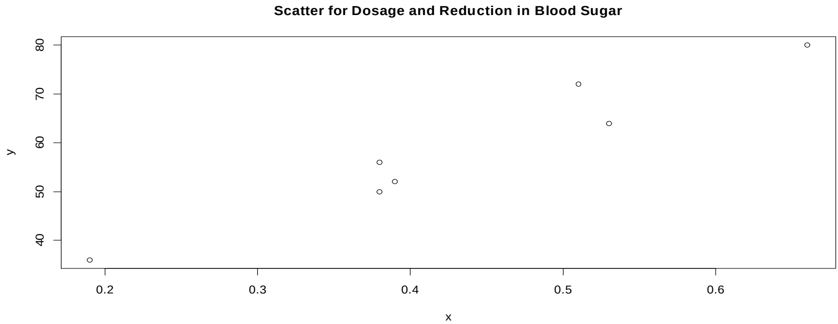
```
x1<-c(0.38,0.51,0.19,0.53,0.39,0.38,0.66)
x2<-c(2,5,8,7,4,6,7)
y<-c(50,72,36,64,52,56,80)
XY<-cbind(x1,x2,y)
cor(XY)
           x1         x2          y
x1 1.00000000 0.03394189 0.97014958
x2 0.03394189 1.00000000 0.08589138
y  0.97014958 0.08589138 1.00000000
```

3.1 Regression Analysis and Diagnostic Testing

We begin with scatter plot

The R codes to plot a scatter diagram is given below:

```
x<-c(0.38,0.51,0.19,0.53,0.39,0.38,0.66)
y<-c(50,72,36,64,52,56,80)
plot(x,y, main="Scatter for Dosage and Reduction in Blood Sugar")
```



Simple Regression Analysis

```
x<-c(0.38,0.51,0.19,0.53,0.39,0.38,0.66)
y<-c(50,72,36,64,52,56,80)
SimpleReg<-lm(y~x)
summary(SimpleReg)

Call:
lm(formula = y ~ x)

Residuals:
```

```
      1       2       3       4       5       6       7 
-3.3685  6.1718  0.8419 -3.7451 -2.3269  2.6315 -0.2048 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   16.948      4.882   3.471 0.017828 *  
x             95.844     10.714   8.945 0.000291 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.913 on 5 degrees of freedom
Multiple R-squared:  0.9412,    Adjusted R-squared:  0.9294 
F-statistic: 80.02 on 1 and 5 DF,  p-value: 0.000291
```

Then the ANOVA table can be obtained as follows:

```
ANOVA<-aov(SimpleReg)
> summary(ANOVA)
            Df Sum Sq Mean Sq F value   Pr(>F)    
x            1 1225.2  1225.2   80.02 0.000291 ***
Residuals    5   76.6    15.3                     
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Multiple Regression and Diagnostic Testing

In a small scale study of the relationship between degree of brand liking (Y) and Moisture content (X_1) and Sweetness (X_2) of the product, the following results were obtained

X_1: 4 4 6 6 8 8

X_2: 2 6 2 6 2 6

Y: 64 81 72 91 83 96

To fit multiple regression with R, is as follows

```
x1<-c(4,4,6,6,8,8)
x2<-c(2,6,2,6,2,6)
y<-c(64,81,72,91,83,96)
MultReg<-lm(y~x1+x2)
summary(MultReg)

Call:
lm(formula = y ~ x1 + x2)
```

```
Residuals:
      1       2       3       4       5       6 
-0.5000  0.1667 -1.0000  1.6667  1.5000 -1.8333 
Coefficients:
            Estimate Std. Error t value Pr(>|t|)   
(Intercept)  39.3333     3.1520   12.48  0.00111 **
x1            4.2500     0.4488    9.47  0.00250 **
x2            4.0833     0.3664   11.14  0.00155 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.795 on 3 degrees of freedom
Multiple R-squared:  0.9862,    Adjusted R-squared:  0.9769 
F-statistic: 106.9 on 2 and 3 DF,  p-value: 0.001627
```

Then the ANOVA table

```
ANOVA<-aov(MultReg)
summary(ANOVA)
            Df Sum Sq Mean Sq F value  Pr(>F)   
x1           1  289.0   289.0   89.69 0.00250 **
x2           1  400.2   400.2  124.19 0.00155 **
Residuals    3    9.7     3.2                   
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Then to test for collinearity or multicollinearity, Serial Correlation and Homoscedasticity.

Multicollinearity Testing

```
library(mctest)
x1<-c(4,4,6,6,8,8)
x2<-c(2,6,2,6,2,6)
y<-c(64,81,72,91,83,96)
X<-cbind(x1,x2)
imcdiag(X,y)
Call:
imcdiag(x = X, y = y)
All Individual Multicollinearity Diagnostics Result

   VIF TOL Wi  Fi Leamer CVIF Klein
x1   1   1  0 Inf      1    1     0
```

```
x2   1   1  0 Inf      1    1     0

1 --> COLLINEARITY is detected
0 --> COLLINEARITY in not detected by the test

* all coefficients have significant t-ratios

R-square of y on all x: 0.9862

* use method argument to check which regressors may be the reason of collinearity
===================================
omcdiag(X,y)
Call:
omcdiag(x = X, y = y)
Overall Multicollinearity Diagnostics
                       MC Results detection
Determinant |X'X|:         1.0000         0
Farrar Chi-Square:         0.0000         0
Red Indicator:                NaN        NA
Sum of Lambda Inverse:     2.0000         0
Theil's Method:           -0.9862         0
Condition Number:          9.3540         0
1 --> COLLINEARITY is detected
0 --> COLLINEARITY in not detected by the test
===================================
Eigvenvalues with INTERCEPT
                   Intercept     x1     x2
Eigenvalues:          2.8156 0.1522 0.0322
Condition Indeces:    1.0000 4.3005 9.3540
```

Serial Correlation Testing

```
library(lmtest)
dwtest(MultReg)
        Durbin-Watson test
data:  MultReg
DW = 2.0747, p-value = 0.4727
alternative hypothesis: true autocorrelation is greater than 0

bgtest(MultReg)
        Breusch-Godfrey test for serial correlation of order up to 1
data:  MultReg
```

```
LM test = 1.5974, df = 1, p-value = 0.2063
```

Test for Heteroscedasticity

```
bptest(MultReg)
        studentized Breusch-Pagan test
data:  MultReg
BP = 5.2043, df = 2, p-value = 0.07412
```

3.2 Robust Regression

Robust regression is used when outliers are suspected

```
crop15<-
read.csv("C:/Users/ADENOMON/Desktop/crop2015.csv",header=T)
crop15
   Production Land.Area  Yield
1    10477.96   5741.88  1.825
2     6339.87   4964.83  1.277
3     7751.61   3150.02  2.461
4       61.70    162.87  0.379
5    47137.00   3179.73 14.824
6     3532.06   2566.33  1.376
7     1678.37   1446.10  1.161
8    57575.73   9170.80  6.278
9     2306.16   3635.74  0.634
10     179.11    428.67  0.418
11    3276.70    826.84  3.963
12     432.94    470.13  0.921
13     764.95    858.05  0.891
14    2067.89   1859.86  1.112
15     997.88    434.45  2.297
16    2208.32    557.50  3.961

library(MASS)
robustreg<-
rlm(Production~Land.Area,data=crop15,psi=psi.bisquare)
summary(robustreg)
Call: rlm(formula = crop15$Production ~ crop15$Land.Area, data
= crop15,
    psi = psi.bisquare)
Residuals:
    Min      1Q  Median      3Q      Max
```

```
-3250.2  -585.3  -307.3  1799.2 43678.6
Coefficients:
                 Value    Std. Error t value
(Intercept)       77.6888 638.6355     0.1216
crop15$Land.Area   1.5069   0.1856     8.1195
Residual standard error: 1507 on 14 degrees of freedom
```

coeftest(robustreg)

```
z test of coefficients:
                  Estimate Std. Error z value  Pr(>|z|)
(Intercept)       77.68876  638.63545  0.1216    0.9032
crop15$Land.Area   1.50689    0.18559  8.1195 4.681e-16 ***
---
Signif.codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Calculation of forecast statistic for robust regression using forecast package

```
accuracy(crop15$Production,robustreg$fitted.values)
          ME        RMSE      MAE        MPE      MAPE
Test set  -5380.791  15259.5   6385.538 -75.20537 123.2302
```

CHAPTER FOUR

4.0 Introduction to ANOVA

ANOVA is used to compare three or more samples

Note here

- ANOVA is quite robust to small deviations from normality
- Normal test are sometimes quite conservative meaning normality may be rejected due to a limited deviation from normality.

Assumptions

- Variable type
- Independence
- Normality
- Equality of Variance

Notes

- If Variances are equal, use ANOVA
- If Variances not equal, use Welch Test
- If normality is not assumed, use Kruskal Wallis Test

4.1 One-Way ANOVA

One-way analysis of variance examines the difference of means among certain number of treatment from an experiment.

Illustration

Nineteen pigs are assigned at random among four experimental groups. Each group is fed a different diet. The data are pig body weights in kilograms, after being raised on these diet. We wish to ask whether the pig weights are the same for all four diets.

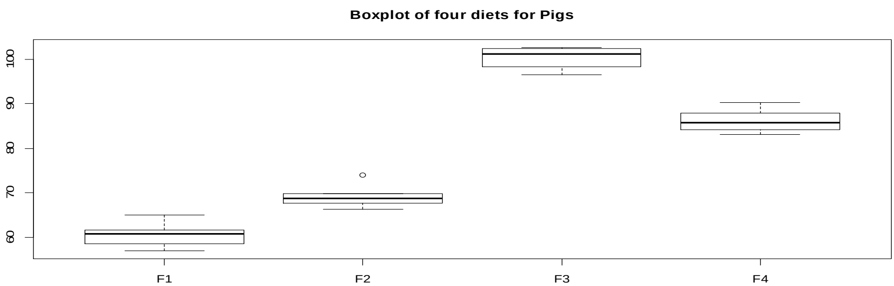
Feed 1	Feed 2	Feed 3	Feed 4
60.8	68.7	102.6	87.9
57.0	67.7	102.1	84.2
65.0	74.0	100.2	83.1
58.6	66. 3	96.5	85.7
61.7	69.8		90. 3

In the example above, the feeds are the treatment, which is a typical example one-way analysis of variance.

```
Response<-c(60.8, 57, 65, 58.6, 61.7, 68.7, 67.7, 74, 66.3,
69.8, 102.6, 102.1, 100.2, 96.5, 87.9, 84.2, 83.1, 85.7, 90.3)
Trt<-c(rep("F1",5),rep("F2",5),rep("F3",4),rep("F4",5))
```

First we plot the boxplot as follows

```
boxplot(Response~Trt,main="Boxplot of four diets for Pigs")
```




```
Data1<-data.frame(Response,Trt)
Oneway<-aov(Response~Trt,data=Data1)
summary(Oneway)
            Df Sum Sq Mean Sq F value   Pr(>F)
Trt          3   4226  1408.8   164.6 1.06e-11 ***
Residuals   15    128     8.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Levene's Test for Homogeneity of Variance

```
library(car)
leveneTest(Response~Trt)
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  3  0.0238 0.9948
      15
Warning message:
In  leveneTest.default(y  =  y,  group  =  group,  ...)  :  group
coerced to factor.
```

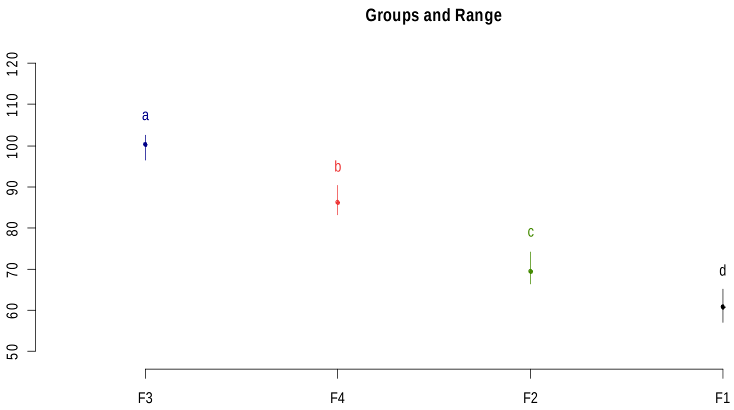
Since p-value=0.9948>0.05, we do not reject H_o we conclude that the variances are equal (Homogenous)

Post ANOVA Test: One Way

```
library(multcomp)
TukeyHSD(Oneway)
  Tukey multiple comparisons of means
    95% family-wise confidence level
Fit: aov(formula = Response ~ Trt)
$Trt
        diff         lwr       upr     p adj
F2-F1   8.68   3.347895 14.012105 0.0014725
F3-F1  39.73  34.074449 45.385551 0.0000000
F4-F1  25.62  20.287895 30.952105 0.0000000
F3-F2  31.05  25.394449 36.705551 0.0000000
F4-F2  16.94  11.607895 22.272105 0.0000009
F4-F3 -14.11 -19.765551 -8.454449 0.0000168
```

LSD Post ANOVA Test

```
LSD.test(Oneway,"Trt",p.adj="bonferroni")
plot(LSD.test(Oneway,"Trt",p.adj="bonferroni")
```



Duncan Post Anova Test

```
duncan.test(Oneway,"Trt")
print(duncan.test(Oneway,"Trt"))
$statistics
   MSerror Df     Mean       CV
  8.556667 15 78.01053 3.749722
$parameters
    test name.t ntr alpha
  Duncan    Trt   4  0.05
$duncan
NULL
$means
   Response      std r  Min   Max    Q25    Q50     Q75
F1    60.62 3.064637 5 57.0  65.0 58.600  60.80  61.700
F2    69.30 2.926602 5 66.3  74.0 67.700  68.70  69.800
F3   100.35 2.767068 4 96.5 102.6 99.275 101.15 102.225
F4    86.24 2.896204 5 83.1  90.3 84.200  85.70  87.900
$comparison
NULL
$groups
   Response groups
F3   100.35      a
F4    86.24      b
F2    69.30      c
F1    60.62      d
```

Normality test of Residuals of One way ANOVA

```
library(fBasics)
jarqueberaTest(Oneway$residuals)
Title:
 Jarque - Bera Normalality Test
Test Results:
  STATISTIC:
    X-squared: 0.9568
  P VALUE:
    Asymptotic p Value: 0.6198
Description:
 Wed Oct 14 23:37:53 2020 by user: ADENOMON
```

4.2 Two-Way ANOVA

For Two-way ANOVA we have the treatments and the blocks. Consider the example below

We wish to perform a two way ANOVA considering the treatment (the feeds) and the blocking effects.

Blocks	Feed 1	Feed 2	Feed 3	Feed 4
1	60.8	68.7	102.6	87.9
2	57.0	67.7	102.1	84.2
3	65.0	74.0	100.2	83.1
4	58.6	66.3	96.5	85.7
5	61.7	69.8	100	90.3

Here we have 5 blocks and 4 treatments. The R code is as follows:

```
Response<-c(60.8, 57, 65, 58.6, 61.7, 68.7, 67.7, 74, 66.3,
69.8, 102.6, 102.1, 100.2, 96.5, 100, 87.9, 84.2, 83.1, 85.7,
90.3)
Trt<-c(rep("F1",5),rep("F2",5),rep("F3",5),rep("F4",5))
Blk<-
c("1","2","3","4","5","1","2","3","4","5","1","2","3","4","5","
1","2","3","4","5")
Data2<-data.frame(Response,Trt,Blk)
Twoway<-aov(Response~Trt+Blk, data=Data2)
summary(Twoway)
```

```
            Df Sum Sq Mean Sq F value   Pr(>F)
Trt          3   4686  1561.9 233.391 6.66e-11 ***
Blk          4     48    12.0   1.799    0.194
Residuals   12     80     6.7
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Post ANOVA test: Two way

```
TukeyHSD(Twoway)
  Tukey multiple comparisons of means
    95% family-wise confidence level
Fit: aov(formula = Response ~ Trt + Blk)
$Trt
        diff        lwr       upr     p adj
F2-F1   8.68   3.822572 13.537428 0.0009262
F3-F1  39.66  34.802572 44.517428 0.0000000
F4-F1  25.62  20.762572 30.477428 0.0000000
F3-F2  30.98  26.122572 35.837428 0.0000000
F4-F2  16.94  12.082572 21.797428 0.0000013
F4-F3 -14.04 -18.897428 -9.182572 0.0000095
$Blk
      diff       lwr      upr     p adj
2-1 -2.250 -8.080511 3.580511 0.7352446
3-1  0.575 -5.255511 6.405511 0.9975718
4-1 -3.225 -9.055511 2.605511 0.4355998
5-1  0.450 -5.380511 6.280511 0.9990675
3-2  2.825 -3.005511 8.655511 0.5557675
4-2 -0.975 -6.805511 4.855511 0.9820403
5-2  2.700 -3.130511 8.530511 0.5950581
4-3 -3.800 -9.630511 2.030511 0.2898145
5-3 -0.125 -5.955511 5.705511 0.9999943
5-4  3.675 -2.155511 9.505511 0.3182459
```

Duncan Test: Two way

```
print(duncan.test(Twoway,"Trt"))
$statistics
   MSerror Df  Mean       CV
  6.692083 12 79.11 3.270012
$parameters
```

```
    test name.t ntr alpha
  Duncan    Trt   4  0.05
$duncan
     Table CriticalRange
2 3.081307      3.564762
3 3.225244      3.731283
4 3.312453      3.832176
$means
   Response      std r  Min   Max   Q25   Q50   Q75
F1    60.62 3.064637 5 57.0  65.0  58.6  60.8  61.7
F2    69.30 2.926602 5 66.3  74.0  67.7  68.7  69.8
F3   100.28 2.401458 5 96.5 102.6 100.0 100.2 102.1
F4    86.24 2.896204 5 83.1  90.3  84.2  85.7  87.9
$comparison
NULL
$groups
   Response groups
F3   100.28      a
F4    86.24      b
F2    69.30      c
F1    60.62      d


print(duncan.test(Twoway,"Blk"))
$statistics
   MSerror Df  Mean       CV
  6.692083 12 79.11 3.270012
$parameters
    test name.t ntr alpha
  Duncan    Blk   5  0.05
$duncan
     Table CriticalRange
2 3.081307      3.985526
3 3.225244      4.171701
4 3.312453      4.284503
5 3.370172      4.359159
$means
  Response      std r  Min   Max    Q25   Q50    Q75
1   80.000 18.88121 4 60.8 102.6 66.725 78.30 91.575
2   77.750 19.71539 4 57.0 102.1 65.025 75.95 88.675
3   80.575 15.02584 4 65.0 100.2 71.750 78.55 87.375
4   76.775 17.40486 4 58.6  96.5 64.375 76.00 88.400
5   80.450 17.74082 4 61.7 100.0 67.775 80.05 92.725
$comparison
```

```
NULL
$groups
  Response groups
3   80.575      a
5   80.450      a
1   80.000      a
2   77.750      a
4   76.775      a
```

4.3 Two-Way ANOVA with Interactions (Factorial Design)

A manufacturer wishes to determine the effectiveness of four types of machines (A,B, C and D) in the production of bolts. To accomplish this, the number of defectives bolts produced by each machine in the days of a given week is obtained for each of two shifts. The results are given in the table below

Factor A Machine	Factor B Shift	Replicates				
		Mon	Tues	Wed	Thurs	Fri
A	1 2	6 5	4 7	5 4	5 6	4 8
B	1 2	10 7	8 9	7 12	7 8	9 8
C	1 2	7 9	5 7	6 5	5 4	9 6
D	1 2	8 5	4 7	6 9	5 7	5 10

To Perform an analysis of variance to determine whether there is a difference between the machines and between the shifts using R. The R code is as follows

```
response<-
c(6,4,5,5,4,5,7,4,6,8,10,8,7,7,9,7,9,12,8,8,7,5,6,5,9,9,7,5,4,6
,8,4,6,5,5,5,7,9,7,10)
factA<-c(rep("A",10),rep("B",10),rep("C",10),rep("D",10))
factB<-
c(rep("1",5),rep("2",5),rep("1",5),rep("2",5),rep("1",5),rep("2
",5),rep("1",5),rep("2",5))
```

```
factExp<-data.frame(response,factA,factB)
Fact<-aov(response~factA+factB+factA*factB, data=factExp)
summary(Fact)
            Df Sum Sq Mean Sq F value  Pr(>F)
factA        3   51.0  17.000   6.415 0.00158 **
factB        1    8.1   8.100   3.057 0.09000 .
factA:factB  3    6.5   2.167   0.818 0.49371
Residuals   32   84.8   2.650
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

4.4 Latin Square Design (also known as Three-Way ANOVA)

Illustration: A farmer wishes to test the effects of four different fertilizers (A, B, C and D) on the yield of wheat. In order to eliminate sources of error due to variability in soil fertility, he uses the fertilizers in a Latin-square arrangement. Perform an analysis of variance between the fertilizers.

A18	C21	D25	B11
D22	B12	A15	C19
B15	A20	C23	D24
C22	D21	B10	A17

```
The R code for Latin square design analysis is as follows:
Response<-c(18, 21,25,11,22,12,15,19,15,20,23,24,22,21,10,17)
Row<-c(rep("1",4),rep("2",4),rep("3",4),rep("4",4))
Col<-
c("1","2","3","4","1","2","3","4","1","2","3","4","1","2","3","
4")
Latin<-
c("A","C","D","B","D","B","A","C","B","A","C","D","C","D","B","
A")
Data3<-data.frame(Response,Row,Col,Latin)
LDesign<-aov(Response~Row+Col+Latin,data=Data3)
summary(LDesign)
            Df Sum Sq Mean Sq F value   Pr(>F)
Row          3  29.19    9.73   4.916 0.046790 *
Col          3   4.69    1.56   0.789 0.542383
Latin        3 284.19   94.73  47.863 0.000139 ***
```

```
Residuals    6  11.87    1.98
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

CHAPTER FIVE

5.0 Non Parametric Statistics

Non parametric statistics are refers to as distribution free statistics which can have small sample size.

- We consider Mann Whitney U test similar to t test in parametric statistics
- Kruskall-wallis similar to one-way ANOVA in parametric statistics
- Friedman Test similar to Two-way ANOVA in parametric statistics

5.1 Mann Whitney U test

```
Fisher<-
ts(read.csv("C:/Users/ADENOMON/Desktop/Fishing.
csv",header=T),start=c(2008,1))
Fisher
Time Series:
Start = 2008
End = 2015
Frequency = 1
     Full.Time Part.Time Occasional   Total
2008    957918    689792      30595 1678305
2009    980715    706208      31694 1718617
2010    923150    562308      34169 1519627
2011    901889    565608      65615 1533112
2012   1014268    596696      68019 1678983
2013    963413    608805      71008 1643226
2014    860379    938343     122929 1921651
2015    740378    635044      81025 1456447
```

We use full and part time columns for illustration

```
Manntest<-
read.csv("C:/Users/ADENOMON/Desktop/manntest.cs
v",header=T)
Manntest
   fishing group
1   957918  full
2   980715  full
3   923150  full
4   901889  full
5  1014268  full
6   963413  full
7   860379  full
8   740378  full
9   689792  part
10  706208  part
11  562308  part
12  565608  part
13  596696  part
14  608805  part
15  938343  part
16  635044  part

wilcox.test(Manntest$fishing~Manntest$group)
        Wilcoxon rank sum test
data:  Manntest$fishing by Manntest$group
W = 60, p-value = 0.001865
alternative hypothesis: true location shift is
not equal to 0
```

5.2 Kruskall Wallis: Non parametric One way ANOVA

```
Kruskaltest<-
read.csv("C:/Users/ADENOMON/Desktop/Kruskaltest
.csv",header=T)
Kruskaltest
   fishing    group
1   957918     full
2   980715     full
3   923150     full
4   901889     full
5  1014268     full
6   963413     full
7   860379     full
8   740378     full
9   689792     part
10  706208     part
11  562308     part
12  565608     part
13  596696     part
14  608805     part
15  938343     part
16  635044     part
17   30595 occasion
18   31694 occasion
19   34169 occasion
20   65615 occasion
21   68019 occasion
22   71008 occasion
23  122929 occasion
24   81025 occasion
```

```
kruskal.test(Kruskaltest$fishing~Kruskaltest$group)
        Kruskal-Wallis rank sum test
data:  Kruskaltest$fishing by Kruskaltest$group
Kruskal-Wallis chi-squared = 19.28, df = 2, p-
value = 6.507e-05
```

Post hoc test of Kruskall Wallis

```
pairwise.wilcox.test(Kruskaltest$fishing,Kruska
ltest$group)
Pairwise comparisons using Wilcoxon rank sum
test
data:  Kruskaltest$fishing and
Kruskaltest$group
         full    occasion
occasion 0.00047 -
part     0.00186 0.00047
P value adjustment method: holm
```

5.3 Friedman test: Non parametric Two way ANOVA

```
Friedtest<-
read.csv("C:/Users/ADENOMON/Desktop/Friedtest.c
sv",header=T)
Friedtest
   fishing    group year
1   957918     full 2008
2   980715     full 2009
3   923150     full 2010
4   901889     full 2011
5  1014268     full 2012
6   963413     full 2013
```

```
7   860379     full 2014
8   740378     full 2015
9   689792     part 2008
10  706208     part 2009
11  562308     part 2010
12  565608     part 2011
13  596696     part 2012
14  608805     part 2013
15  938343     part 2014
16  635044     part 2015
17   30595 occasion 2008
18   31694 occasion 2009
19   34169 occasion 2010
20   65615 occasion 2011
21   68019 occasion 2012
22   71008 occasion 2013
23  122929 occasion 2014
24   81025 occasion 2015

friedman.test(Friedtest$fishing,Friedtest$group,Fri
edtest$year)
        Friedman rank sum test
data:  Friedtest$fishing, Friedtest$group and
Friedtest$year
Friedman chi-squared = 14.25, df = 2,
p-value = 0.0008047
```

Post hoc test for Friedman test

```
pairwise.wilcox.test(Friedtest$fishing,Friedtes
t$year)
        Pairwise comparisons using Wilcoxon
rank sum test
data:  Friedtest$fishing and Friedtest$year
     2008 2009 2010 2011 2012 2013 2014
2009 1    -    -    -    -    -    -
2010 1    1    -    -    -    -    -
2011 1    1    1    -    -    -    -
2012 1    1    1    1    -    -    -
2013 1    1    1    1    1    -    -
2014 1    1    1    1    1    1    -
2015 1    1    1    1    1    1    1
P value adjustment method: holm

pairwise.wilcox.test(Friedtest$fishing,Friedtes
t$group)
        Pairwise comparisons using Wilcoxon
rank sum test
data:  Friedtest$fishing and Friedtest$group
         full    occasion
occasion 0.00047 -
part     0.00186 0.00047
P value adjustment method: holm
```

CHAPTER SIX

6.0 Bootstrapping in R

The boot package provides extensive facilities for bootstrapping and is related to resampling methods. You can bootstrap a single statistic (e.g median), or a vector (e.g regression weights).

```
bootobject<-boot(data=,statistic=, R)
```

data is a vector, matrix or data frame, statistic is a function that provide k statistics, R is the number of bootstrap replicates.

Illustration

The following are the dosage of a drug and reduction in blood sugar level from 7 patients.

Dosage(X): 0.38 0.51 0.19 0.53 0.39 0.38 0.66

Reduction in Blood Sugar (Y): 50 72 36 64 52 56 80

The R codes to run the following simple correlations are below

```
x<-c(0.38,0.51,0.19,0.53,0.39,0.38,0.66)
y<-c(50,72,36,64,52,56,80)
Cdata<-data.frame(y,x)
```

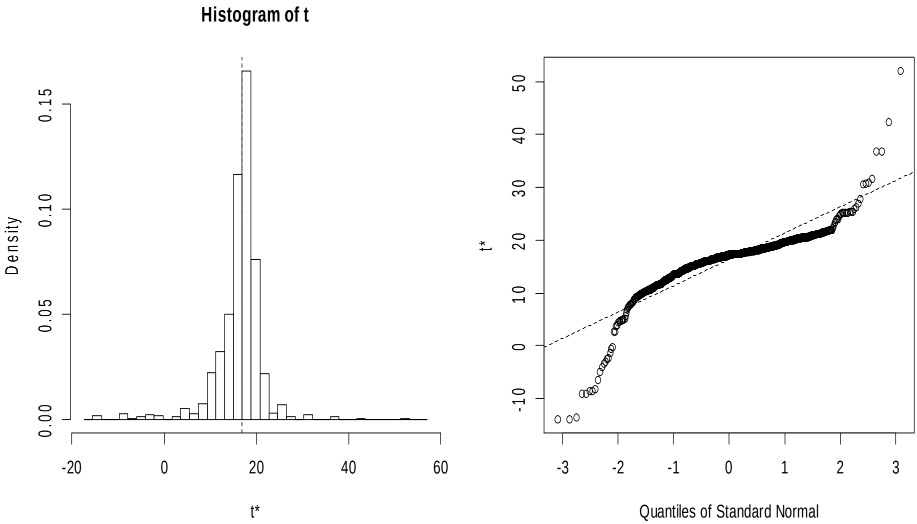
Bootstrap the Regression Coefficients

```
bs<-function(formula,data,indices){d<-
data[indices,]
fit<-lm(formula,data=d)
return(coef(fit))
}
```

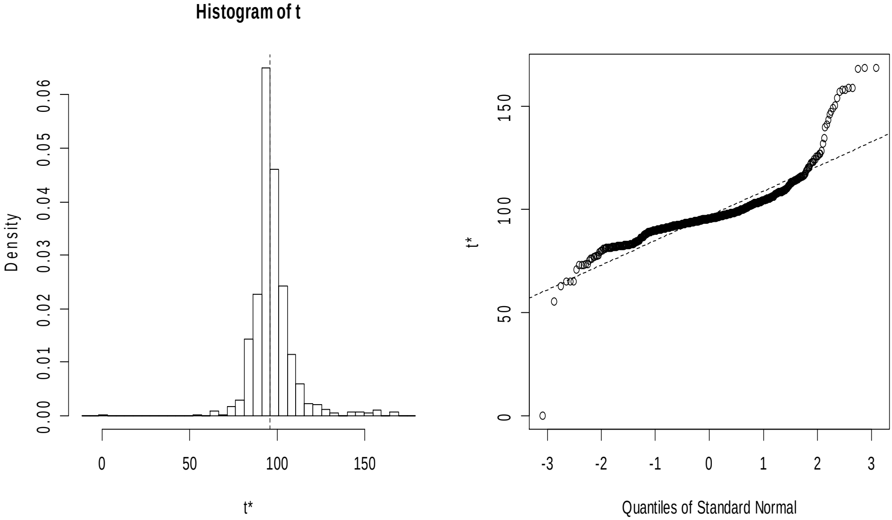
```
results<- boot(data=Cdata, statistic=bs,
R=1000,formula=y~x)

results
ORDINARY NONPARAMETRIC BOOTSTRAP
Call:
boot(data = Cdata, statistic = bs, R = 1000,
formula = y ~ x)
Bootstrap Statistics :
    original     bias    std. error
t1* 16.94773 -0.1345952    9.786349
t2* 95.84404  0.3100640   20.646486

summary(results)
     R original bootBias  bootSE bootMed
1 1000   16.948  -0.6841  5.7537  17.016
2 1000   95.844   1.4357 14.5128  95.524
plot(results,index=1)# intercept
```



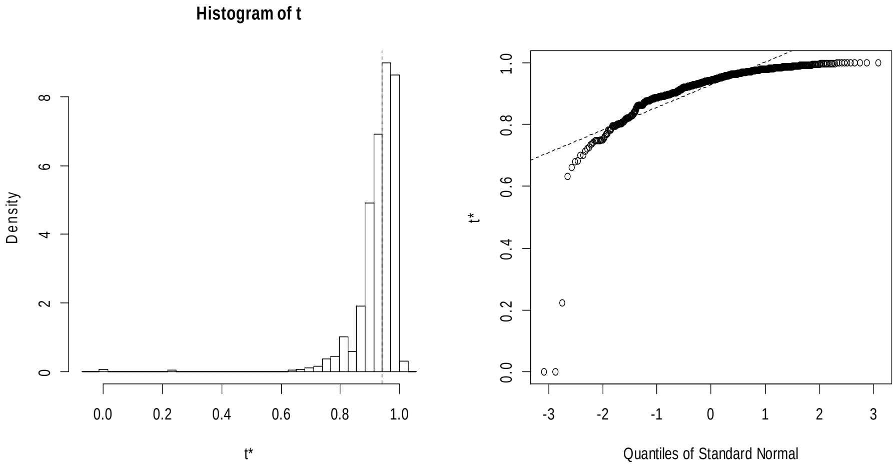
```
plot(results,index=2)# dosage
```



Bootstrap the R-square from Regression Model

```
rsq<-function(formula,data,indices){d<-
data[indices,]
fit<-lm(formula,data=d)
return(summary(fit)$r.square)
}
results<- boot(data=Cdata, statistic=rsq,
R=1000,formula=y~x)

summary(results)
  R     original bootBias bootSE   bootMed
1 1000  0.94119 -0.012528 0.073144 0.94333
plot(results)
```


Histogram of t
Density
0
2
4
6
8
0.0
0.2
0.4
0.6
0.8
1.0
t*
0.0
0.2
0.4
0.6
0.8
1.0
t*
-3
-2
-1
0
1
2
3
Quantiles of Standard Normal

Take Home Practice Exercises

Use R to Analyze the following Exercises

1. The three factors in the experiment are A= type of silica added to the mix, B= temperature of a water bath, and C= amount of time the material spends in the water bath. Each factor occurs at two levels, designated "High" and "Low". The data are given in table below.

Battery Separator Data

	C=Low		C=High	
A	B=Low	B=High	B=Low	B=High
Low	40.9 42.2 41.3	46.3 47.0 48.2	48.6 49.5 46.6	69.0 66.3 66.1
High	36.5 34.8 35.7	53.3 55.4 56.3	59.6 56.4 58.8	75.2 72.5 73.2

2. Given a Latin square design as

A(2.7)	B(2.6)	C(1.9)
B(2.2)	C(0.2)	A(2.3)
C(1.9)	A(2.1)	B(2.4)

3. Suppose we have a randomized block design (Two-way without interaction) below:

$$\begin{array}{c} \\ \\ t_1 \\ t_2 \\ t_3 \end{array}\begin{array}{c} \text{Block} \\ \begin{array}{cccc} b_1 & b_2 & b_3 & b_4 \end{array} \\ \begin{pmatrix} 7 & 8 & 6 & 10 \\ 10 & 9 & 12 & 14 \\ 20 & 15 & 25 & 26 \end{pmatrix} \end{array}$$

4. An experiment of four experimental diet (Treatments)-One way ANOVA.

	Diets		
1	2	3	4
7.0	5.3	4.9	8.8
9.9	5.7	7.6	8.9
8.5	4.7	5.5	8.1
5.1	3.5	2.8	3.3
10.3	7.7	8.4	9.1

5. The following data were obtained from an experiment

X_1	3	7	4	2	8	9	10	3	1	4	2
X_2	2	2	2	3	3	4	5	5	6	6	6
Y	4	3	8	18	22	24	24	18	13	10	16

Obtain the following:

i. Scatter of Y and X_1 and Y and X_2.
ii. Spearman Rank correlation of Y and X_2.
iii. Kendall Correlation coefficients of Y and X_1.
iv. The correlation Matrix of Y, X_1 and X_2.

6. Construct the frequency table for the data below. Hence Plot the Bar chart and Histogram for the data below

X	2	2	2	3	3	4	5	5	6	6	6

Also test for normality of the data.

7. Values of some export crops in 1965

Crop	Cocoa	Palm Produce	Groundnut
Value (₦m)	85.4	80.2	106.2

Represent the data with Pie Chart.

Bibliography

Emenogu, N. G. and Adenomon, M. O. (2018): Design and Analysis of Experiments with Examples in R. Niger State, Nigeria: Jube-Evans Books & Publication.

Everitt, B. S. (2002): The Cambridge Dictionary of Statistics (2nd ed). New York: Cambridge University Press.

Marques de Sá, J. P.(2007): Applied Statistics Using SPSS, STATISTICA, MATLAB and R (2nd ed). New York: Springer.

Nelson, P. R.; Coffin, M. & Copeland, K. A. F. (2003): Introductory Statistics for Engineering Experimentation. New York: Elsevier Academic Press.

Oyejola, B. A. (2003): Design & Analysis of Experiment for Biology and Agricultural Students. Kwara State: Olad Publishers.

Upton, G. & Cook, I. (2002): Oxford Dictionary of Statistics. New York: Oxford University Press.

Zar, J. H. (1999): Biostatistical Analysis. India: Dorling Kindersley Pvt. Ltd

Feed Back

Comments and, Statistical Consulting and Computing, feel free to contact at

admonsagie@gmail.com
admonsagie@yahoo.com
adenomonmo@nsuk.edu.ng
+2347036990145
+2348150775104

Foundation of Laboratory for Econometrics & Applied Statistics of Nigeria **(Aka FOUND-LEAS-IN-NIGERIA)**

…we Found Leas in Nigeria, Come Invest in it…

Objectives

- Statistical Advocacy in Nigeria
- Support women and girls in Mathematical and Statistical Sciences in Nigeria
- Support activities in World Statistics and Mathematics Day
- Engaged in Statistical and mathematical capacity building in Nigeria.
- Support indigent students in Mathematical and Statistical Sciences in Nigeria
- Organised free lectures on best practices in the field of Statistical Sciences.
- Lots more.

Group opens to the field of Statistics, Mathematics, Economics, Data science and physical sciences.

Join through the facebook link below

https://www.facebook.com/groups/566434690484084/