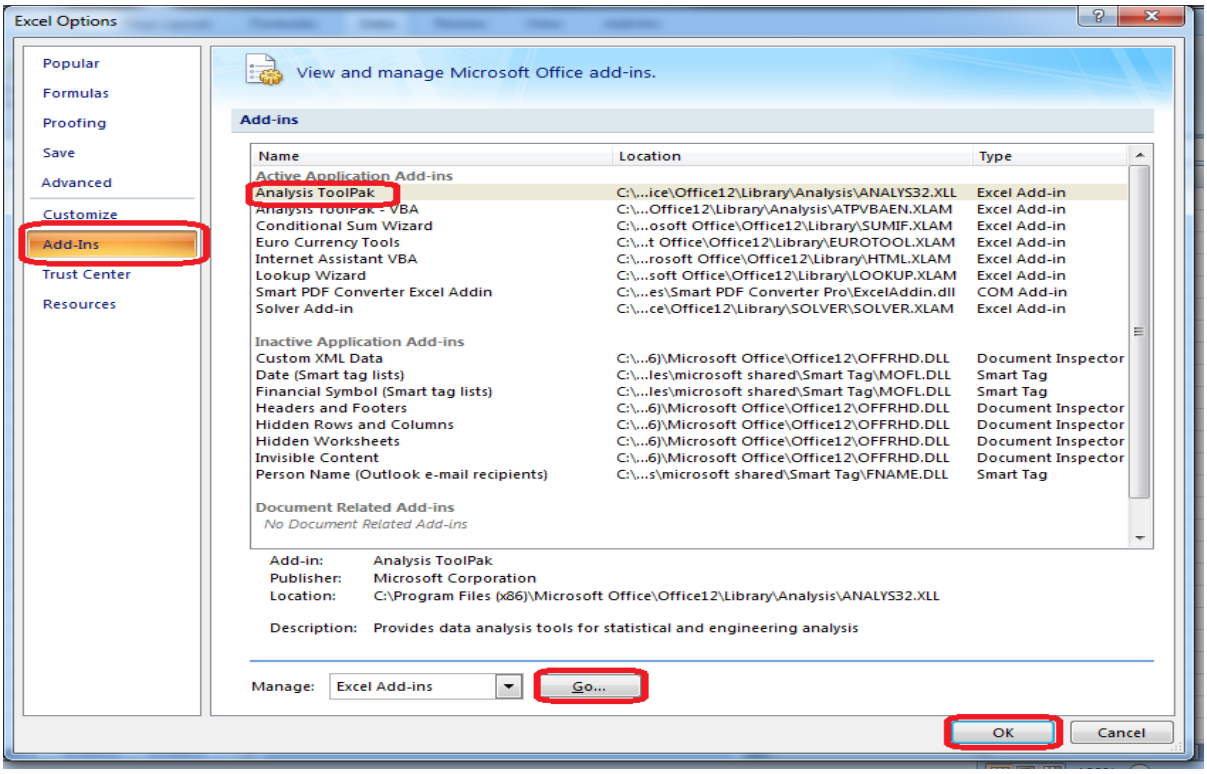
Practical 1

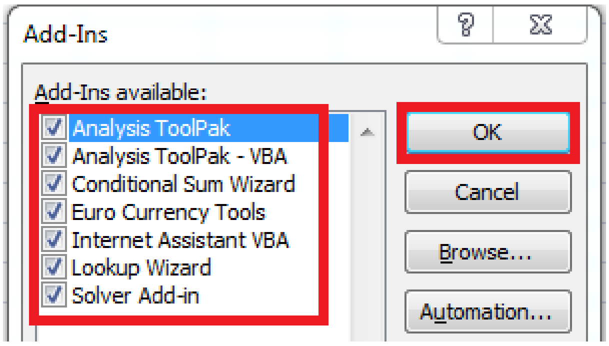
Aim: Write a program for obtaining descriptive statistics of data.

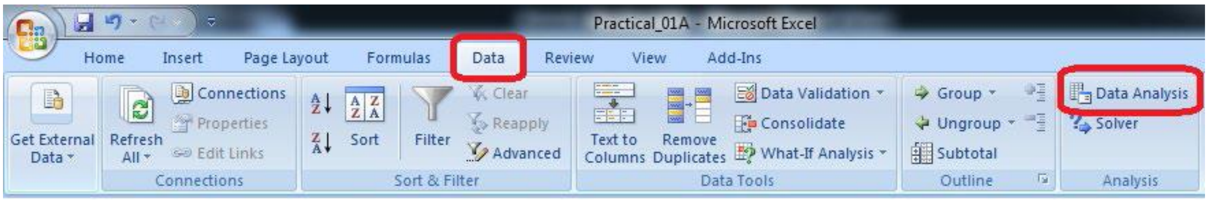
Program/Steps to obtain descriptive statistics of data.

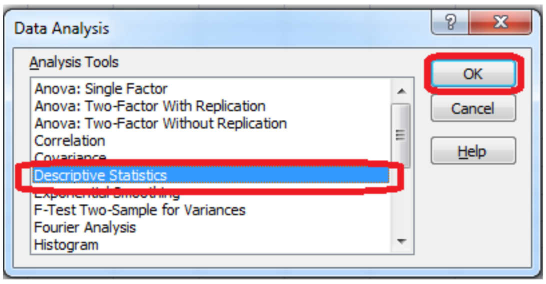
Using Excel

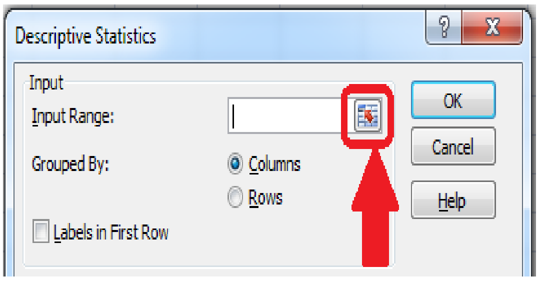
Go to File Menu -> Options ->Add-Ins -> Select Analysis Tool Pak -> Press OK



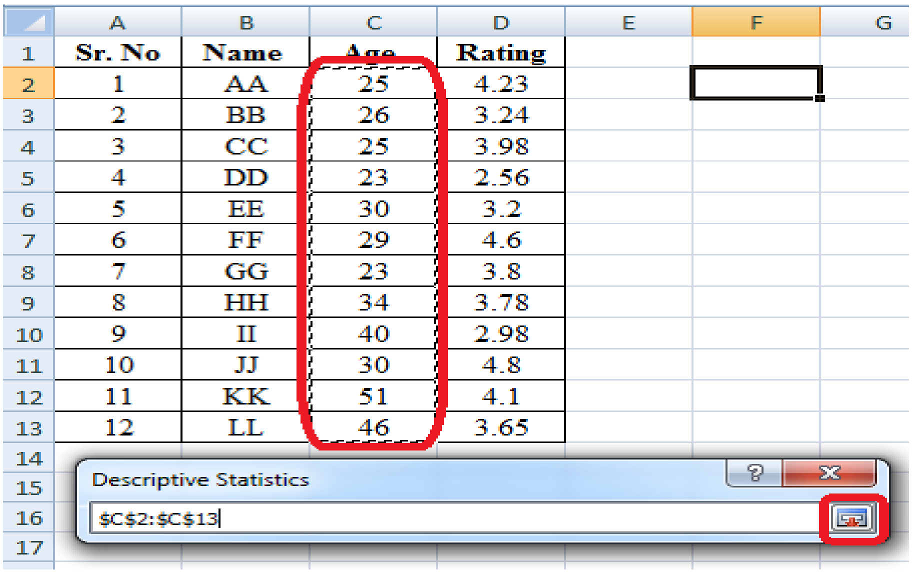




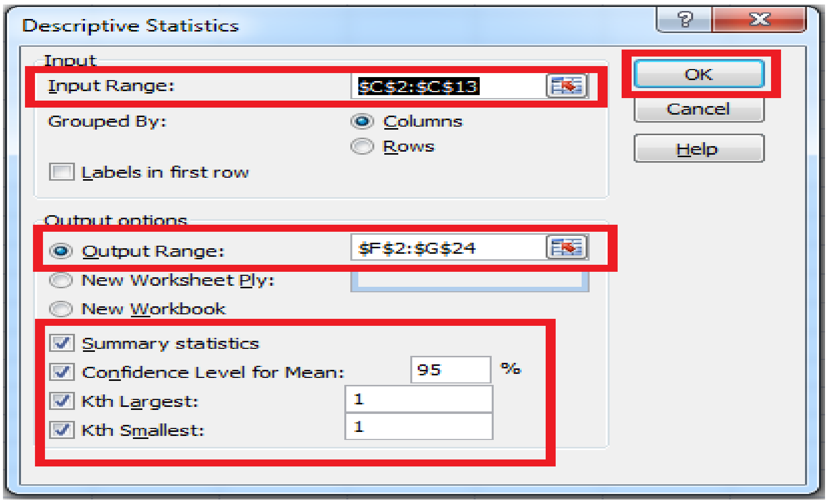




Select the data range from the excel worksheet.



	A	B	C	D	E	F	G
1	Sr. No	Name	Age	Rating			
2	1	AA	25	4.23			
3	2	BB	26	3.24			
4	3	CC	25	3.98			
5	4	DD	23	2.56			
6	5	EE	30	3.2			
7	6	FF	29	4.6			
8	7	GG	23	3.8			
9	8	HH	34	3.78			
10	9	II	40	2.98			
11	10	JJ	30	4.8			
12	11	KK	51	4.1			
13	12	LL	46	3.65			



Output:

	A	B	C	D	E	F	G
1	**Sr. No**	**Name**	**Age**	**Rating**			
2	1	AA	25	4.23		*Column1*	
3	2	BB	26	3.24			
4	3	CC	25	3.98		Mean	31.83333
5	4	DD	23	2.56		Standard Error	2.665246
6	5	EE	30	3.2		Median	29.5
7	6	FF	29	4.6		Mode	25
8	7	GG	23	3.8		Standard Deviation	9.232682
9	8	HH	34	3.78		Sample Variance	85.24242
10	9	II	40	2.98		Kurtosis	0.24931
11	10	JJ	30	4.8		Skewness	1.135089
12	11	KK	51	4.1		Range	28
13	12	LL	46	3.65		Minimum	23
14						Maximum	51
15						Sum	382
16						Count	12
17						Largest(1)	51
18						Smallest(1)	23
19						Confidence Level(95.0%)	5.866167

Practical 2

Aim:Import data from different data sources (from Excel, csv, mysql, sql server, oracle to R/Python/Excel)

#Read data from SQLite3

import pandas as pd

import sqlite3 as sq

Change database name/file if needed

sInputFileName='utility.db'

sInputTable='Country_Code'

conn = sq.connect(sInputFileName)

sSQL='select * FROM ' + sInputTable + ';'

InputData=pd.read_sql_query(sSQL, conn)

print('Input Data Values ====================================')

print(InputData)

print('===')

#Read data from MySQL

#Install Python-MySQL connector(Download from Resource folder)

Click Start> MySQL CLI> Login > Create database > create table > Insert records in table

```
mysql> create database mydb1;
Query OK, 1 row affected (0.01 sec)

mysql> use mydb1;
Database changed
mysql> create table test( name varchar(10), eid int);
Query OK, 0 rows affected (0.01 sec)

mysql> insert into test values('Abc', 1);
Query OK, 1 row affected (0.00 sec)

mysql> insert into test values('Def', 2);
Query OK, 1 row affected (0.01 sec)
```

#To install pymysql package run pip command on cmd

#pip3 install pymysql

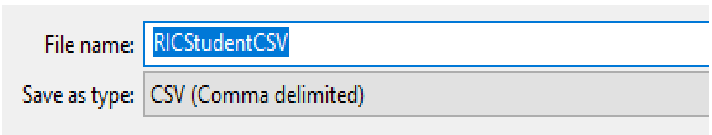
```
import pymysql
db = pymysql.connect("localhost","root","123456","mydb1" )
cursor = db.cursor()
sql = """SELECT * from TEST"""
try:
        cursor.execute(sql)
        results = cursor.fetchall()
      # Now print fetched result
        for row in results:
           print(row)
        print("success")
        db.commit()
except:
      db.rollback()
db.close()
```

```
===================== RESTART: A:\RIC\programs\dbprog.py
('Abc', 1)
('Def', 2)
success
```

#Read data from CSV

Create a csv file with random data or uses any existing CSV file.

	A	B	C
1	Math	Reading	Writing
2	48	68	63
3	62	81	72
4	79	80	78
5	76	83	79
6	59	64	62
7	69	84	85
8	70	84	83
9	46	48	41
10	61	78	80
11	86	78	77



Save file as CSV.

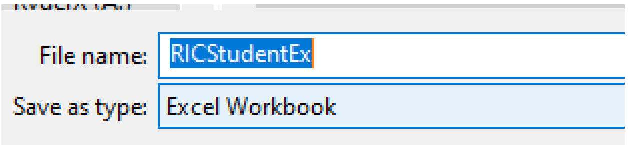
#to install pandas run pip3 command on cmd

```
#pip3 install pandas
#note keep RICStudentCSV.csv file and program in the same directory
import pandas as pd
marks = pd.read_csv('RICStudentCSV.csv')
print(marks)
print("Summary of data")
print(marks.describe())
```

```
==================== RESTART: A:/RIC/programs/ReadCSV.py
   Math  Reading  Writing
0    48       68       63
1    62       81       72
2    79       80       78
3    76       83       79
4    59       64       62
5    69       84       85
6    70       84       83
7    46       48       41
8    61       78       80
9    86       78       77
Summary of data
            Math    Reading   Writing
count  10.000000  10.000000  10.00000
mean   65.600000  74.800000  72.00000
std    12.937456  11.564313  13.35831
min    46.000000  48.000000  41.00000
25%    59.500000  70.500000  65.25000
50%    65.500000  79.000000  77.50000
75%    74.500000  82.500000  79.75000
max    86.000000  84.000000  85.00000
```

#Read data from Excel

Save the previous file as Excel.



import pandas as pd

marks = **pd.read_excel**('RICStudentEx.xlsx')

print(marks)

print("Summary of data")

print(marks.describe())

Ouput:-

```
==================== RESTART: A:/RIC/programs/ReadExcel.py
   Math  Reading  Writing
0    48       68       63
1    62       81       72
2    79       80       78
3    76       83       79
4    59       64       62
5    69       84       85
6    70       84       83
7    46       48       41
8    61       78       80
9    86       78       77
Summary of data
            Math    Reading   Writing
count  10.000000  10.000000  10.00000
mean   65.600000  74.800000  72.00000
std    12.937456  11.564313  13.35831
min    46.000000  48.000000  41.00000
25%    59.500000  70.500000  65.25000
50%    65.500000  79.000000  77.50000
75%    74.500000  82.500000  79.75000
max    86.000000  84.000000  85.00000
```

Practical 3

Aim:Design a survey form for a given case study, collect the primary data and analyse it

Case 1:

A researcher wants to conduct a Survey in colleges on Use of ICT in higher education from Mumbai, Thane and Navi Mumbai. The survey focuses on access to and use of ICT in teaching and learning, as well as on attitudes towards the use of ICT in teaching and learning.

Design questionnaire addressed to teachers seeks information about the target class, his experience using ICT for teaching, access to ICT infrastructure, support available, ICT based activities and material used, obstacles to the use of ICT in teaching, learning activities with the target class, your skills and attitudes to ICT, and some personal background information.

Arrange question in following groups:

1. Information about the target class you teach
2. Experience with ICT for teaching
3. ICT access for teaching
4. Support to teachers for ICT use
5. ICT based activities and material used for teaching
6. Obstacles to using ICT in teaching and learning
7. Learning activities with the target class
8. Teacher skills
9. Teacher opinions and attitudes
10. Personal background information

Case 2:

A research agency wants to study the perception about App based taxi service in Mumbai, Thane and Navi Mumbai. The survey focuses on customers attitude towards app base taxi service as well as on attitudes towards regular taxi cab.

Design questionnaire seeks information about the target taxi service, his experience using taxi services, access, support available, obstacles and some personal background information, with the following objectives:

1. To find out the customer satisfaction towards the App based-taxi services.
2. To find the level of convenience and comfort with App based -taxi services.
3. To know their opinion about the tariff system and promptness of service.

4. To ascertain the customer view towards the driver behaviour and courtesy.
5. To provide inputs to enhance the services to delight the customers.
6. To examine relationship between service quality factors and taxi passenger satisfaction.
7. To suggest better regulations for transportation authorities regarding customer protection and effective monitoring of taxi services.

Case 3:

A popular electronic store want to conduct a survey to develop awareness of branded laptop baseline estimates and determine popularity of different company's laptop. It suggests steps to be initiated or strengthened in the field of demand in a region. The key indicators are among the general population, demand branded laptop and the problem users.

The objectives of this particular study are:-

1. To know the preferences of different types of branded laptops by students and professionals.
2. To study which factor influence for choosing different types of branded laptops.

3. To know about the level of satisfaction towards different types of branded laptops.
4. To identify the perception of consumers towards the laptop positioning strategy.
5. To know the consumer preference towards laptop in the present era.

Using the collected data for analysis

	A	B	C	D	E	F	G	H	I	J	K
1	Respondent ID	Question1	question2	Question3	Question4	Question5	Question6	Question7	Question8	Question9	Question10
2	1	Agree	Strongly Agree	Disagree	strongly Disagree	Agree	Strongly Agree	Disagree	strongly Disagree	Strongly Agree	Disagree
3	2	Agree	Strongly Agree	Disagree	strongly Disagree	Agree	Agree	Strongly Agree	Disagree	strongly Disagree	Disagree
4	3	Agree	Strongly Agree	Disagree	strongly Disagree	Agree	Agree	Strongly Agree	Agree	Strongly Agree	Disagree
5	4	Agree	Strongly Agree	Disagree	strongly Disagree	Agree	Agree	Strongly Agree	Agree	Strongly Agree	Agree
6	5	Agree	Strongly Agree	Disagree	strongly Disagree	Agree	Agree	Strongly Agree	Agree	Strongly Agree	Agree
7	6	Agree	Strongly Agree	Disagree	strongly Disagree	Agree	Agree	Strongly Agree	Agree	Strongly Agree	Disagree
8	7	strongly Disagree	Agree	Strongly Agree	Disagree	Agree	Agree	Strongly Agree	Disagree	strongly Disagree	Disagree
9	8	strongly Disagree	Agree	Strongly Agree	Disagree	Agree	Strongly Agree	Disagree	strongly Disagree	Strongly Agree	Disagree
10	9	strongly Disagree	Agree	Disagree	strongly Disagree	Strongly Agree	Strongly Agree	Disagree	strongly Disagree	Strongly Agree	Disagree
11	10	strongly Disagree	Agree	Disagree	strongly Disagree	Strongly Agree	Strongly Agree	Disagree	strongly Disagree	Strongly Agree	Disagree
12	11	strongly Disagree	Agree	Agree	Strongly Agree	Strongly Agree	Strongly Agree	Disagree	strongly Disagree	Strongly Agree	Disagree
13	12	strongly Disagree	strongly Disagree	Agree	Agree	strongly Disagree	Strongly Agree	Disagree	strongly Disagree	Strongly Agree	Disagree
14	13	strongly Disagree	strongly Disagree	Agree	Strongly Agree	Disagree	Strongly Agree	Agree	Strongly Agree	Strongly Agree	Disagree
15	14	strongly	Strongly	Agree	Strongly	Disagree	strongly	Agree	Strongly	Strongly	Disagree

Sheet1 Sheet2 Sheet3

	A	B	C	D	E	F	G	H	I	J	K
22											
23	count(N)	20	20	20	20	20	20	20	20	20	20
24	Not Answer	0	0	0	0	0	0	0	0	0	0
25	Toal	20	20	20	20	20	20	20	20	20	20
26	agree	11									
27											
28	disagree	1	1	9	2	4	1	8	2	0	15
29	strongly disagree	8	5	1	10	3	2	3	7	2	2
30	agree	11	5	8	3	10	10	2	9	3	2
31	strongly agree	0	9	2	5	3	7	7	2	15	1
32	total	20	20	20	20	20	20	20	20	20	20
33	percentage										
34	disagree(%)	5%	5%	45%	10%	20%	5%	40%	10%	0%	75%
35	strongly disagree(%)	40%	25%	5%	50%	15%	10%	15%	35%	10%	10%
36	agree(%)	55%	25%	40%	15%	50%	50%	10%	45%	15%	10%
37	strongly agree(%)	0%	45%	10%	25%	15%	35%	35%	10%	75%	5%
38	total	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

Sheet1 Sheet2 Sheet3

Formula for operation

count(N) = COUNTA(B2:B21)

Not Answer = COUNT(D2:D21)

Toal = SUM(C23:C24)

Disagree = COUNTIF(B$2:B$21,$A28)

strongly disagree = COUNTIF(E$2:E$21,$A29)

Agree =COUNTIF(D$2:D$21,$A31)

strongly agree = COUNTIF(D$2:D$21,$A31)

Total = SUM(B28:B31)

Percentage

disagree(%) =B28/B32

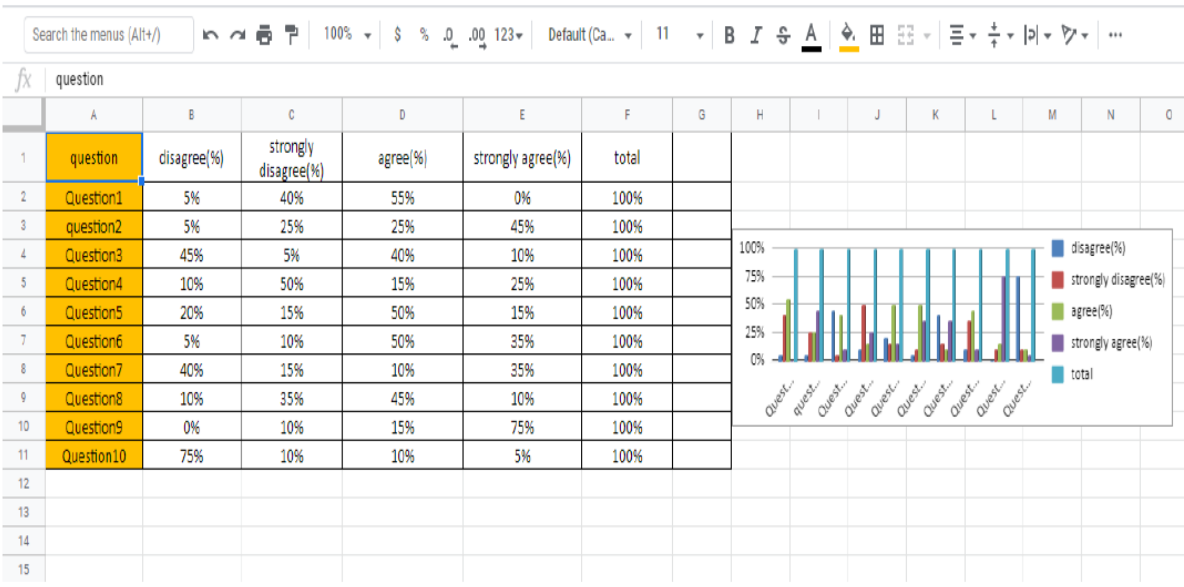
strongly disagree(%) = B29/B32

agree(%) = B30/B32

strongly agree(%) = =B31/B32

Total = SUM(B34:B37)

Output:-



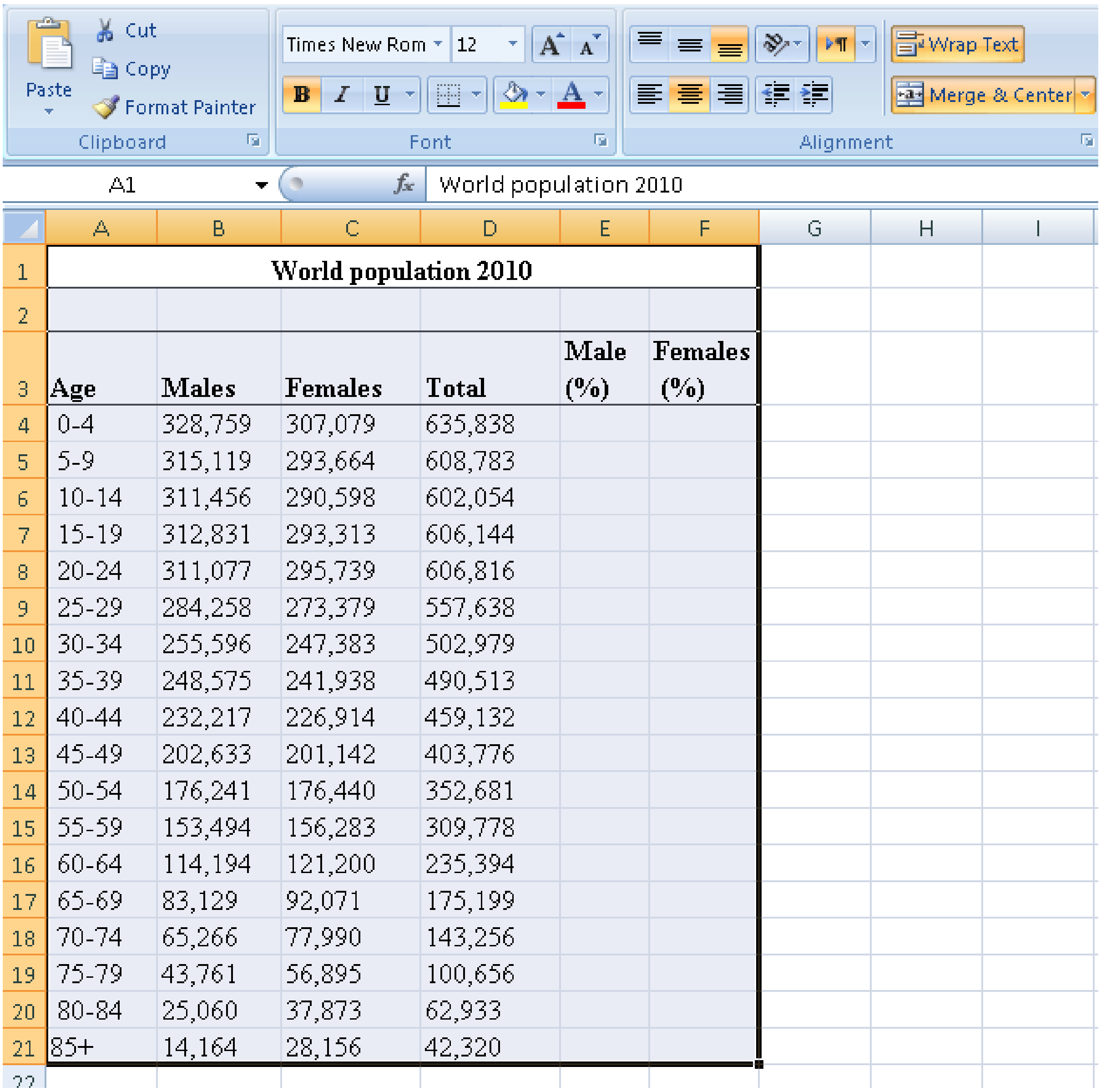
question	disagree(%)	strongly disagree(%)	agree(%)	strongly agree(%)	total
Question1	5%	40%	55%	0%	100%
question2	5%	25%	25%	45%	100%
Question3	45%	5%	40%	10%	100%
Question4	10%	50%	15%	25%	100%
Question5	20%	15%	50%	15%	100%
Question6	5%	10%	50%	35%	100%
Question7	40%	15%	10%	35%	100%
Question8	10%	35%	45%	10%	100%
Question9	0%	10%	15%	75%	100%
Question10	75%	10%	10%	5%	100%

Practical 4

Aim:Perform analysis of given secondary data.
Steps in Secondary Data Analysis
1. **Determine your research question** – Knowing exactly what you are looking for.
2. **Locating data**– Knowing what is out there and whether you can gain access to it. A quick Internet search, possibly with the help of a librarian, will reveal a wealth of options.
3. **Evaluating relevance of the data** – Considering things like the data's original purpose, when it was collected, population, sampling strategy/sample, data collection protocols, operationalization of concepts, questions asked, and form/shape of the data.
4. **Assessing credibility of the data** – Establishing the credentials of the original researchers, searching for full explication of methods including any problems encountered, determining how consistent the data is with data from other sources, and discovering whether the data has been used in any credible published research.
5. **Analysis –** This will generally involve a range of statistical processes.

Example: Analyze the given Population Census Data for Planning and Decision Making by using the size and composition of populations.

Output:-



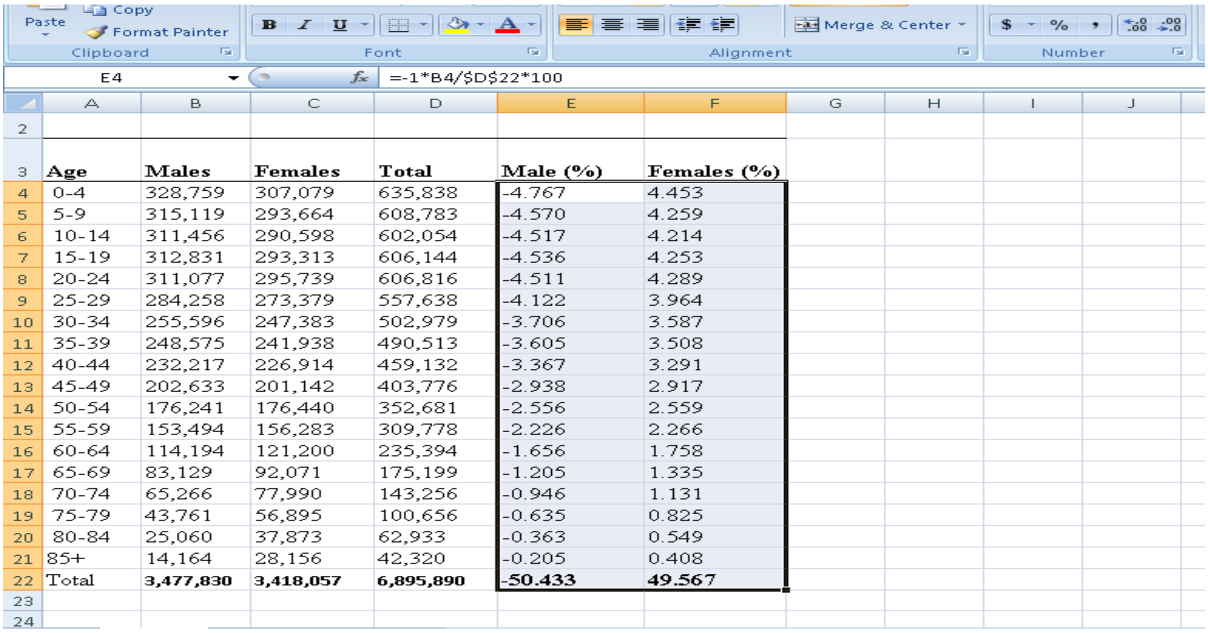
World population 2010					
Age	**Males**	**Females**	**Total**	**Male (%)**	**Females (%)**
0-4	328,759	307,079	635,838		
5-9	315,119	293,664	608,783		
10-14	311,456	290,598	602,054		
15-19	312,831	293,313	606,144		
20-24	311,077	295,739	606,816		
25-29	284,258	273,379	557,638		
30-34	255,596	247,383	502,979		
35-39	248,575	241,938	490,513		
40-44	232,217	226,914	459,132		
45-49	202,633	201,142	403,776		
50-54	176,241	176,440	352,681		
55-59	153,494	156,283	309,778		
60-64	114,194	121,200	235,394		
65-69	83,129	92,071	175,199		
70-74	65,266	77,990	143,256		
75-79	43,761	56,895	100,656		
80-84	25,060	37,873	62,933		
85+	14,164	28,156	42,320		

Put the cursor in cell **B22** and click on the **AutoSum** and then click **Enter**. This will calculate the total population. Then copy the formula in cell **D22** across the row **22.**
(Total_population)
To calculate the percent of males in cell **E4**, enter the formula
[-1*100*Male_count*Total_population]
=-1*100*B4/$D$22
And copy the formula in cell **E4** down to cell **E21.**
To calculate the percent of females in cell **F4**, enter the formula

[1*100*Female_count*Total_population]
=**100*C4/$D$22**. Copy the formula in cell **F4** down to cell **F21.**

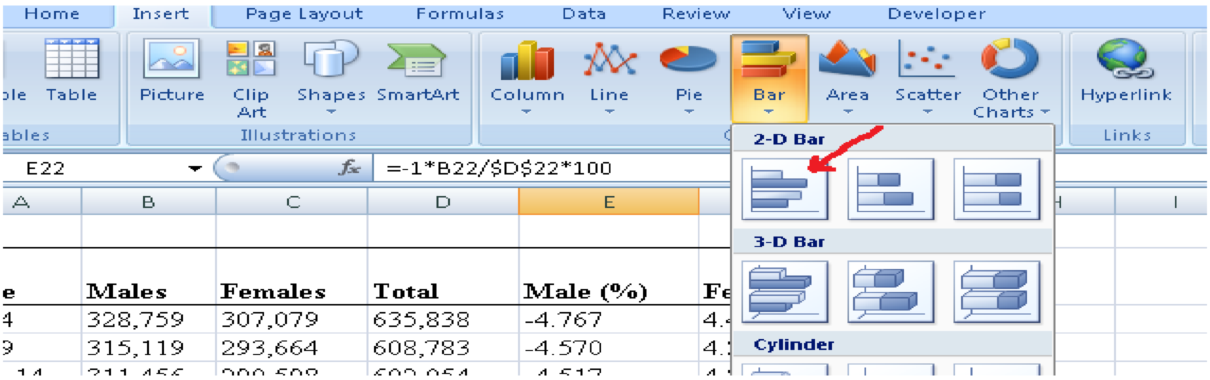
This gives percentage in +ve for female and –ve foe male.

To build the population pyramid, we need to choose a horizontal bar chart with two series of data (% male and % female) and the age labels in column A as the **Category X-axis** labels. Highlight the range **A3:A21**, hold down the CTRL key and highlight the range **E3:F21**



Age	Males	Females	Total	Male (%)	Females (%)
0-4	328,759	307,079	635,838	-4.767	4.453
5-9	315,119	293,664	608,783	-4.570	4.259
10-14	311,456	290,598	602,054	-4.517	4.214
15-19	312,831	293,313	606,144	-4.536	4.253
20-24	311,077	295,739	606,816	-4.511	4.289
25-29	284,258	273,379	557,638	-4.122	3.964
30-34	255,596	247,383	502,979	-3.706	3.587
35-39	248,575	241,938	490,513	-3.605	3.508
40-44	232,217	226,914	459,132	-3.367	3.291
45-49	202,633	201,142	403,776	-2.938	2.917
50-54	176,241	176,440	352,681	-2.556	2.559
55-59	153,494	156,283	309,778	-2.226	2.266
60-64	114,194	121,200	235,394	-1.656	1.758
65-69	83,129	92,071	175,199	-1.205	1.335
70-74	65,266	77,990	143,256	-0.946	1.131
75-79	43,761	56,895	100,656	-0.635	0.825
80-84	25,060	37,873	62,933	-0.363	0.549
85+	14,164	28,156	42,320	-0.205	0.408
Total	3,477,830	3,418,057	6,895,890	-50.433	49.567

Under **inset** tab, under horizontal bar charts select **clustered bar chart**



Put the tip of your mouse arrow on the **Y-axis** (vertical axis) so it says "Category Axis", right click and chose **Format Axis**

Choose **Axis options** tab and set the major and minor tick mark type to **None**, Axis labels to **Low**, and click **OK**.

Click on any of the bars in your pyramid, click right and select "format data series". Set the **Overlap** to **100** and **Gap Width** to **0**. Click **OK**.

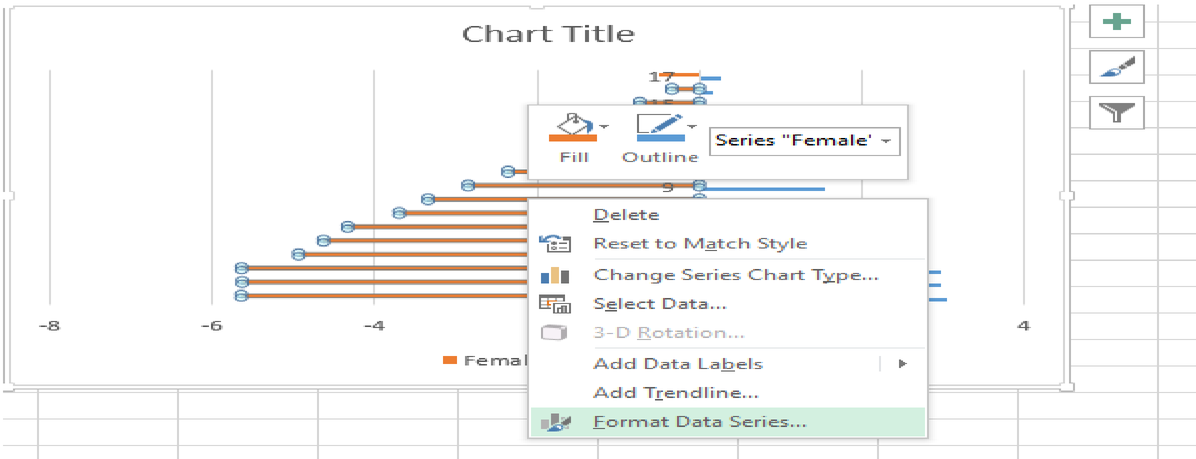
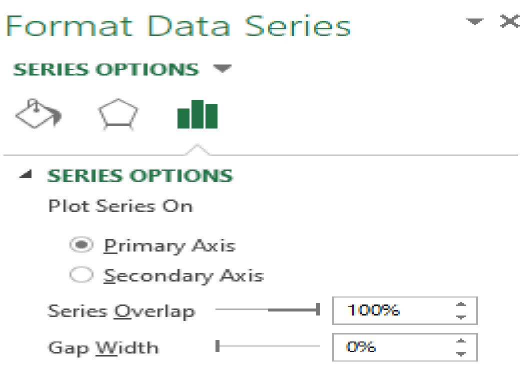
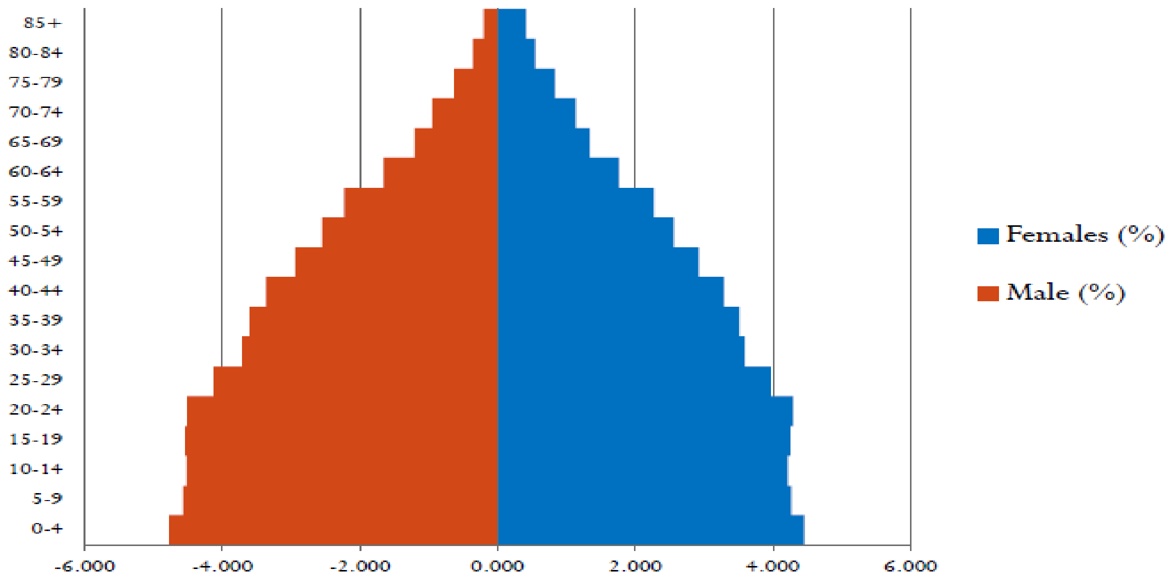


Chart Title
Fill
Outline
Series "Female'
Delete
Reset to Match Style
Change Series Chart Type...
Select Data...
3-D Rotation...
Add Data Labels
Add Trendline...
Format Data Series...
-8
-6
-4
4
Femal


Format Data Series
SERIES OPTIONS
SERIES OPTIONS
Plot Series On
Primary Axis
Secondary Axis
Series Overlap
100%
Gap Width
0%


85+
80-84
75-79
70-74
65-69
60-64
55-59
50-54
45-49
40-44
35-39
30-34
25-29
20-24
15-19
10-14
5-9
0-4
-6.000
-4.000
-2.000
0.000
2.000
4.000
6.000
Females (%)
Male (%)

Practical 5

Aim: Perform testing of hypothesis using one sample t-test.

One sample t-test : The One Sample *t* Test determines whether the sample mean is statistically different from a known or hypothesised population mean. The One Sample *t* Test is a parametric test.

H0: Mean age of given sample is 30.

H1: Mean age of given sample is not 30

#pip3 install scipy

#pip3 install numpy

```
from scipy.stats import ttest_1samp
import numpy as np
ages = np.genfromtxt('ages.csv')
print(ages)
ages_mean = np.mean(ages)
print("Mean age:",ages_mean)
print("Test 1: m=30")
tset, pval = ttest_1samp(ages, 30)
print('p-values - ',pval)
if pval< 0.05:
   print("we reject null hypothesis")
else:
   print("we fail to reject null hypothesis")
```

Output:-

```
==================== RESTART: A:/RIC/programs/RICttest.py ====================
[20. 30. 25. 13. 16. 17. 34. 35. 38. 43. 45. 48. 49. 50. 51. 54. 55. 56.
 59. 61. 62. 18. 22. 29.]
Mean age: 38.75
Test 1: m=30
p-values -  0.01333239479255858
we reject null hypothesis
```

#Test 2

H0: Mean age of given sample is 38.

H1: Mean age of given sample is not 38.

```
from scipy.stats import ttest_1samp
import numpy as np
ages = np.genfromtxt('ages.csv')
print(ages)
ages_mean = np.mean(ages)
print("Mean age:",ages_mean)
print("Test 2: m=38")
tset, pval = ttest_1samp(ages, 38)
print('p-values - ',pval)
if pval< 0.05:
   print("we reject null hypothesis")
else:
   print("we fail to reject null hypothesis")
```

```
==================== RESTART: A:/RIC/programs/RICttest.py ====================
[20. 30. 25. 13. 16. 17. 34. 35. 38. 43. 45. 48. 49. 50. 51. 54. 55. 56.
 59. 61. 62. 18. 22. 29.]
Mean age: 38.75
Test 2: m=38
p-values -  0.8202593087020069
we fail to reject null hypothesis
```

Practical 6

Aim: Write a program for t-test comparing two means for independent samples.

The *t* distribution provides a good way to perform one sample tests on the mean when the population variance is not known provided the population is normal or the sample is sufficiently large so that the Central Limit Theorem applies.

Two Sample t Test

Example: A college Princiapal informed classroom teachers that some of their students showedunusual potential for intellectual gains. One months later the students identified to teachers ashaving potentional for unusual intellectual gains showed significiantly greater gains performanceon a test said to measure IQ than did students who were not so identified. Below are the data forthe students:

Experimental	Comparison	
35	2	
40	27	
12	38	
15	31	
21	1	
14	19	
46	1	
10	34	
28	3	
48	1	
16	2	
30	3	
32	2	
48	1	
31	2	
22	1	
12	3	
39	29	
19	37	
25	2	
27.15	11.95	Mean
12.51	14.61	Sd

Experimental Data

To calculate Standard Mean go to cell A22 and type =SUM(A2:A21)/20

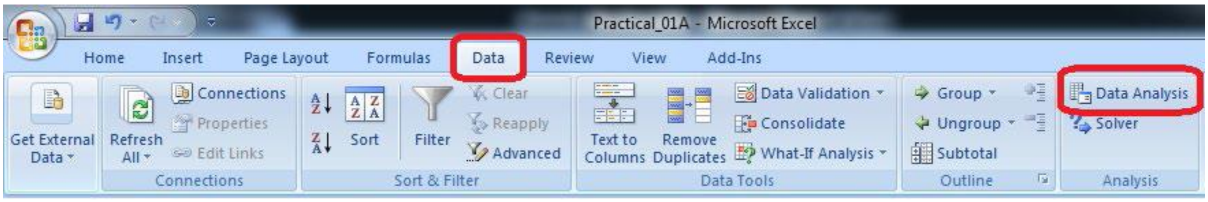
To calculate Standard Deviation go to cell A23 and type =STDEV(A2:A21)

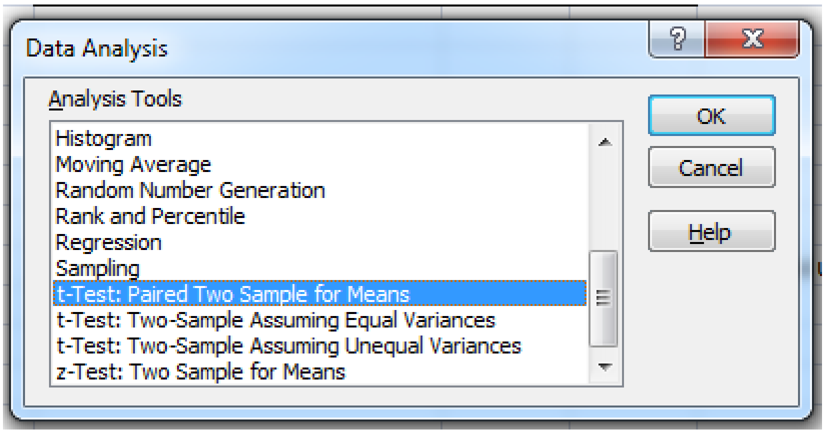
Comparison Data

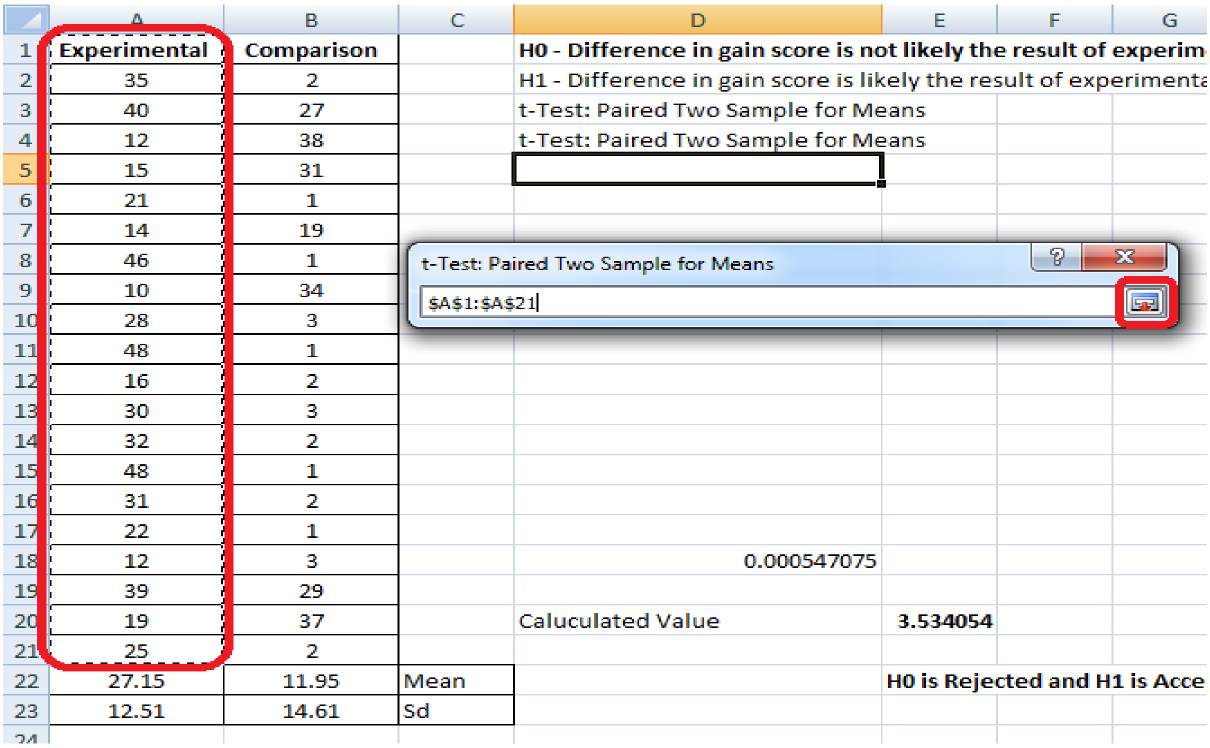
To calculate Standard Mean go to cell B22 and type =SUM(B2:B21)/20

To calculate Standard Deviation go to cell B23 and type =STDEV(B2:B21)

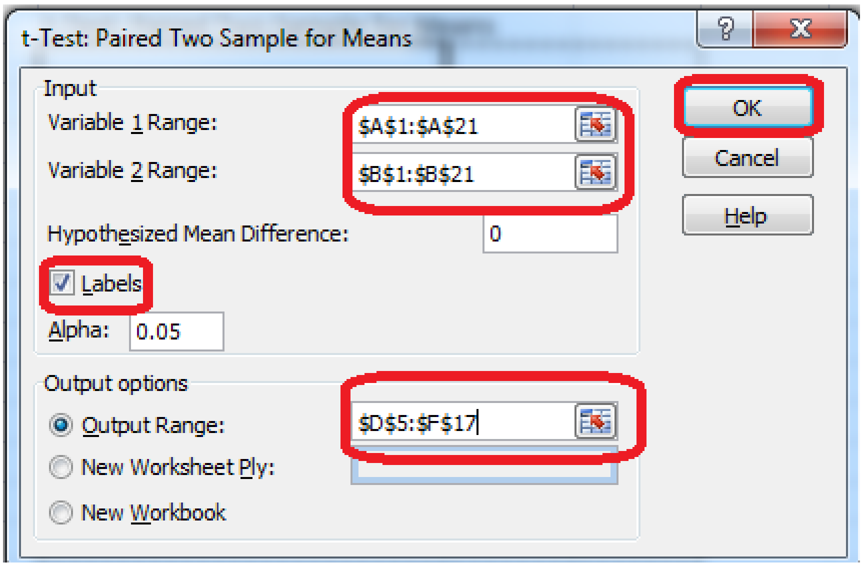
To find T-Test Statistics go to data Data Analysis







	A	B	C	D	E	F	G
1	Experimental	Comparison		H0 - Difference in gain score is not likely the result of experim			
2	35	2		H1 - Difference in gain score is likely the result of experimenta			
3	40	27		t-Test: Paired Two Sample for Means			
4	12	38		t-Test: Paired Two Sample for Means			
5	15	31					
6	21	1					
7	14	19					
8	46	1					
9	10	34					
10	28	3					
11	48	1					
12	16	2					
13	30	3					
14	32	2					
15	48	1					
16	31	2					
17	22	1					
18	12	3		0.000547075			
19	39	29					
20	19	37		Caluculated Value	3.534054		
21	25	2					
22	27.15	11.95	Mean		H0 is Rejected and H1 is Acce		
23	12.51	14.61	Sd				



To caluculate the T-Test square value go to cell E20 and type
=(A22-B22)/SQRT((A23*A23)/COUNT(A2:A21)+(B23*B23)/COUNT(A2:A21))
Now go to cell E20 and type
=IF(E20<E12,"H0 is Accepted", "H0 is Rejected and H1 is Accepted")
Our calculated value is larger than the tabled value at alpha = .01, so we reject the null hypothesisand accept the alternative hypothesis, namely, that the difference in gain scores is likely the resultof the experimental treatment and not the result of chance variation.

Output:

	A	B	C	D	E	F	G	H	I	J	K
1	Experimental	Comparison		**H0 - Difference in gain score is not likely the result of experimental treatment.**							
2	35	2		H1 - Difference in gain score is likely the result of experimental treatment and not the result of change variation.							
3	40	27		t-Test: Paired Two Sample for Means							
4	12	38		t-Test: Paired Two Sample for Means							
5	15	31		t-Test: Paired Two Sample for Means							
6	21	1									
7	14	19			*Experimental*	*Comparison*					
8	46	1		Mean	27.15	11.95					
9	10	34		Variance	156.45	213.5236842					
10	28	3		Observations	20	20					
11	48	1		Pearson Correlation	-0.395904927						
12	16	2		Hypothesized Mean Difference	0						
13	30	3		df	19						
14	32	2		t Stat	2.996289153						
15	48	1		P(T<=t) one-tail	0.003711226						
16	31	2		t Critical one-tail	1.729132792						
17	22	1		P(T<=t) two-tail	0.007422452						
18	12	3		t Critical two-tail	2.09302405						
19	39	29									
20	19	37		Caluculated Value	3.534053898						
21	25	2									
22	27.15	11.95	Mean		**H0 is Rejected and H1 is Accepted**						
23	12.51	14.61	Sd								

Practical 7

Aim: Perform testing of hypothesis using paired t-test.

The paired sample t-test is also called dependent sample t-test. It's an univariate test that tests for a significant difference between 2 related variables. An example of this is if you where to collect the blood pressure for an individual before and after some treatment, condition, or time point. The data set contains blood pressure readings before and after an intervention. These are variables "bp_before" and "bp_after".

The hypothesis being test is:
- **H0** - The mean difference between sample 1 and sample 2 is equal to 0.
- **H1** - The mean difference between sample 1 and sample 2 is not equal to 0

```
from scipy import stats

import matplotlib.pyplot as plt

import pandas as pd

df = pd.read_csv("blood_pressure.csv")

print(df[['bp_before','bp_after']].describe())

tst,pval=stats.ttest_rel(df['bp_before'], df['bp_after'])

if pval< 0.05:

    print("we reject null hypothesis")

else:

    print("we fail to reject null hypothesis")
```

Output:-

```
================= RESTART: A:/RIC/programs/RICPairedTest.py
        bp_before    bp_after
count  120.000000  120.000000
mean   156.450000  151.358333
std     11.389845   14.177622
min    138.000000  125.000000
25%    147.000000  140.750000
50%    154.500000  149.500000
75%    164.000000  161.000000
max    185.000000  185.000000
we reject null hypothesis
```

Practical 8:

Aim: Perform testing of hypothesis using chi-squared godness-of-fit test.

Problem

Ansystem administrator needs to upgrade the computers for his division. He wants to know what sort of computer system his workers prefer. He gives three choices: Windows, Mac, or Linux. Test the hypothesis or theory that an equal percentage of the population prefers each type of computer system .

System	O	Ei	$\sum \frac{(O_i - E_i)^2}{Ei}$
Windows	20	33.33%	
Mac	60	33.33%	
Linux	20	33.33%	

H0 : The population distribution of the variable is the same as the proposed distribution
HA : The distributions are different
To calculate the Chi –Squred value for Windows go to cell D2 and type
=((B2-C2)*(B2-C2))/C2
To calculate the Chi –Squred value for Mac go to cell D3 and type
=((B3-C3)*(B3-C3))/C3
To calculate the Chi –Squred value for Mac go to cell D3 and type
=((B4-C4)*(B4-C4))/C4

Go to Cell D5 for $\sum \frac{(O_i - E_i)^2}{Ei}$ and type
=SUM(D2:D4)
To get the table value for Chi-Square for α = 0.05 and dof = 2, go to cell D7 and type
=CHIINV(0.05,2)
At cell D8 type =IF(D5>D7, "H0 Accepted","H0 Rejected")

Output:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	System	O	Ei	$\sum \frac{(O_i - E_i)^2}{Ei}$										
2	Windows	20	33.33	5.333333		H0 : The population distribution of the variable is the same as the proposed distribution								
3	Mac	60	33.33	21.33333		H1 - : The distributions are different								
4	Linux	20	33.33	5.333333										
5	Total	100	100	32										
6														
7			Table Value	5.991465										
8			H0 Accepted											

Practical 9

Aim: Perform testing of hypothesis using chi-squared test of independence.

In a study to understatnd the permormacne of M. Sc. IT Part -1 class, a college selects a random sample of 100 students. Each student was asked his grade obtained in B. Sc. IT.

Null Hypothesis - H0 : The performance of girls students is same as boys students.
Alternate Hypothesis - H1 : The performance of boys and girls students are different.
Open Excel Workbook

	O	A	B	C	D	Total	$\sum\frac{(O_i - E_i)^2}{Ei}$
Girls	11	7	5	5	11	**39**	6.075
Boys	30	4	3	10	14	**61**	6.075
Total	41	11	8	15	25	**100**	**12.150**
Ei	**20.5**	**5.5**	**4**	**7.5**	**12.5**	**50**	

To get the table value go to cell T11 and type **=CHIINV(0.05,4)**
Go to cell O13 and type =IF(T8>=T11," H0 is Accepted", "H0 is Rejected")

M	N	O	P	Q	R	S	T
H0 : Performance of boys and girls are equal							
Frequency Table							$\frac{(O_i - E_i)^2}{}$
	O	**A**	**B**	**C**	**D**	**Total**	**Ei**
Girls	11	7	5	5	11	**39**	6.075
Boys	30	4	3	10	14	**61**	6.075
Total	41	11	8	15	25	**100**	**12.150**
Ei	**20.5**	**5.5**	**4**	**7.5**	**12.5**	**50**	
Critcal Value of α =0.05 for df = (2-1) * (5-1)							9.487729
Decesion		**H0 is Accepted**					

Practical 10A:

Aim: Perform testing of hypothesis using Z-test.

Use a Z test if:
• Your sample size is greater than 30. Otherwise, use a t test.
• Data points should be independent from each other. In other words, one data point isn’t related or doesn’t affect another data point.
• Your data should be normally distributed. However, for large sample sizes (over 30) this doesn’t always matter.
• Your data should be randomly selected from a population, where each item has an equal chance of being selected.
• Sample sizes should be equal if at all possible.

Ho - Blood pressure has a mean of 156 units

Program Code for one-sample Z test.

```
from statsmodels.stats import weightstats as stests

import pandas as pd

from scipy import stats

df = pd.read_csv("blood_pressure.csv")

print(df['bp_before'].describe())

ztest ,pval = stests.ztest(df['bp_before'], x2=None, value=156)

print("pval:",float(pval))

if pval< 0.05:

    print("we reject null hypothesis")

else:

    print("we fail to reject null hypothesis")
```

Output:-

```
=================== RESTART: A:/RIC/programs/RIC1SampleZ.py
count    120.000000
mean     156.450000
std       11.389845
min      138.000000
25%      147.000000
50%      154.500000
75%      164.000000
max      185.000000
Name: bp_before, dtype: float64
pval: 0.6651614730255063
we fail to reject null hypothesis
```

Practical 10B

Aim:Two-sample Z test

In two sample z-test , similar to t-test here we are checking two independent data groups and deciding whether sample mean of two group is equal or not.

H0 : mean of two group is 0

H1 : mean of two group is not 0

```
import pandas as pd

from statsmodels.stats import weightstats as stests

df = pd.read_csv("blood_pressure.csv")

print(df[['bp_before','bp_after']].describe())

ztest,pval=stests.ztest(df['bp_before'],x2=df['bp_after'],value=0,alternative= 'two-sided')

print("pval:",float(pval))

if pval< 0.05:

   print("we reject null hypothesis")

else:

   print("we fail to reject null hypothesis")
```

```
=================== RESTART: A:/RIC/programs/RIC2SampleZ.py
        bp_before   bp_after
count  120.000000  120.000000
mean   156.450000  151.358333
std     11.389845   14.177622
min    138.000000  125.000000
25%    147.000000  140.750000
50%    154.500000  149.500000
75%    164.000000  161.000000
max    185.000000  185.000000
pval: 0.002162306611369422
we reject null hypothesis
```

Practical 11

Aim: Perform testing of hypothesis using One-way ANOVA.

ANOVA Assumptions

• The dependent variable (SAT scores in our example) should be continuous.
• The independent variables (districts in our example) should be two or more categorical groups.
• There must be different participants in each group with no participant being in more than one group. In our case, each school cannot be in more than one district.
• The dependent variable should be approximately normally distributed for each category.
• Variances of each group are approximately equal.

From our data exploration, we can see that the average SAT scores are quite different for each district. Since we have five different groups, we cannot use the t-test, use the 1-way ANOVA test anyway just to understand the concepts.

Using Excel

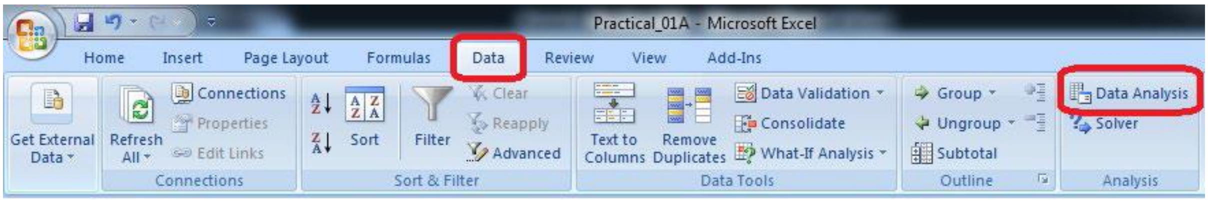
H0 - There are no significant differences between the Subject's mean SAT scores.

µ1 = µ2 = µ3 = µ4 = µ5

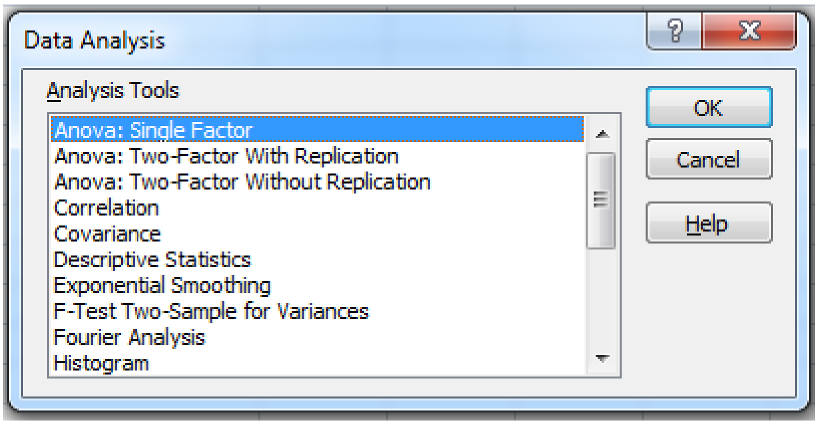
H1 - There is a significant difference between the Subject's mean SAT scores.

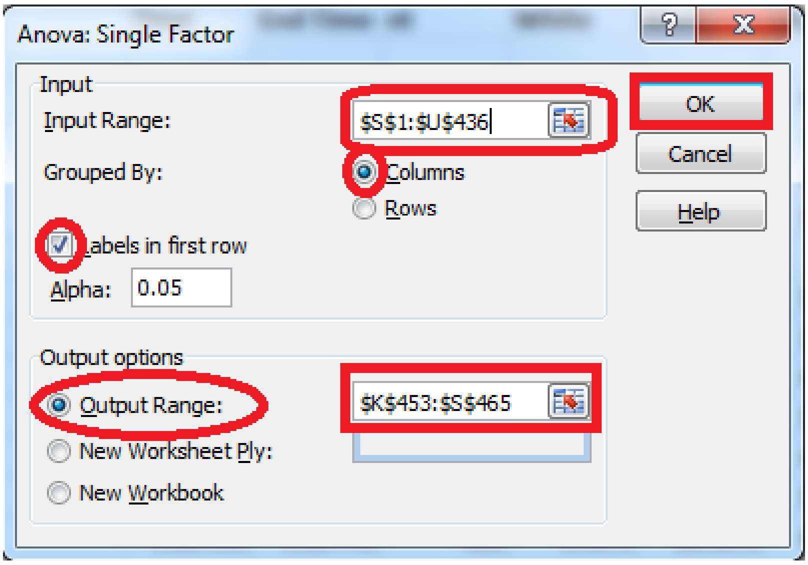
If there is at least one group with a significant difference with another group, the null hypothesis will be rejected.

To perform ANOVA go to data □Data Analysis



Input Range : $S$1:$U$436*(Select columns to be analyzed in group)*
Output Range :$K$453:$S$465



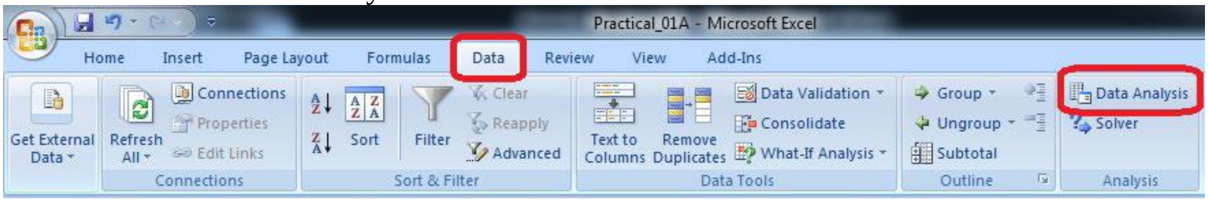


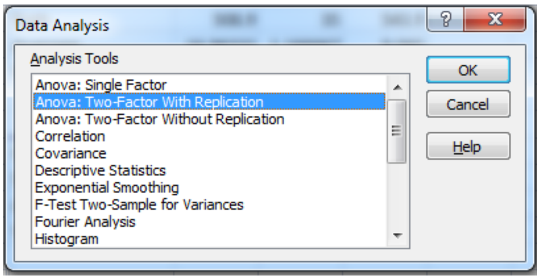
Anova: Single Factor						
SUMMARY						
Groups	*Count*	*Sum*	*Average*	*Variance*		
Average Score (SAT Math)	375	162354	432.944	5177.144		
Average Score (SAT Reading)	375	159189	424.504	3829.267		
Average Score (SAT Writing)	375	156922	418.4587	4166.522		
ANOVA						
Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	39700.57	2	19850.28	4.520698	0.01108	3.003745
Within Groups	4926677	1122	4390.977			
Total	4966377	1124				

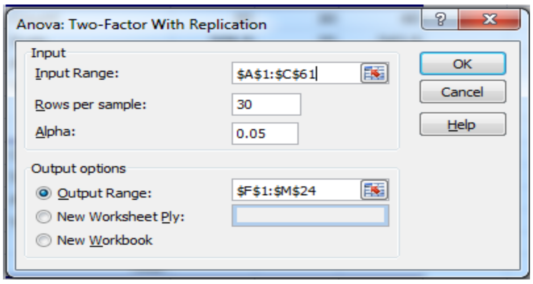
Since the resulting p-value is less than 0.05. The null hypothesis (H0) is rejected and conclude that there is a significant difference between the SAT scores for each subject.

Practical 12

Aim: Perform testing of hypothesis using Two-way ANOVA.
Using Excel:
Go to Data tab -> Data Analysis







Input Range - $A$1:$C$61(select values along with column name, only numeric columns)
ToothGrowth.csv
Rows Per Sample – 30 (Beacause 30 Patients are given each dose)
Alpha – 0.05
Output Range - $F$1:$M$24
Output:

Anova: Two-Factor With Replication						
SUMMARY	len	dose	Total			
1						
Count	30	30	60			
Sum	508.9	35	543.9			
Average	16.96333	1.166667	9.065			
Variance	68.32723	0.402299	97.22333			
31						
Count	30	30	60			
Sum	619.9	35	654.9			
Average	20.66333	1.166667	10.915			
Variance	43.63344	0.402299	118.2854			
Total						
Count	60	60				
Sum	1128.8	70				
Average	18.81333	1.166667				
Variance	58.51202	0.39548				
ANOVA						
Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Sample	102.675	1	102.675	3.642079	0.058808	3.922879
Columns	9342.145	1	9342.145	331.3838	8.55E-36	3.922879
Interaction	102.675	1	102.675	3.642079	0.058808	3.922879
Within	3270.193	116	28.19132			
Total	12817.69	119				

P-value = 0.0588079 column in the ANOVA Source of Variation table at the bottom of the output. The p-values for both medicine dose and interaction are greater than significance level (0.05), these factors are statistically significant.

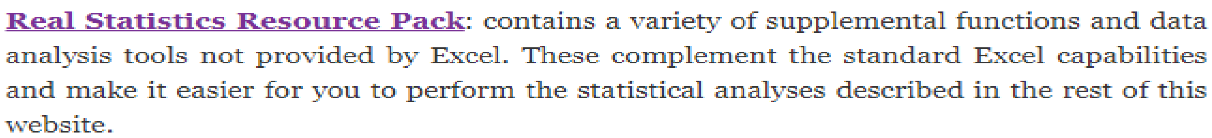
Practical 13

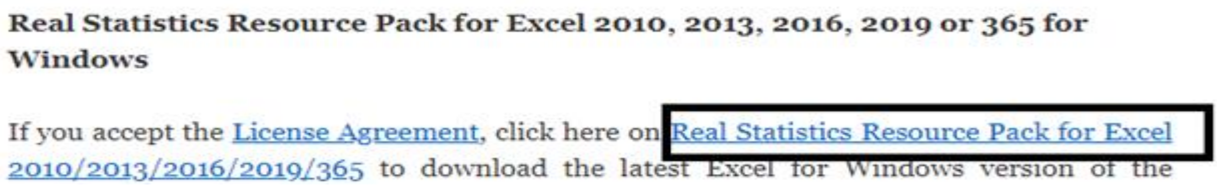
Aim: Perform testing of hypothesis using MANOVA.

MANOVA is the acronym for Multivariate Analysis of Variance. When analyzing data, we may encounter situations where we have there multiple response variables (dependent variables). In MANOVA there also some assumptions, like ANOVA. Before performing MANOVA we have to check the following assumptions are satisfied or not.
• The samples, while drawing, should be independent of each other.
• The dependent variables are continuous in nature and the independent variables are categorical.
• The dependent variables should follow a multivariate normal distribution.
• The population variance-covariance matrices of each group are same, i.e. groups are homogeneous.

Go to http://www.real-statistics.com/free-download/

1. Download Real Statistics Resource Pack


Real Statistics Resource Pack: contains a variety of supplemental functions and data analysis tools not provided by Excel. These complement the standard Excel capabilities and make it easier for you to perform the statistical analyses described in the rest of this website.


Real Statistics Resource Pack for Excel 2010, 2013, 2016, 2019 or 365 for Windows

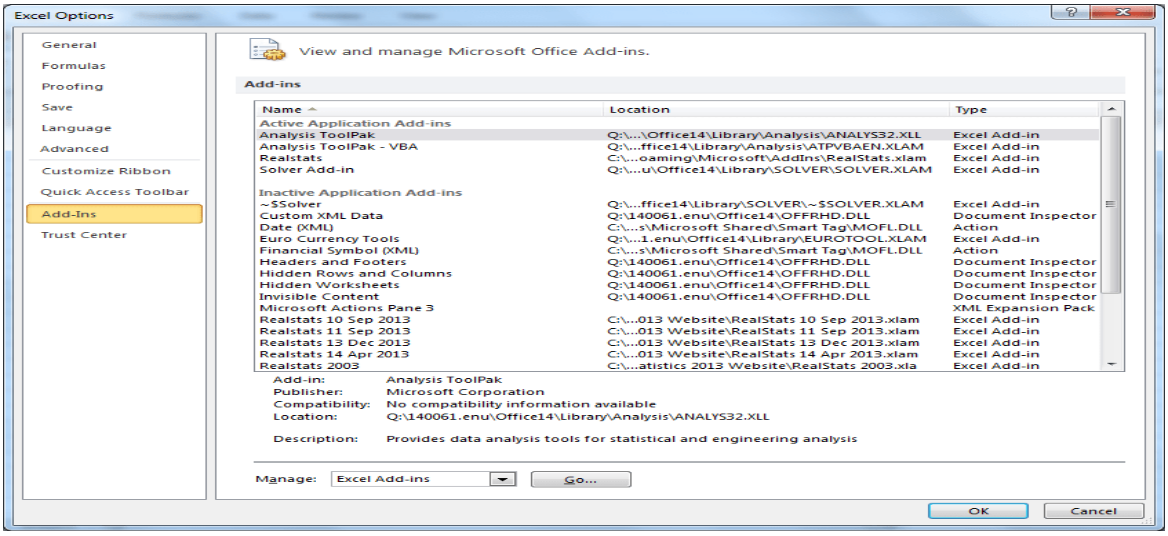
If you accept the License Agreement, click here on Real Statistics Resource Pack for Excel 2010/2013/2016/2019/365 to download the latest Excel for Windows version of the

Or

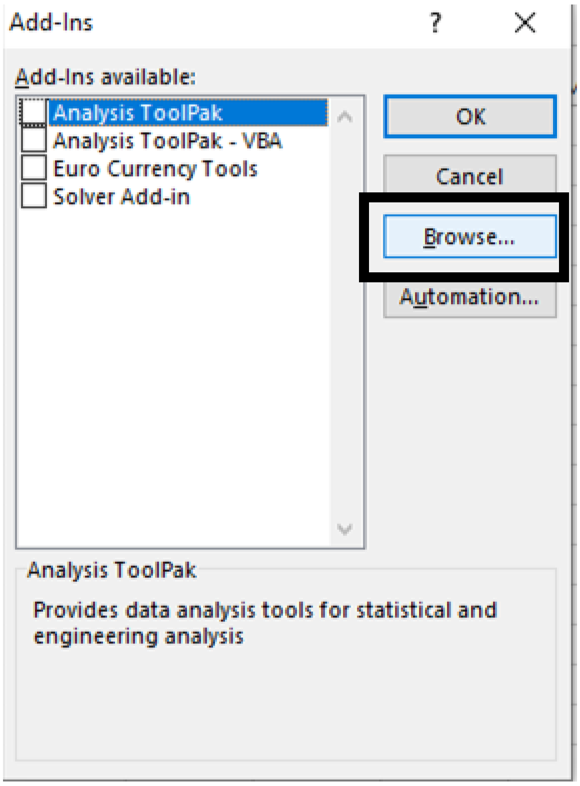
http://www.real-statistics.com/wp-content/uploads/2019/11/XRealStats.xlam

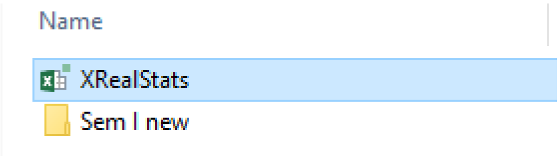
Install Add-in in excel. Select **File** > **Help|Options** > **Add-Ins** and click on the **Go** button at the bottom of the window (see Figure 1).

Add-ins -> Analysis Pack -> Go

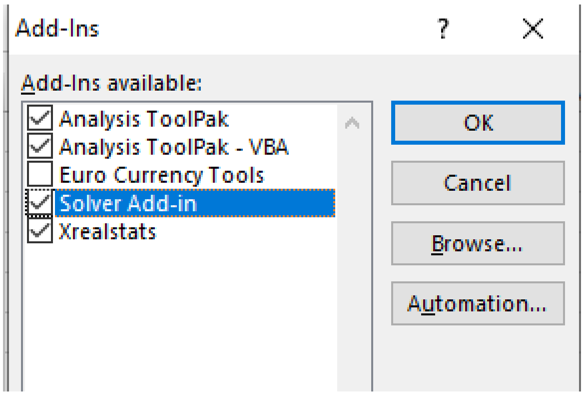


Click on browse and select XrealStats file (previously downloaded).





Select the following Add-Ins. Click OK.

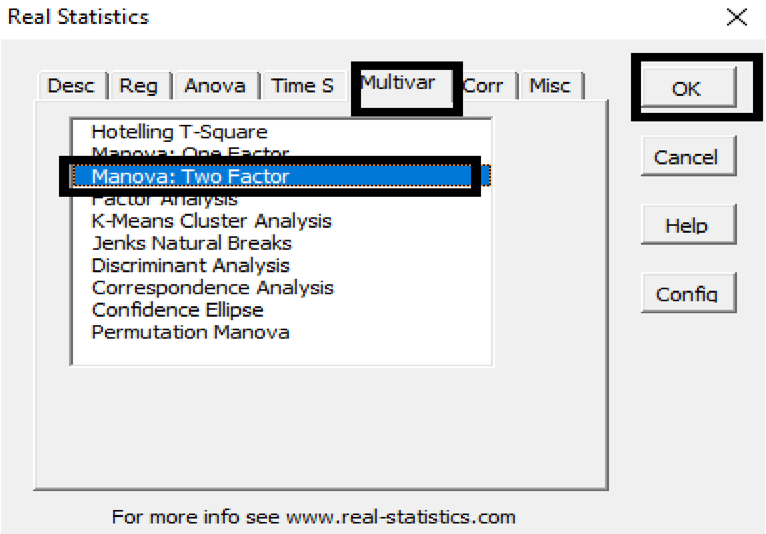


Now create an excel sheet with following data.

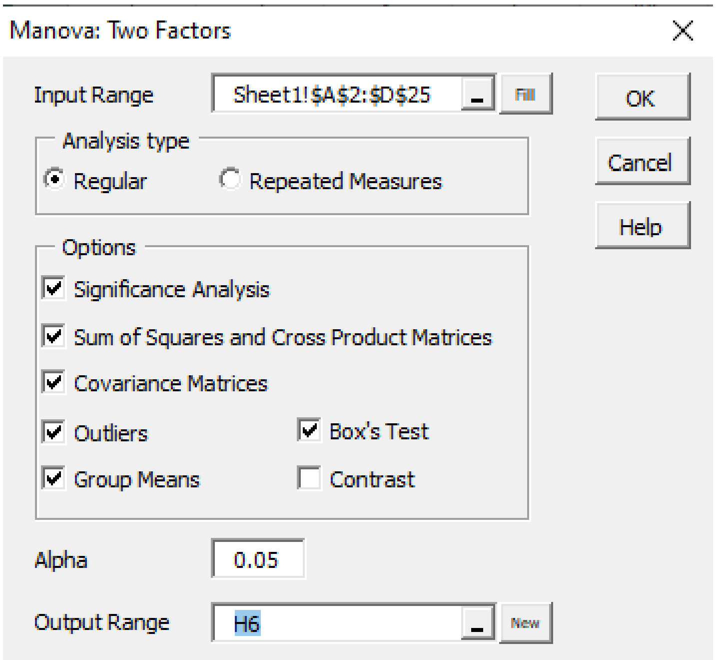
A study was conducted to see the impact of social-economic class (rich, middle, poor) and gender (male, female) on kindness and optimism using on a sample of 24 people based on the data in Figure 1.

	A	B	C	D
3	gender	economic	kindness	optimism
4	male	wealthy	5	3
5	male	wealthy	4	6
6	male	wealthy	3	4
7	male	wealthy	2	4
8	male	middle	4	6
9	male	middle	3	6
10	male	middle	5	4
11	male	middle	5	5
12	male	poor	7	5
13	male	poor	4	3
14	male	poor	3	1
15	male	poor	7	2
16	female	wealthy	2	3
17	female	wealthy	3	5
18	female	wealthy	5	3
19	female	wealthy	4	2
20	female	middle	9	8
21	female	middle	6	5
22	female	middle	7	6
23	female	middle	8	9
24	female	poor	8	9
25	female	poor	9	8
26	female	poor	3	7
27	female	poor	5	7

Press ctrl-m to open Real Statistics menu.



Select the data excluding column names. Select a cell for output.



Output:

Two-Way MANOVA						
fact A	*stat*	*df1*	*df2*	*F*	*p-value*	*part eta-sq*
Pillai Trac	0.190764	2	16	1.885866	0.183909	0.190764
Wilk's Lam	0.809236	2	16	1.885866	0.183909	0.190764
Hotelling	0.235733	2	16	1.885866	0.183909	0.190764
Roy's Lg R	0.235733					
fact B	*stat*	*df1*	*df2*	*F*	*p-value*	*part eta-sq*
Pillai Trac	0.340249	4	34	1.742501	0.163458	0.170125
Wilk's Lam	0.8181	4	32	1.778757	0.157443	0.1819
Hotelling	0.479878	4	30	1.799541	0.155008	0.193509
Roy's Lg R	0.448078					

SSCP Matrices	
Tot	
104.9565	59.86957
59.86957	110.6087
Row (A)	
12.5247	15.41502
15.41502	18.97233
Column (B)	
31.15295	22.95885
22.95885	19.37655

SSCP Matrices			Group Covariance Matrices	
Tot			female	middle
104.96	59.87		1.6667	2
59.87	110.61		2	3.3333
Row (A)			female	poor
12.525	15.415		7.5833	2.0833
15.415	18.972		2.0833	0.9167
Column (B)			female	wealthy
31.153	22.959		1.6667	-0.5
22.959	19.377		-0.5	1.5833
Interaction (AB)			male	middle
11.029	4.7457		0.9167	-0.75
4.7457	40.593		-0.75	0.9167
Res			male	poor
50.25	16.75		4.25	2.0833
16.75	31.667		2.0833	2.9167
			male	wealthy
			1	1
			1	1.3333

Covariance Matrix	
4.7708	2.7213
2.7213	5.0277
Inverse of Covariance Matrix	
0.3032	-0.164
-0.164	0.2877
Mean vector	
5	5

Practical 14

Aim: Perform the Random sampling for the given data and analyse it.

Example 1: From a population of 10 women and 10 men as given in the table in Figure 1 on the left below, create a random sample of 6 people for Group 1 and a periodic sample consisting of every 3rd woman for Group 2.
You need to run the sampling data analysis tool twice, once to create Group 1 and again to create Group 2. For Group 1 you select all 20 population cells as the Input Range and Random as the Sampling Method with 6 for the Random Number of Samples. For Group 2 you select the 10 cells in the Women column as Input Range and Periodic with Period 3.
Open existing excel sheet with population data
Sample Sheet looks as given below:

	A	B	C	D	E	F	G	H	I	J	K
1	Sr. No	Roll No	Student's Name	Gender	Grade		Sr. No	Roll No	Student's Name	Gender	Grade
2	1	1	Gaborone	m	O		62	3	Maun	f	O
3	2	2	Francistown	m	O		63	7	Tete	f	O
4	3	5	Niamey	m	O		64	9	Chimoio	f	O
5	4	13	Maxixe	m	O		65	11	Pemba	f	O
6	5	16	Tema	m	O		66	14	Chibuto	f	O
7	6	17	Kumasi	m	O		67	25	Mampong	f	O
8	7	34	Blida	m	O		68	36	Tlemcen	f	O
9	8	35	Oran	m	O		69	40	Adrar	f	O
10	9	38	Saefda	m	O		70	41	Tindouf	f	O
11	10	42	Constantine	m	O		71	46	Skikda	f	O
12	11	43	Annaba	m	O		72	47	Ouargla	f	O
13	12	45	Bejaefa	m	O		73	10	Matola	f	D
14	13	48	Medea	m	O		74	20	Legon	f	D
15	14	49	Djelfa	m	O		75	21	Sunyani	f	D
16	15	50	Tipaza	m	O		76	72	Teenas	f	D
17	16	51	Bechar	m	O		77	73	Kouba	f	D
18	17	54	Mostaganem	m	O		78	75	Hussen Dey	f	D
19	18	55	Tiaret	m	O		79	77	Khenchela	f	D
20	19	56	Bouira	m	O		80	82	Hassi Bahbah	f	D
21	20	59	Tebessa	m	O		81	84	Baraki	f	D
22	21	61	El Harrach	m	O		82	91	Boudouaou	f	D
23	22	62	Mila	m	O		83	95	Tadjenanet	f	D
24	23	65	Fouka	m	O		84	4	Molepolole	f	C

Set Cell O1 = Male and Cell O2 = Female
To generate a random sample for male students from given population go to Cell O1 and type =INDEX(E$2:E$62,RANK(B2,B$2:B$62))
Drag the formula to the desired no of cell to select random sample.
Now, to generate a random sample for female students go to cell P1 and type =INDEX(K$2:K$40,RANK(H2,H$2:H$40))
Drag teh formula to the desired no of cell to select random sample.
Output:

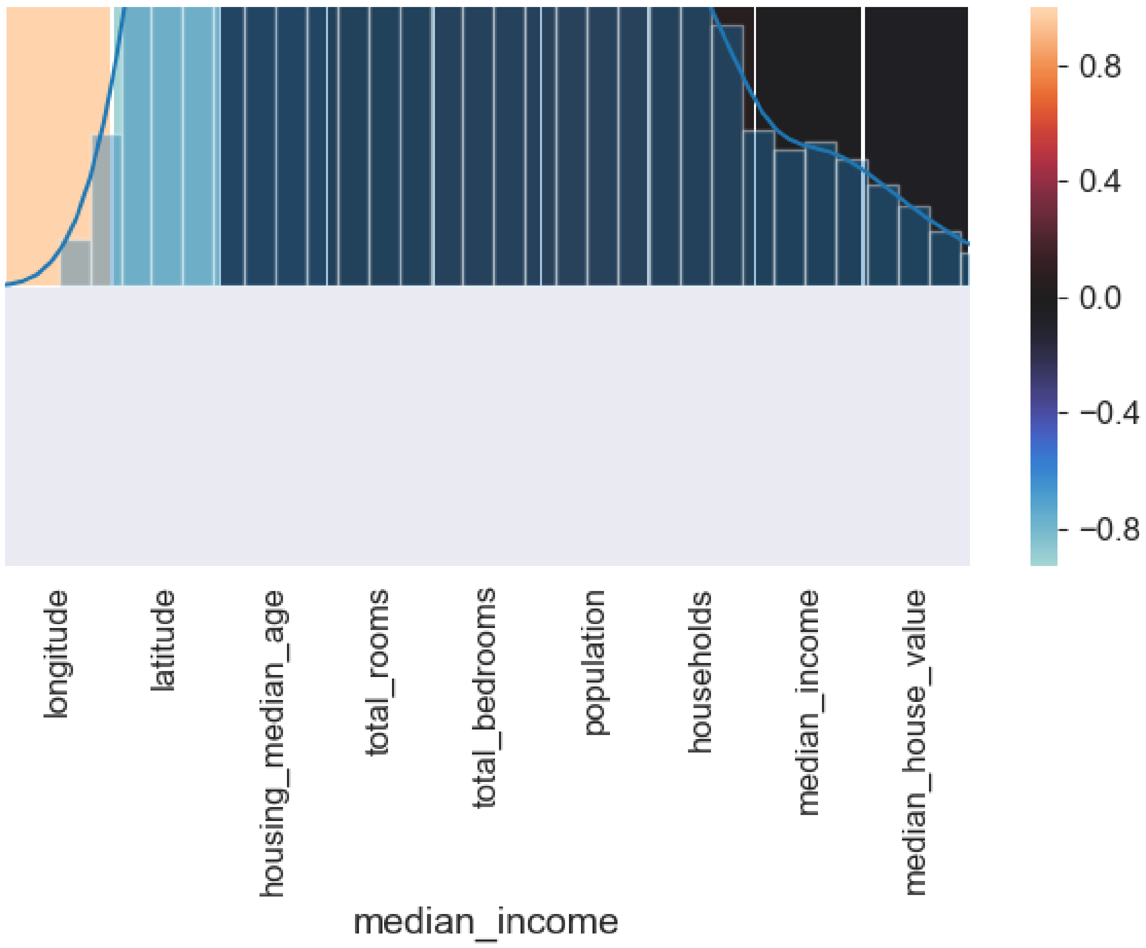
O	P
Male	**Female**
A	A
A	A
A	A
B	A
C	B
C	C
D	C
D	C
D	C
D	C
D	D
D	A
D	B
D	B
O	D
O	D
O	D
O	D
O	O
O	O
O	O
O	O
O	A

Practical 15

Aim: Perform the Stratified sampling for the given data and analyse it.

We are to carry out a **hypothetical** housing quality survey across Lagos state, Nigeria. And we looking at a total of 5000 houses (hypothetically). We don’t just go to one local government and select 5000 houses, rather we ensure that the 5000 houses are a representative of the whole 20 local government areas Lagos state is comprised of. This is called stratified sampling. The population is divided into homogenous strata and the right number of instances is sampled from each stratum to guarantee that the test-set (which in this case is the 5000 houses) is a representative of the overall population. If we used random sampling, there would be a significant chance of having bias in the survey results.

```
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
plt.rcParams['axes.labelsize'] = 14
plt.rcParams['xtick.labelsize'] = 12
plt.rcParams['ytick.labelsize'] = 12
import seaborn as sns
color = sns.color_palette()
sns.set_style('darkgrid')
housing =pd.read_csv('housing.csv')
print(housing.head())
print(housing.info())
#creating a heatmap of the attributes in the dataset
correlation_matrix = housing.corr()
plt.subplots(figsize=(8,6))
sns.heatmap(correlation_matrix, center=0, annot=True, linewidths=.3)
corr =housing.corr()
print(corr['median_house_value'].sort_values(ascending=False))
sns.distplot(housing.median_income)
plt.show()
```


0.8
0.4
0.0
−0.4
−0.8
longitude
latitude
housing_median_age
total_rooms
total_bedrooms
population
households
median_income
median_house_value
median_income

```
================== RESTART: A:/RIC/programs/RICSampling.py =========
   longitude  latitude  ...  median_house_value  ocean_proximity
0    -122.23     37.88  ...            452600.0         NEAR BAY
1    -122.22     37.86  ...            358500.0         NEAR BAY
2    -122.24     37.85  ...            352100.0         NEAR BAY
3    -122.25     37.85  ...            341300.0         NEAR BAY
4    -122.25     37.85  ...            342200.0         NEAR BAY

[5 rows x 10 columns]
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
longitude             20640 non-null float64
latitude              20640 non-null float64
housing_median_age    20640 non-null float64
total_rooms           20640 non-null float64
total_bedrooms        20433 non-null float64
population            20640 non-null float64
households            20640 non-null float64
median_income         20640 non-null float64
median_house_value    20640 non-null float64
ocean_proximity       20640 non-null object
dtypes: float64(9), object(1)
memory usage: 1.6+ MB
None
median_house_value    1.000000
median_income         0.688075
total_rooms           0.134153
housing_median_age    0.105623
households            0.065843
total_bedrooms        0.049686
population           -0.024650
longitude            -0.045967
latitude             -0.144160
Name: median_house_value, dtype: float64
```

Practical 16

Aim: Write a program for computing different correlation.

Correlation is usually defined as a measure of the linear relationship between two quantitative variables.

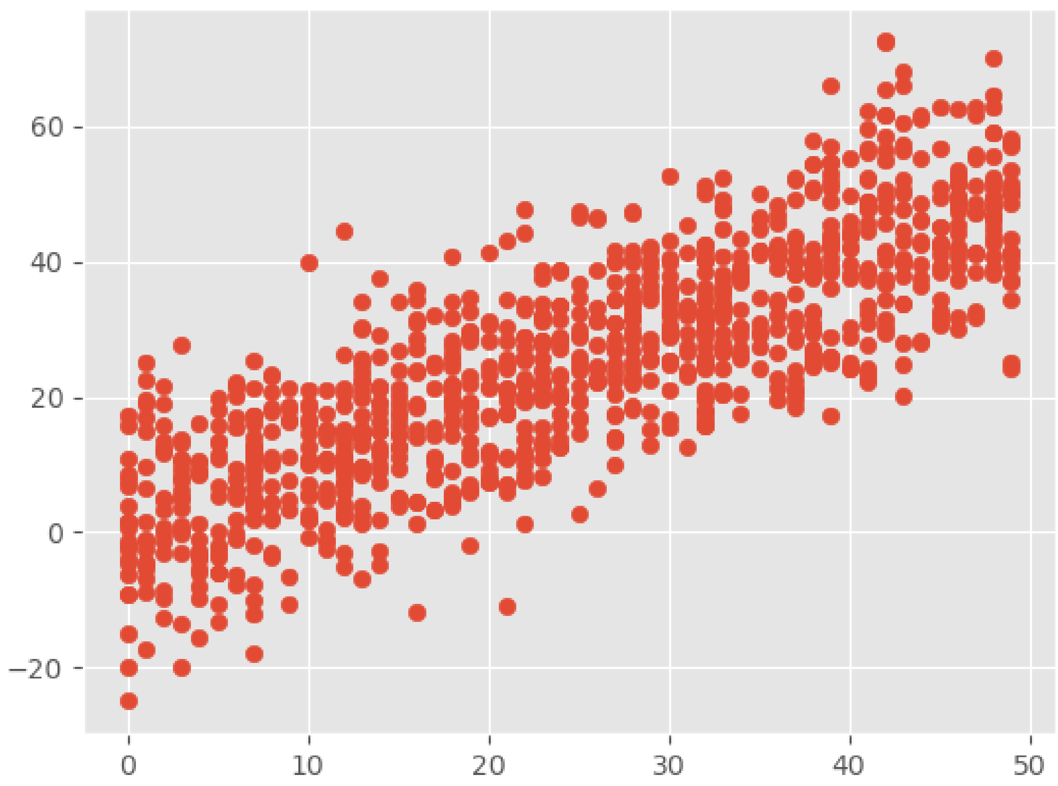
When the values of one variable increase as the values of the other increase, this is known as positive correlation.

When the values of one variable decrease as the values of another increase to form an inverse relationship, this is known as negative correlation.

A weak correlation is one where on average the values of one variable are related to the other, but there are many exceptions.

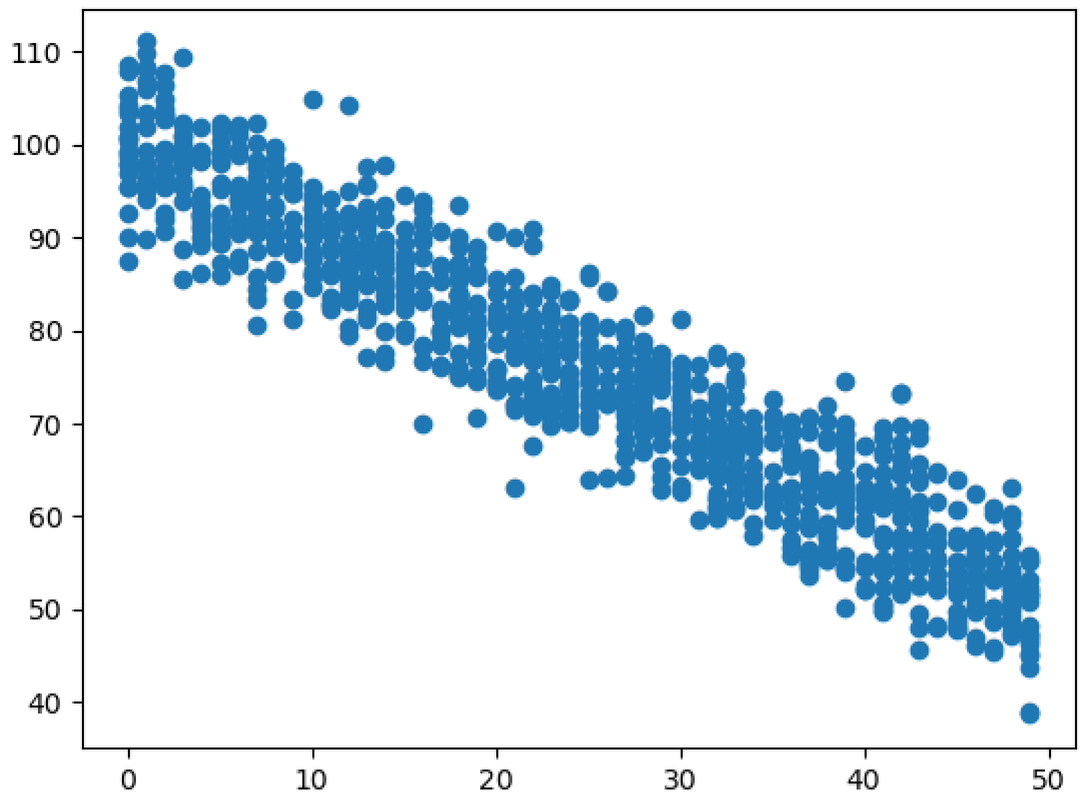
Positive Correlation:

```
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(1)
# 1000 random integers between 0 and 50
x = np.random.randint(0, 50, 1000)
# Positive Correlation with some noise
y = x + np.random.normal(0, 10, 1000)
np.corrcoef(x, y)
plt.style.use('ggplot')
plt.scatter(x, y)
plt.show()
```



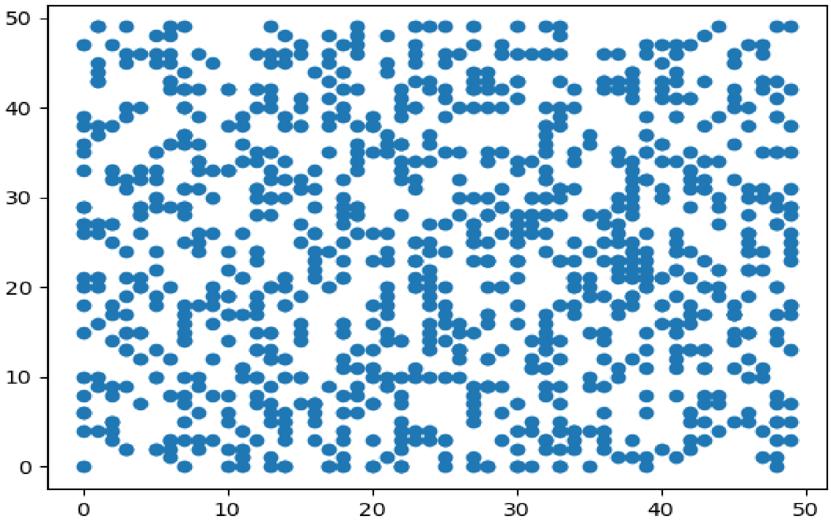
Negative Correlation:

```
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(1)
# 1000 random integers between 0 and 50
x = np.random.randint(0, 50, 1000)
# Negative Correlation with some noise
y = 100 - x + np.random.normal(0, 5, 1000)
np.corrcoef(x, y)
plt.scatter(x, y)
plt.show()
```



No/Weak Correlation:

```
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(1)
x = np.random.randint(0, 50, 1000)
y = np.random.randint(0, 50, 1000)
np.corrcoef(x, y)
plt.scatter(x, y)
plt.show()
```



Practical 17

Aim: Perform Linear regression for prediction.

Step 1: Import libraries and dataset.

Import the important libraries and the dataset we are using to perform Polynomial Regression.

Step 2: Dividing the dataset into 2 components.
Divide dataset into two components that is X and y. X will contain the Column between 1 and 2. y will contain the 2 column.

Step 3: Fitting Linear Regression to the dataset
Fitting the linear Regression model On two components.

Step 4: Fitting Polynomial Regression to the dataset
Fitting the Polynomial Regression model on two components X and y.

Step 5: In this step we are Visualising the Linear Regression results using scatter plot.

	A	B	C
1	sno	Temperat	Pressure
2	1	0	0.0002
3	2	20	0.0012
4	3	40	0.006
5	4	60	0.03
6	5	80	0.09
7	6	100	0.27

data.csv

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

 # Step 1 :Import libraries and dataset

datas = pd.read_csv('data.csv')

print(datas)

#Step 2: Dividing the dataset into 2 components

X = datas.iloc[:, 1:2].values

y = datas.iloc[:, 2].values

#Step 3: Fitting Linear Regression to the dataset

from sklearn.linear_model import LinearRegression

lin = LinearRegression()

lin.fit(X, y)

 # Step 4: Fitting Polynomial Regression to the dataset

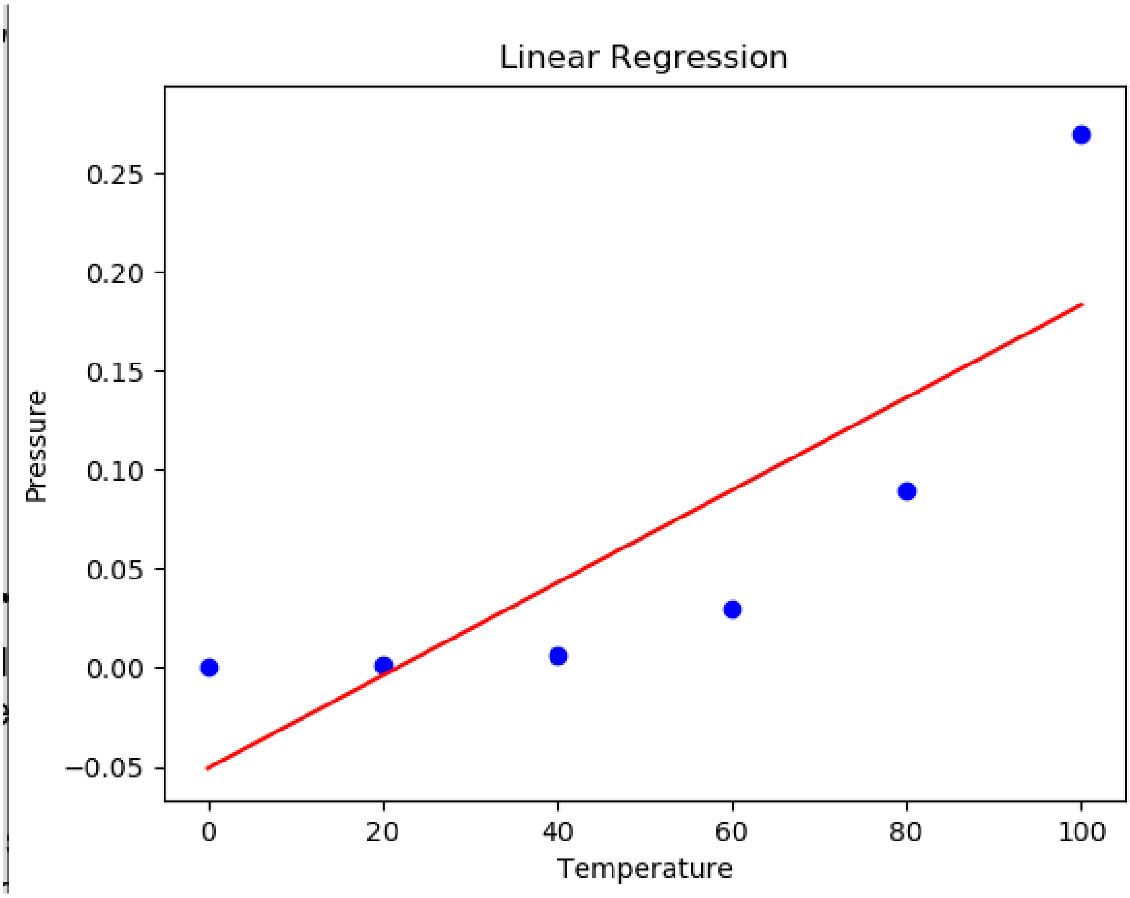
```
from sklearn.preprocessing import PolynomialFeatures
poly = PolynomialFeatures(degree = 4)
X_poly = poly.fit_transform(X)
poly.fit(X_poly, y)
lin2 = LinearRegression()
lin2.fit(X_poly, y)
```

 # Step 5: Visualising the Linear Regression results

```
plt.scatter(X, y, color = 'blue')
```

plt.plot(X, lin.predict(X), color = 'red')

```
plt.title('Linear Regression')
plt.xlabel('Temperature')
plt.ylabel('Pressure')
plt.show()
```



Practical 18

Aim: Perform Polynomial Regression for prediction.

Step 1: Import libraries and dataset
Import the important libraries and the dataset we are using to perform Polynomial Regression.

Step 2: Dividing the dataset into 2 components
Divide dataset into two components that is X and y.X will contain the Column between 1 and 2. y will contain the 2 column.

Step 3: Fitting Linear Regression to the dataset
Fitting the linear Regression model On two components.

Step 4: Fitting Polynomial Regression to the dataset
Fitting the Polynomial Regression model on two components X and y.

Step 5: Visualising the Polynomial Regression results using scatter plot.

```
import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

# Step 1 :Import libraries and dataset

datas = pd.read_csv('data.csv')

print(datas )

#Step 2: Dividing the dataset into 2 components

X = datas.iloc[:, 1:2].values

y = datas.iloc[:, 2].values

#Step 3: Fitting Linear Regression to the dataset

from sklearn.linear_model import LinearRegression

lin = LinearRegression()

lin.fit(X, y)

# Step 4: Fitting Polynomial Regression to the dataset

from sklearn.preprocessing import PolynomialFeatures

poly = PolynomialFeatures(degree = 4)

X_poly = poly.fit_transform(X)

poly.fit(X_poly, y)

lin2 = LinearRegression()

lin2.fit(X_poly, y)
```

 # Step 5: Visualising the Linear Regression results

```
# Visualising the Polynomial Regression results
plt.scatter(X, y, color = 'blue')
plt.plot(X, lin2.predict(poly.fit_transform(X)), color = 'red')
plt.title('Polynomial Regression')
plt.xlabel('Temperature')
plt.ylabel('Pressure')
plt.show()

plt.show()
```