

# STATISTICAL AND VISUALIZATION ANALYSIS

FOR MINIMUM DATA

## Abstract

This project aims into statistical and visualization analysis techniques using both R and Python for a minimal dataset comprising of a individuals internal and semester marks up to III semester. Statistical analysis plays a crucial role in decision-making processes, understanding data distributions and relationships. Through the integration of R and Python, this project demonstrates the visualization of minimal datasets. The findings showcase the significance of statistical analysis and visualization in extracting meaningful insights from limited data, thus highlighting their utility in data exploration and decision-making processes.

## Introduction

Statistical analysis involves the collection, organization, analysis, interpretation, and presentation of data. It includes descriptive statistics to summarize data and inferential statistics to make inferences and predictions from the data. Common statistical techniques include hypothesis testing, regression, coefficient of variation, correlation analysis, and analysis of variance. In this project we will see certain statistic analyses using R

- Coefficient of variation
- Coefficient of correlation
- Rank correlation
- Paired-t test

Also, This project explores analysing data through numpy and pandas by performing various tasks such as reading, cleaning and implementing descriptive analysis, data visualization using Python, showcasing techniques to create informative plots, charts, and graphs. By importing libraries such as Matplotlib, Seaborn, Plotly, and Bokeh.

These libraries allows users to create static, interactive, and animated visualizations to explore and communicate insights from their data. This project focuses on utilizing Matplotlib to conduct an data analysis for following charts

- Line
- Bar
- Pie

### **Statistic analyses using R**

**Data:** The data collected for this project is marks secured by an individual up to III semester which is in csv file. The below link gives you the data

<C:\Users\kumar\OneDrive\Documents\MARKS.csv>

- **Coefficient of variation**

The coefficient of variation (CV) is a statistical measure used to quantify the relative variability of a dataset compared to its mean. It is calculated as the ratio of the standard deviation to the mean, expressed as a percentage.

Mathematically, the coefficient of variation (CV) is defined as:

$$c \cdot v = \left( \frac{SD}{M} \right) \times 100\%$$

The coefficient of variation allows for the comparison of variability between datasets with different units or scales. A lower CV indicates less variability relative to the mean, while a higher CV suggests greater variability relative to the mean. This makes the coefficient of variation a useful tool in comparing the dispersion of datasets with different means or units of measurement.

## PROBLEM

The Marks secured by an Individual in internal and semester are as follows.

INTERNAL	86	60	68	76	86	82	60	96	86	88	84	62	96	86	92
SEMESTER	73	63	71	81	82	70	68	97	76	84	80	60	94	80	91

Find which marks is more consistent in scoring.

### INTERNALS

X= 86,60,68,76,86,82,60,96,86,88,84,62,96,86,92

$$c \cdot v = \left( \frac{SD}{M} \right) \times 100\%$$

Calculated SD= 12.4319

Calculated Mean= 80.5333

$$Cv=(12.4319/80.5333) \times 100\%$$

$$cv = 15.436980462813424 \%$$

### SEMESTER

Y= 73,63,71,81,82,70,68,97,76,84,80,60,94,80,91

$$c \cdot v = \left( \frac{SD}{M} \right) \times 100\%$$

Calculated SD= 10.843

Calculated Mean= 78

$$Cv=( 10.843/78) \times 100\%$$

$$cv = 13.901328138568445\%$$

INTERNAL MARKS ARE MORE CONSISTENT THAN SEMESTER.

## Using R

The Marks secured by an Individual in internal and semester are as follows.

INTERNAL	86	60	68	76	86	82	60	96	86	88	84	62	96	86	92
SEMESTER	73	63	71	81	82	70	68	97	76	84	80	60	94	80	91

Find which marks is more consistent in scoring.

### CODE:

```
x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
```

```
mean(x)
```

```
sd(x)
```

```
cv=((sd(x)/mean(x))*100)
```

```
cv
```

```
y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
```

```
mean(y)
```

```
sd(y)
```

```
cv=((sd(y)/mean(y))*100)
```

```
cv
```

### RESULT:

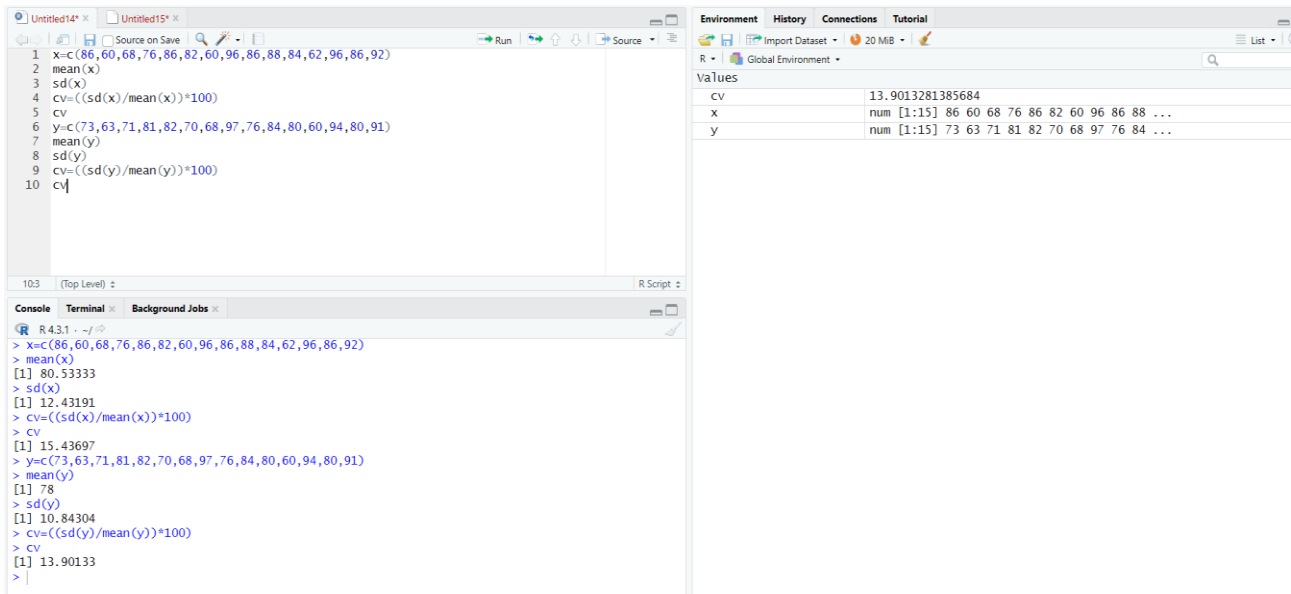
For dataset x (internal marks), the CV is approximately 15.43697%.

For dataset y (semester marks), the CV is approximately 13.90133%.

### CONCLUSION:

Comparing the coefficients of variation, we observe that the internal marks dataset (x) has a lower CV than the semester marks dataset (y). Based on these results, we can conclude that the internal marks is relatively more consistent compared to the semester marks.

## OUTPUT:



The screenshot displays the RStudio environment. The script editor on the left contains the following R code:

```
1 x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
2 mean(x)
3 sd(x)
4 cv=((sd(x)/mean(x))*100)
5 cv
6 y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
7 mean(y)
8 sd(y)
9 cv=((sd(y)/mean(y))*100)
10 cv
```

The console on the bottom left shows the execution of these commands:

```
> x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
> mean(x)
[1] 80.53333
> sd(x)
[1] 12.43191
> cv=((sd(x)/mean(x))*100)
> cv
[1] 15.43697
> y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
> mean(y)
[1] 78
> sd(y)
[1] 10.84304
> cv=((sd(y)/mean(y))*100)
> cv
[1] 13.90133
>
```

The Environment pane on the right shows the following values:

Variable	Value
cv	13.9013281385684
x	num [1:15] 86 60 68 76 86 82 60 96 86 88 ...
y	num [1:15] 73 63 71 81 82 70 68 97 76 84 ...

- **Coefficient of correlation**

The coefficient of correlation, often denoted as  $r$ , is a statistical measure that quantifies the strength and direction of the linear relationship between two variables. It ranges from -1 to 1, where:

- $r = 1$ :

Perfect positive correlation, indicating that as one variable increases, the other variable also increases proportionally.

- $r = 0$ :

No correlation, indicating that there is no linear relationship between the two variables.

- $r = -1$ :

Perfect negative correlation, indicating that as one variable increases, the other variable decreases proportionally.

The coefficient of correlation is calculated using the formula for Pearson correlation coefficient,

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2} \sqrt{\sum(y_i - \bar{y})^2}}$$

Where,

$x_i$  and  $y_i$  are individual data points.

$\bar{x}$  and  $\bar{y}$  are mean of variables  $x$  and  $y$ .

### PROBLEM:

The Marks secured by an Individual in internal and semester are as follows.

INTERNAL	86	60	68	76	86	82	60	96	86	88	84	62	96	86	92
SEMESTER	73	63	71	81	82	70	68	97	76	84	80	60	94	80	91

Find which marks is more consistent in scoring.

$X - M_x$	$Y - M_y$	$(X - M_x)^2$	$(Y - M_y)^2$	$(X - M_x)(Y - M_y)$
5.467	-5.000	29.884	25.000	-27.333
-20.533	-15.000	421.618	225.000	308.000
-12.533	-7.000	157.084	49.000	87.733
-4.533	3.000	20.551	9.000	-13.600
5.467	4.000	29.884	16.000	21.867
1.467	-8.000	2.151	64.000	-11.733
-20.533	-10.000	421.618	100.000	205.333
15.467	19.000	239.218	361.000	293.867
5.467	-2.000	29.884	4.000	-10.933
7.467	6.000	55.751	36.000	44.800
3.467	2.000	12.018	4.000	6.933
-18.533	-18.000	343.484	324.000	333.600
15.467	16.000	239.218	256.000	247.467
5.467	2.000	29.884	4.000	10.933
11.467	13.000	131.484	169.000	149.067
<b>80.533</b>	<b>78.000</b>	<b>2163.733</b>	<b>1646.000</b>	<b>1646.000</b>

## Key

X: X Values

Y: Y Values

$M_x$ : Mean of X Values

$M_y$ : Mean of Y Values

$x - M_x$  &  $y - M_y$ : Deviation scores

$(x - M_x)^2, (y - M_y)^2$ : Deviation Squared

$(x - M_x)(y - M_y)$  Product of Deviation Scores

## Calculation

X Values

$$\Sigma = 1208$$

$$\text{Mean} = 80.533$$

$$\Sigma(x - M_x)^2 = 2163.733$$

Y Values

$$\Sigma = 1170$$

$$\text{Mean} = 78$$

$$\Sigma(y - M_y)^2 = 1646$$

X and Y Combined

$$N = 15$$

$$\Sigma(x - M_x)(y - M_y) = 1646$$

R Calculation

$$r = \frac{\Sigma(x - M_x)(y - M_y)}{\sqrt{\Sigma(x - M_x)^2} \sqrt{\Sigma(y - M_y)^2}}$$

$$r = 1646 / \sqrt{(2163.733)(1646)} = 0.8722$$

$$r = 0.8722$$

## Using R

Calculate the coefficient of correlation of the following data

INTERNAL	86	60	68	76	86	82	60	96	86	88	84	62	96	86	92
SEMESTER	73	63	71	81	82	70	68	97	76	84	80	60	94	80	91

## CODE:

```
x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
```

```
y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
```

```
cor(x,y,method="pearson")
```

## RESULT:

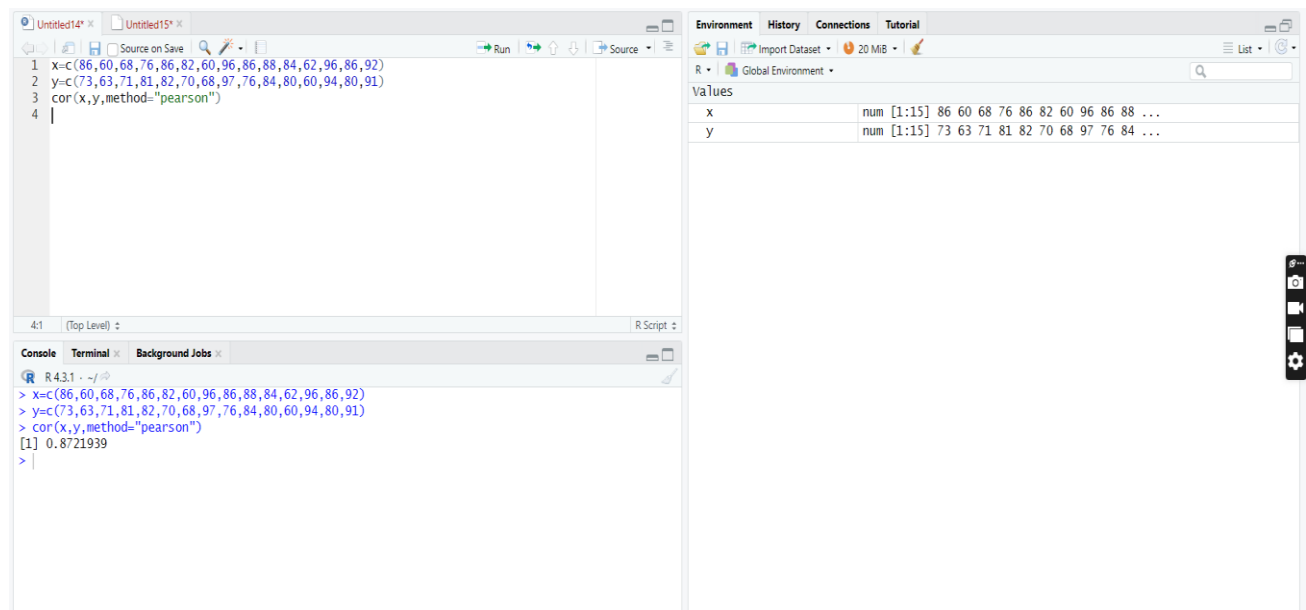
The Pearson correlation coefficient for the given data is approximately 0.8721939

## CONCLUSION:

This is a strong positive correlation, which means that high X variable scores go with high Y variable scores.



## OUTPUT:



The screenshot shows the RStudio interface. The script editor on the left contains the following R code:

```
1 x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
2 y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
3 cor(x,y,method="pearson")
4
```

The console at the bottom shows the execution of the code:

```
> x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
> y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
> cor(x,y,method="pearson")
[1] 0.8721939
>
```

The Environment pane on the right shows the variables x and y as numeric vectors of length 15.

Variable	Value
x	num [1:15] 86 60 68 76 86 82 60 96 86 88 ...
y	num [1:15] 73 63 71 81 82 70 68 97 76 84 ...

- **Rank correlation**

The Spearman's coefficient correlation, represented by  $\rho$  or by  $r_R$ , is a nonparametric measure of the strength and direction of the association that exists between two ranked variables. It determines the degree to which a relationship is monotonic, i.e., whether there is a monotonic component of the association between two continuous or ordered variables.

Formula of Spearman's Rank Correlation Coefficient

$$\rho = 1 - \frac{6\sum d^2}{n(n^2 - 1)}$$

$\rho$ : the strength of the rank correlation between variables

$\sum d^2$ : sum of the squared differences between x and y variable ranks

$n$ : sample size

## PROBLEM:

Calculate the Rank correlation of the following data

INTERNAL	86	60	68	76	86	82	60	96	86	88	84	62	96	86	92
SEMESTER	73	63	71	81	82	70	68	97	76	84	80	60	94	80	91

<i>x</i>	<i>y</i>	<i>R<sub>x</sub></i>	<i>R<sub>y</sub></i>	<i>d=R<sub>x</sub>-R<sub>y</sub></i>	<i>d<sup>2</sup></i>
86	73	6.5	10	-3.5	12.25
60	63	14.5	14	0.5	0.25
68	71	12	11	1	1
76	81	11	6	5	25
86	82	6.5	5	1.5	2.25
82	70	10	12	-2	4
60	68	14.5	13	1.5	2.25
96	97	1.5	1	0.5	0.25
86	76	6.5	9	-2.5	6.25
88	84	4	4	0	0
84	80	9	7.5	1.5	2.25
62	60	13	15	-2	4
96	94	1.5	2	-0.5	0.25
86	80	6.5	7.5	-1	1
92	91	3	3	0	0

61

$$r = 1 - \frac{6 \cdot \left( \sum d^2 + \sum \frac{m(m^2-1)}{12} \right)}{n(n^2-1)}$$

$$= 1 - \frac{6 \cdot \left( 61 + \frac{2 \cdot (2^2-1)}{12} + \frac{4 \cdot (4^2-1)}{12} + \frac{2 \cdot (2^2-1)}{12} + \frac{2 \cdot (2^2-1)}{12} \right)}{15 \cdot (15^2-1)}$$

$$= 1 - \frac{6 \cdot (61 + 0.5 + 5 + 0.5 + 0.5)}{15 \cdot (225-1)}$$

$$= 1 - \frac{405}{3360}$$

$$= 1 - 0.120536$$

$$= 0.879464$$

## Using R

Calculate the Rank correlation of the following data

INTERNAL	86	60	68	76	86	82	60	96	86	88	84	62	96	86	92
SEMESTER	73	63	71	81	82	70	68	97	76	84	80	60	94	80	91

**CODE:**

```
x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
```

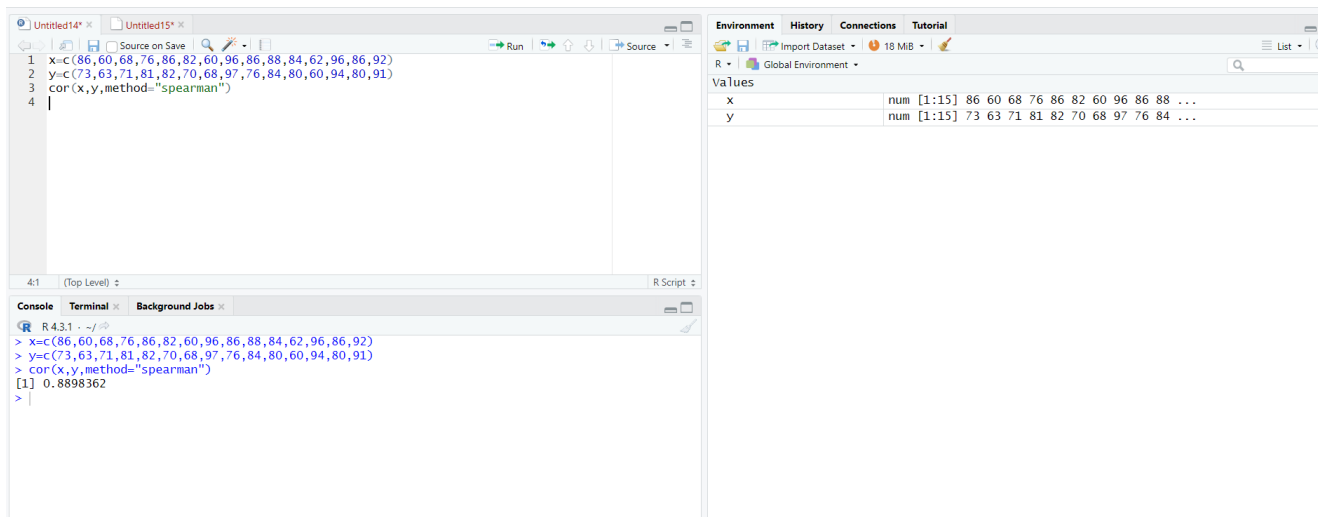
```
y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
```

```
cor(x,y,method="spearman")
```

**RESULT:**

The Spearman rank correlation coefficient for the given data is approximately 0.8898362.

**OUTPUT:**



The screenshot displays the R Studio environment. The script editor on the left contains the following code:

```
1 x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
2 y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
3 cor(x,y,method="spearman")
4 |
```

The Environment pane on the right shows the variables x and y, both of type 'num' (numeric) and length 15.

The Console pane at the bottom shows the execution output:

```
R 4.3.1 ~ /
> x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
> y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
> cor(x,y,method="spearman")
[1] 0.8898362
> |
```

- **Paired-t test**

A **paired samples t-test** is used to compare the means of two samples when each observation in one sample can be paired with an observation in the other sample.

**Define the hypotheses.**

We will perform the paired samples t-test with the following hypotheses:

- **H<sub>0</sub>:**  $\mu_1 = \mu_2$  (the two population means are equal)
- **H<sub>1</sub>:**  $\mu_1 \neq \mu_2$  (the two population means are not equal)

**Calculate the test statistic  $t$ .**

$$t = \frac{\bar{d}}{(\sigma/\sqrt{n})}$$

**Calculate the p-value of the test statistic  $t$ .**

According to the T Score to P Value Calculator, the p-value associated with  $t$  and degrees of freedom =  $n-1$ .

**Draw a conclusion.**

IF p-value is less than our significance level  $\alpha = 0.05$ , we reject the null hypothesis. IF p-value is greater than our significance level, we accept null hypothesis.

**PROBLEM:**

The Marks secured by an Individual in internal and semester are as follows. Test whether there is any improvement in semester marks than internal marks at 1% level of significance

INTERNAL	86	60	68	76	86	82	60	96	86	88	84	62	96	86	92
SEMESTER	73	63	71	81	82	70	68	97	76	84	80	60	94	80	91

## CALCULATION

$$H_0: \mu = \mu_1$$

$$H_1: \mu \neq \mu_1$$

X	Y	Diff(x-y)
86	73	-13
60	63	3
68	71	3
76	81	5
86	82	-4
82	70	-12
60	68	8
96	97	1
86	76	-10
88	84	-4
84	80	-4
62	60	-2
96	94	-2
86	80	-6
92	91	-1
		<b>M=-2.533</b>

$$\text{Sigma}=6.0812$$

$$n=15$$

$$t = \frac{\bar{d}}{(\sigma/\sqrt{n})}$$

$$t = \frac{2.533}{\frac{6.0812}{\sqrt{15}}}$$

$$t = 1.6134$$

$$\text{p-value}=2.976843$$

### RESULT:

Since calculated value is lesser than tabulated value, we accept H<sub>0</sub> and reject H<sub>1</sub>. Therefore, we conclude that there is no significant change in both marks.

## Using R

The Marks secured by an Individual in internal and semester are as follows. Test whether there is any improvement in semester marks than internal marks at 1% level of significance

INTERNAL	86	60	68	76	86	82	60	96	86	88	84	62	96	86	92
SEMESTER	73	63	71	81	82	70	68	97	76	84	80	60	94	80	91

### CODE:

```
x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
```

```
y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
```

```
d=x-y
```

```
d
```

```
dbar=mean(d)
```

```
dbar
```

```
sigma=sd(d)
```

```
sigma
```

```
n=15
```

```
t=dbar/(sigma/sqrt(n))
```

```
t
```

```
alpha=0.01
```

```
t.half.alpha=qt(1-alpha/2,df=n-1)
```

```
t.half.alpha
```

## RESULT:

$$H_0: \mu = \mu_1$$

$$H_1: \mu \neq \mu_1$$

Test statistic:  $t = \bar{d} / (\sigma / \sqrt{n})$

Calculated value = 1.613426

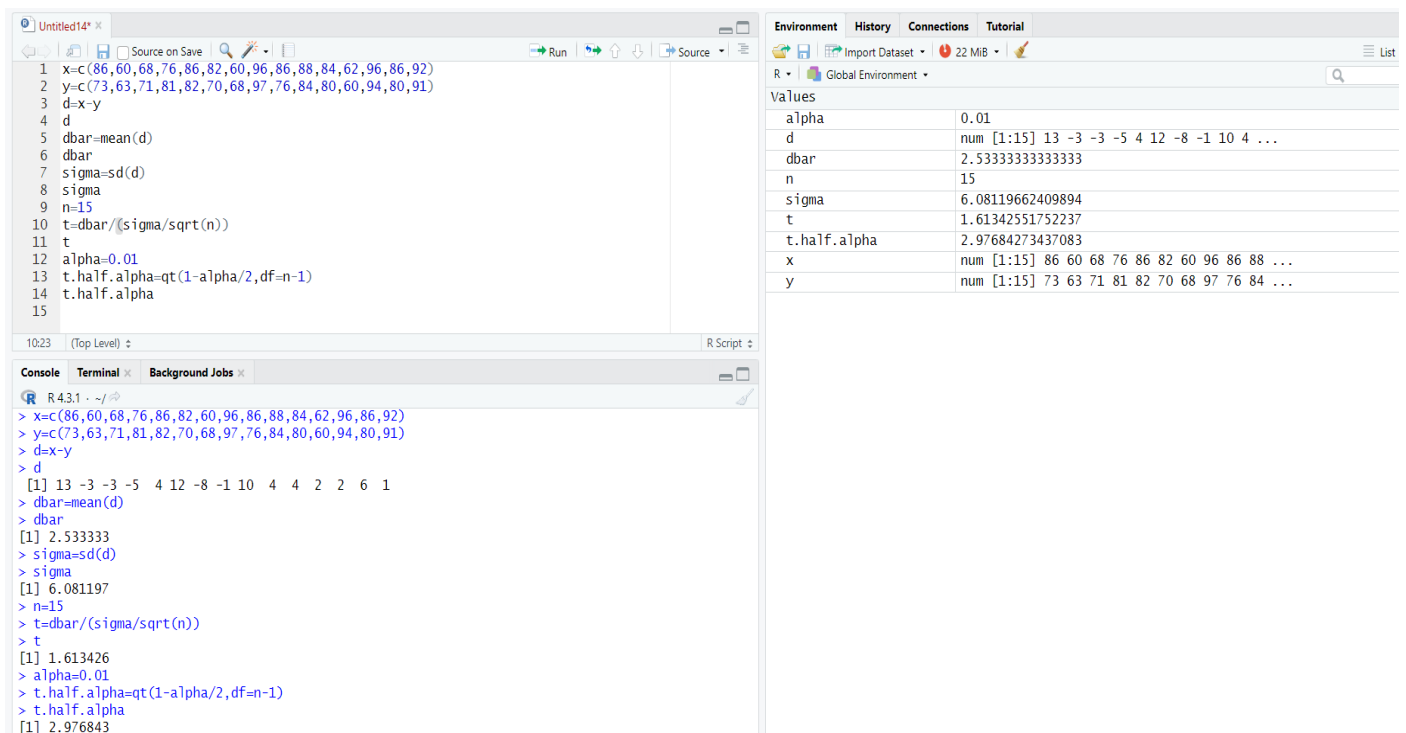
Tabulated value = 2.976843

## Conclusion:

Since calculated value is lesser than tabulated value, we accept  $H_0$  and reject  $H_1$ .

Therefore, we conclude that there is no significant change in both marks.

## OUTPUT:



```
1 x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
2 y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
3 d=x-y
4 d
5 dbar=mean(d)
6 dbar
7 sigma=sd(d)
8 sigma
9 n=15
10 t=dbar/(sigma/sqrt(n))
11 t
12 alpha=0.01
13 t.half.alpha=qt(1-alpha/2,df=n-1)
14 t.half.alpha
15
```

Environment

Variable	Value
alpha	0.01
d	num [1:15] 13 -3 -3 -5 4 12 -8 -1 10 4 ...
dbar	2.53333333333333
n	15
sigma	6.08119662409894
t	1.61342551752237
t.half.alpha	2.97684273437083
x	num [1:15] 86 60 68 76 86 82 60 96 86 88 ...
y	num [1:15] 73 63 71 81 82 70 68 97 76 84 ...

Console

```
R 4.3.1 ~ - / ~
> x=c(86,60,68,76,86,82,60,96,86,88,84,62,96,86,92)
> y=c(73,63,71,81,82,70,68,97,76,84,80,60,94,80,91)
> d=x-y
> d
[1] 13 -3 -3 -5 4 12 -8 -1 10 4 4 2 2 6 1
> dbar=mean(d)
> dbar
[1] 2.533333
> sigma=sd(d)
> sigma
[1] 6.081197
> n=15
> t=dbar/(sigma/sqrt(n))
> t
[1] 1.613426
> alpha=0.01
> t.half.alpha=qt(1-alpha/2,df=n-1)
> t.half.alpha
[1] 2.976843
```

## Data analysis using numpy and pandas

Performed few basic analysis for the chosen data set using python library such as numpy and pandas

### 1. Reading the data:

```
import pandas as pd
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
data=pd.read_csv(r'C:\Users\kumar\OneDrive\Documents\MARK  
S.csv')
```

```
data
```

	SUBJECTS	INTERNALS	SEMESTER
0	FRENCH_1	86	73
1	ENGLISH_1	60	63
2	ALGEBRA AND TRIGNOMETRY	68	71
3	C PROGRAMMING	76	81
4	DIFFERENTIAL CALCULUS	86	82
5	FRENCH_2	82	70
6	ENGLISH_2	60	68
7	INTERNAL CALCULUS	96	97
8	DIFFERENTIAL CALCULUS	86	76
9	NUMERICAL METHODS	88	84
10	FRENCH_3	84	80
11	ENGLISH_3	62	60
12	VECTOR ANALYSIS	96	94
13	MATHETICAL STATISTICS	86	80
14	FINANCIAL MATHEMATICS	92	91



## 2.cleaning data

data.isnull()

	SUBJECTS	INTERNALS	SEMESTER
0	False	False	False
1	False	False	False
2	False	False	False
3	False	False	False
4	False	False	False
5	False	False	False
6	False	False	False
7	False	False	False
8	False	False	False
9	False	False	False
10	False	False	False
11	False	False	False
12	False	False	False
13	False	False	False
14	False	False	False

data.isnull().sum()

```
SUBJECTS    0
INTERNALS    0
SEMESTER     0
dtype: int64
```

data.notnull()

	SUBJECTS	INTERNALS	SEMESTER
0	True	True	True
1	True	True	True
2	True	True	True
3	True	True	True
4	True	True	True
5	True	True	True
6	True	True	True
7	True	True	True
8	True	True	True
9	True	True	True
10	True	True	True
11	True	True	True
12	True	True	True
13	True	True	True
14	True	True	True

data.notnull().sum()

```
SUBJECTS    15
INTERNALS    15
SEMESTER     15
dtype: int64
```

```
#imputation by mean
br = data['INTERNALS'].mean()
data['INTERNALS'].fillna(value=br, inplace=True)
dr = data['SEMESTER'].mean()
data['SEMESTER'].fillna(value=dr, inplace=True)
print( 'internals mean',br)
print('semester mean',dr)

internals mean 80.53333333333333
semester mean 78.0
```

## 2.Descriptive analysis:

```
data.describe()
```

	INTERNALS	SEMESTER
<b>count</b>	15.000000	15.000000
<b>mean</b>	80.533333	78.000000
<b>std</b>	12.431910	10.843036
<b>min</b>	60.000000	60.000000
<b>25%</b>	72.000000	70.500000
<b>50%</b>	86.000000	80.000000
<b>75%</b>	87.000000	83.000000
<b>max</b>	96.000000	97.000000

```
data.describe(include='all')
```

	SUBJECTS	INTERNALS	SEMESTER
count	15	15.000000	15.000000
unique	14	NaN	NaN
top	DIFFERENTIAL CALCULUS	NaN	NaN
freq	2	NaN	NaN
mean	NaN	80.533333	78.000000
std	NaN	12.431910	10.843036
min	NaN	60.000000	60.000000
25%	NaN	72.000000	70.500000
50%	NaN	86.000000	80.000000
75%	NaN	87.000000	83.000000
max	NaN	96.000000	97.000000

```
data.describe(include=['object'])
```

	SUBJECTS
count	15
unique	14
top	DIFFERENTIAL CALCULUS
freq	2

```
data.count()
```

```
SUBJECTS    15
INTERNALS   15
SEMESTER    15
dtype: int64
```

data.min()

```
SUBJECTS      ALGEBRA AND TRIGNOMETRY
INTERNALS                                60
SEMESTER                                60
dtype: object
```

data.max()

```
SUBJECTS      VECTOR ANALYSIS
INTERNALS                                96
SEMESTER                                97
dtype: object
```

## **Data visualization using Python**

- Line

A Python Matplotlib line chart, also known as a line plot or line graph, is a visualization technique used to represent data points as a series of connected straight line segments. This type of chart is commonly used to display trends or relationships between continuous data points over a specified range. created a line chart for internal and semester marks in which blue line represents internals and orange represents semester marks . this is done by using plot() funcions in python matplotlib.

### **CODE:**

```
import pandas as pd
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
data=pd.read_csv(r'C:\Users\kumar\OneDrive\Documents\MARKS.csv')
```

```
x=data['SUBJECTS']
```

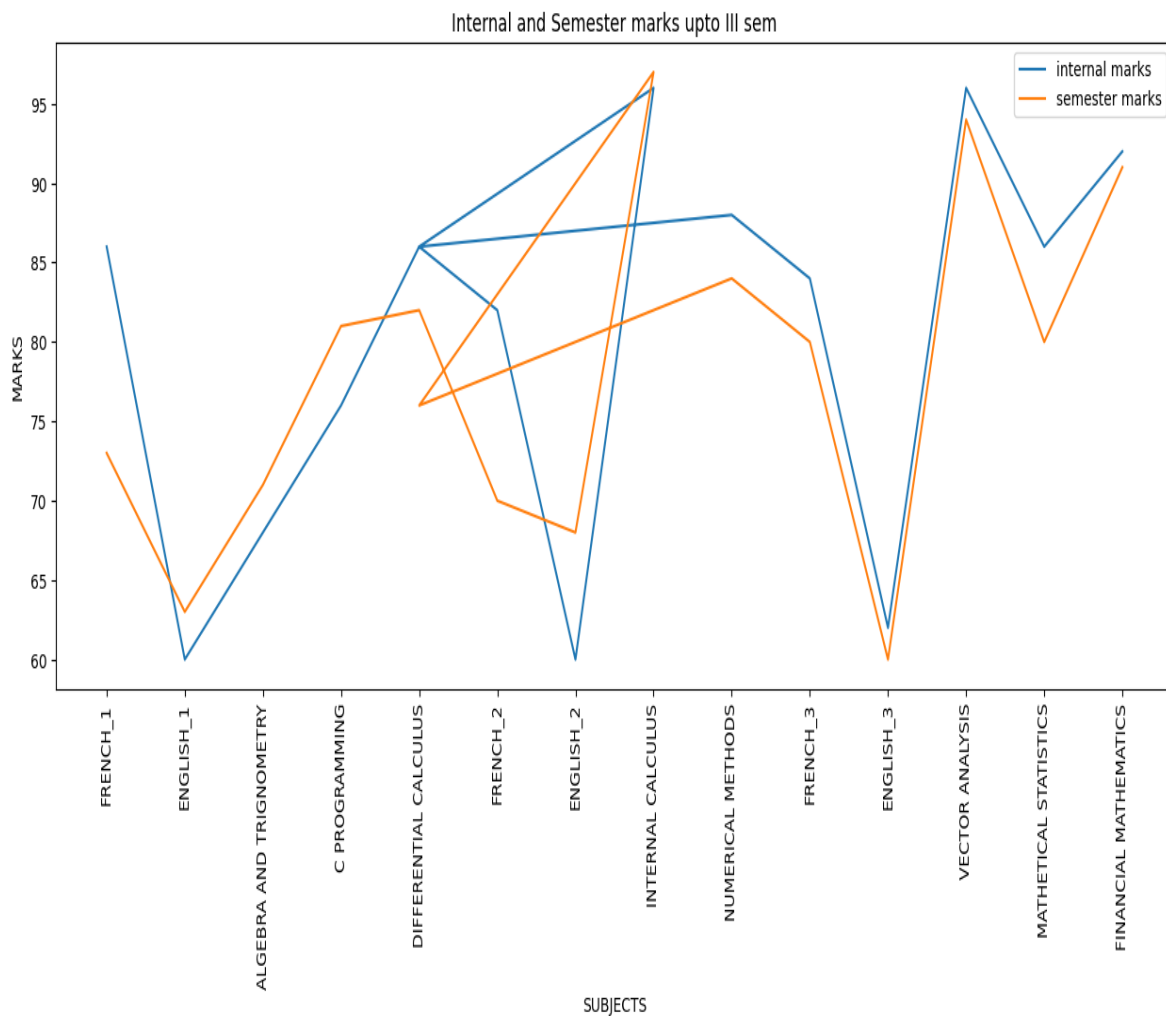
```
y=data['INTERNALS']
```

```

z=data['SEMESTER']
plt.figure(figsize=(9,6))
plt.plot(x,y,label='internal marks')
plt.plot(x,z,label='semester marks')
plt.xlabel('SUBJECTS')
plt.ylabel('MARKS')
plt.xticks(rotation='vertical')
plt.title('Internal and Semester marks upto III sem')
plt.legend()
plt.show()

```

## OUTPUT:



## **INFERENCE:**

Both internal and semester marks are consistent and integral calculus is high scored subject.

- **Bar chart**

A Python Matplotlib bar chart is a type of visualization used to represent categorical data with rectangular bars. The length of each bar corresponds to the value of a particular category, allowing for easy comparison between different categories. created a separate bar diagrams for internal and semester marks.by using bar() function in python matplotlib.

## **CODE:**

```
#internals plot
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

data=pd.read_csv(r'C:\Users\kumar\OneDrive\Documents\MARKS.csv')
x=data['SUBJECTS']
y=data['INTERNALS']
z=data['SEMESTER']
plt.figure(figsize=(9,6))
plt.bar(x,y,label='internal marks',color='red')
plt.xticks(rotation='vertical')
plt.xlabel('SUBJECTS')
plt.ylabel('MARKS')
plt.legend()
```

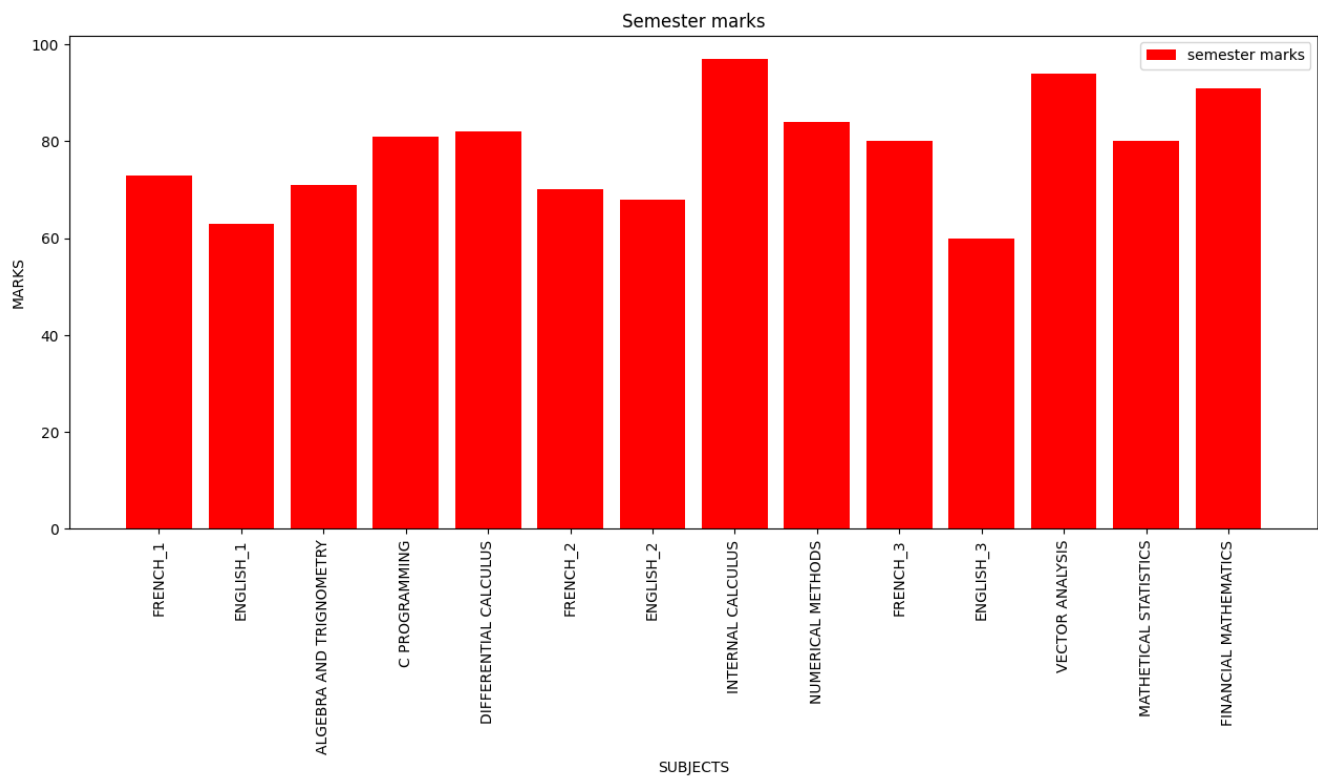
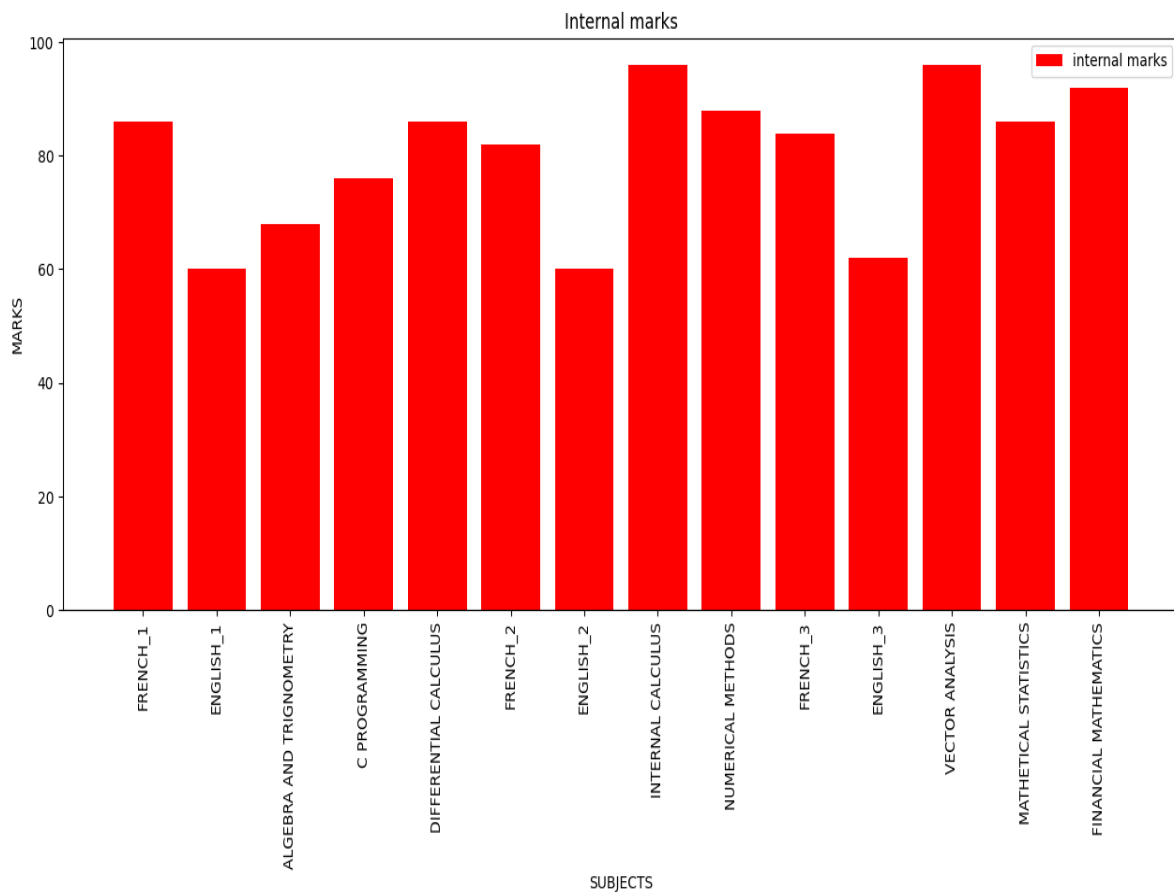
```
plt.title('Internal marks')
plt.show()
#semester plot
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
data=pd.read_csv(r'C:\Users\kumar\OneDrive\Documents\MARKS.csv')
x=data['SUBJECTS']
y=data['INTERNALS']
z=data['SEMESTER']
plt.figure(figsize=(9,6))
plt.bar(x,z,label='semester marks',color='red')
plt.xticks(rotation='vertical')
plt.xlabel('SUBJECTS')
plt.ylabel('MARKS')
plt.legend()
plt.title('Semester marks')
plt.show()
```

### **INFERENCE:**

Internals has more scorings than semester marks .



## OUTPUT:



- **pie chart**

A Python Matplotlib pie chart is a circular statistical graphic used to display categorical data. The arc length of each slice is proportional to the quantity it represents. Pie charts are useful for showing the proportion or distribution of different categories within a dataset. Created a distribution of marks based on subject for the chosen data using pie() function in python matplotlib.

**CODE:**

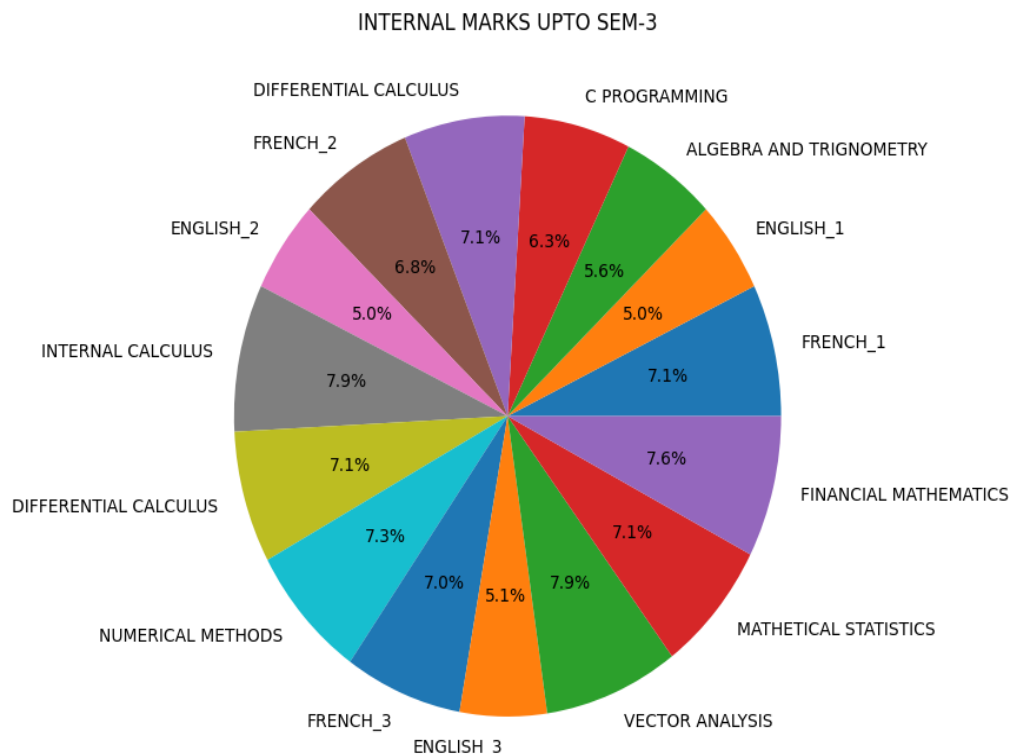
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

data=pd.read_csv(r'C:\Users\kumar\OneDrive\Documents\MARKS.csv')
x=data['SUBJECTS']
y=data['INTERNALS']
plt.title('INTERNAL MARKS UPTO SEM-3')
plt.pie(y,labels=x,autopct='%1.1f%%')
plt.show()
```

**INFERENCE:**

Has an most equal distribution of marks with slight difference.

## OUTPUT:



- **SPECIAL CHART(LOLLIPOP CHART)**

A Lollipop chart, also known as a Dot-Whisker plot, is a combination of a scatter plot and a bar chart. It displays data points as dots (or circles) on a horizontal or vertical axis and connects them to the baseline with a line or a stem. Lollipop charts are commonly used to compare individual data points or to emphasize specific values within a dataset. Created a lollipop chart for both internal and semester marks using stem() function of matplotlib.

## CODE:

```
#internal chart
```

```
import pandas as pd
```

```
import numpy as np
import matplotlib.pyplot as plt
data=pd.read_csv(r'C:\Users\kumar\OneDrive\Documents\MARKS.csv')
x=data['SUBJECTS']
y=data['INTERNALS']
z=data['SEMESTER']
plt.figure(figsize=(9,6))
plt.stem(x,y,linefmt='|')
plt.plot(x,y,'*',label='internal marks',color='red')
plt.xticks(rotation='vertical')
plt.xlabel('SUBJECTS')
plt.ylabel('MARKS')
plt.title('Internal and Semester marks upto III sem')
plt.legend()
plt.show()

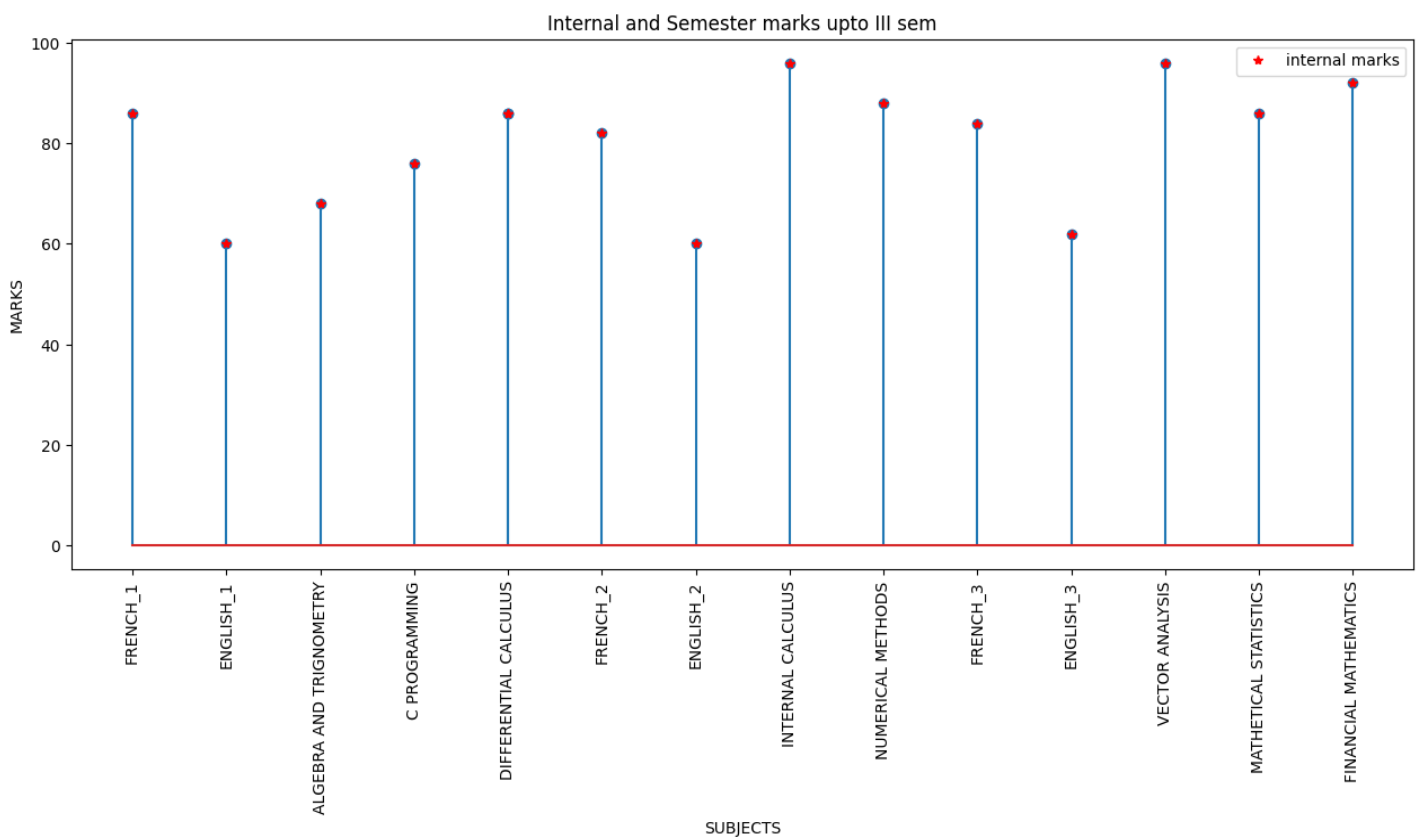
#semester chart
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
data=pd.read_csv(r'C:\Users\kumar\OneDrive\Documents\MARKS.csv')
x=data['SUBJECTS']
y=data['INTERNALS']
z=data['SEMESTER']
```

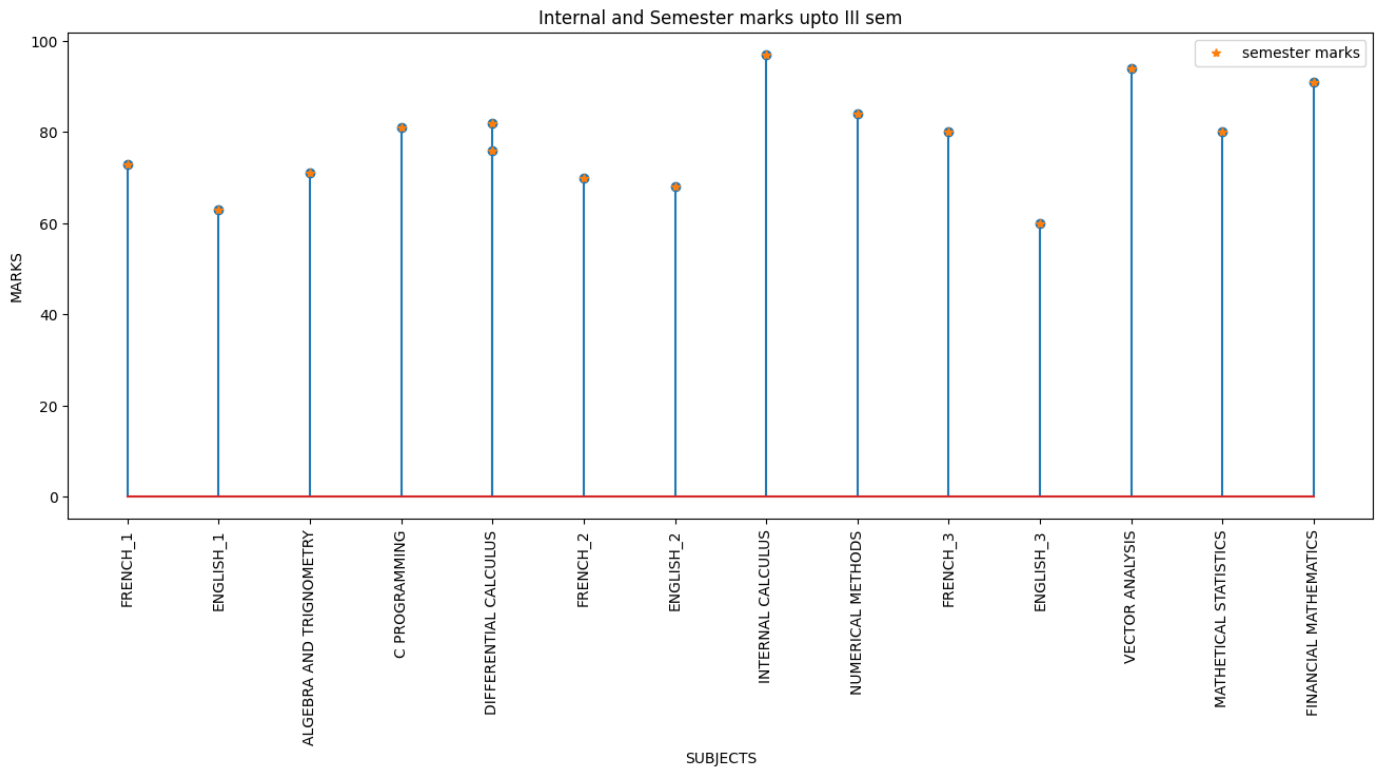
```

plt.figure(figsize=(9,6))
plt.stem(x,z,linefmt='|')
plt.plot(x,z,'*',label='semester marks')
plt.xticks(rotation='vertical')
plt.xlabel('SUBJECTS')
plt.ylabel('MARKS')
plt.title('Internal and Semester marks upto III sem')
plt.legend()
plt.show()

```

## OUPUT:





## INFERENCE:

From internals we find that marks scored in English are more over same.

Highest score-integral calculus(internals)

Second highest score-vector analysis

Highest score-integral calculus(semester)

Second highest score-vector analysis

## CONCLUSION:

In conclusion, this analytics project successfully utilized Python libraries such as NumPy and Pandas to analyse and derive insights from the dataset. By employing various statistical methods, data visualization techniques, particular conclusions were drawn for the chosen data.

## **REFERENCE:**

[Rank Correlation: Spearman Coefficient, Methods, Formula, Examples \(toppr.com\)](https://www.toppr.com)

[Correlation Coefficient - Definition, Formula, Properties and Examples \(byjus.com\)](https://byjus.com)

[Paired T-Test - Definition, Formula, Solved Examples, and FAQs \(cuemath.com\)](https://www.cuemath.com)

[Paired Samples t-test: Definition, Formula, and Example - Statology](#)

[Spearman's Rank Correlation Coefficient: Definition, Meaning \(embibe.com\)](https://www.embibe.com)

## **BOOKS:**

Fabio Nelli. Python for Data Analytics,

S.P.Gupta (2008), Statistical Methods, 25th Edition, Sultan Chand & Sons, New Delhi (Unit III, IV & V).

**BY,**  
**SRINIDHI K**  
**Bsc Mathematics-II**

