Statistics

Based on data science and data analytics stats simplify into below keywords

1. Data Gathering

    -> Extracting the data from different sources and files

2. Organising the data

    -> CLeaning the data based on requirements and Stats

3. Summarizing the data

    -> Representing the data in the form graphical and writting report

4. Interprecting the data

    -> Conculding the data

Statistics

Statistics are of two types

1. Descriptive Statistics

2. Inferential Statistics

Descriptive Statistics

Performing the below

1. Data Gathering

2. Organising data

3. Summarizing data

On Which data

-> Applying all methods on the Population data

Inferential Statistics

Perofrming below method

1. Interprecting data

    -> We concluding the data

on which data

-> Applying on sample data which taken from population

    => Random sample

    => sequential

Discriptive Statistics

1. Measure of central Trendency

2. Measure of Dispersion

Measure of Central Trendency

-> we are finding the middle value

    1. mean

    2. median

    3. mode

Measure of Dispersion

-> Dispersion Means deviation or difference

1. range
2. standard deviation
3. variance

Quartiles

Loading the Data

In [2]:

```
import numpy as np
import pandas as pd
import seaborn as sns

df = sns.load_dataset('iris')
df.head()
```

Out[2]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

I . Measure of Central Trendency

we are finding the middle value

1. mean
2. median
3. mode
4. Relation between mean, median and mode

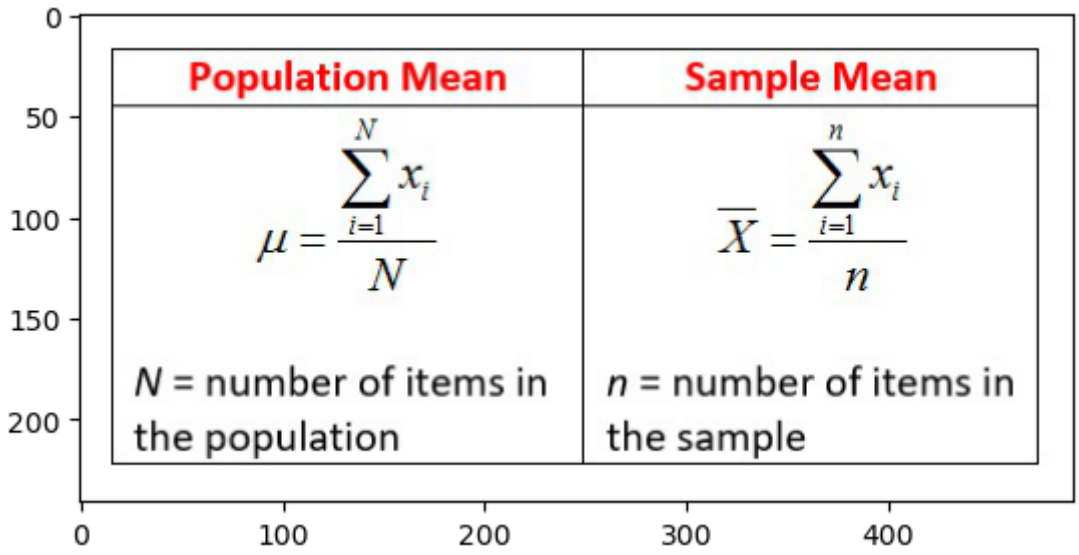
1. Mean

1. Mean is nothing but average
2. sum of all elements by total number of elements
3. Mean is represnted by using mue or xbar
4. xbar means sample mean
5. mue means poplution mean
6. Mean can apply only on Continuous data

Displaying Mean Formula

In [3]:

```
import matplotlib.image as mpimg
import matplotlib.pyplot as plt
# Read Images
img = mpimg.imread(r"C:\Users\divesh\Divesh Classes\Class_3PM\Python\mean.jpeg")
# Output Images
plt.imshow(img)
plt.show()
```



Creating Mean function to apply on the data

In [4]:

```
def mean(data,col):
    m = round(data[col].mean(),2)
    return m
```

In [5]:

```
mean(df,'sepal_length')
```

Out[5]:

```
5.84
```

Applying Pandas Mean Function

In [6]:

```
round(df[['sepal_length','petal_length']].mean(),2)
```

Out[6]:

```
sepal_length    5.84
petal_length    3.76
dtype: float64
```

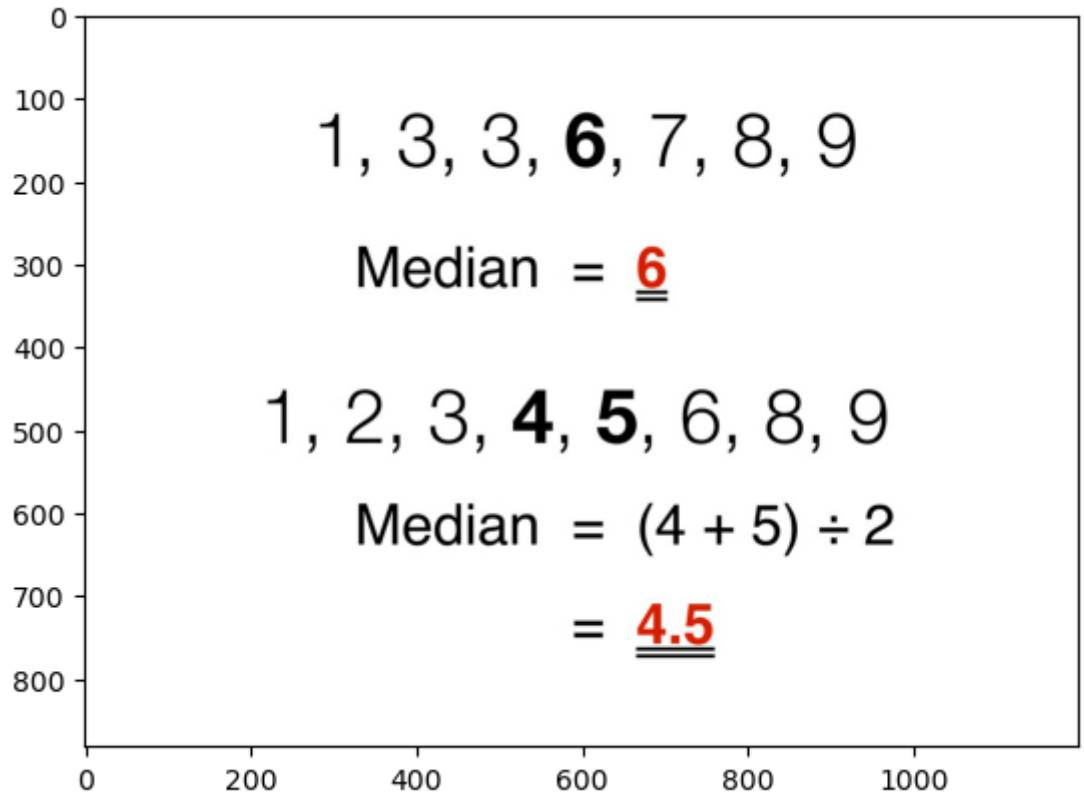
2. Median

```
    -> Middle value

        -> arange values in the ascending order

        -> find the middle count value

            -> 1,3,2,4,5

                -> 1,2,3,4,5 => 3 => odd count

            -> 1,2,4,5,6,3

                -> 1,2,3,4,5,6 => 3,4 => 3+4/2 => 3.5

    -> Median going to apply quantitative data
```

Displaying the Median Formula

In [7]:

```
# Read Images
img = mpimg.imread(r"C:\Users\divesh\Divesh Classes\Class_3PM\Python\median.jpeg")
# Output Images
plt.imshow(img)
plt.show()
```



Creating Median function to apply on the data

In [8]:

```
def median(data,col):
    mn = round(data[col].median(),2)
    return mn
```

In [9]:

```
median(df,'sepal_length')
```

Out[9]:

```
5.8
```

Applying Pandas Median Function

In [10]:

```
round(df[['sepal_length','petal_length']].median(),2)
```

Out[10]:

```
sepal_length    5.80
petal_length    4.35
dtype: float64
```

3. Mode

-> most repeated value

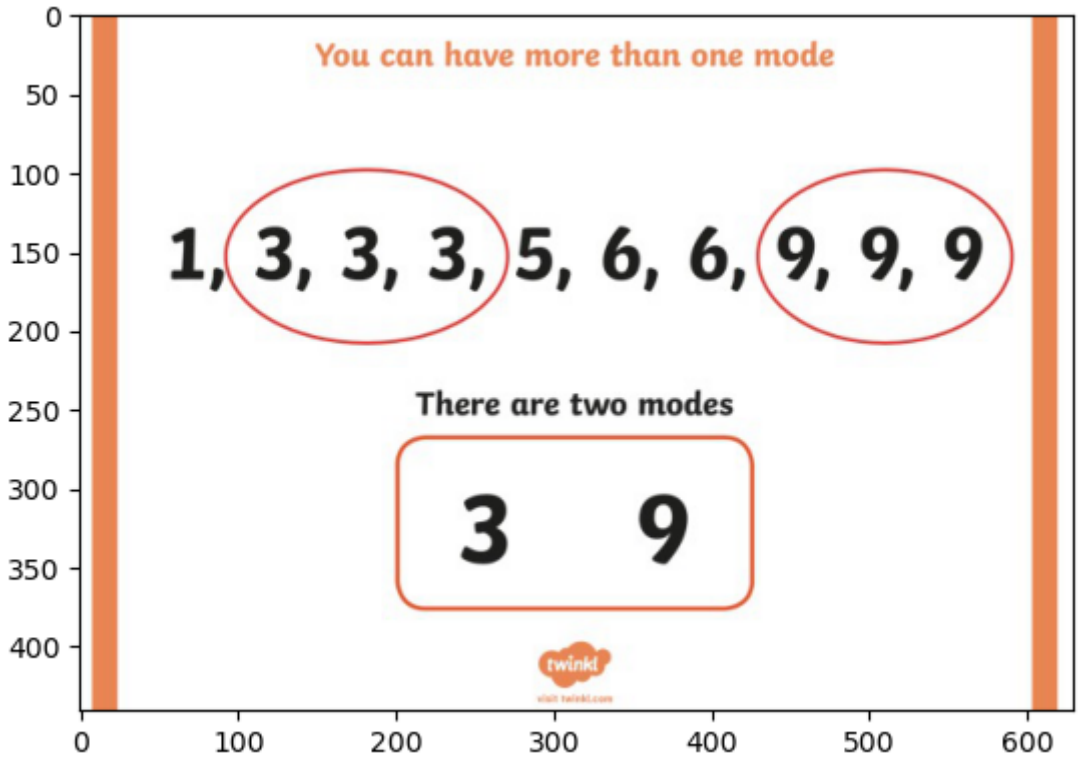
-> We can get one or more mode values

-> mode apply on Qualitative data and descrete data

Displaying Mode Formula

In [11]:

```
# Read Images
img = mpimg.imread(r"C:\Users\divesh\Divesh Classes\Class_3PM\Python\mode.jpeg")
# Output Images
plt.imshow(img)
plt.show()# Read Images
```



Creating mode Formula

In [12]:

```
def mode(data,col):
    md = data[col].mode()
    return md
```

In [13]:

```
mode(df,'species')
```

Out[13]:

```
0        setosa
1    versicolor
2     virginica
Name: species, dtype: object
```

Applying by using pandas mode

In [14]:

```
df['species'].mode()
```

Out[14]:

```
0        setosa
1    versicolor
2     virginica
Name: species, dtype: object
```

4. Relation between mean median mode

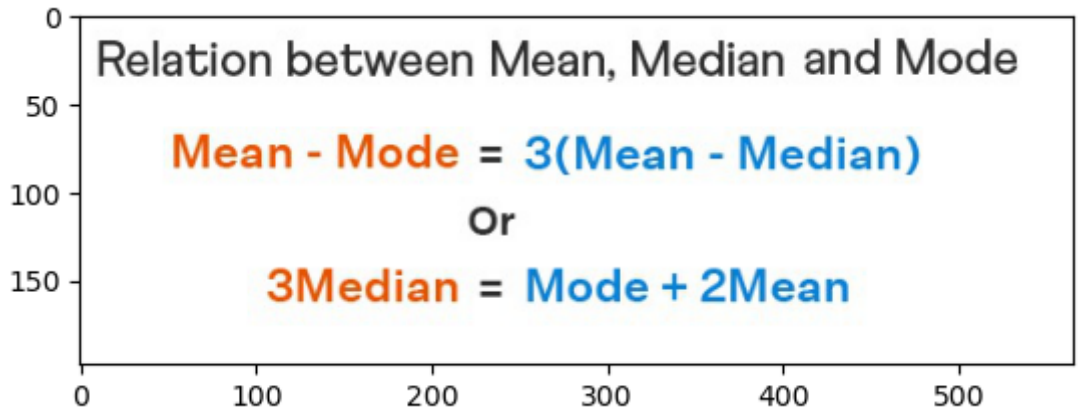
this one we can apply on quantitave data

mode = 3*median* - 2mean

Displaying the Relation between mean, median and mode formula

In [15]:

```
import matplotlib.image as mpimg
import matplotlib.pyplot as plt
# Read Images
img = mpimg.imread(r"C:\Users\divesh\Divesh Classes\Class_3PM\Python\relation.jpeg")
# Output Images
plt.imshow(img)
plt.show()
```



Creating the mode formula

In [16]:

```
def mode_qn(data,col):
    mean = round(data[col].mean(),2)
    median = round(data[col].median(),2)
    mode = round(3*median - 2*mean,2)
    return mean,median,mode
```

In [17]:

```
mode_qn(df,'sepal_length')
```

Out[17]:

```
(5.84, 5.8, 5.72)
```

II. Measure of Dispersion

Dispersion means -> Deviation -> difference

1. Range
2. Standard Deviation
3. Variance

1. Range

In set of data, difference between max value and min value

```
    -> Range = Max - Min
```

In [18]:

```
def range(data,col):
    maxi = round(data[col].max(),2)
    mini = round(data[col].min(),2)
    range_a = round(maxi - mini)
    return range_a
```

In [19]:

```
range(df,'sepal_length')
```

Out[19]:

4

2. Standard Deviation:

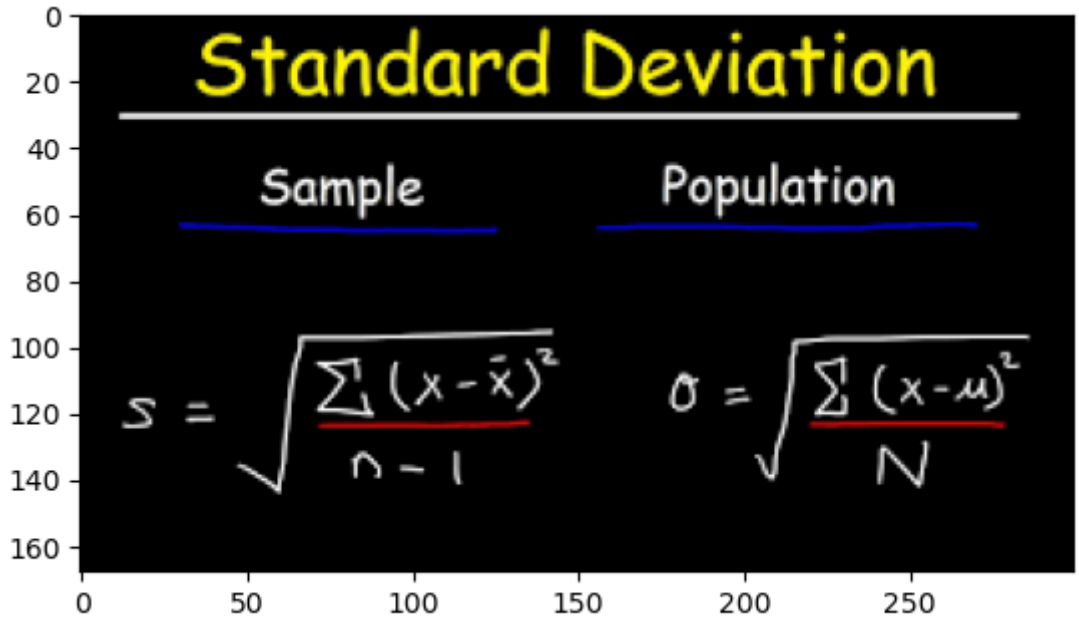
Standard Deviation is a measure that used to quantify the amount of variation or dispersion of a set of data.

```
    -> represents by using

        => s   (Latin Letter)

        => sigma (Greek Letter)
```

Displaying the Formula for standard deviation

In [20]:

```
import matplotlib.image as mpimg
import matplotlib.pyplot as plt
# Read Images
img = mpimg.imread(r"C:\Users\divesh\Divesh Classes\Class_7PM\standard deviation.png")
# Output Images
plt.imshow(img)
plt.show()
```



Creating the standard deviation formula

In [21]:

```
def std(data,col):
    std = round(data[col].std(),2)
    return std
```

In [22]:

```
std(df,'sepal_length')
```

Out[22]:

```
0.83
```

In [23]:

```
round(df[['sepal_length','sepal_width']].std(),2)
```

Out[23]:

```
sepal_length    0.83
sepal_width     0.44
dtype: float64
```

3. Variance

```
    -> Variance is nothing but square root of standard deviation
```

Creating the variance formula

In [24]:

```
def variance(data,col):
    var = round(data[col].var(),2)
    return var
```

In [25]:

```
variance(df,'sepal_length')
```

Out[25]:

```
0.69
```

Applying from the pandas

In [26]:

```
round(df[['sepal_length','sepal_width']].var(),2)
```

Out[26]:

```
sepal_length    0.69
sepal_width     0.19
dtype: float64
```

III. Quartile

Data info

In statistics we are dividing data into few equal parts.

```
    -> Percentile -> 100

    -> Dec -> 10

    -> Oct -> 8

    -> Quartile -> 4
```

In Pertile and Quartiles only

Quartiles

-> Breaking whole data into 4 equal parts

-> Q1 => 25% -> 25%

-> Q2 => 25% -> 50%

-> Q3 => 25% -> 75%

-> Q4 => 25% -> 100%

In [27]:

```
def quartile(data,col):
    q1 = round(data[col].quantile(0.25),2)
    q2 = round(data[col].quantile(0.5),2)
    q3 = round(data[col].quantile(0.75),2)
    return q1,q2,q3
```

In [28]:

```
quartile(df,'sepal_length')
```

Out[28]:

```
(5.1, 5.8, 6.4)
```

In [29]:

```
df[['sepal_length','sepal_width']].quantile(0.25)
```

Out[29]:

```
sepal_length    5.1
sepal_width     2.8
Name: 0.25, dtype: float64
```

In [30]:

```
df[['sepal_length','sepal_width']].quantile(0.5)
```

Out[30]:

```
sepal_length    5.8
sepal_width     3.0
Name: 0.5, dtype: float64
```

In [31]:

```
df[['sepal_length','sepal_width']].quantile(0.75)
```

Out[31]:

```
sepal_length    6.4
sepal_width     3.3
Name: 0.75, dtype: float64
```

In [ ]:

```
```