## Question no. 7 - Calculate Mean, Median, Mode, Variance, Standard Deviation, Range & comment about the values / draw inferences, for the given dataset¶

For Points, Score, Weigh>

Find Mean, Median, Mode, Variance, Standard Deviation, and Range and also Comment about the values/ Draw some inferences.

Use Q7.csv file

```
#importing required library
import pandas as pd
import numpy as np

#Reding the CSV file
df=pd.read_csv("Q7.csv")

#Know the Dataset parameters
df.head()
```

```
           Unnamed: 0  Points  Score  Weigh
0           Mazda RX4    3.90  2.620  16.46
1       Mazda RX4 Wag    3.90  2.875  17.02
2          Datsun 710    3.85  2.320  18.61
3      Hornet 4 Drive    3.08  3.215  19.44
4   Hornet Sportabout    3.15  3.440  17.02
```

```
#calculating mean
df.mean()
```

```
Points     3.596563
Score      3.217250
Weigh     17.848750
dtype: float64
```

```
#calculating median
df.median()
```

```
Points     3.695
Score      3.325
Weigh     17.710
dtype: float64
```

```
#calculating mode
df[['Points','Score','Weigh']].mode()
```

```
   Points  Score  Weigh
0    3.07   3.44  17.02
1    3.92    NaN  18.90
```

```
#calculating variance
df.var()
```

```
Points    0.285881
Score     0.957379
Weigh     3.193166
dtype: float64
```

```
#calculating Standard Deviation
df.std()
```

```
Points    0.534679
Score     0.978457
Weigh     1.786943
dtype: float64
```

```
#calculating range of following features
range_points=df['Points'].max()-df['Points'].min()
range_score=df['Score'].max()-df['Score'].min()
range_weigh=df['Weigh'].max()-df['Weigh'].min()
print(range_points)
print(range_score)
print(range_weigh)
```

```
2.17
3.9109999999999996
8.399999999999999
```

## Question no. 9 - Calculate Skewness, Kurtosis & draw inferences on the following data¶

Cars speed and distance
Use Q9_a.csv

SP and Weight(WT)

Use Q9_b.csv

## (a)

```
#importing library
import scipy.stats
```

```
#loading the dataset
cf=pd.read_csv("Q9_a.csv")
```

```
#know about the dataset
cf.head()
```

```
   Index  speed  dist
0      1      4     2
1      2      4    10
2      3      7     4
3      4      7    22
4      5      8    16
```

```
#calculating skewness
cf.skew()
```

```
Index    0.000000
speed   -0.117510
dist     0.806895
dtype: float64
```

```
#calculating kurtosis
cf.kurt()
```

```
Index   -1.200000
speed   -0.508994
dist     0.405053
dtype: float64
```

## Question no. 9(b)

```
#loading the dataset
sp=pd.read_csv("Q9_b.csv")
```

```
#Know about the dataset
sp.head()
```

```
   Unnamed: 0          SP         WT
0           1  104.185353  28.762059
1           2  105.461264  30.466833
2           3  105.461264  30.193597
3           4  113.461264  30.632114
4           5  104.461264  29.889149
```

```
#calculating skewness
sp.skew()
```

```
Unnamed: 0    0.000000
SP            1.611450
WT           -0.614753
dtype: float64
```

```
#calculating kurtosis
sp.kurt()
```

```
Unnamed: 0   -1.200000
SP            2.977329
WT            0.950291
dtype: float64
```

## Question no. 11- Suppose we want to estimate the average weight of an adult male in Mexico. We draw a random sample of 2,000 men from a population of 3,000,000 men and weigh them. We find that the average person in our sample weighs 200 pounds, and the standard deviation of the sample is 30 pounds. Calculate 94%,98%,96% confidence interval?

```
#importing library
import pandas as pd
import numpy as np
import scipy
from scipy import stats

#calculating 94% confidence interval of the given data in the question
#mean=200
#standart deviation=30
CI=stats.t.interval(0.94,1999,200,30)
print('94% Confidence interval of the data is',np.round(CI,3))
```

```
94% Confidence interval of the data is [143.544 256.456]
```

```
CP_94=stats.t.ppf(0.97,df=1999)
CP_98=stats.t.ppf(0.99,df=1999)
CP_96=stats.t.ppf(0.98,df=1999)
print('t94 is ',np.round(CP_94,4))
print('t98 is ',np.round(CP_98,4))
print('t96 is ',np.round(CP_96,4))
```

```
t94 is  1.8819
t98 is  2.3282
t96 is  2.0551
```

## Question no. 12 - Below are the scores obtained by a student in tests

34,36,36,38,38,39,39,40,40,41,41,41,41,42,42,45,49,56

Find mean, median, variance, standard deviation.

What can we say about the student marks?

```
#Making dataframe of the students data
Marks=pd.DataFrame([34,36,36,38,38,39,39,40,40,41,41,41,41,42,42,45,49
,56])

#calculating mean of the data
Marks.mean()
```

```
0    41.0
dtype: float64
```

```
#claculating median of the data
Marks.median()
```

```
0    40.5
dtype: float64
```

```
#calculating variance of the data
Marks.var()
```

```
0    25.529412
dtype: float64
```

```
#calculating standard deviation of the data
Marks.std()
```

```
0    5.052664
dtype: float64
```

## Question no.20 -Calculate probability from the given dataset for the below cases¶

Data _set: Cars.csv

Calculate the probability of MPG of Cars for the below cases.

MPG <- Cars$MPG

P(MPG>38)

P(MPG<40)

P(20<MPG<50)

```
#import library
import pandas as pd
import numpy as np
import scipy
from scipy import stats

cars=pd.read_csv("cars.csv")

cars.head()
```

```
   HP        MPG  VOL          SP         WT
0  49  53.700681   89  104.185353  28.762059
1  55  50.013401   92  105.461264  30.466833
2  55  50.013401   92  105.461264  30.193597
3  70  45.696322   92  113.461264  30.632114
4  53  50.504232   92  104.461264  29.889149

cars.shape

(81, 5)

cars_mean=cars['MPG'].mean()
print('mean of MPG',np.round(cars_mean,4))

mean of MPG 34.4221

cars_std=cars['MPG'].std()
print('std of MPG',np.round(cars_std,4))

std of MPG 9.1314

#probability of MPG>38
P38=1-stats.norm.cdf(38,cars_mean,cars_std)
print('P(MPG>38)-',np.round(P38,4))

P(MPG>38)- 0.3476

#probability of MPG<40
P40=stats.norm.cdf(40,cars_mean,cars_std)
print('P(MPG<40)-',np.round(P40,4))

P(MPG<40)- 0.7293

#probability fo P(20<MPG<50)
P20_50=(stats.norm.cdf(50,cars_mean,cars_std)-
stats.norm.cdf(20,cars_mean,cars_std))
print(" P(20<MPG<50) is",np.round(P20_50,4))

 P(20<MPG<50) is 0.8989
```

## Question no. 21 - Check whether the data follows normal distribution

## (a) Check whether the MPG of Cars follows Normal Distribution

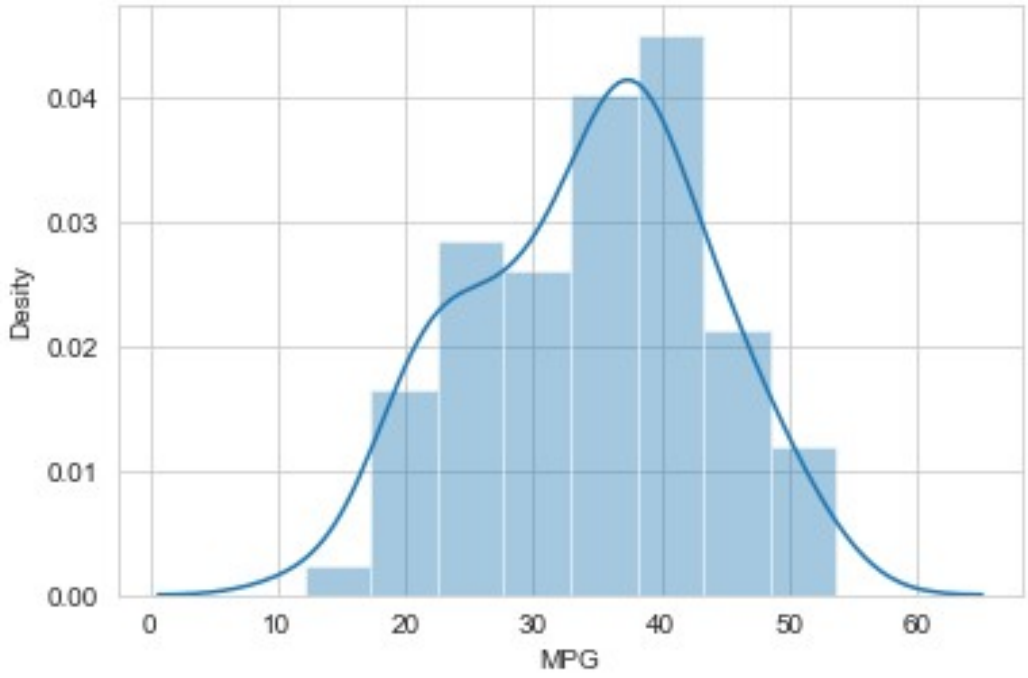
Dataset: Cars.csv

```
#import library
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
import warnings
warnings.filterwarnings("ignore")

cars=pd.read_csv("cars.csv")

sns.set_style('whitegrid')
sns.distplot(cars['MPG'])
plt.xlabel("MPG")
plt.ylabel("Desity")
plt.show()
```




```
cars['MPG'].skew()
```

```
-0.17794674747025727
```

## Question no. 21 (b)

## B) Check Whether the Adipose Tissue (AT) and Waist Circumference(Waist) from wc-at data set follows Normal Distribution

Dataset: wc-at.csv

```
#import library
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

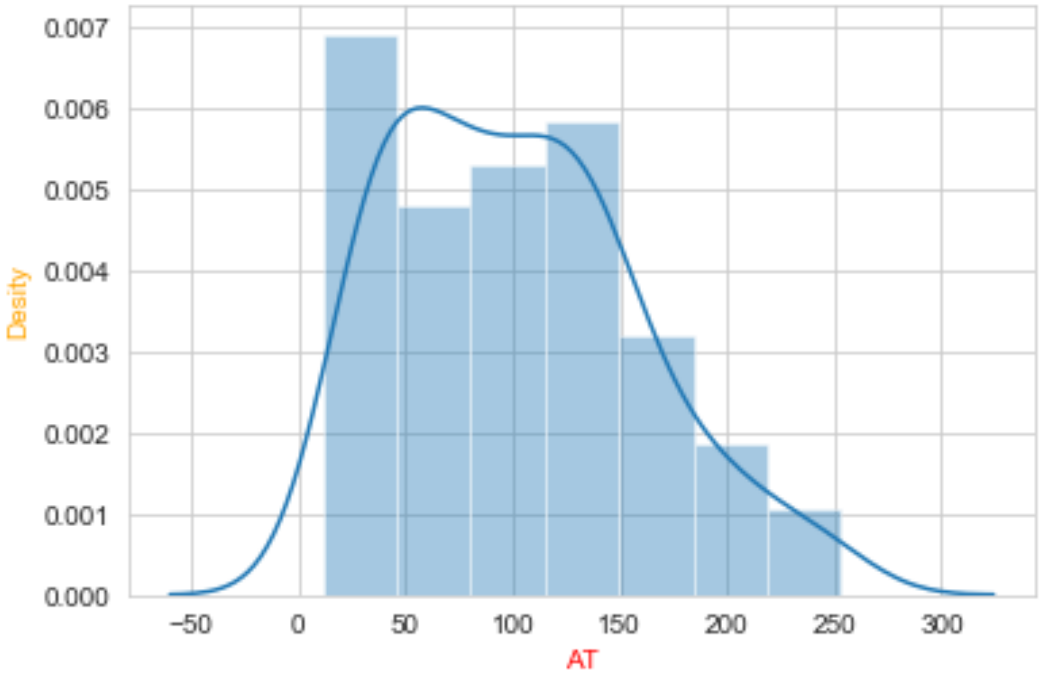
```
import warnings
warnings.filterwarnings("ignore")

wcat=pd.read_csv("wc-at.csv")

wcat.head()
```

```
   Waist     AT
0  74.75  25.72
1  72.60  25.89
2  81.80  42.60
3  83.95  42.80
4  74.65  29.84
```

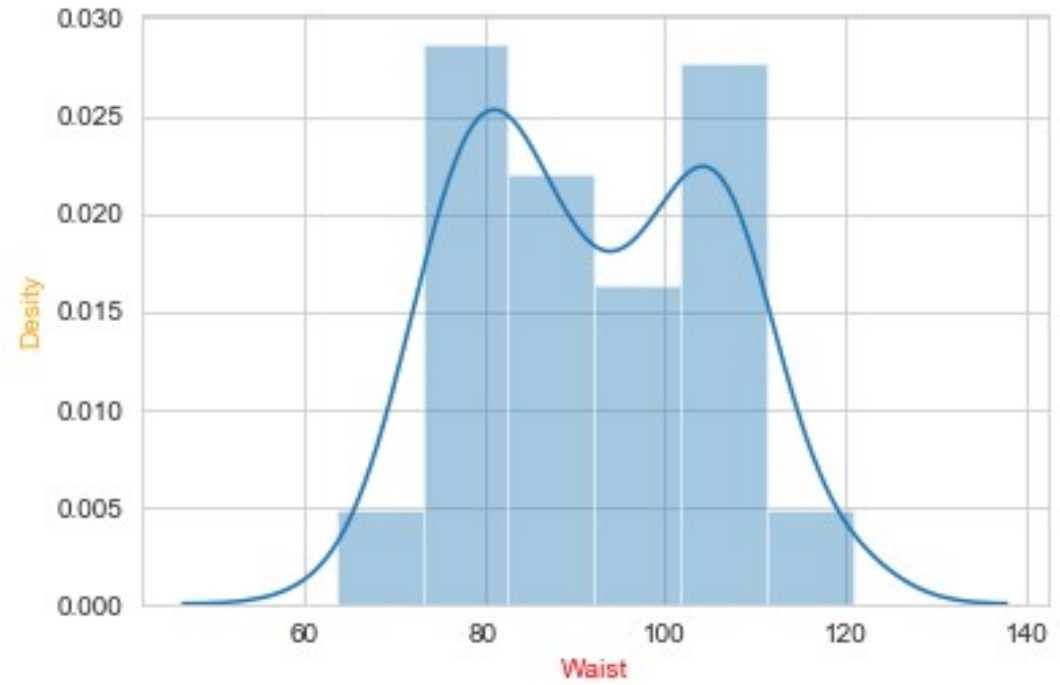
```
sns.set_style('whitegrid')
sns.distplot(wcat['AT'])
plt.xlabel("AT",color='red')
plt.ylabel("Desity",color='orange')
plt.show()
```




```
wcat['AT'].skew()
```

```
0.584869324127853
```

```
sns.set_style('whitegrid')
sns.distplot(wcat['Waist'])
plt.xlabel("Waist",color='red')
plt.ylabel("Desity",color='orange')
plt.show()
```




```
wcat['Waist'].skew()
```

```
0.1340560824786468
```

## Question no. 22

## Calculate the Z scores of 90% confidence interval, 94% confidence interval, 60% confidence interval

```
import scipy
from scipy import stats
import pandas as pd
import numpy as np
```

```
#calculating Z score of 90% confidence interval
Z90=stats.norm.ppf(0.90)
print('Z score of 90% confidence interval is',np.round(Z90,4))
```

```
Z score of 90% confidence interval is 1.2816
```

```
#calculating Z score of 94% confidence interval
Z94=stats.norm.ppf(0.94)
print('Z score of 94% confidence interval is',np.round(Z94,4))
```

```
Z score of 94% confidence interval is 1.5548
```

```
#calculating Z score of 60% confidence interval
Z60=stats.norm.ppf(0.60)
print('Z score of 60% confidence interval is',np.round(Z60,4))
```

```
Z score of 60% confidence interval is 0.2533
```

## Question no. 23

## Calculate the t scores of 95% confidence interval, 96% confidence interval, 99% confidence interval for sample size of 25

```
#calculation T score for 95% CI
T95=stats.t.ppf(0.95,24)
print('T score for 95% confidence interval is',np.round(T95,4))
```

```
T score for 95% confidence interval is 1.7109
```

```
#calculation T score for 96% CI
T96=stats.t.ppf(0.96,24)
print('T score for 96% confidence interval is',np.round(T96,4))
```

```
T score for 96% confidence interval is 1.8281
```

```
#calculation T score for 99% CI
T99=stats.t.ppf(0.99,24)
print('T score for 99% confidence interval is',np.round(T99,4))
```

```
T score for 99% confidence interval is 2.4922
```

## Q 24) A Government company claims that an average light bulb lasts 270 days. A researcher randomly selects 18 bulbs for testing. The sampled bulbs last an average of 260 days, with a standard deviation of 90 days. If the CEO's claim were true, what is the probability that 18 randomly selected bulbs would have an average life of no more than 260 days

```
import numpy as np
import scipy
from scipy import stats
# h0 -> n>260
# h1 -> n<260

#Mean=270
#s_mean=260
#std=90
#n=18
```

```
t=(260-270)/(90/np.sqrt(18))

pvalue=stats.t.cdf(np.round(t,3),df=17)

print('The probabilty of gitting an average life of no more than 260
days are',np.round((pvalue)*100,3))
```

```
The probabilty of gitting an average life of no more than 260 days are
32.181
```