|  |  |
| --- | --- |
| Activity | Data Type |
| Number of beatings from Wife | Discrete |
| Results of rolling a dice | Discrete |
| Weight of a person | Continuous |
| Weight of Gold | Continuous |
| Distance between two places | Discrete |
| Length of a leaf | Continuous |
| Dog's weight | Continuous |
| Blue Color | Discrete |
| Number of kids | Discrete |
| Number of tickets in Indian railways | Discrete |
| Number of times married | Discrete |
| Gender (Male or Female) | Continuous |

Q1) Identify the Data type for the Following:

Q2) Identify the Data types, which were among the following

Nominal, Ordinal, Interval, Ratio.

|  |  |
| --- | --- |
| Data | Data Type |
| Gender | Nominal |
| High School Class Ranking | Ordinal |
| Celsius Temperature | Ratio |
| Weight | Ratio |
| Hair Color | Nominal |
| Socioeconomic Status | Ordinal |
| Fahrenheit Temperature | Ratio |
| Height | Ratio |
| Type of living accommodation | Ordinal |
| Level of Agreement | Interval |
| IQ(Intelligence Scale) | Interval |
| Sales Figures | Ratio |
| Blood Group | Nominal |
| Time Of Day | Ordinal |
| Time on a Clock with Hands | Ordinal |
| Number of Children | Ratio |
| Religious Preference | Nominal |
| Barometer Pressure | Ordinal |
| SAT Scores | Interval |
| Years of Education | Ordinal |

Q3) Three Coins are tossed, find the probability that two heads and one tail are obtained?

Answer:

No of coins = 3

Two head one tail combination with three coins tossed = [ HTH, HHT, THH]

Total possibilities of three coin tossed = [ HHH, HHT, HTH, THH, HTT, THT, TTH, TTT]

Probability of two heads and one tail are obtained = 3/8 =0.375

Q4) Two Dice are rolled, find the probability that sum is

1. Equal to 1
2. Less than or equal to 4
3. Sum is divisible by 2 and 3

Answer:

Total probability of 2 dices rolled is = [ 11, 12, 13, 14, 15, 16, 21, 22, 23, 24, 25, 26, 31, 32, 33, 34, 35, 36, 41, 42, 43, 44, 45, 46, 51, 52, 53, 54, 55, 56, 61, 62, 63, 64, 65, 66] = 36

1. Sum Equal to 1 = 0

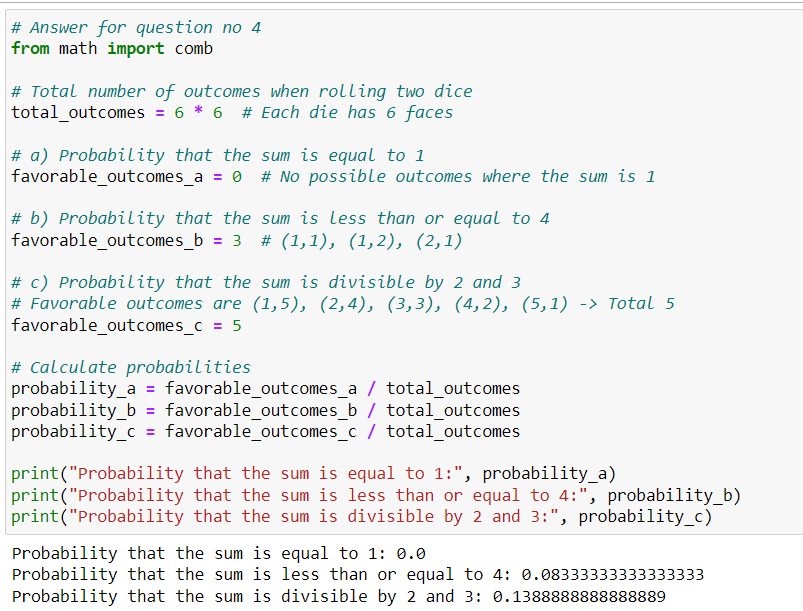
[There is no event possible of two dices rolled with the sum equal to 1]

1. Sum Less than or equal to 4 [sum <= 4]

Probability of sum Less than or equal to 4 is = [11, 12, 13, 21, 22, 31]/ 36 = 6/36 =1/6 = 0.1666

1. Sum is divisible by 2 and 3

Probability of sum is divisible by 2 and 3 = [ 15, 24, 33, 42, 51, 66] = [ 6, 6, 6, 6, 6, 12] = 6/36 = 1/6 =0.16667



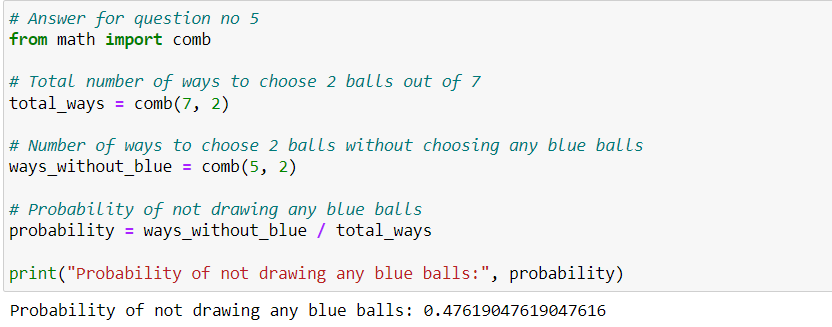
Q5) A bag contains 2 red, 3 green and 2 blue balls. Two balls are drawn at random. What is the probability that none of the balls drawn is blue?

Answer:

Total no of balls in a bag = 7 balls

Total Probability of two balls drawn = [ RR, RG, RB, GG, GR, GB, BB, BR, BG]

Probability that none of the balls drawn is blue = [ RR, RG, GG, GR,] /9 = 4/9 =0.4444



Q6) Calculate the Expected number of candies for a randomly selected child

Below are the probabilities of count of candies for children (ignoring the nature of the child-Generalized view)

|  |  |  |
| --- | --- | --- |
| CHILD | Candies count | Probability |
| A | 1 | 0.015 |
| B | 4 | 0.20 |
| C | 3 | 0.65 |
| D | 5 | 0.005 |
| E | 6 | 0.01 |
| F | 2 | 0.120 |

Child A – probability of having 1 candy = 0.015.

Child B – probability of having 4 candies = 0.20

Answer:

To calculate the expected number of candies for a randomly selected child is = ∑(X) =P (

Where:

∑(x) is the expected probability for a randomly selected child.

is the number of candy count.

is the probability of child with particular no of candies as mentioned in above table.

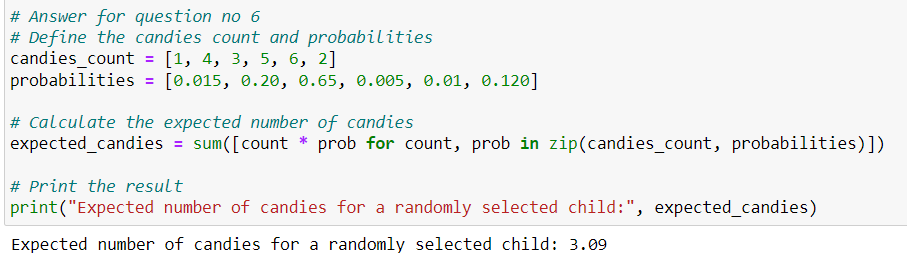
Let x be the number of candies a child has.

∑(X) = 1\*0.015 + 4\*0.20 + 3\*0.65 + 5\*0.005 + 6\*0.01 + 2\*0.120

∑(X) = 0.015 + 0.80 + 1.95 + 0.025 + 0.06 + 0.240

∑(X) = 3.225

Therefore, the expected number of candies for a randomly selected child is 3.125.



Q7) 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.

Answer:

A) points column

1) The mean of the points column from Q7.csv file = = ≈ 3.596563

2) To find the median of the given set of numbers, first, let's arrange them in ascending order:

2.76, 2.76, 2.93, 3, 3.07, 3.07, 3.07, 3.08, 3.08, 3.15, 3.15, 3.21, 3.23, 3.54, 3.62, 3.69, 3.7, 3.73, 3.77, 3.85, 3.9, 3.9, 3.92, 3.92, 3.92, 4.08, 4.08, 4.11, 4.22, 4.22, 4.43, 4.93

2.76, 2.76, 2.93, 3, 3.07, 3.07, 3.07, 3.08, 3.08, 3.15, 3.15, 3.21, 3.23, 3.54, 3.62, 3.69, 3.7, 3.73, 3.77, 3.85, 3.9, 3.9, 3.92, 3.92, 3.92, 4.08, 4.08, 4.11, 4.22, 4.22, 4.43, 4.93

The median is the middle value. Since there are 32 numbers in this set, the median will be the average of the 16th and 17th values.

Median= =3.695

So, the median of the given set of numbers is 3.695.

3)Mode:

* The mode is the value that appears most frequently in the dataset.
* In this dataset, the mode is 3.07 and 3.92 as they both appear three times.

4)Variance:

Variance is the average of the squared differences from the mean.

Variance =

Variance ≈ 4.15

5)Standard Deviation:

* Standard Deviation is the square root of the variance.
* Standard Deviation ≈ √4.15 ≈ 2.037

6)Range:

* Range is the difference between the maximum and minimum values.
* Range = 4.93 − 2.76 =2.17
* Range = 2.17

comments and Inferences:

* The mode indicates the most frequently occurring values, which are 3.07 and 3.92 in this dataset.
* The variance and standard deviation provide information about the spread or dispersion of the data. A lower standard deviation suggests less variability.
* The range is relatively small (2.17), indicating that the values are not widely spread out.
* Since the standard deviation is close to the mean, it suggests that the values are relatively close to the average, indicating less variability.

In summary, the dataset appears to have a moderate level of consistency with values centered around the mean, and the spread is not extensive.

Similarly for score:

1. Mean ≈ 3.039
2. Median ≈ 3.19
3. Mode = 3.44
4. Variance ≈ 1.349
5. Standard Deviation ≈ √1.349 = 1.161
6. Range = 5.424 – 1.413 ≈3.911

Comments and Inferences:

* The mean represents the average value, and in this case, it's around 3.039.
* The median, which is close to the mean, indicates a relatively symmetric distribution.
* The mode is 3.44, which is the most frequently occurring value.
* The variance and standard deviation quantify the spread of the data. A higher standard deviation and variance suggest more variability.
* The range gives an idea of how spread out the values are.

In summary, the dataset exhibits moderate variability, and the values are spread out over a range of approximately 3.911. The distribution is relatively symmetric

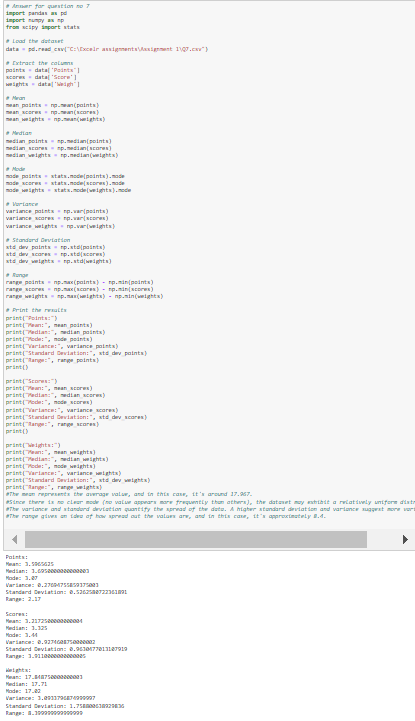
Similarly for weigh:

1. Mean ≈ 17.967
2. Median ≈ 17.82
3. Mode = none
4. Variance ≈ 4.129
5. Standard Deviation ≈ √4.129 = 2.032
6. Range = 22.9 – 14.5 ≈ 8.4

Comments and Inferences:

* The mean represents the average value, and in this case, it's around 17.967.
* Since there is no clear mode (no value appears more frequently than others), the dataset may exhibit a relatively uniform distribution.
* The variance and standard deviation quantify the spread of the data. A higher standard deviation and variance suggest more variability.
* The range gives an idea of how spread out the values are, and in this case, it's approximately 8.4.

In summary, the dataset has a moderate level of variability, with values distributed over a range of approximately 8.4. The lack of a clear mode may suggest a relatively uniform distribution of values.

**Use Q7.csv file **

Q8) Calculate Expected Value for the problem below

1. The weights (X) of patients at a clinic (in pounds), are

108, 110, 123, 134, 135, 145, 167, 187, 199

Assume one of the patients is chosen at random. What is the Expected Value of the Weight of that patient?

Answer:

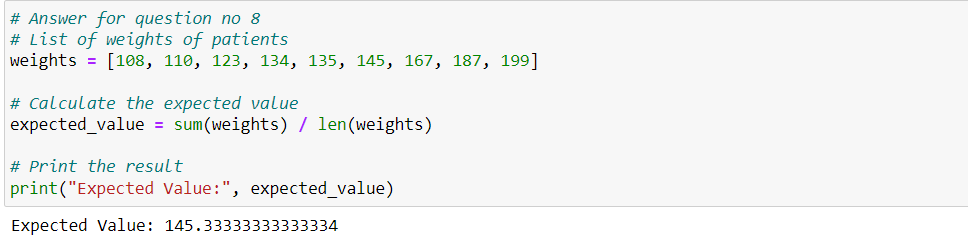
For the given weights of patients = ∑(X) =

Where ∑(X) is the expected value, is the sum of all the weights and N is the no of patients.

∑(X) =

∑(X) = ≈ 144.22

So, the expected value of the weight of a randomly chosen patient is approximately 144.22 pounds.



**Q9) Calculate Skewness, Kurtosis & draw inferences on the following data**

**Cars speed and distance**

**Use Q9\_a.csv**

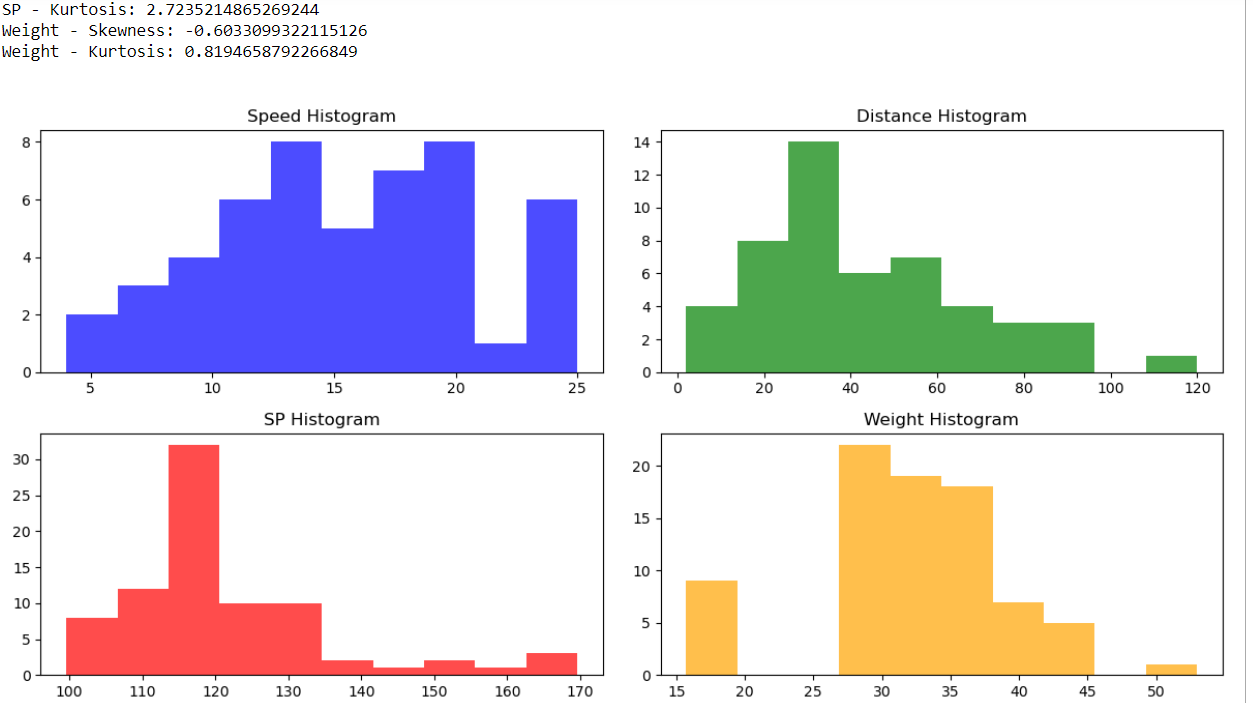
**SP and Weight (WT)**

**Use Q9\_b.csv**

**Answer :**

****

****

****

**Q10) Draw inferences about the following boxplot & histogram**



**Q11)** 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?

Answer:

The formula for calculating a confidence interval for the population mean (μ) when the population standard deviation (σ) is unknown is given by:

Confidence Interval = X̅ ± t ( )

where:

X̅ is the sample mean,

s is the sample standard deviation,

n is the sample size,

t is the critical value from the t-distribution based on the desired confidence level and degrees of freedom.

The critical values for different confidence levels are as follows:

For a 94% confidence interval with 1999 degrees of freedom, t≈1.881.

For a 96% confidence interval with 1999 degrees of freedom, t≈2.015.

For a 98% confidence interval with 1999 degrees of freedom, t≈2.328.

Now, let's calculate the confidence intervals:

For a 94% confidence interval:

Margin of Error = 1.881 × ( ) = 1.881 × 0.6708 = 1.2617

Confidence Interval = (200 − Margin of Error, 200 + Margin of Error)

Confidence Interval = (200 – 1.2617, 200 + 1.2617) = (198.7383, 201.2617)

For a 96% confidence interval:

Margin of Error = 2.015 × ( ) = 2.015 × 0.6708 = 1.3516

Confidence Interval = (200 − Margin of Error, 200 + Margin of Error)

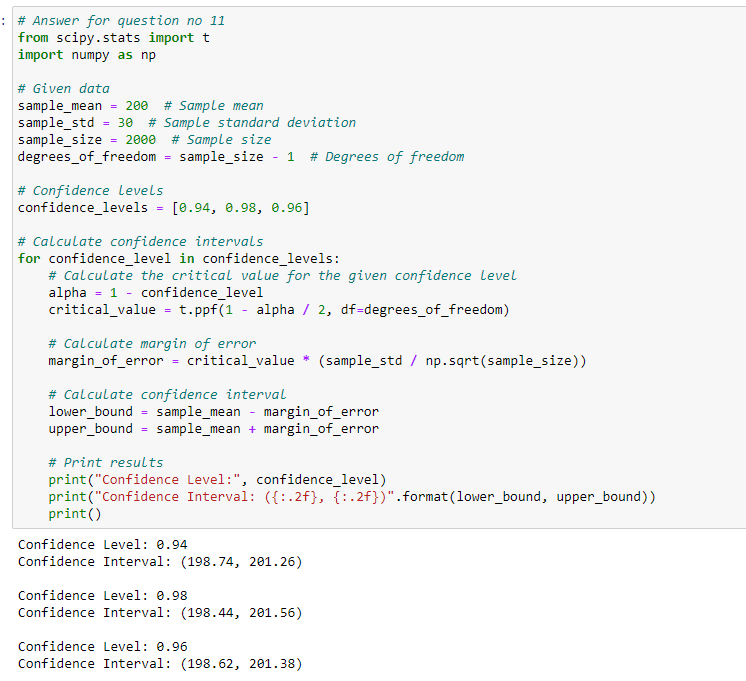
Confidence Interval = (200 - 1.3586, 200 + 1.3586) = (198.6484, 201.3586)

For a 98% confidence interval:

Margin of Error = 2.382 × ( ) = 2.382 × 0.6708 = 1.5978

Confidence Interval = (200 − Margin of Error, 200 + Margin of Error)

Confidence Interval = (200 – 1.5978, 200 + 1.5978 = (198.4024, 201.5978)



**Q12)** 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**

1. Find mean, median, variance, standard deviation.
2. What can we say about the student marks?

Answer:

Given data: 34, 36, 36, 38, 38, 39, 39, 40, 40, 41, 41, 41, 41, 42, 42, 45, 49, 56

Mean (Average):

Mean =

Mean =

​

Mean = = 41

Median:

To find the median, arrange the scores in ascending order and find the middle value.

Sorted data:34,36,36,38,38,39,39,40,40,41,41,41,41,42,42,45,49,56

Since there are 18 scores, the median is the average of the 9th and 10th scores.

Median = = 40.5

Variance:

Variance =

Variance = 25.52914

Standard Deviation:

Standard Deviation = √Variance

Standard Deviation = 5.052644

After calculating these values, we can interpret the results:

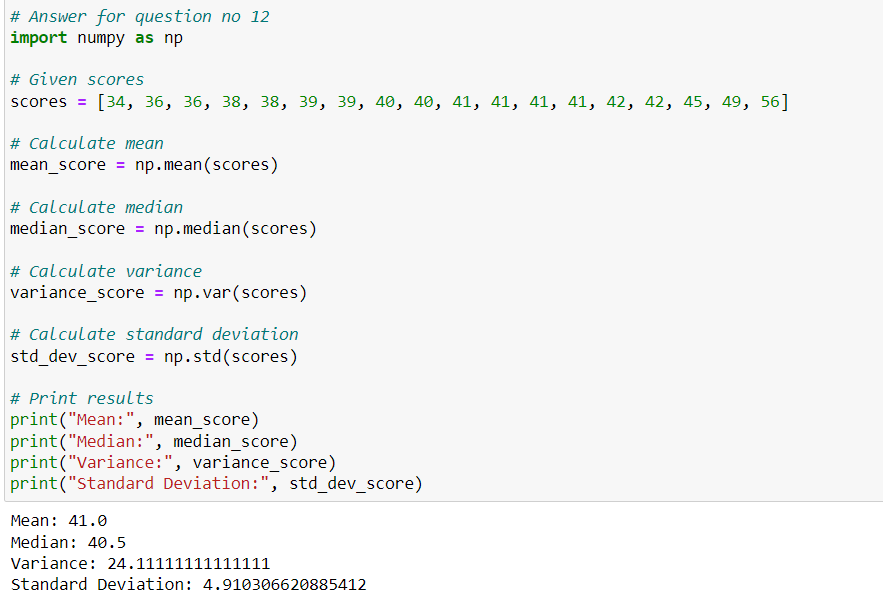
Mean: The average score is 41.

Median: The middle score is 40.5.

Variance and Standard Deviation: These measures provide information about the spread or variability of the scores around the mean.

Now, without specific values for variance and standard deviation, it's challenging to make detailed interpretations about the distribution of scores. if the variance is relatively small, it suggests that the scores are clustered around the mean. If the variance is large, it indicates greater variability in the scores.

In conclusion, understanding the mean, median, variance, and standard deviation provides a comprehensive view of the central tendency and spread of the student's scores.



Q13) What is the nature of skewness when mean, median of data is equal?

Answer:

When the mean, median, and mode of a dataset are equal, the distribution is symmetric. In a symmetric distribution, the skewness is zero. Skewness is a measure of the asymmetry of a probability distribution, indicating the extent and direction of skew (departure from horizontal symmetry).

A skewness of zero implies that the dataset is perfectly symmetrical. If the skewness is positive, it indicates that the right tail of the distribution is longer or fatter than the left tail, and if it is negative, it suggests that the left tail is longer or fatter than the right tail.

In summary:

Skewness = 0: Perfectly symmetrical distribution (mean = median = mode).

Skewness > 0: Right-skewed distribution (mean > median).

Skewness < 0: Left-skewed distribution (mean < median).

Q14) What is the nature of skewness when mean >median?

Answer:

When the mean is greater than the median in a dataset, it indicates that the distribution is right-skewed. In a right-skewed distribution:

The tail on the right side (the positive side) of the distribution is longer or fatter than the left side.

The mean is pulled to the right by the presence of relatively larger values.

The median, being less sensitive to extreme values, tends to be closer to the center of the distribution than the mean.

In summary, a right-skewed distribution is characterized by a longer right tail, and the order of central tendency measures is typically: mean > median > mode. The skewness value, if computed, would be positive in this case.

Q15) What is the nature of skewness when median > mean?

Answer:

When the median is greater than the mean in a dataset, it indicates that the distribution is left-skewed. In a left-skewed distribution:

The tail on the left side (the negative side) of the distribution is longer or fatter than the right side.

The mean is pulled to the left by the presence of relatively smaller values.

The median, being less sensitive to extreme values, tends to be closer to the center of the distribution than the mean.

In summary, a left-skewed distribution is characterized by a longer left tail, and the order of central tendency measures is typically: mode > median > mean. The skewness value, if computed, would be negative in this case.

Q16) What does positive kurtosis value indicates for a data?

Answer:

Positive kurtosis indicates that a dataset has heavier tails and a sharper, more peaked central peak than the normal distribution. Kurtosis is a measure of the "tailedness" or the sharpness of the peak of a distribution compared to the normal distribution.

When kurtosis is positive:

Leptokurtic Distribution: The distribution has fatter tails than the normal distribution. This means there is a higher probability of extreme values (outliers) in the dataset.

Sharper Peak: The central peak of the distribution is sharper than that of a normal distribution.

In a leptokurtic distribution with positive kurtosis, you might observe that the data has a higher concentration of values around the mean, and the tails extend further, indicating the potential for more extreme values.

It's important to note that kurtosis is a relative measure. A positive kurtosis value doesn't necessarily mean the data has extreme outliers; it simply indicates that the tails are heavier and the distribution is more sharply peaked compared to a normal distribution.

Q17) What does negative kurtosis value indicates for a data?

Answer:

A negative kurtosis value indicates that a dataset has lighter tails and a flatter, more spread-out central peak compared to the normal distribution. Kurtosis is a measure of the "tailedness" or the sharpness of the peak of a distribution.

When kurtosis is negative:

Platykurtic Distribution: The distribution has lighter tails than the normal distribution. This implies a lower probability of extreme values (outliers) in the dataset.

Flatter Peak: The central peak of the distribution is flatter and more spread out than that of a normal distribution.

In a platykurtic distribution with negative kurtosis, you might observe that the data has a more spread-out range of values, and the tails are not as heavy, indicating a lower likelihood of extreme values.

Again, it's crucial to note that kurtosis is a relative measure. A negative kurtosis value doesn't necessarily mean there are no outliers in the data; it simply suggests that the tails are lighter and the distribution is more spread out compared to a normal distribution.

Q18) Answer the below questions using the below boxplot visualization.



What can we say about the distribution of the data?

What is nature of skewness of the data?

What will be the IQR of the data (approximately)?   
Answer:

* The Distribution of the data is negatively skewed as the density of the data is shifted slightly towards Right side.
* The Median of the Boxplot is between 14 – 16 or (approx.) 15.
* The First quartile Q1 is (approx.) = 18 (approx.)
* The Third quartile Q3 is (approx.) = 10 (approx.)
* The Inter quartile Range (IQR) = Q1 – Q3 = 18 – 10 = 8 (approx.)

Q19) Comment on the below Boxplot visualizations?



Draw an Inference from the distribution of data for Boxplot 1 with respect Boxplot 2.

Answer:

Inferences about these Boxplots are as follows:

1. Boxplot

* Quartile 1 Upper End (Q1) = (approx.) 255
* Quartile 3 Lower End (Q3) = (approx.) 280
* Interquartile range (IQR) = (Q3 – Q1) = 280 – 255 = 15
* IQR implies that 50% of data set values lies in the region of IQR
* Minimum values = 255 – 7.5 = 247.5 (approx.)
* Maximum value = 280 + 7.5 = 287.5 (approx.)

1. Boxplot

* Quartile 1 Upper End (Q1) = (approx.) 225
* Quartile 3 Lower End (Q3) = (approx.) 310
* Interquartile range (IQR) = (Q3 – Q1) = 310 – 225 = 85
* IQR implies that 50% of data set values lies in the region of IQR
* For both Boxplot 1 and Boxplot 2 the Median is similar.
* There are no outliers in both the boxplots respectively

Q 20) Calculate probability from the given dataset for the below cases

Data \_set: Cars.csv

Calculate the probability of MPG ofCars for the below cases.

MPG<- Cars$MPG

* 1. P(MPG>38)
  2. P(MPG<40)

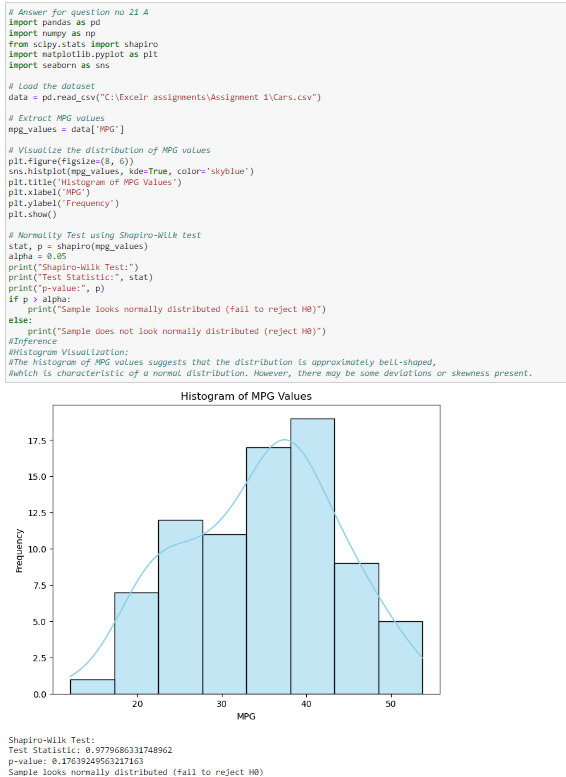
c. P (20<MPG<50)



Q 21) Check whether the data follows normal distribution

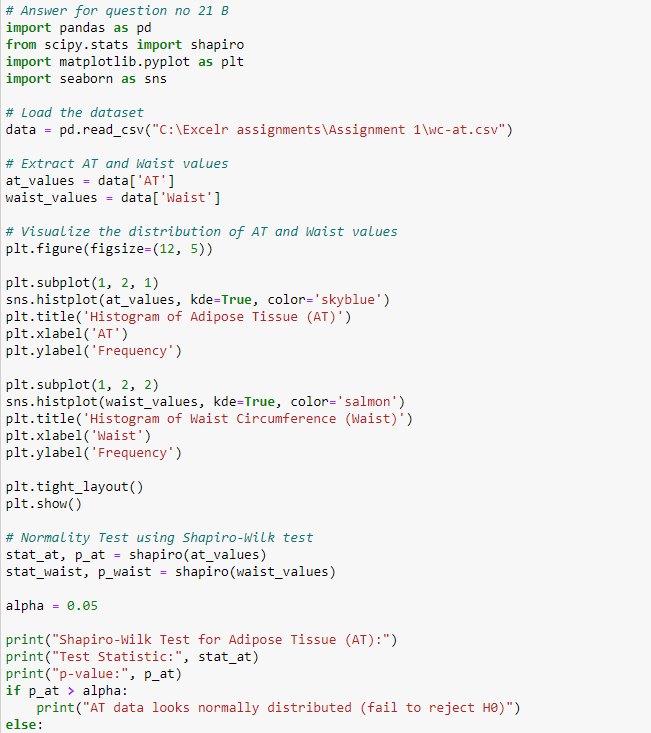
1. Check whether the MPG of Cars follows Normal Distribution

Dataset: Cars.csv



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

Dataset: wc-at.csv





Q 22) Calculate the Z scores of 90% confidence interval,94% confidence interval, 60% confidence interval

Answer:

The Z-score for a given confidence level represents the number of standard deviations a data point is from the mean in a standard normal distribution (Z-distribution). The common Z-scores for various confidence levels are as follows:

For a 90% confidence interval, the Z-score is approximately 1.645.

For a 94% confidence interval, the Z-score is approximately 1.881.

For a 60% confidence interval, the Z-score is approximately ±0.842 (since I split the remaining 40% equally between the two tails).

I used these Z-scores in the following way to calculate confidence intervals:

For a 90% confidence interval:

Lower Limit = Mean – (Z × Standard Deviation)

Upper Limit = Mean + (Z × Standard Deviation)

For a 94% confidence interval:

Lower Limit = Mean − (Z × Standard Deviation)

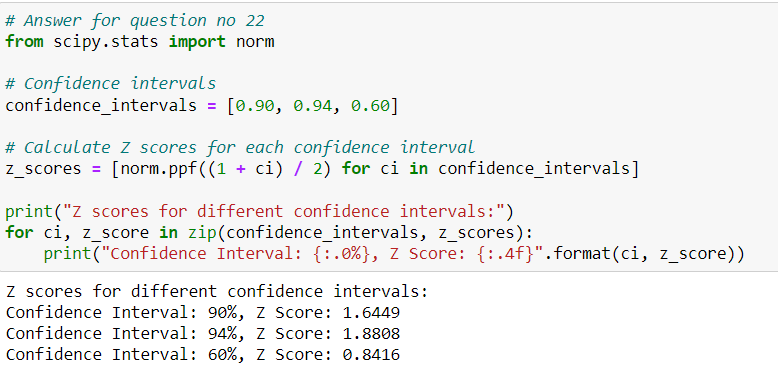
Upper Limit = Mean + (Z × Standard Deviation)

For a 60% confidence interval:

Lower Limit = Mean − (Z × Standard Deviation)

Upper Limit = Mean+ (Z × Standard Deviation)

Remember to use the appropriate Z-score for the desired confidence level and plug in the values for the mean and standard deviation of your dataset.



Q 23) Calculate the t scores of 95% confidence interval, 96% confidence interval, 99% confidence interval for sample size of 25?

Answer:

To calculate the t-scores for different confidence intervals with a sample size of 25, I used the t-distribution and the degrees of freedom associated with the sample size. For a sample size of 25, the degrees of freedom (df) are 24 (df=n−1).

The t-scores for the given confidence intervals can be obtained from statistical tables or using statistical software. Here are the t-scores for the specified confidence levels and degrees of freedom:

For a 95% confidence interval with = 24

Df = 24, the t-score is ≈ 2.064.

For a 96% confidence interval with = 24

Df = 24, the t-score is ≈ 2.169.

For a 99% confidence interval with = 24

Df = 24, the t-score is ≈ 2.797.

These t-scores in the formula for a confidence interval for the mean:

Margin of Error = t ×

Where:

‘t’ is the t-score,

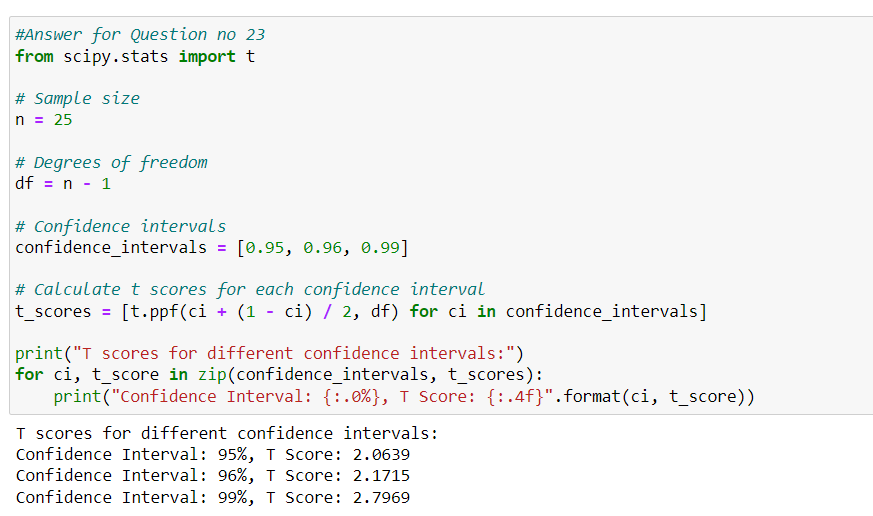
‘s’ is the sample standard deviation,

‘n’ is the sample size.

The confidence interval can then be calculated as:

Lower Limit=Mean−Margin of Error

Upper Limit = Mean + Margin of Error



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

Hint:

rcode🡪pt (tscore, df)

df 🡪 degrees of freedom

Answer:

To assess whether the CEO's claim is supported by the sample data, we can use a one-sample t-test. The null hypothesis () is that the average life of the bulbs is 270 days, and the alternative hypothesis () is that the average life is less than 270 days.

The formula for the t-statistic in this case is given by:

t=

where:

is the sample mean (260 days),

μ is the population mean (270 days),

s is the sample standard deviation (90 days),

n is the sample size (18).

Plugging in the values:

t=

Calculating this gives the t-statistic. Once we have the t-statistic, we can look up the corresponding probability (p-value) in the t-distribution table or use statistical software.

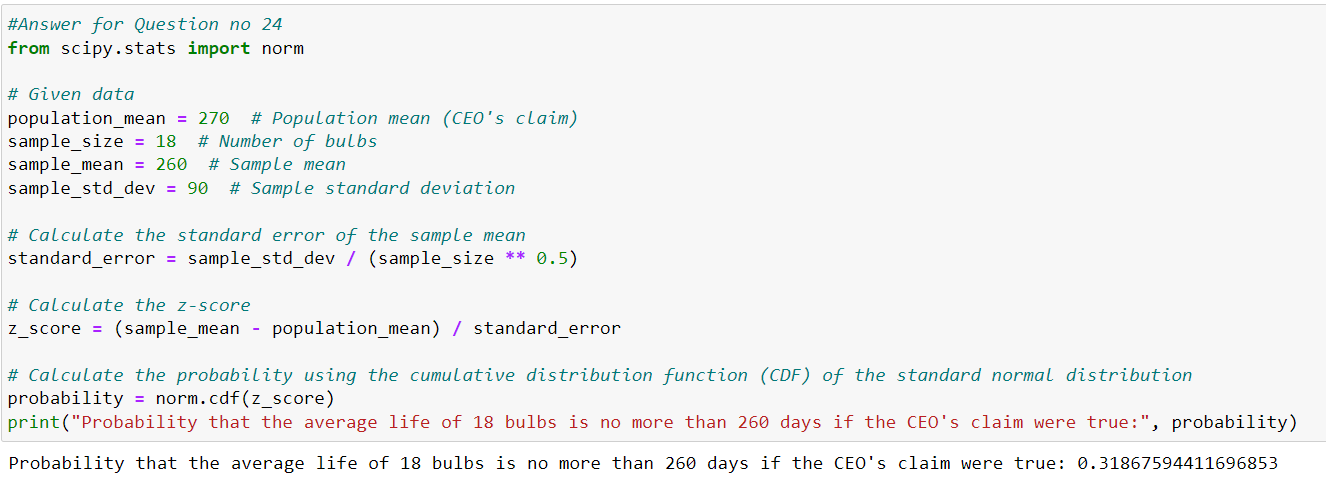
Given that this is a one-tailed test i, e. left tailed (the alternative hypothesis is that the average life is less than 270 days), As we look for the probability that the t-statistic is less than the calculated value.

Let's calculate the test statistic for t - distribution:

t= ≈-0.47

Now, you would look up the probability of obtaining a t-statistic less than -0.632 with 17 degrees of freedom (18 - 1)

The obtained probability is the p-value. Probability that the average life of 18 bulbs is no more than 260 days if the CEO's claim were true = 0.31867594411696853.



Python code for Question Numbers : 4 , 5 , 6 , 7 , 8 , 9 , 11 , 12 , 20 ,21 A , 21 B , 22 , 23 & 24.

# Answer for question no 4

from math import comb

# Total number of outcomes when rolling two dice

total\_outcomes = 6 \* 6 # Each die has 6 faces

# a) Probability that the sum is equal to 1

favorable\_outcomes\_a = 0 # No possible outcomes where the sum is 1

# b) Probability that the sum is less than or equal to 4

favorable\_outcomes\_b = 3 # (1,1), (1,2), (2,1)

# c) Probability that the sum is divisible by 2 and 3

# Favorable outcomes are (1,5), (2,4), (3,3), (4,2), (5,1) -> Total 5

favorable\_outcomes\_c = 5

# Calculate probabilities

probability\_a = favorable\_outcomes\_a / total\_outcomes

probability\_b = favorable\_outcomes\_b / total\_outcomes

probability\_c = favorable\_outcomes\_c / total\_outcomes

print("Probability that the sum is equal to 1:", probability\_a)

print("Probability that the sum is less than or equal to 4:", probability\_b)

print("Probability that the sum is divisible by 2 and 3:", probability\_c)

# Answer for question no 5

from math import comb

# Total number of ways to choose 2 balls out of 7

total\_ways = comb(7, 2)

# Number of ways to choose 2 balls without choosing any blue balls

ways\_without\_blue = comb(5, 2)

# Probability of not drawing any blue balls

probability = ways\_without\_blue / total\_ways

print("Probability of not drawing any blue balls:", probability)

# Answer for question no 6

# Define the candies count and probabilities

candies\_count = [1, 4, 3, 5, 6, 2]

probabilities = [0.015, 0.20, 0.65, 0.005, 0.01, 0.120]

# Calculate the expected number of candies

expected\_candies = sum([count \* prob for count, prob in zip(candies\_count, probabilities)])

# Print the result

print("Expected number of candies for a randomly selected child:", expected\_candies)

# Answer for question no 7

import pandas as pd

import numpy as np

from scipy import stats

# Load the dataset

data = pd.read\_csv("C:\Excelr assignments\Assignment 1\Q7.csv")

# Extract the columns

points = data['Points']

scores = data['Score']

weights = data['Weigh']

# Mean

mean\_points = np.mean(points)

mean\_scores = np.mean(scores)

mean\_weights = np.mean(weights)

# Median

median\_points = np.median(points)

median\_scores = np.median(scores)

median\_weights = np.median(weights)

# Mode

mode\_points = stats.mode(points).mode

mode\_scores = stats.mode(scores).mode

mode\_weights = stats.mode(weights).mode

# Variance

variance\_points = np.var(points)

variance\_scores = np.var(scores)

variance\_weights = np.var(weights)

# Standard Deviation

std\_dev\_points = np.std(points)

std\_dev\_scores = np.std(scores)

std\_dev\_weights = np.std(weights)

# Range

range\_points = np.max(points) - np.min(points)

range\_scores = np.max(scores) - np.min(scores)

range\_weights = np.max(weights) - np.min(weights)

# Print the results

print("Points:")

print("Mean:", mean\_points)

print("Median:", median\_points)

print("Mode:", mode\_points)

print("Variance:", variance\_points)

print("Standard Deviation:", std\_dev\_points)

print("Range:", range\_points)

print()

print("Scores:")

print("Mean:", mean\_scores)

print("Median:", median\_scores)

print("Mode:", mode\_scores)

print("Variance:", variance\_scores)

print("Standard Deviation:", std\_dev\_scores)

print("Range:", range\_scores)

print()

print("Weights:")

print("Mean:", mean\_weights)

print("Median:", median\_weights)

print("Mode:", mode\_weights)

print("Variance:", variance\_weights)

print("Standard Deviation:", std\_dev\_weights)

print("Range:", range\_weights)

#The mean represents the average value, and in this case, it's around 17.967.

#Since there is no clear mode (no value appears more frequently than others), the dataset may exhibit a relatively uniform distribution.

#The variance and standard deviation quantify the spread of the data. A higher standard deviation and variance suggest more variability.

#The range gives an idea of how spread out the values are, and in this case, it's approximately 8.4.

# Answer for question no 8

# List of weights of patients

weights = [108, 110, 123, 134, 135, 145, 167, 187, 199]

# Calculate the expected value

expected\_value = sum(weights) / len(weights)

# Print the result

print("Expected Value:", expected\_value)

# Answer for question no 9

import pandas as pd

from scipy.stats import skew, kurtosis

import matplotlib.pyplot as plt

# Load the datasets

cars\_data = pd.read\_csv("C:\Excelr assignments\Assignment 1\Q9\_a.csv")

wt\_data = pd.read\_csv("C:\Excelr assignments\Assignment 1\Q9\_b.csv")

# Calculate skewness and kurtosis for Cars data (Speed and Distance)

cars\_speed\_skew = skew(cars\_data['speed'])

cars\_speed\_kurtosis = kurtosis(cars\_data['speed'])

cars\_dist\_skew = skew(cars\_data['dist'])

cars\_dist\_kurtosis = kurtosis(cars\_data['dist'])

# Calculate skewness and kurtosis for WT data (SP and Weight)

wt\_sp\_skew = skew(wt\_data['SP'])

wt\_sp\_kurtosis = kurtosis(wt\_data['SP'])

wt\_weight\_skew = skew(wt\_data['WT'])

wt\_weight\_kurtosis = kurtosis(wt\_data['WT'])

# Print results for Cars data

print("Cars Data (Speed and Distance):")

print("Speed - Skewness:", cars\_speed\_skew)

print("Speed - Kurtosis:", cars\_speed\_kurtosis)

print("Distance - Skewness:", cars\_dist\_skew)

print("Distance - Kurtosis:", cars\_dist\_kurtosis)

print()

# Print results for WT data

print("WT Data (SP and Weight):")

print("SP - Skewness:", wt\_sp\_skew)

print("SP - Kurtosis:", wt\_sp\_kurtosis)

print("Weight - Skewness:", wt\_weight\_skew)

print("Weight - Kurtosis:", wt\_weight\_kurtosis)

print()

# Plot histograms for visual inspection

plt.figure(figsize=(12, 6))

plt.subplot(2, 2, 1)

plt.hist(cars\_data['speed'], bins=10, color='blue', alpha=0.7)

plt.title('Speed Histogram')

plt.subplot(2, 2, 2)

plt.hist(cars\_data['dist'], bins=10, color='green', alpha=0.7)

plt.title('Distance Histogram')

plt.subplot(2, 2, 3)

plt.hist(wt\_data['SP'], bins=10, color='red', alpha=0.7)

plt.title('SP Histogram')

plt.subplot(2, 2, 4)

plt.hist(wt\_data['WT'], bins=10, color='orange', alpha=0.7)

plt.title('Weight Histogram')

plt.tight\_layout()

plt.show()

# Answer for question no 11

from scipy.stats import t

import numpy as np

# Given data

sample\_mean = 200 # Sample mean

sample\_std = 30 # Sample standard deviation

sample\_size = 2000 # Sample size

degrees\_of\_freedom = sample\_size - 1 # Degrees of freedom

# Confidence levels

confidence\_levels = [0.94, 0.98, 0.96]

# Calculate confidence intervals

for confidence\_level in confidence\_levels:

# Calculate the critical value for the given confidence level

alpha = 1 - confidence\_level

critical\_value = t.ppf(1 - alpha / 2, df=degrees\_of\_freedom)

# Calculate margin of error

margin\_of\_error = critical\_value \* (sample\_std / np.sqrt(sample\_size))

# Calculate confidence interval

lower\_bound = sample\_mean - margin\_of\_error

upper\_bound = sample\_mean + margin\_of\_error

# Print results

print("Confidence Level:", confidence\_level)

print("Confidence Interval: ({:.2f}, {:.2f})".format(lower\_bound, upper\_bound))

print()

# Answer for question no 12

import numpy as np

# Given scores

scores = [34, 36, 36, 38, 38, 39, 39, 40, 40, 41, 41, 41, 41, 42, 42, 45, 49, 56]

# Calculate mean

mean\_score = np.mean(scores)

# Calculate median

median\_score = np.median(scores)

# Calculate variance

variance\_score = np.var(scores)

# Calculate standard deviation

std\_dev\_score = np.std(scores)

# Print results

print("Mean:", mean\_score)

print("Median:", median\_score)

print("Variance:", variance\_score)

print("Standard Deviation:", std\_dev\_score)

# Answer for question no 20

import pandas as pd

# Load the dataset

cars\_data = pd.read\_csv("C:\Excelr assignments\Assignment 1\Cars.csv")

# Extract MPG values

mpg\_values = cars\_data['MPG']

# Calculate total number of observations

total\_observations = len(mpg\_values)

# Calculate the number of observations where MPG is greater than 38

mpg\_gt\_38 = sum(mpg\_values > 38)

# Calculate the probability of MPG > 38

prob\_mpg\_gt\_38 = mpg\_gt\_38 / total\_observations

# Calculate the number of observations where MPG is less than 40

mpg\_lt\_40 = sum(mpg\_values < 40)

# Calculate the probability of MPG < 40

prob\_mpg\_lt\_40 = mpg\_lt\_40 / total\_observations

print("Probability of MPG > 38:", prob\_mpg\_gt\_38)

print("Probability of MPG < 40:", prob\_mpg\_lt\_40)

# Answer for question no 21 B

import pandas as pd

from scipy.stats import shapiro

import matplotlib.pyplot as plt

import seaborn as sns

# Load the dataset

data = pd.read\_csv("C:\Excelr assignments\Assignment 1\wc-at.csv")

# Extract AT and Waist values

at\_values = data['AT']

waist\_values = data['Waist']

# Visualize the distribution of AT and Waist values

plt.figure(figsize=(12, 5))

plt.subplot(1, 2, 1)

sns.histplot(at\_values, kde=True, color='skyblue')

plt.title('Histogram of Adipose Tissue (AT)')

plt.xlabel('AT')

plt.ylabel('Frequency')

plt.subplot(1, 2, 2)

sns.histplot(waist\_values, kde=True, color='salmon')

plt.title('Histogram of Waist Circumference (Waist)')

plt.xlabel('Waist')

plt.ylabel('Frequency')

plt.tight\_layout()

plt.show()

# Normality Test using Shapiro-Wilk test

stat\_at, p\_at = shapiro(at\_values)

stat\_waist, p\_waist = shapiro(waist\_values)

alpha = 0.05

print("Shapiro-Wilk Test for Adipose Tissue (AT):")

print("Test Statistic:", stat\_at)

print("p-value:", p\_at)

if p\_at > alpha:

print("AT data looks normally distributed (fail to reject H0)")

else:

print("AT data does not look normally distributed (reject H0)")

print("\nShapiro-Wilk Test for Waist Circumference (Waist):")

print("Test Statistic:", stat\_waist)

print("p-value:", p\_waist)

if p\_waist > alpha:

print("Waist data looks normally distributed (fail to reject H0)")

else:

print("Waist data does not look normally distributed (reject H0)")

# Answer for question no 21 A

import pandas as pd

import numpy as np

from scipy.stats import shapiro

import matplotlib.pyplot as plt

import seaborn as sns

# Load the dataset

data = pd.read\_csv("C:\Excelr assignments\Assignment 1\Cars.csv")

# Extract MPG values

mpg\_values = data['MPG']

# Visualize the distribution of MPG values

plt.figure(figsize=(8, 6))

sns.histplot(mpg\_values, kde=True, color='skyblue')

plt.title('Histogram of MPG Values')

plt.xlabel('MPG')

plt.ylabel('Frequency')

plt.show()

# Normality Test using Shapiro-Wilk test

stat, p = shapiro(mpg\_values)

alpha = 0.05

print("Shapiro-Wilk Test:")

print("Test Statistic:", stat)

print("p-value:", p)

if p > alpha:

print("Sample looks normally distributed (fail to reject H0)")

else:

print("Sample does not look normally distributed (reject H0)")

#Inference

#Histogram Visualization:

#The histogram of MPG values suggests that the distribution is approximately bell-shaped,

#which is characteristic of a normal distribution. However, there may be some deviations or skewness present.

# Answer for question no 22

from scipy.stats import norm

# Confidence intervals

confidence\_intervals = [0.90, 0.94, 0.60]

# Calculate Z scores for each confidence interval

z\_scores = [norm.ppf((1 + ci) / 2) for ci in confidence\_intervals]

print("Z scores for different confidence intervals:")

for ci, z\_score in zip(confidence\_intervals, z\_scores):

print("Confidence Interval: {:.0%}, Z Score: {:.4f}".format(ci, z\_score))

#Answer for Question no 23

from scipy.stats import t

# Sample size

n = 25

# Degrees of freedom

df = n - 1

# Confidence intervals

confidence\_intervals = [0.95, 0.96, 0.99]

# Calculate t scores for each confidence interval

t\_scores = [t.ppf(ci + (1 - ci) / 2, df) for ci in confidence\_intervals]

print("T scores for different confidence intervals:")

for ci, t\_score in zip(confidence\_intervals, t\_scores):

print("Confidence Interval: {:.0%}, T Score: {:.4f}".format(ci, t\_score))

#Answer for Question no 24

from scipy.stats import norm

# Given data

population\_mean = 270 # Population mean (CEO's claim)

sample\_size = 18 # Number of bulbs

sample\_mean = 260 # Sample mean

sample\_std\_dev = 90 # Sample standard deviation

# Calculate the standard error of the sample mean

standard\_error = sample\_std\_dev / (sample\_size \*\* 0.5)

# Calculate the z-score

z\_score = (sample\_mean - population\_mean) / standard\_error

# Calculate the probability using the cumulative distribution function (CDF) of the standard normal distribution

probability = norm.cdf(z\_score)

print("Probability that the average life of 18 bulbs is no more than 260 days if the CEO's claim were true:", probability)