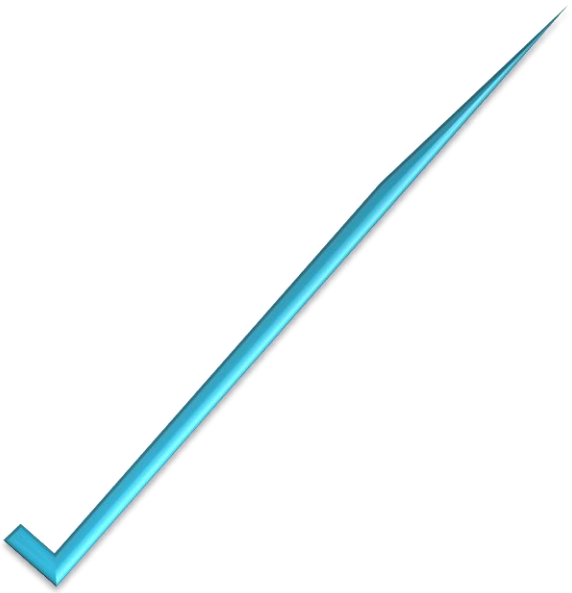
Major Assignment

Data Science “Statistics”

NAME: Md Zeeshan Rasheed



Submission date: 15-08-2023

1. According to a study, the daily average time spent by a user on a social media website is 50 minutes. To test the claim of this study, Ramesh, a researcher, takes a sample of 25 website users and finds out that the mean time spent by the sample users is 60 minutes and the sample standard deviation is 30 minutes. Based on this information, the null and the alternative hypotheses will be: Ho = The average time spent by the users is 50 minutes H1 = The average time spent by the users is not 50 minutes Use a 5% significance level to test this hypothesis.

In [1]:
```
import math
import scipy.stats as stats

n=25
mu=50
x_bar=60
s=30
alpha=0.05

# Calculate the t-statistic
t=(x_bar-mu)
k=(s/math.sqrt(n))
z=t/k


# Find the critical t-value

df=n-1
t_crit= abs(stats.t.ppf(alpha/2, df))


# Check if the calculated t-value is outside the critical region

if abs(z)>t_crit:
    print("The average time spent by the user is not 50 minutes")

else:
    print("The average time spent by the user is 50 minutes")
```

```
The average time spent by the user is 50 minutes
```

Sol1. The average time spent by the users is 50 minutes

2. Height of 7 students (in cm) is given below. What is the median? 168 170 169 160 162 164 162.

In [2]:
```
l_median=[160,162,162,164,168,169,170]
```

In [3]:
```
import numpy as np
np.median(l_median)
```

Out[3]:
```
164.0
```

Sol2. Median is 164

3. Below are the observations of the marks of a student. Find the value of mode. 84 85 89 92 93 89 87 89 92

In [4]:
```
data=[ 84,85,89,92,93,89,87,89,92]

mode=stats.mode(data)
mode
```

Out[4]:
```
ModeResult(mode=array([89]), count=array([3]))
```

Sol3. The average time spent by the users is 50 minutes

4. From the table given below, what is the mean of marks obtained by 20 students? Marks Xi: 3, 4, 5, 6 ,7 ,8 ,9 ,10

No. of Students fi: 1, 2, 2, 4, 5, 3, 2, 1

In [5]:
```
data={"Marks" : [3,4,5,6,7,8,9,10],
     "No. of students" : [1,2,2,4,5,3,2,1]}
```

In [6]:
```
import pandas as pd
```

In [7]:
```
df=pd.DataFrame(data)
```

In [8]:
```
df["No. of students"].sum()
```

Out[8]:
```
20
```

In [9]:
```
df["xiFi"]=df["Marks"]*df["No. of students"]
```

In [10]:
```
df
```

Out[10]:

	Marks	No. of students	xiFi
0	3	1	3
1	4	2	8
2	5	2	10
3	6	4	24
4	7	5	35
5	8	3	24
6	9	2	18
7	10	1	10

In [11]:
```
df["xiFi"].mean()
```

Out[11]:
```
16.5
```

Sol4. Mean is 16.5

5. For a certain type of computer, the length of time between charges of the battery is normally distributed with a mean of 50 hours and a standard deviation of 15 hours. John owns one of these computers and wants to know the probability that the length of time will be between 50 and 70 hours.

Sol5. Z – Score = (X – μ) / σ

Where, X is a normal random variable
μis the average or the mean
σis the standard deviation
For X = 50:
Z1 = (50 - 50) / 15 = 0
For X = 70:
Z2 = (70 - 50) / 15 = 1.33333…
The probability that Z is between 0 and 1.333… is:
P(0 ≤ Z ≤ 1.33)
0.5000 ≤ Z ≤ 0.9082 Therefore, the probability that the length of time between charges computer battery will be between 50 and 70 hours is 0.4082 or approximately 40.82%.

6. Find the range of the following. g = [10, 23, 12, 21, 14, 17, 16, 11, 15, 19]

Sol6. Range = Highest – Lowest Range = 23 – 10 Range = 13

7. It is estimated that 50% of emails are spam emails. Some software has been applied to filter these spam emails before they reach your inbox. A certain brand of software claims that it can detect 99% of spam emails, and the probability for a false positive (a non-spam 2 email detected as spam) is 5%. Now if an email is detected as spam, then what is the probability that it is in fact a non-spam email?

Sol7. P(spam) = 50% or 0.50
P(not a spam) = 50% or 0.50
P(positive|spam) = 99% or 0.99
P(positive|not a spam) = 5% or 0.05
We want to calculate the probability that an email is a non-spam email (NS) given that the software detected it as spam (F). This is written as P(not a spam|positive).
P(A|B) = P(A) * P(B|A) / P(B)
P(not a spam|positive) = P(not a spam) * P(positive|not a spam) / P(positive)
We have to find P(positive)
Therefore, P(positive) = P(positive|spam) * P(spam) + P(positive|not a spam) * P(not a spam)
= 0.99 * 0.50 + 0.05 * 0.50
= 0.52
Now, P(not a spam|positive) = P(not a spam) * P(positive|not a spam) / P(positive)
= 0.50 * 0.05 / 0.52
= 0.048
The probability that an email detected as spam by the software is actually a non-spam email is 0.048 or approximately 4.8%. The software claims to have a 99% detection rate for spam emails, but there is still a chance that it will misclassify a non-spam email as spam.

8. Given the following distribution of returns, determine the lower quartile: {10 25 12 21 19 17 16 11 15 19}.

In [12]:
```
qr=[10,25,12,21,19,17,16,11,15,19]
```

In [13]:
```
list.sort(qr)
qr
```

Out[13]:
```
[10, 11, 12, 15, 16, 17, 19, 19, 21, 25]
```

In [14]:
```
np.quantile(qr,[0,0.25,0.5,0.75,1])
```

Out[14]:
```
array([10.  , 12.75, 16.5 , 19.  , 25.  ])
```

Sol8. Lower Quartile 12.75

9. For a Binomial distribution, the number of trials(n) is 25, and the probability of success is 0.3. What's the variability of the distribution?

Sol9. Probability of success = 0.3
Probability of failure = 1 - Probability of success
= 1 – 03
= 0.7 Variance = trials * Probability of success * Probability of failure
= 25 * 0.3 * 0.7
= 5.25
Standard Deviation = $\sqrt{Variance}$
= $\sqrt{}$**5.25**
= 2.29

10. Download the Cell Phone Survey Dataset and perform the below mentioned operations on the dataset

- **Checking datatypes of each column in the dataset**

In [15]:
```
import statistics as sst
```

In [16]:
```
df1=pd.read_csv("E:\Cell Phone Survey.csv")
```

In [17]:
```
df1.head()
```

Out[17]:

	Gender	Carrier	Type	Usage	Signal strength	Value for the Dollar	Customer Service
0	M	AT&T	Smart	High	5	4	4
1	M	AT&T	Smart	High	5	4	2
2	M	AT&T	Smart	Average	4	4	4
3	M	AT&T	Smart	Very high	2	3	3
4	M	AT&T	Smart	Very high	5	5	2

In [18]:
```
# Checking datatypes of each columns in the dataset.
df1.dtypes
```

Out[18]:
```
Gender                  object
Carrier                 object
Type                    object
Usage                   object
Signal strength          int64
Value for the Dollar     int64
Customer Service         int64
dtype: object
```

- **Find Mean of Signal strength column using Pandas and Statistics library**

In [19]:
```
print("The mean using pandas function for signal Strength is ", round(df1["Signal strength"].mean(),2))
```

```
The mean using pandas function for signal Strength is  3.31
```

In [20]:
```
print("The mean using pandas function for signal Strength is ", round(sst.mean(df1["Signal strength"]))
```

```
The mean using pandas function for signal Strength is  3
```

- **Find Median of Customer Service column using Pandas and Statistics library**

In [21]:
```
print("The mean using pandas function for signal Strength is ", round(df1["Customer Service"].median(),
```

```
The mean using pandas function for signal Strength is  3.0
```

In [22]:
```
print("The mean using pandas function for signal Strength is ", round(sst.median(df1["Customer Service"
```

```
The mean using pandas function for signal Strength is  3
```

- **Find Mode of Signal strength column using Pandas and Statistics library**

In [23]:
```
print("The mean using pandas function for signal Strength is ", round(df1["Signal strength"].mode(),2))
```

```
The mean using pandas function for signal Strength is  0     3
Name: Signal strength, dtype: int64
```

In [24]:
```
print("The mean using pandas function for signal Strength is ", round(sst.mode(df1["Signal strength"]))
```

```
The mean using pandas function for signal Strength is  3
```

- **Find Standard deviation of Customer Service column using Pandas and Statistics library**

In [25]:
```
print("The mean using pandas function for signal Strength is ", round(df1["Customer Service"].std(),2))
```

```
The mean using pandas function for signal Strength is  0.96
```

In [26]:
```
print("The mean using pandas function for signal Strength is ", round(sst.stdev(df1["Customer Service"]
```

```
The mean using pandas function for signal Strength is  1
```

- **Find Variance of Customer Service column using Pandas and Statistics library**

In [27]:
```
print("The mean using pandas function for signal Strength is ", round(df1["Customer Service"].var(),2))
```

```
The mean using pandas function for signal Strength is  0.93
```

In [28]:
```
print("The mean using pandas function for signal Strength is ", round(sst.variance(df1["Customer Servic
```

```
The mean using pandas function for signal Strength is  1
```

- **Calculate Percentiles of Value for the Dollar column using Numpy.**

In [29]:
```
np.percentile(df1["Value for the Dollar"],[0,25,50,75,100])
```

Out[29]:
```
array([1., 3., 3., 4., 5.])
```

- **Calculate Range of Value for the Dollar column using Pandas.**

In [30]:
```
range=df1["Value for the Dollar"].max()-df1["Value for the Dollar"].min()
print("Range for Value for the Dollar is ", range)
```

```
Range for Value for the Dollar is  4
```

- **Calculate IQR of Value for the Dollar column using Pandas.**

In [31]:
```
Q3=df1["Value for the Dollar"].quantile(0.75)
Q1=df1["Value for the Dollar"].quantile(0.25)
IQR= Q3-Q1
print("The IQR of Value for the Dollar column is ", IQR)
```

```
The IQR of Value for the Dollar column is  1.0
```

In [ ]:

****