Data Science with Python
Career Program

ASSIGNMENT -04
STATISTICS
[MAJOR]

Presented
By Rajesh Kumar

Q1) **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

```python
import scipy.stats as stats

# Define the sample information
sample_size = 25
sample_mean = 60
sample_std = 30
population_mean = 50

# Calculate the t-statistic and p-value
t_statistic, p_value = stats.ttest_1samp([sample_mean]*sample_size, population_mean)

# Define the significance level
alpha = 0.05

# Print the results
print("Null Hypothesis (Ho): The average time spent by the users is 50 minutes")
print("Alternative Hypothesis (H1): The average time spent by the users is not 50 minutes")
print("Significance level: 5%")

if p_value < alpha:
    print("Result: Reject the null hypothesis")
else:
    print("Result: Fail to reject the null hypothesis")
```

```
Null Hypothesis (Ho): The average time spent by the users is 50 minutes
Alternative Hypothesis (H1): The average time spent by the users is not 50 minutes
Significance level: 5%
Result: Reject the null hypothesis
```

Q2) **Height of 7 students (in cm) is given below. What is the median?**
168 170 169 160 162 164 162.

+ Code + Text

```
[ ]  import statistics as stats

     heights= [168, 170, 169, 160, 162, 164, 162]
     median_height=stats.median(heights)

     print("The median height Of students is:", median_height)
```

```
The median height Of students is: 164
```

Q3) **Below are the observations of the marks of a student. Find the value of mode.**
84 85 89 92 93 89 87 89 92

```
[ ] import statistics as stats

    marks= [84, 85, 89, 92, 93, 89, 87, 89, 92]
    try:
        mode_value=stats.mode(marks)
        print("The mode value is:", mode_value)
    except stats.StatsError as e:
        print("There is no unique mode:", e)
```

```
The mode value is: 89
```

```
[ ] import statistics as stats

    marks= [84, 85, 89, 92, 93, 89, 87, 89, 92]
    mode_value = stats.mode(marks)

    print("The mode value is:", mode_value)
```

```
The mode value is: 89
```

Q4) From the table given below, what is the mean of marks obtained by 20 students?

Marks Xi	No. of students(fi)
3	1
4	2
5	2
6	4
7	5
8	3
9	2
10	1
Total	20

+ Code + Text

```
[ ]  marks= [3, 4, 5, 6, 7, 8, 9, 10]
     frequencies= [1, 2, 2, 4, 5, 3, 2, 1]

     sum_products=sum(mark*freq for mark, freq in zip(marks, frequencies))

     total_students=sum(frequencies)

     mean=sum_products/total_students

     print("The mean of marks obtained by the 20 students is:", mean)
```

```
The mean of marks obtained by the 20 students is: 6.6
```

Q5) **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.**

```
[ ] import scipy.stats as stats

    mean=50             # Mean of the normal distribution
    std_dev=15          # Standard deviation of the normal distribution

    # Calculate the z-scores for the lower and upper limits
    z_lower= (50-mean) /std_dev
    z_upper= (70-mean) /std_dev

    # Calculate the probabilities using the cumulative distribution function (CDF)
    prob_lower=stats.norm.cdf(z_lower)
    prob_upper=stats.norm.cdf(z_upper)

    # Calculate the probability between 50 and 70 hours
    prob_between=prob_upper-prob_lower

    print("The probability that the length of time will be between 50 and 70 hours is:", prob_between)

    The probability that the length of time will be between 50 and 70 hours is: 0.4087887802741321
```

OR

```
0s
from scipy.stats import norm

mean = 50
std_dev = 15

# Calculate the probability using the CDF
probability = norm.cdf(70, loc=mean, scale=std_dev) - norm.cdf(50, loc=mean, scale=std_dev)

# Print the result
print("The probability that the length of time will be between 50 and 70 hours is:", probability)

The probability that the length of time will be between 50 and 70 hours is: 0.4087887802741321
```

Q6) Find the range of the following.

g = [10, 23, 12, 21, 14, 17, 16, 11, 15, 19]

```
[ ]  g = [10, 23, 12, 21, 14, 17, 16, 11, 15, 19]

     range_value=max(g) -min(g)

     print("The range of the dataset is:", range_value)

     The range of the dataset is: 13
```

Q7) **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 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?**

```
[10] P_Spam = 0.5
     P_D_given_Spam = 0.99
     P_D_given_NotSpam = 0.05

     # Calculate P(D)
     P_D = P_D_given_Spam * P_Spam + P_D_given_NotSpam * (1 - P_Spam)

     # Calculate P(¬S|D)
     P_NotSpam_given_D = (P_D_given_NotSpam * (1 - P_Spam)) / P_D

     # Print the result
     print("The probability that an email is not spam given that it is detected as spam:", P_NotSpam_given_D)

     The probability that an email is not spam given that it is detected as spam: 0.04807692307692308
```

Q8) **Given the following distribution of returns, determine the lower quartile:**

{10 25 12 21 19 17 16 11 15 19}

```
[12] import numpy as np

     # Define the dataset
     dataset = [10, 25, 12, 21, 19, 17, 16, 11, 15, 19]

     # Calculate the lower quartile
     lower_quartile = np.percentile(dataset, 25)

     # Print the result
     print("Lower Quartile:", lower_quartile)
```

```
Lower Quartile: 12.75
```

Q9) 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?

```
[13] # Define the values
     n = 25
     p = 0.3

     # Calculate the variability
     variability = n * p * (1 - p)

     # Print the result
     print("Variability of the distribution:", variability)
```

```
Variability of the distribution: 5.25
```

Q10) **Using the ‘Cell Phone Survey Dataset’ perform the below mentioned operations on the dataset:-**

1) Checking datatypes of each column in the dataset

```
[17] import pandas as pd
     import numpy as np
     import statistics
     from scipy import stats
```

```
[19] # Load the dataset
     df = pd.read_csv('Cell Phone Survey.csv')
```

```
     # Checking datatypes of each column
     print("Datatypes of each column:")
     print(df.dtypes)
```

```
Datatypes of each column:
Gender                  object
Carrier                 object
Type                    object
Usage                   object
Signal strength          int64
Value for the Dollar     int64
Customer Service         int64
dtype: object
```

2) Find Mean of Signal strength column using Pandas and Statistics library

```
[24] # Find Mean of Signal strength column
     signal_strength_mean_pandas = df['Signal strength'].mean()
     signal_strength_mean_stats = statistics.mean(df['Signal strength'])
     print("Mean of Signal strength (Pandas):", signal_strength_mean_pandas)
     print("Mean of Signal strength (Statistics):", signal_strength_mean_stats)

     Mean of Signal strength (Pandas): 3.3076923076923075
     Mean of Signal strength (Statistics): 3.3076923076923075
```

3) Find the Median of Customer Service column using Pandas and Statistics library

```
[26] # Find Median of Customer Service column
     customer_service_median_pandas = df['Customer Service'].median()
     customer_service_median_stats = statistics.median(df['Customer Service'])
     print("Median of Customer Service (Pandas):", customer_service_median_pandas)
     print("Median of Customer Service (Statistics):", customer_service_median_stats)
```

```
Median of Customer Service (Pandas): 3.0
Median of Customer Service (Statistics): 3.0
```

4) Find Mode of Signal strength column using Pandas and Statistics library.

```
[29] # Find Mode of Signal strength column
     signal_strength_mode_pandas = df['Signal strength'].mode().values
     signal_strength_mode_stats = statistics.mode(df['Signal strength'])
     print("Mode of Signal strength (Pandas):", signal_strength_mode_pandas)
     print("Mode of Signal strength (Statistics):", signal_strength_mode_stats)
```

```
Mode of Signal strength (Pandas): [3]
Mode of Signal strength (Statistics): 3
```

5) **Find Standard deviation of Customer Service column using Pandas and Statistics library.**

```
[30] # Find Standard deviation of Customer Service column
     customer_service_std_pandas = df['Customer Service'].std()
     customer_service_std_stats = statistics.stdev(df['Customer Service'])
     print("Standard deviation of Customer Service (Pandas):", customer_service_std_pandas)
     print("Standard deviation of Customer Service (Statistics):", customer_service_std_stats)
```

```
Standard deviation of Customer Service (Pandas): 0.9623375261979595
Standard deviation of Customer Service (Statistics): 0.9623375261979595
```

6) Find Variance of Customer Service column using Pandas and Statistics library

```
[31] # Find Variance of Customer Service column
     customer_service_var_pandas = df['Customer Service'].var()
     customer_service_var_stats = statistics.variance(df['Customer Service'])
     print("Variance of Customer Service (Pandas):", customer_service_var_pandas)
     print("Variance of Customer Service (Statistics):", customer_service_var_stats)
```

```
Variance of Customer Service (Pandas): 0.9260935143288084
Variance of Customer Service (Statistics): 0.9260935143288085
```

7) Calculate Percentiles of Value for the Dollar column using Numpy

```
[33] # Calculate Percentiles of Value for the Dollar column
     value_percentiles = np.percentile(df['Value for the Dollar'], [25, 50, 75])
     print("Percentiles of Value for the Dollar (Numpy):", value_percentiles)
```

```
Percentiles of Value for the Dollar (Numpy): [3. 3. 4.]
```

8) Calculate Range of Value for the Dollar column using Pandas.

```
[36] # Calculate Range of Value for the Dollar column
     value_range_pandas = df['Value for the Dollar'].max() - df['Value for the Dollar'].min()
     print("Range of Value for the Dollar (Pandas):", value_range_pandas)
```

```
Range of Value for the Dollar (Pandas): 4
```

9) Calculate IQR of Value for the Dollar column using Pandas.

```
[38] # Calculate IQR of Value for the Dollar column
     value_iqr_pandas = df['Value for the Dollar'].quantile(0.75) - df['Value for the Dollar'].quantile(0.25)
     print("IQR of Value for the Dollar (Pandas):", value_iqr_pandas)
```

```
IQR of Value for the Dollar (Pandas): 1.0
```

10) Hypothesis Testing - Using the data in the Cell Phone Survey dataset, apply ANOVA to determine if the mean response for Value for dollar is the same for different types of cell phones.

```
[50] # Hypothesis Testing - ANOVA
     anova_result = stats.f_oneway(df['Value for the Dollar'][df['Type'] == 'Android'],
                                   df['Value for the Dollar'][df['Type'] == 'iOS'],
                                   df['Value for the Dollar'][df['Type'] == 'Other'])
     print("Hypothesis Testing - ANOVA Result:")
     print(anova_result)
```

```
Hypothesis Testing - ANOVA Result:
F_onewayResult(statistic=nan, pvalue=nan)
```

OR

```
[49] # Perform ANOVA
     anova_result = stats.f_oneway(df['Value for the Dollar'][df['Type'] == 'Android'],
                                   df['Value for the Dollar'][df['Type'] == 'iOS'],
                                   df['Value for the Dollar'][df['Type'] == 'Other'])

     # Define the significance level
     alpha = 0.05

     # Print the results
     print("Hypothesis Testing - ANOVA Result:")
     print("Null Hypothesis (Ho): The mean response for Value for dollar is the same for different types of cell phones")
     print("Alternative Hypothesis (H1): The mean response for Value for dollar is different for at least one pair of cell phone types")
     print("Significance level:", alpha)

     if anova_result.pvalue < alpha:
         print("Result: Reject the null hypothesis")
     else:
         print("Result: Fail to reject the null hypothesis")
```

```
Hypothesis Testing - ANOVA Result:
Null Hypothesis (Ho): The mean response for Value for dollar is the same for different types of cell phones
Alternative Hypothesis (H1): The mean response for Value for dollar is different for at least one pair of cell phone types
Significance level: 0.05
Result: Fail to reject the null hypothesis
```


Thank
you