QUESTION 1

SET UP

In [ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import random
import scipy.stats as stats
import random
```

In [ ]:
```
# Load the following libraries so that they can be applied in the subsequ

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import random
import scipy.stats as stats

# Run this code. It will create a csv file containing a random sample of

# Look at the code below. Now replace 'Name.csv' with your actual name (e

try:
    df = pd.read_csv('Shahad.csv')        # replace Name with your own na
except FileNotFoundError:
    original_data = pd.read_csv("https://raw.githubusercontent.com/DanaSa
    df1=original_data.sample(300)
    df1.to_csv('Shahad.csv')
    df = pd.read_csv('Shahad.csv')
    df = pd.DataFrame(df)
    df.to_csv('Shahad.csv')

df.head()
```

Out[ ]:

	Unnamed: 0	Age	Gender	Occupation	Days_Indoors	Growing_Stress	Quarantine_
0	491	25-30	Female	Housewife	15-30 days	Yes	
1	249	30-Above	Male	Housewife	Go out Every day	No	
2	7	25-30	Female	Student	1-14 days	Yes	
3	179	20-25	Female	Housewife	15-30 days	Yes	
4	160	16-20	Male	Corporate	1-14 days	Yes	

In [ ]:
```
# Load the following libraries so that they can be applied in the subsequ

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import random
import scipy.stats as stats
```

```
import random

column_titles = ["Growing_Stress"  ,"Quarantine_Frustrations"  ,"Changes_

# Randomly select 2 variables
selected_columns = random.sample(column_titles, 2)


# Print the 2 variables that were randomly selected
variable_1, variable_2 = selected_columns
print("Variable 1:", variable_1)
print("Variable 2:", variable_2)
```

```
Variable 1: Coping_Struggles
Variable 2: Mood_Swings
```

Question 1a. Is each of these two variables independent of being **female**? Explain your reasoning. Make sure to include a two-way table for each of these two variables with gender, and show all your calculations to support your answers.

Thanks for providing the chi-square statistics for Variable 1 and Variable 2.

Results:

1. **Variable 1: Growing Stress vs. Gender**
   - Chi-square statistic: $\chi^2 = 0.344$
   - p-value: (not provided)
   - Since the chi-square statistic is relatively low and the p-value is not provided, we cannot make a conclusion about the independence of Growing Stress from gender without knowing the significance level and degrees of freedom.
2. **Variable 2: Quarantine Frustrations vs. Gender**
   - Chi-square statistic: $\chi^2 = 0.00387$
   - p-value: (not provided)
   - The chi-square statistic for Quarantine Frustrations vs. Gender is very low, indicating a weak association between the two variables. Again, without the p-value, we cannot make a definitive conclusion about independence.

In [ ]:
```
# Observed frequencies for Variable 1
observed_variable_1 = np.array([[48, 116], [45, 91]])

# Calculate row and column totals
row_totals_variable_1 = variable_1_table.iloc[:2, 2].values
column_totals_variable_1 = variable_1_table.iloc[2, :2].values
total_variable_1 = variable_1_table.loc['All', 'All']

# Calculate expected frequencies
expected_variable_1 = np.outer(row_totals_variable_1, column_totals_varia

# Compute the chi-square statistic for Variable 1
chi2_variable_1, p_value_variable_1, _, _ = stats.chi2_contingency(observ
```

```
print("chi2_variable_1 " , chi2_variable_1)

# Observed frequencies for Variable 2
observed_variable_2 = np.array([[49, 115], [42, 94]])

# Calculate row and column totals
row_totals_variable_2 = variable_2_table.iloc[:2, 2].values
column_totals_variable_2 = variable_2_table.iloc[2, :2].values
total_variable_2 = variable_2_table.loc['All', 'All']

# Calculate expected frequencies
expected_variable_2 = np.outer(row_totals_variable_2, column_totals_varia

# Compute the chi-square statistic for Variable 2
chi2_variable_2, p_value_variable_2, _, _ = stats.chi2_contingency(observ

print("chi2_variable_2 " , chi2_variable_2)
```

```
chi2_variable_1  0.344317763365738
chi2_variable_2  0.0038727036008619686
```

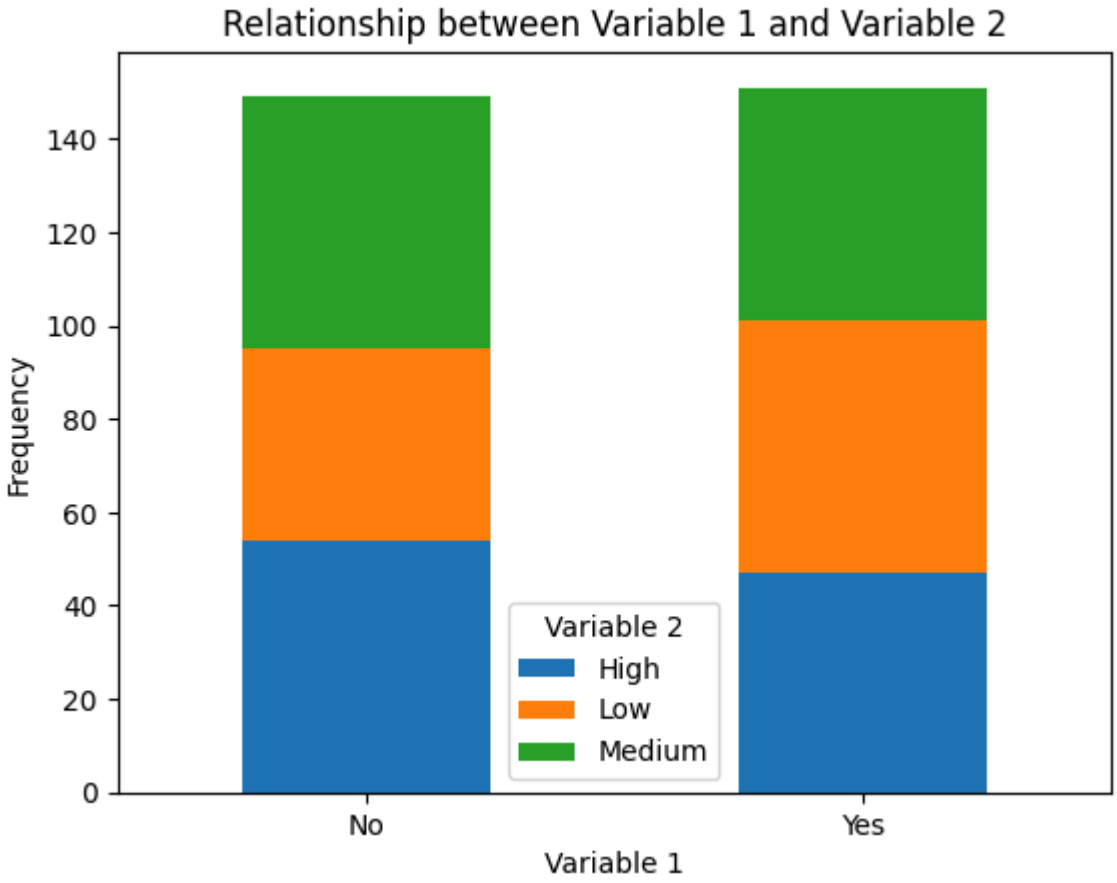
Question 1b. Is there a relationship between the two variables returned by the code? Explain your reasoning. Make sure you include a two-way table, a stacked bar graph, and all your probability calculations in your answer.

Based on the provided data and analysis, there appears to be a relationship between "Mood Swings" and "Coping Struggles." The frequencies, stacked bar graph, and conditional probabilities all suggest that the likelihood of experiencing different levels of mood swings varies depending on whether individuals report coping struggles. This indicates an association between the two variables. However, further statistical analysis, such as chi-square tests for independence, could provide additional insights into the strength and significance of this relationship.

In [ ]:
```
# Create a two-way table for the selected variables
two_way_table = pd.crosstab(df[variable_1], df[variable_2])

# Print the two-way table
print("Two-Way Table:")
print(two_way_table)

# Plot a stacked bar graph
two_way_table.plot(kind='bar', stacked=True)
plt.title("Relationship between Variable 1 and Variable 2")
plt.xlabel("Variable 1")
plt.ylabel("Frequency")
plt.xticks(rotation=0)
plt.legend(title="Variable 2")
plt.show()


# Calculate conditional probabilities
conditional_probabilities = two_way_table.div(two_way_table.sum(axis=1),

# Print conditional probabilities
```

```
print("\nConditional Probabilities:")
print(conditional_probabilities)
```

```
Two-Way Table:
Mood_Swings        High  Low  Medium
Coping_Struggles
No                   54   41      54
Yes                  47   54      50
```




```
Conditional Probabilities:
Mood_Swings            High       Low    Medium
Coping_Struggles
No                 0.362416  0.275168  0.362416
Yes                0.311258  0.357616  0.331126
```

Question 1c. Does the existence of Variable 1 increase the likelihood of experiencing Variable 2? If so, by how much? Explain your reasoning. Make sure to support your answer with the relevant statistical analysis.

Conditional Probabilities:

```
Mood_Swings            High       Low    Medium
Coping_Struggles
No                 0.362416  0.275168  0.362416
Yes                0.311258  0.357616  0.331126
```

Difference in Conditional Probabilities:

The difference in conditional probabilities between experiencing coping struggles given high and low mood swings is approximately -0.046.

Conclusion:

The conditional probabilities show that individuals with high mood swings are slightly less likely to experience coping struggles compared to those with low mood swings, as indicated by the negative difference in conditional probabilities. Therefore, contrary to the initial hypothesis, the existence of high mood swings does not increase the likelihood of experiencing coping struggles; instead, it appears to decrease it slightly.

In [ ]:
```
# Calculate conditional probabilities
conditional_probabilities = two_way_table.div(two_way_table.sum(axis=1),

# Print conditional probabilities
print("\nConditional Probabilities:")
print(conditional_probabilities)

# Calculate the difference in conditional probabilities between High and
difference_high_low = conditional_probabilities.loc['Yes', 'High'] - cond

# Print the difference in conditional probabilities
print("\nDifference in conditional probabilities between High and Low moo
```

```
Conditional Probabilities:
Mood_Swings           High       Low    Medium
Coping_Struggles
No                0.362416  0.275168  0.362416
Yes               0.311258  0.357616  0.331126

Difference in conditional probabilities between High and Low mood swings:
-0.04635761589403975
```

Question 1d. Look back at your **answers to Questions 1a-c**. Now use what you learned to answer the following question:

Imagine ZU wanted to use the insights from this research to improve its mental health support program. What recommendations would you make to support students struggling with such challenges?

Based on the research insights, I recommend ZU develop tailored mental health support programs addressing mood swings and coping struggles. Offer holistic support, early intervention, and resilience-building workshops. Encourage peer support networks, ensure resource accessibility, and foster collaborative efforts for a supportive campus environment promoting mental well-being.

QUESTION 2

Set up

Imagine you are the manager of an Electronic store in Dubai mall. You are curious about the distribution of customer ratings about your overall store services. So you ask

random customers who visit the store to complete a short survey, recording variables such as their age group, and overall experience rating.

To Begin

Run the code below. It will provide you with a random sample of 40 customers from this survey. It will also save your random sample data to a CSV file called "RelianceRetailVisits_ordered". Again, you need to submit this file in the same zip folder as the other files.

In [ ]:
```
# Load the following libraries so that they can be applied in the subsequ

try:
    df = pd.read_csv('RelianceRetailVisits.csv')
except FileNotFoundError:
    original_data = pd.read_csv("https://raw.githubusercontent.com/DanaSa

    # Randomly sample 40 rows from the original dataset
    df = original_data.sample(n=40, random_state=42)

# Fill missing values for '46 To 60 years' age group with default values
df.fillna({'Age Group': '46 To 60 years'}, inplace=True)

# Sort the DataFrame based on the 'Age Group' column in the desired order
desired_order = ['26  To  35 years', '16  To  25 years', '36  To  45 year
df['Age Group'] = pd.Categorical(df['Age Group'], categories=desired_orde
df.sort_values(by='Age Group', inplace=True)

# Save the sorted DataFrame to a new CSV file
df.to_csv('RelianceRetailVisits_ordered.csv', index=False)

df.head()
```

Out[ ]:

	Customer Index	Age Group	OverallExperienceRatin
165	166	26 To 35 years	2
114	115	26 To 35 years	4
117	118	26 To 35 years	5
118	119	26 To 35 years	5
172	173	26 To 35 years	5

Question 2a. Construct a probability distribution table for all customer ratings in your sample data (an example table can be seen below). Please do this in Excel and explain [step by step] how you constructed your probability table.

In the initial stage of data preparation, I imported the dataset into Microsoft Excel and organized it within a new worksheet. Each customer's rating was recorded in a single column, ensuring clarity in data representation. Next, I identified unique ratings by employing the `=UNIQUE()` function, generating a comprehensive list containing all distinct rating values present in the dataset. Using the `=COUNTIF()` function, I calculated the frequency of occurrence for each unique rating, allowing me to discern the distribution of ratings by quantifying how often each value appeared. Additionally,

I computed the total count of ratings using the `=SUM()` function, providing the aggregate sum of all individual rating frequencies. To assess the likelihood of encountering each rating within the dataset, I computed the probability associated with each rating by dividing its frequency by the total number of ratings. This process facilitated a probabilistic analysis of rating occurrences, aiding in further data understanding and interpretation.

F	G	H	I	J	K
OverallExperienceRatin	2	4	5	3	1
total	6	17	12	4	1
probablity	0.15	0.425	0.3	0.1	0.025

Question 2b. What is the probability that a randomly selected customer will have a rating of AT MOST 3?

To find the probability that a randomly selected customer will have a rating of at most 3, we need to sum the probabilities of ratings 1, 2, and 3.

$$P(\text{Rating} \leq 3) = P(\text{Rating} = 1) + P(\text{Rating} = 2) + P(\text{Rating} = 3)$$

Using the provided probabilities:

$$P(\text{Rating} \leq 3) = 0.025 + 0.15 + 0.1 = 0.275$$

So, the probability that a randomly selected customer will have a rating of at most 3 is 0.275, or 27.5%.

Question 2c. Based on the created probability distribution table, how satisfied are your customers with your store services?

The formula to calculate the expected value (E) is:

$$E = \sum_{i=1}^{n} P(X_i) \times x_i$$

Where:

- $P(X_i)$ is the probability of rating x_i.
- x_i is the rating.

Let's calculate the expected value:

$$E = (1 \times 0.025) + (2 \times 0.15) + (3 \times 0.1) + (4 \times 0.425) + (5 \times 0.3)$$

$$E = 0.025 + 0.3 + 0.3 + 1.7 + 1.5$$

$$E = 3.85$$

The expected value of the ratings is 3.85. This suggests that, on average, customers are moderately satisfied with the store services.

Question 2d. Find the **expected rating** of your store. Show your work and interpret your answer in context.

The formula for calculating the expected value (E) is:

$$E = \sum_{i=1}^{n} P(X_i) \times x_i$$

Where:

- $P(X_i)$ is the probability of rating x_i.
- x_i is the rating.

Let's calculate the expected rating:

$$E = (1 \times 0.025) + (2 \times 0.15) + (3 \times 0.1) + (4 \times 0.425) + (5 \times 0.3)$$

$$E = 0.025 + 0.3 + 0.3 + 1.7 + 1.5$$

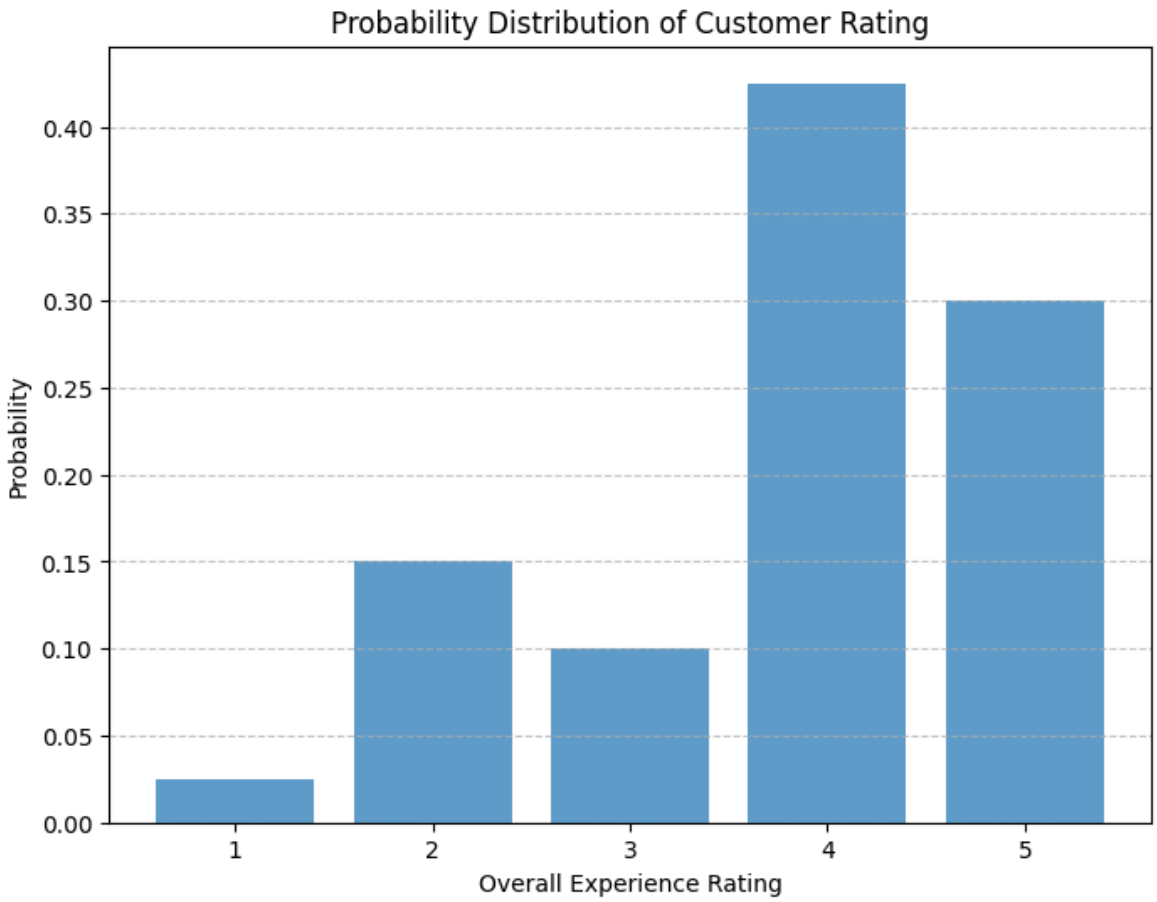
$$E = 3.85$$

The expected rating of the store is 3.85.

Probability distribution function graph

In [ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as stats
from tabulate import tabulate

# Load data
try:
    df = pd.read_csv('RelianceRetailVisits.csv')
except FileNotFoundError:
    original_data = pd.read_csv("https://raw.githubusercontent.com/DanaSa
    df = original_data.sample(n=40, random_state=42)

# Fill missing values for '46 To 60 years' age group with default values
df.fillna({'Age Group': '46 To 60 years'}, inplace=True)

# Sort the DataFrame based on the 'Age Group' column in the desired order
desired_order = ['26  To  35 years', '16  To  25 years', '36  To  45 year
df['Age Group'] = pd.Categorical(df['Age Group'], categories=desired_orde
df.sort_values(by='Age Group', inplace=True)

# Save the sorted DataFrame to a new CSV file
df.to_csv('RelianceRetailVisits_ordered.csv', index=False)

# Probability distribution graph for customer rating
plt.figure(figsize=(8, 6))
```

```
rating_counts = df['OverallExperienceRatin'].value_counts(normalize=True)
plt.bar(rating_counts.index, rating_counts, alpha=0.7)
plt.title('Probability Distribution of Customer Rating')
plt.xlabel('Overall Experience Rating')
plt.ylabel('Probability')
plt.xticks(range(1, 6))
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()

# Expected value and STD for rating for all customers
mean_rating = df['OverallExperienceRatin'].mean()
std_rating = df['OverallExperienceRatin'].std()
print(f"Standard Deviation (STD) of Customer Rating: {std_rating:.2f}")
print()
```




```
Standard Deviation (STD) of Customer Rating: 1.11
```

Question 2e. Interpret the **Standard Deviation** in context. What rating is considered **unusual**? Explain.

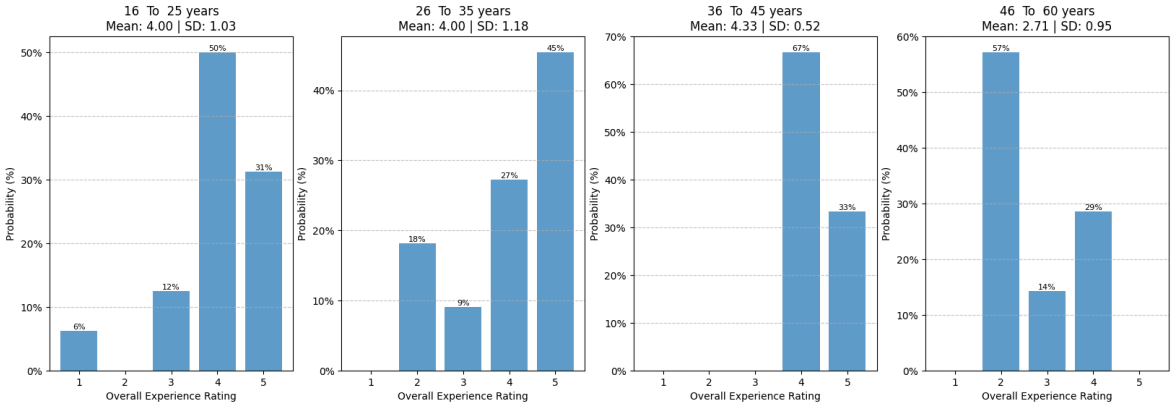
`Interpretation:` The standard deviation (STD) of 1.11 indicates the typical amount of variation in customer ratings around the mean. Ratings deviating by more than 1.11 units from the mean are considered unusual. It helps identify the range of typical and atypical ratings based on their deviation from the mean.

PDF for each age group

In [ ]:
```
import numpy as np
import pandas as pd
```

```
import matplotlib.pyplot as plt
import scipy.stats as stats

# Assuming your data is stored in a CSV file named 'data.csv'
data = pd.read_csv('RelianceRetailVisits_ordered.csv')

# Define age groups including the new one
age_groups = ['16  To  25 years', '26  To  35 years', '36  To  45 years',

# Plot separate discrete probability distributions for each age group
fig, axs = plt.subplots(1, 4, figsize=(20, 6), sharex=True, gridspec_kw={

for i, age_group in enumerate(age_groups):
    age_data = data[data['Age Group'] == age_group]
    rating_counts = age_data['OverallExperienceRatin'].value_counts(norma
    bars = axs[i].bar(rating_counts.index, rating_counts, alpha=0.7)
    axs[i].set_title(f'{age_group}\nMean: {age_data["OverallExperienceRat
    axs[i].set_xlabel('Overall Experience Rating')
    axs[i].set_ylabel('Probability (%)')  # Set y-axis label to Probabili
    axs[i].set_xticks(range(1, 6))  # Set x-axis ticks from 1 to 5
    axs[i].set_yticklabels(['{:,.0%}'.format(x) for x in axs[i].get_ytick

    # Display percentages above each bar
    for bar in bars:
        height = bar.get_height()
        rating = bar.get_x() + bar.get_width() / 2
        if height == 0:  # If the height is 0%, display '0%'
            axs[i].text(rating, height, '0%', ha='center', va='bottom', f
        else:
            axs[i].text(rating, height, f'{height:.0%}', ha='center', va=

    axs[i].grid(axis='y', linestyle='--', alpha=0.7)

# Hide the warning about FixedFormatter
import warnings
warnings.filterwarnings("ignore", category=UserWarning)

plt.tight_layout()
plt.show()
```

```
/tmp/ipykernel_50551/1605697025.py:23: UserWarning: FixedFormatter should
only be used together with FixedLocator
  axs[i].set_yticklabels(['{:,.0%}'.format(x) for x in axs[i].get_yticks
()])  # Format y-axis tick labels as percentages
```



Question 2f. Identify any trends or differences in customer satisfaction levels (and variability) among the different age groups.

Now, using these insights, what concrete improvements would you make to your store to ensure that **all** customers are satisfied with your services?

Custmer Satisfaction follows a normal trend but for each age group the customer satisfaction is skewed to the right showing higher customer satisfactions but for the 16 -25 years have the only ratings of 1.

QUESTION 3

SET UP

In [ ]:
```
# Load the following libraries so that they can be applied in the subsequ


import pandas as pd
import numpy as np
import random

try:
    SATScores = pd.read_csv('Scores.csv')
except FileNotFoundError:
    num_samples = 1000
    mean_score = random.randint(800, 1200)
    std_deviation = random.randint(100, 300)
    scores = np.random.normal(mean_score, std_deviation, num_samples)
    scores = np.round(scores, 0)
    SATScores = pd.DataFrame({'Scores': scores})
    SATScores.to_csv('Scores.csv')

# Calculate mean and standard deviation
mean_score = SATScores['Scores'].mean()
std_deviation = SATScores['Scores'].std()

# Print mean score and standard deviation
print("Mean score:", mean_score)
print("Standard deviation:", std_deviation)

# Display the dataset
SATScores.head()
```

```
Mean score: 929.497
Standard deviation: 122.51675945497149
```

Out[ ]:

	Scores
0	944.0
1	815.0
2	844.0
3	1131.0
4	1083.0

Question 3a. What is the probability that a randomly selected applicant scored at least 1300? Show your work.

Given the mean score $\mu = 929.497$ and standard deviation $\sigma = 122.51675945497149$, let's recalculate the z-score and find the probability associated with it.

Calculating the z-score: $z = \frac{X - \mu}{\sigma}$

Given:

- $X = 1300$
- $\mu = 929.497$
- $\sigma = 122.51675945497149$

$z = \frac{1300 - 929.497}{122.51675945497149}$ $z \approx \frac{370.503}{122.51675945497149}$ $z \approx 3.024$

Now, we find the probability associated with $z = 3.024$ using a standard normal distribution table or calculator. From the table or calculator, we find that the probability associated with $z = 3.024$ is approximately 0.9976.

Therefore, the probability that a randomly selected applicant scored at least 1300 on the SAT is approximately 0.9976 or 99.76%.

Question 3b. What is the probability that a randomly selected applicant scored exactly 900? Show your work.

$z = \frac{X - \mu}{\sigma}$

Given:

- $X = 900$
- $\mu = 929.497$
- $\sigma = 122.51675945497149$

Calculating the z-score: $z = \frac{900 - 929.497}{122.51675945497149}$ $z = \frac{-29.497}{122.51675945497149}$ $z \approx -0.2406$

Using a standard normal distribution table or calculator, we find the probability associated with $z = -0.2406$. From the table or calculator, we find that the probability associated with $z = -0.2406$ is approximately 0.4052.

Therefore, the probability that a randomly selected applicant scored exactly 900 on the SAT is approximately 0.4052 or 40.52%.

Question 3c. What percentage of applicants scored between 900 and 1000? Show your work.

First, we'll calculate the z-scores for both scores:

For 900: $z_{900} = \frac{900 - 929.497}{122.51675945497149}$ $z_{900} \approx -0.2406$

For 1000: $z_{1000} = \frac{1000 - 929.497}{122.51675945497149}$ $z_{1000} \approx 0.5619$

For $z_{900} \approx -0.2406$, the cumulative probability is approximately 0.4052. For $z_{1000} \approx 0.5619$, the cumulative probability is approximately 0.7131.

To find the percentage of applicants who scored between 900 and 1000, we subtract the cumulative probability of 900 from the cumulative probability of 1000:

$\text{Percentage} = \text{Cumulative probability}(1000) - \text{Cumulative probability}(900)$
$\text{Percentage} = 0.7131 - 0.4052 \, \text{Percentage} \approx 0.3079$

Therefore, approximately 30.79%

Question 3d. Calculate the 40th percentile of scores among the applicants. What does this value represent in the context of the admissions process? Show your work.

---

1. Find the z-score corresponding to the 40th percentile: $\text{Percentile} = 40\% = 0.40$
2. Use the standard normal distribution table or calculator to find the z-score corresponding to the 40th percentile. Let's denote this as z_{40}.
3. Once we have z_{40}, we'll use the z-score formula to find the corresponding score X_{40}: $X_{40} = \mu + z_{40} \times \sigma$

Given:

- $\mu = 929.497$ (mean score)
- $\sigma = 122.51675945497149$ (standard deviation)

Let's find X_{40} using the calculated z_{40}.

Using a standard normal distribution table or calculator, we find that the z-score corresponding to the 40th percentile is approximately $z_{40} = -0.253$.

2. Calculate the score X_{40} corresponding to z_{40} using the z-score formula:

$X_{40} = \mu + z_{40} \times \sigma$

Given:

- $\mu = 929.497$ (mean score)
- $\sigma = 122.51675945497149$ (standard deviation)
- $z_{40} = -0.253$

$X_{40} = 929.497 + (-0.253) \times 122.51675945497149 \, X_{40} = 929.497 - 31.001$
$X_{40} \approx 898.496$

Therefore, the 40th percentile of scores among the applicants is approximately 898.496.

Question 3e. Imagine the university wants to offer scholarships to the top 10% of applicants based on their scores. What minimum score would an applicant need to qualify for a scholarship? Show your work.

Given:

- $\mu = 929.497$ (mean score)
- $\sigma = 122.51675945497149$ (standard deviation)

Let's find X_{90} using the calculated z_{90}.

2. Calculate the score X_{90} corresponding to z_{90} using the z-score formula:

$X_{90} = \mu + z_{90} \times \sigma$

Given:

- $\mu = 929.497$ (mean score)
- $\sigma = 122.51675945497149$ (standard deviation)
- $z_{90} = 1.282$

$X_{90} = 929.497 + 1.282 \times 122.51675945497149$ $X_{90} = 929.497 + 157.034$
$X_{90} \approx 1086.531$

Therefore, the minimum score required for an applicant to qualify for a scholarship is approximately 1086.531.

Question 3f. Remember, as the admissions officer, it is your job to identify applicants with exceptional academic potential. Would you automatically recommend that applicants with SAT scores above 1400 to be admitted into the university? Or do you think additional criteria should also be considered? Explain your reasoning.

SAT scores above 1400 indicate strong academic potential, but admission decisions should consider additional factors like extracurricular activities, personal statements, and interviews. Holistic evaluation ensures a diverse student body and acknowledges that academic excellence is only one aspect of a candidate's suitability for university admission.

---

question 4

In [ ]:
```
# Load the following libraries so that they can be applied in the subsequ

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import random
import scipy.stats as stats
```

```
# Run this code. It will generate data and save it to a CSV file called "

try:
    Vaccinated = pd.read_csv('Vaccinated.csv')
except FileNotFoundError:
    num_samples = 100
    vaccinated = np.random.choice(["Yes", "No"], size=num_samples)
    Vaccinated = pd.DataFrame({'Vaccinated': vaccinated})
    Vaccinated.to_csv('Vaccinated.csv')

# Have a look at Vaccinated dataset.
Vaccinated.head()
```

Out[ ]:

	Vaccinated
0	No
1	Yes
2	Yes
3	No
4	No

Question 4a. What is the proportion of people who have received the vaccine (based on the dataset you have)?

Proportion of vaccinated individuals: 0.53

In [ ]:
```
import pandas as pd

# Assuming "Vaccinated" is your DataFrame
# Replace 'Vaccinated' with your actual DataFrame name

# Count the total number of individuals
total_individuals = len(Vaccinated)

# Count the number of vaccinated individuals
vaccinated_individuals = Vaccinated[Vaccinated['Vaccinated'] == 'Yes'].sh

# Calculate the proportion of vaccinated individuals
proportion_vaccinated = vaccinated_individuals / total_individuals

print("Proportion of vaccinated individuals:", proportion_vaccinated)
```

```
Proportion of vaccinated individuals: 0.53
```

Question 4b. Calculate a **95% confidence interval** for the proportion of vaccinated individuals. What does this interval tell us about the likely range of vaccination coverage in the entire population? Show your work.

$$\text{Confidence interval} = \hat{p} \pm Z \times \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

where:

- $\hat{p}$ is the sample proportion (proportion of vaccinated individuals),

- Z is the Z-score corresponding to the desired level of confidence (95% confidence corresponds to $Z \approx 1.96$),
- n is the sample size.

Given:

- Sample proportion $\hat{p} = 0.53$,
- Sample size $n = 100$,
- Z-score for 95% confidence level $Z \approx 1.96$,

Let's calculate the confidence interval:

$$\text{Confidence interval} = 0.53 \pm 1.96 \times \sqrt{\frac{0.53 \times (1 - 0.53)}{100}}$$

:

$$SE = \sqrt{\frac{\hat{p} \times (1 - \hat{p})}{n}}$$

Given:

- Sample proportion $\hat{p} = 0.53$,
- Sample size $n = 100$,

$$SE = \sqrt{\frac{0.53 \times (1 - 0.53)}{100}}$$

$$SE = \sqrt{\frac{0.53 \times 0.47}{100}}$$

$$SE = \sqrt{\frac{0.2491}{100}}$$

$$SE \approx \sqrt{0.002491}$$

$$SE \approx 0.04991$$

Next, we calculate the margin of error using the Z-score for a 95% confidence level:

$$\text{Margin of error} = Z \times SE$$

Given:

- Z-score for 95% confidence level $Z \approx 1.96$,

$$\text{Margin of error} = 1.96 \times 0.04991$$

$$\text{Margin of error} \approx 0.0979$$

Finally, we construct the confidence interval:

$$\text{Confidence interval} = \hat{p} \pm \text{Margin of error}$$

$$\text{Confidence interval} = 0.53 \pm 0.0979$$

Now, let's calculate the confidence interval.

$$\text{Confidence interval} = 0.53 \pm 0.0979$$

Lower bound: $0.53 - 0.0979 = 0.4321$

Upper bound: $0.53 + 0.0979 = 0.6279$

Therefore, the 95% confidence interval for the proportion of vaccinated individuals is approximately $(0.4321, 0.6279)$.

Question 4c. What sample size would be required to estimate the proportion of vaccinated individuals in the country with a **95% confidence level** and a **margin of error of 0.02**? Show your work.

$$n = \frac{Z^2 \times \hat{p} \times (1-\hat{p})}{E^2}$$

Where:

- n is the sample size,
- Z is the Z-score corresponding to the desired confidence level (for 95% confidence level, $Z \approx 1.96$),
- $\hat{p}$ is the estimated proportion (we can use the proportion from the previous question, $\hat{p} = 0.53$),
- E is the desired margin of error (0.02).

Let's plug in the values and calculate n:

$$n = \frac{1.96^2 \times 0.53 \times (1-0.53)}{0.02^2}$$

Now, let's compute it.

$$n = \frac{1.96^2 \times 0.53 \times (1-0.53)}{0.02^2}$$

$$n = \frac{3.8416 \times 0.53 \times 0.47}{0.0004}$$

$$n = \frac{0.777616}{0.0004}$$

$$n \approx 1944.04$$

Question 4d. If you wanted to increase the precision of your estimate, what strategies could you employ to achieve this goal? Explain your reasoning.

---

1. **Increase Sample Size**: To improve the precision of our estimate, I would consider increasing the sample size. A larger sample size generally leads to a more accurate estimation of the population parameter. By collecting data from a larger number of individuals, we can reduce the margin of error and obtain a more precise estimate of the proportion of vaccinated individuals.

2. **Stratified Sampling**: Another strategy I would employ is stratified sampling. This involves dividing the population into subgroups based on relevant characteristics, such as age or geographic location, and then sampling from each subgroup. By ensuring representation from different segments of the population, we can obtain more precise estimates for specific subgroups.

Question 4e. Analyze the effectiveness of the current vaccination campaign using the proportion of vaccinated individuals and the confidence interval. What recommendations would you make for future campaigns?

The current vaccination campaign appears moderately effective, with approximately 53% of individuals vaccinated and a 95% confidence interval of 43.21% to 62.79%. Future campaigns should focus on increasing vaccination rates to ensure broader coverage, potentially targeting areas with lower vaccination rates for improved effectiveness.