Avesh Raza Nagauri 9082425683 aveshnagauri5@gmail.com colab link for project https://colab.research.google.com/drive/1-Hh79_dKVM3lAsWuPcWJ6e6XmrCgfR_y#scrollTo=O3GSom-3Jptb

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

```
df = pd.read_csv('aerofit_treadmill.txt')
```

```
'''There seems to be a problem with the PDF, as the visualizations and recommendations
are not displaying properly. Please use the provided Colab link to review the project.'''
```

```
'There seems to be a problem with the PDF, as the visualizations and recommendations \nare not displaying properly. Please use
the provided Colab link to review the project.'
```

```
df.head()
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
0	KP281	18	Male	14	Single	3	4	29562	112
1	KP281	19	Male	15	Single	2	3	31836	75
2	KP281	19	Female	14	Partnered	4	3	30699	66
3	KP281	19	Male	12	Single	3	3	32973	85
4	KP281	20	Male	13	Partnered	4	2	35247	47

```
df.shape
```

```
(180, 9)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 180 entries, 0 to 179
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Product        180 non-null    object
 1   Age            180 non-null    int64
 2   Gender         180 non-null    object
 3   Education      180 non-null    int64
 4   MaritalStatus  180 non-null    object
 5   Usage          180 non-null    int64
 6   Fitness        180 non-null    int64
 7   Income         180 non-null    int64
 8   Miles          180 non-null    int64
dtypes: int64(6), object(3)
memory usage: 12.8+ KB
```

```
df.describe()
```

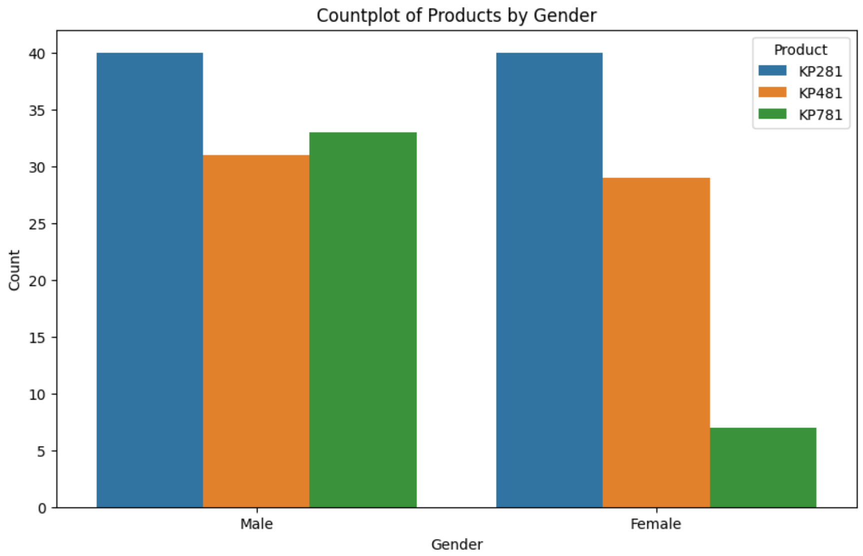
	Age	Education	Usage	Fitness	Income	Miles
count	180.000000	180.000000	180.000000	180.000000	180.000000	180.000000
mean	28.788889	15.572222	3.455556	3.311111	53719.577778	103.194444
std	6.943498	1.617055	1.084797	0.958869	16506.684226	51.863605
min	18.000000	12.000000	2.000000	1.000000	29562.000000	21.000000
25%	24.000000	14.000000	3.000000	3.000000	44058.750000	66.000000
50%	26.000000	16.000000	3.000000	3.000000	50596.500000	94.000000
75%	33.000000	16.000000	4.000000	4.000000	58668.000000	114.750000
max	50.000000	21.000000	7.000000	5.000000	104581.000000	360.000000

```
df.isna().sum()
```

```
Product          0
Age              0
```

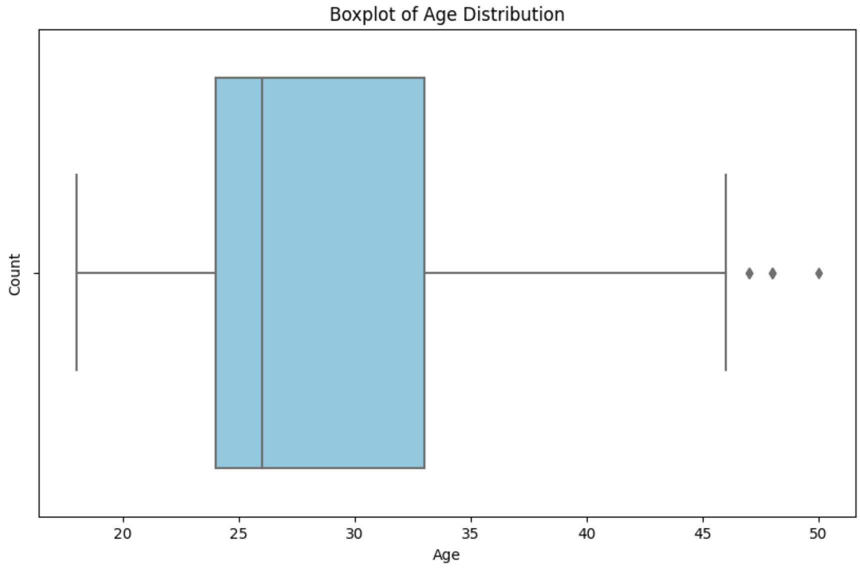
```
Gender           0
Education        0
MaritalStatus    0
Usage            0
Fitness          0
Income           0
Miles            0
dtype: int64
```

```
# Visual representation of product counts for each gender using a countplot.

plt.figure(figsize=(10, 6))

sns.countplot(x='Gender', data=df, hue='Product')

plt.title('Countplot of Products by Gender')
plt.xlabel('Gender')
plt.ylabel('Count')
plt.legend(title='Product')

plt.show()
```



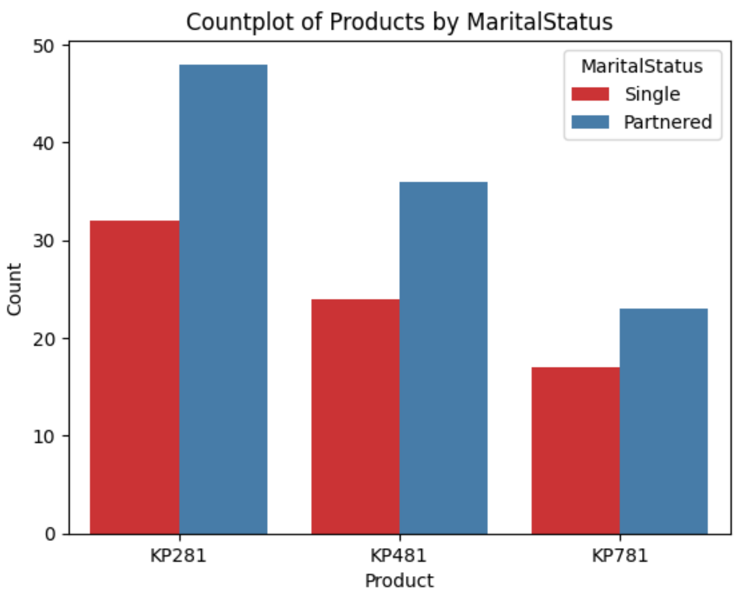

```
'''
Insight: The data visualization reveals a compelling pattern—KP281 emerges as the top
performer across both genders, closely followed by KP481 in the second position and KP781
securing the third spot in sales.
'''
```

```
# Identifying the age bracket with the highest customer footfall.

plt.figure(figsize=(10, 6))

sns.boxplot(x='Age', data=df, color='skyblue')

plt.title('Boxplot of Age Distribution')
plt.xlabel('Age')
plt.ylabel('Count')

plt.show()
```




```
'''
Insight: The predominant consumer demographic falls within the age range of 23 to 33,
with a notable presence of outliers around the age of 50.
'''
```

```
# Visual representation of product counts based on marital status using a countplot.

sns.countplot(x='Product', data=df, hue='MaritalStatus', palette = 'Set1')
plt.xlabel('Product')
plt.ylabel('Count')
plt.title('Countplot of Products by MaritalStatus')

plt.show()
```




```
'''
Insight: There is a notable trend where customers with a partner exhibit a higher
propensity to purchase products compared to their single counterparts across various
product categories.
'''
```

```
df['Income'].describe()
```

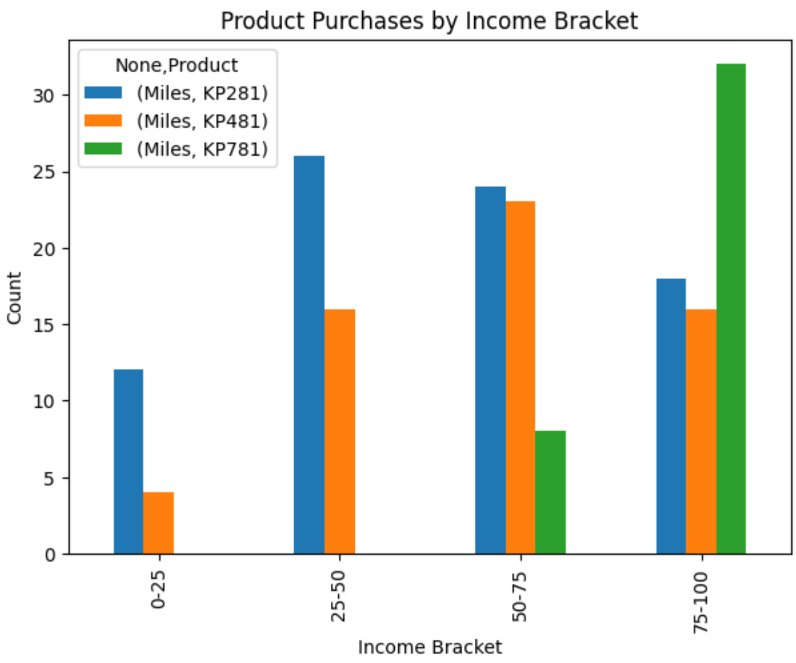
```
count      180.000000
mean     53719.577778
std      16506.684226
min      29562.000000
```

```
25%       44058.750000
50%       50596.500000
75%       58668.000000
max      104581.000000
Name: Income, dtype: float64
```

```
# Examining customers with varying income levels to identify patterns or trends in their behavior or preferences.

selected_columns = ['Product', 'Income']

income_bins = [0, 40000,50000, 60000,70000, 80000, float('inf')]
income_labels = ['< 40000', '40000 - 50000','50000 - 60000','60000 - 70000', '70000 - 80000', '80000+']

# Creating a new column 'Income Bracket' based on the defined bins
df['Income Bracket'] = pd.cut(df['Income'], bins=income_bins, labels=income_labels, right=False)

# Creating a pivot table
pivot_table = pd.pivot_table(df[selected_columns + ['Income Bracket']], index='Income Bracket',
                             columns='Product', aggfunc=len, fill_value=0)


pivot_table.reset_index()
```

	Income Bracket	Income		
Product		**KP281**	**KP481**	**KP781**
0	< 40000	23	9	0
1	40000 - 50000	25	21	5
2	50000 - 60000	26	23	6
3	60000 - 70000	6	7	6
4	70000 - 80000	0	0	4
5	80000+	0	0	19

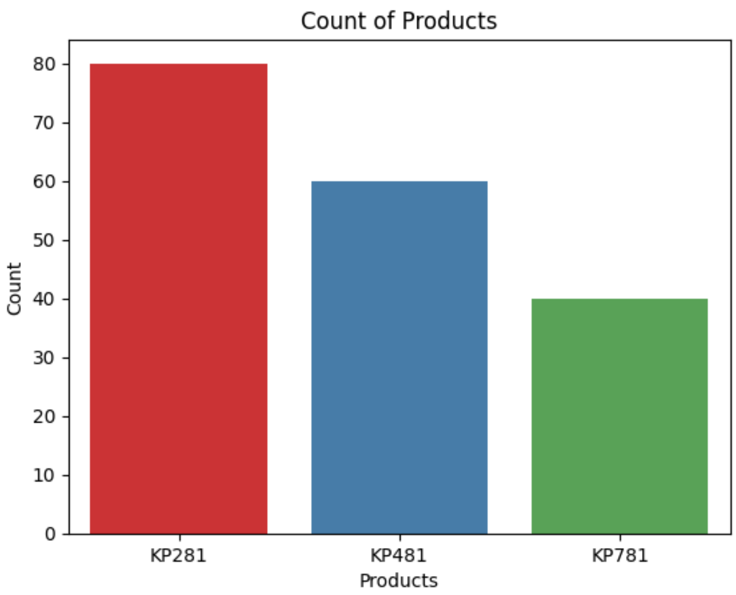
```
# Creating a grouped bar chart to visualize the data distribution.

pivot_table.plot(kind='bar', figsize = (7,5))

plt.title('Product Purchases by Income Bracket')
plt.xlabel('Income Bracket')
plt.ylabel('Count')

plt.show()
```




```
df['Product'].value_counts().to_frame()
```

	Product
KP281	80
KP481	60
KP781	40

```
# Bar plot depicting the sales count for each product.

sns.countplot(x='Product', data=df, palette='Set1')

plt.title("Count of Products")
plt.xlabel("Products")
plt.ylabel("Count")

plt.show()
```




```
'''
Insight: The sales data highlights a clear hierarchy among products, with KP281 leading
the pack with approximately 80 units sold, followed by KP481 at around 60 units, and KP781
trailing with approximately 40 units.
'''
```

```
# Counting the occurrences of each unique value in every column of the table.

df['MaritalStatus'].value_counts().to_frame()
```

	MaritalStatus
Partnered	107
Single	73

```
'''
Insight: The consumer base analysis reveals that 107 individuals have a partner, while 73 are
identified as single, providing valuable insights into the distribution of relationship
statuses among the customer demographic.
'''
```

```
df['Age'].value_counts().to_frame().head()
```

	Age
25	25
23	18
24	12
26	12
28	9

```
df['Gender'].value_counts().to_frame()
```

	Gender
Male	104
Female	76

```
'''
Insight: The demographic breakdown indicates that the consumer base consists of 104 males
and 76 females, shedding light on the gender distribution within the analyzed dataset.
'''
```

```
df['Education'].value_counts().to_frame()
```

	Education
16	85
14	55
18	23
15	5
13	5
12	3
21	3
20	1

```
df['Usage'].value_counts().to_frame()
```

	Usage
3	69
4	52
2	33
5	17
6	7
7	2

```
# How many customers have purchased the KP781 product?

x = df['Product'] == 'KP781'
x.value_counts().to_frame()
```

	Product
False	140
True	40

```
'''
The analysis indicates a significant likelihood of customers opting for the premium
product 'KP781,' with a probability of 0.22, equivalent to 22%. This insight underscores
the product's appeal and suggests potential opportunities for strategic marketing efforts
or promotions to further capitalize on this preference and drive sales. Understanding and
leveraging such probabilities can contribute to informed decision-making and targeted
business strategies.
'''
```

```
df['Miles'].describe()
```

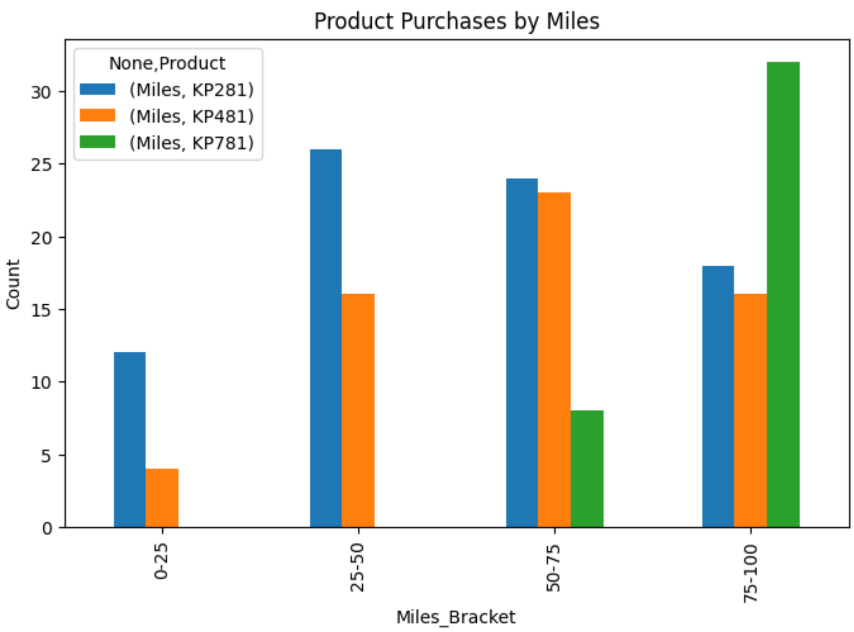
```
count    180.000000
mean     103.194444
std       51.863605
min       21.000000
25%       66.000000
50%       94.000000
75%      114.750000
```

```
max      360.000000
Name: Miles, dtype: float64
```

```
# Exploring the correlation between miles traveled by customers and their product preferences.

selected_columns = ['Product','Miles']

mile_bins = [25,50,75,100,float('inf')]
mile_labels = ['0-25', '25-50', '50-75', '75-100']

df['Miles_Bracket'] = pd.cut(df['Miles'], bins = mile_bins, labels = mile_labels)
pivot_table = pd.pivot_table(df[selected_columns + ['Miles_Bracket']], index = 'Miles_Bracket',
                             columns = 'Product', aggfunc = len)

pivot_table.reset_index()
```

	Miles_Bracket	Miles		
Product		**KP281**	**KP481**	**KP781**
0	0-25	12.0	4.0	NaN
1	25-50	26.0	16.0	NaN
2	50-75	24.0	23.0	8.0
3	75-100	18.0	16.0	32.0

```
# Visualization of product purchases categorized by miles, presented in a plot.

pivot_table.plot(kind ='bar', figsize = (8,5))
plt.title('Product Purchases by Miles')
plt.xlabel('Miles_Bracket')
plt.ylabel('Count')

plt.show()
```



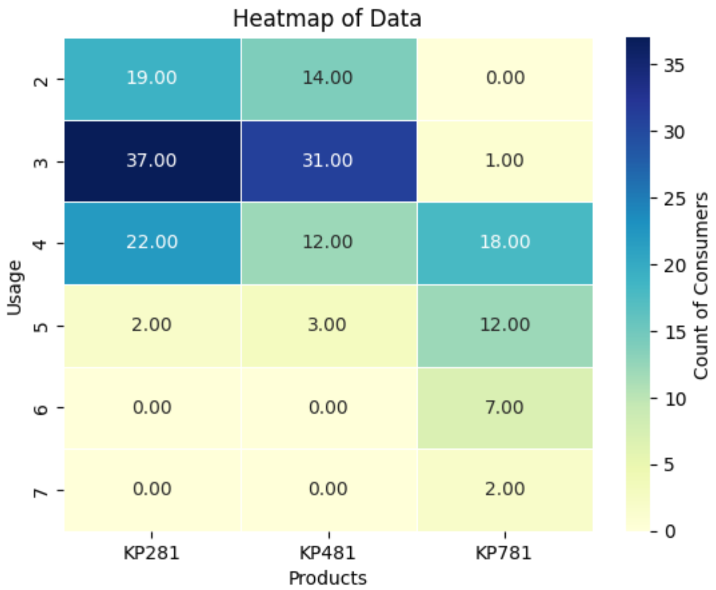

```
'''
The data visualization indicates a noticeable trend: as customers' mileage increases,
there is a corresponding shift towards premium products, specifically KP781.
'''
```

```
'\nThe data visualization indicates a noticeable trend: as customers' mileage inc
reases, \nthere is a corresponding shift towards premium products, specifically K
```

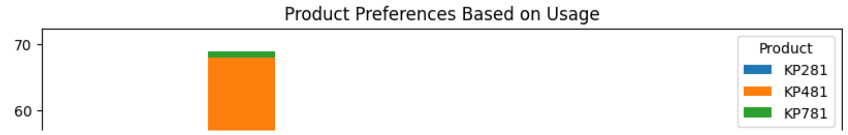
```
# Investigating the impact of consumers' weekly product usage on their purchasing behavior.

usage_heatmap = df.groupby('Usage')['Product'].value_counts().unstack().fillna(0)
usage_heatmap
```

Product	KP281	KP481	KP781
Usage			
2	19.0	14.0	0.0
3	37.0	31.0	1.0
4	22.0	12.0	18.0
5	2.0	3.0	12.0

```
# Product preferences influenced by usage patterns depicted in a heatmap.

sns.heatmap(usage_heatmap, cmap="YlGnBu", annot=True, fmt=".2f", linewidths=0.5, cbar_kws={'label': 'Count of Consumers'})

plt.title('Heatmap of Data')
plt.xlabel('Products')
plt.ylabel('Usage')

plt.show()
```




```
# Barplot illustrating consumer product preferences based on usage patterns.

usage_heatmap.plot(kind='bar', stacked=True, figsize=(10, 6))

plt.title('Product Preferences Based on Usage')
plt.xlabel('Usage')
plt.ylabel('Count of Customers')
plt.legend(title='Product')

plt.show()
```

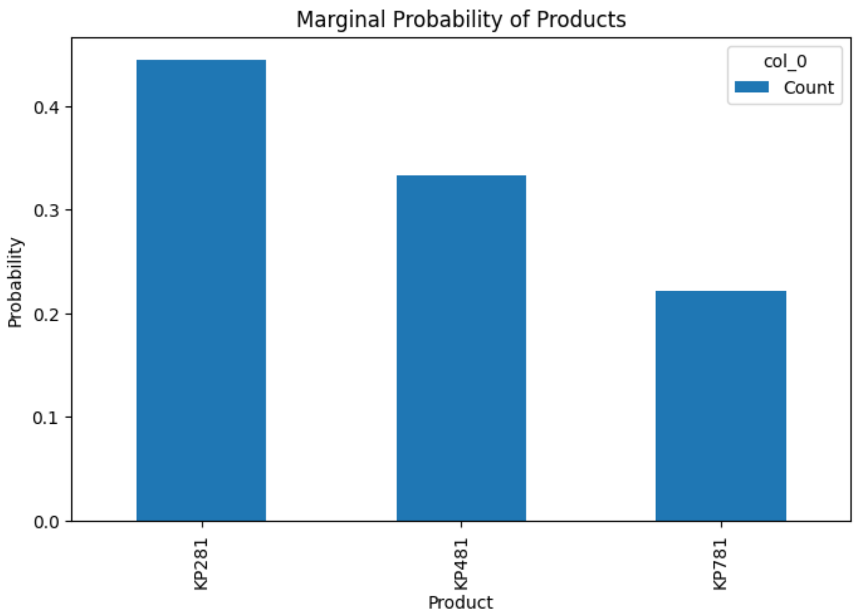



```
'''
Insight: The visualization highlights a distinct trend – customers who use our products
more frequently per week tend to opt for the 'KP781' premium product. However, a noteworthy
observation is the limited number of customers engaging with our products in a week for
5 to 7 times, suggesting potential areas for targeted engagement and promotion efforts.
'''
```

```
# What percentage of customers have acquired KP281, KP481, or KP781?

marginal_prob_table = pd.crosstab(index=df['Product'], columns='Count', normalize=True)
marginal_prob_table
```

col_0	Count
Product	
KP281	0.444444
KP481	0.333333
KP781	0.222222

```
# Barplot illustrating the probability distribution of products.

marginal_prob_table.plot(kind='bar', figsize=(8, 5))
plt.title('Marginal Probability of Products')
plt.xlabel('Product')
plt.ylabel('Probability')
plt.show()
```

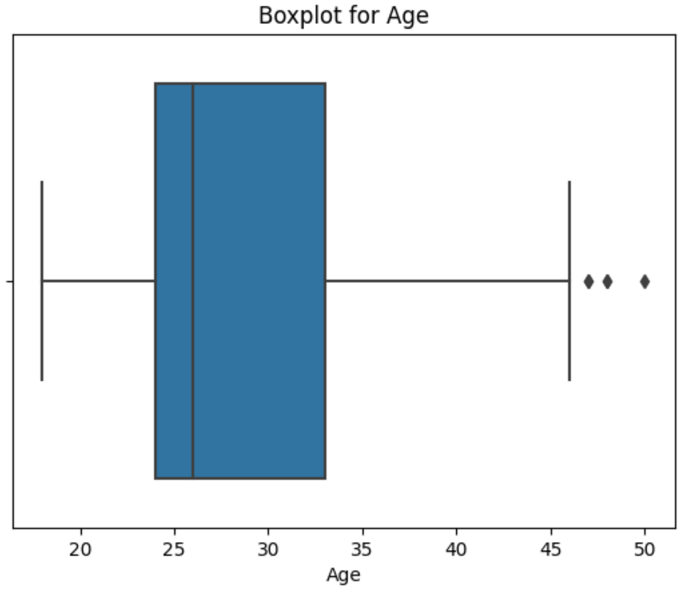


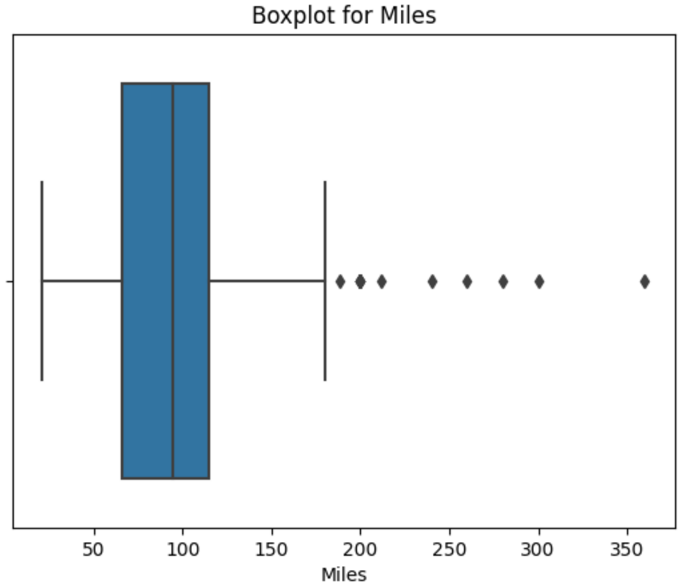

```
'''
Insight: The product selection pattern among customers reveals distinct preferences.
Approximately 44.4% of customers favor KP281, followed by 33.3% opting for KP481, and
22.2% choosing KP781. Understanding these probabilities provides valuable insights into
consumer choices and can inform targeted strategies for product promotion and marketing
efforts.
'''
```

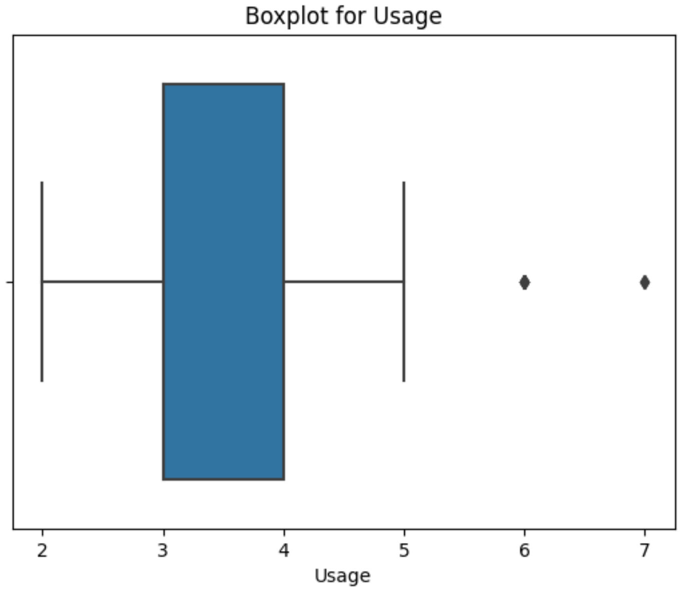
```
df.describe()
```

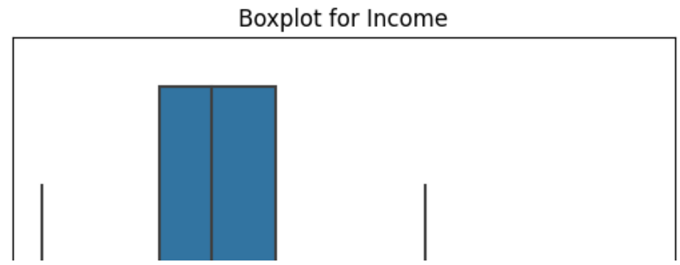
	Age	Education	Usage	Fitness	Income	Miles
count	180.000000	180.000000	180.000000	180.000000	180.000000	180.000000
mean	28.788889	15.572222	3.455556	3.311111	53719.577778	103.194444
std	6.943498	1.617055	1.084797	0.958869	16506.684226	51.863605
min	18.000000	12.000000	2.000000	1.000000	29562.000000	21.000000
25%	24.000000	14.000000	3.000000	3.000000	44058.750000	66.000000

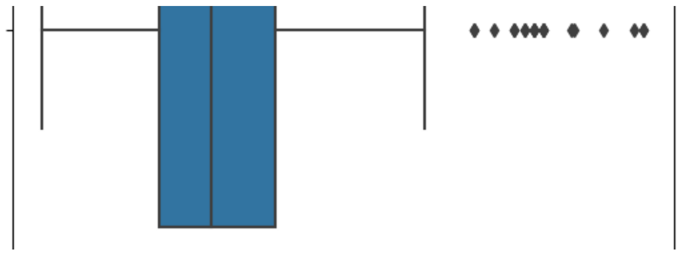
```
# Detecting Outliers (using boxplot, “describe” method by checking the difference
# between mean and median)

numerical_columns = ['Age', 'Miles', 'Usage','Income','Fitness']

# Boxplot for each numerical column
for column in numerical_columns:
    sns.boxplot(x=df[column])
    plt.title(f'Boxplot for {column}')
    plt.show()

# Using describe method to check mean-median difference
desc_stats = df.describe().transpose()
mean_median_diff = desc_stats['mean'] - desc_stats['50%']

print("Descriptive Statistics:")
print(desc_stats)
print("\nDifference between mean and median:")
print(mean_median_diff)
```


Boxplot for Age
20
25
30
35
40
45
50
Age


Boxplot for Miles
50
100
150
200
250
300
350
Miles


Boxplot for Usage
2
3
4
5
6
7
Usage


Boxplot for Income



```
# Likelihood of a male customer purchasing a KP781 treadmill.

result = df[(df['Gender'] == 'Male') & (df['Product'] == 'KP781')]
result.count()
```

```
Product          33
Age              33
Gender           33
Education        33
MaritalStatus    33
Usage            33
Fitness          33
Income           33
Miles            33
dtype: int64
```

```
percent = 33/180*100
percent
```

```
18.333333333333332
```

```
'''
Insight: The analysis indicates that the probability of a male customer opting for a
'KP781' product is 0.1833, equivalent to 18%. This probability insight sheds light on
the likelihood of male customers choosing the specific 'KP781' product within the given
dataset.
'''
```

```
(180, 9)
```

```
# Likelihood of a female customer purchasing a KP781 treadmill.

result = df[(df['Gender'] == 'Female') & (df['Product'] == 'KP781')]
result.count()
```

```
Product          7
Age              7
Gender           7
Education        7
MaritalStatus    7
Usage            7
Fitness          7
Income           7
Miles            7
dtype: int64

Fitness         0.311111
```

```
percent = 7/180*100
percent
```

```
3.888888888888889
```

```
'''
Insight: The data suggests that the probability of a female customer selecting a 'KP781'
product is 0.0388, translating to 3.88%. This probability insight provides valuable
information about the likelihood of female customers opting for the specific 'KP781'
product within the observed dataset.
'''
```

Recommendations

```
'''
1. Targeted Marketing Campaigns: Given the distinct preferences observed in product
selection among different demographics and usage patterns, consider implementing
targeted marketing campaigns. Tailor promotions to align with the identified preferences
```

```
targeted marketing campaigns. Tailor promotions to align with the identified preferences,
such as highlighting the features or benefits of the preferred 'KP781' product for
specific customer segments.

2. Engagement Strategies for 5 to 7 Times Usage: Recognizing the limited engagement in
the 5 to 7 times per week usage range, develop strategies to boost customer interaction
within this frequency bracket. This could involve promoting additional features, offering
incentives, or introducing loyalty programs to encourage more frequent product use.

3. Diversification of Product Offerings: While 'KP281' is the top choice, diversify
the product offerings to cater to a broader audience. Introduce variations or complementary
products that align with consumer preferences, ensuring a well-rounded product portfolio.

4. Promotional Packages: Create bundled promotional packages that combine preferred products
like 'KP781' with others, providing added value for customers and potentially encouraging them
to explore and purchase additional products.

5. Customer Engagement Surveys: Conduct surveys or feedback sessions to understand customer
expectations, preferences, and satisfaction levels. Use the insights gathered to refine product
features, enhance customer experience, and address any potential concerns or areas for improvement.

6. Market Expansion: Explore opportunities to expand the market reach by identifying untapped
demographics or regions. This could involve assessing the feasibility of introducing products tailored
to specific local preferences or cultural considerations.

7. Customer Education Initiatives: Since product preferences vary, initiate educational campaigns to
inform customers about the unique features and benefits of each product. This can empower customers to
make informed decisions and potentially lead to increased satisfaction and loyalty.

8. Collaborations and Partnerships:** Explore partnerships or collaborations with fitness
influencers, gyms, or wellness platforms to enhance brand visibility and credibility.
Leveraging influencers can be particularly effective in reaching and influencing specific
target demographics.

9. E-commerce Optimization: If applicable, optimize the e-commerce platform for a seamless
and user-friendly experience. Ensure that product information is clear, and the checkout process
is efficient, making it easy for customers to browse, select, and purchase products.

10. Data-driven Decision Making: Continue leveraging insights from data analytics to inform
strategic decisions. Regularly analyze sales patterns, customer behavior, and market trends
to stay adaptable and responsive to changing consumer preferences.

'''
```