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

aerofit= pd.read_csv('aerofit_treadmill.txt')

aerofit.head(2)
```

```
  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
```

```
aerofit.isnull().sum()
```

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

There are no Null Values in the dataset

```
aerofit.shape

(180, 9)
```

The data set has 180 rows and 9 columns

Detect Outliers (using boxplot, "describe" method by checking the difference between mean and median))

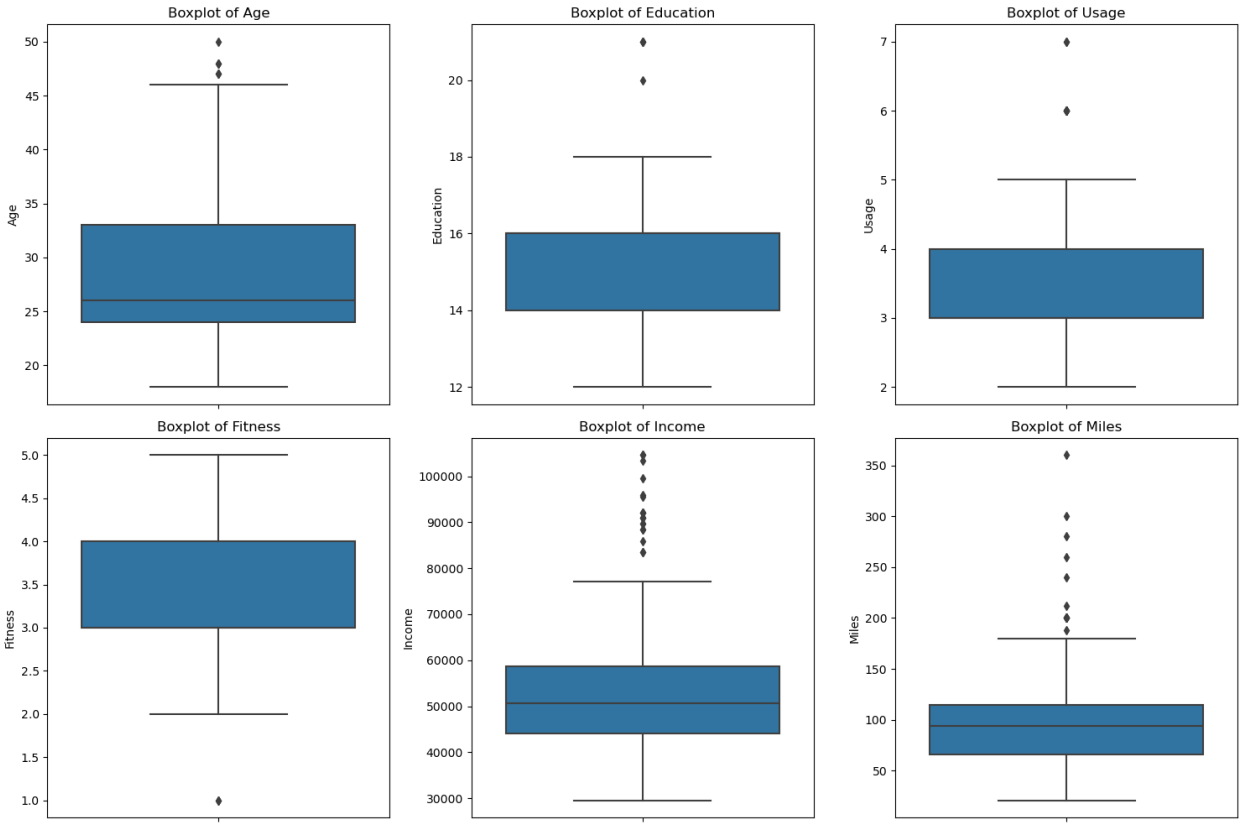
```
aerofit.describe()
```

```
              Age   Education       Usage     Fitness                 
Income  \
count  180.000000  180.000000  180.000000  180.000000     180.000000

mean    28.788889   15.572222    3.455556    3.311111   53719.577778

std      6.943498    1.617055    1.084797    0.958869   16506.684226
```

```
min     18.000000   12.000000    2.000000    1.000000   29562.000000

25%     24.000000   14.000000    3.000000    3.000000   44058.750000

50%     26.000000   16.000000    3.000000    3.000000   50596.500000

75%     33.000000   16.000000    4.000000    4.000000   58668.000000

max     50.000000   21.000000    7.000000    5.000000  104581.000000
```

```
              Miles
count  180.000000
mean   103.194444
std     51.863605
min     21.000000
25%     66.000000
50%     94.000000
75%    114.750000
max    360.000000
```

```
plt.figure(figsize=(15, 10))

columns = ['Age', 'Education', 'Usage', 'Fitness', 'Income', 'Miles']

for i, col in enumerate(columns):
    plt.subplot(2, 3, i + 1)
    sns.boxplot(y=aerofit[col])
    plt.title(f'Boxplot of {col}')

plt.tight_layout()
plt.show()

for col in columns:
    mean = aerofit[col].mean()
    median = aerofit[col].median()
    print(f"{col}: Mean = {mean}, Median = {median}, Difference =
{abs(mean - median)}")
```




```
Age: Mean = 28.788888888888888, Median = 26.0, Difference =
2.788888888888888
Education: Mean = 15.572222222222223, Median = 16.0, Difference =
0.42777777777777715
Usage: Mean = 3.4555555555555557, Median = 3.0, Difference =
0.4555555555555557
Fitness: Mean = 3.311111111111111, Median = 3.0, Difference =
0.3111111111111109
Income: Mean = 53719.57777777778, Median = 50596.5, Difference =
3123.0777777777766
Miles: Mean = 103.19444444444444, Median = 94.0, Difference =
9.194444444444443
```

Attribute Insights

1. Age
   - **Mean**: 28.79 years
   - **Median**: 26.0 years
   - **Difference**: 2.79 years
   - **Boxplot**:
     - The boxplot for Age shows several outliers on the higher end (ages above the whisker).
   - **Interpretation**:

- The distribution is slightly right-skewed with older outliers. The central tendency is around the mid-20s, but there are some significantly older individuals affecting the mean.

2. Education
- **Mean**: 15.57 years
- **Median**: 16.0 years
- **Difference**: 0.43 years
- **Boxplot**:
  - The boxplot for Education shows no significant outliers.
- **Interpretation**:
  - The distribution is symmetric, with the mean and median being close. Most individuals have around 15-16 years of education, indicating a well-educated group.

3. Usage
- **Mean**: 3.46 times
- **Median**: 3.0 times
- **Difference**: 0.46 times
- **Boxplot**:
  - The boxplot for Usage shows some minor outliers on the higher end (usage above the whisker).
- **Interpretation**:
  - The distribution is slightly right-skewed, with most individuals using the service around 3-4 times. The outliers suggest a few individuals use the service significantly more often.

4. Fitness
- **Mean**: 3.31
- **Median**: 3.0
- **Difference**: 0.31
- **Boxplot**:
  - The boxplot for Fitness shows some minor outliers.
- **Interpretation**:
  - The distribution is slightly right-skewed with most fitness levels being around 3-4. There are some outliers indicating individuals with higher fitness levels.

5. Income
- **Mean**: $53,719.58
- **Median**: $50,596.50
- **Difference**: $3,123.08
- **Boxplot**:
  - The boxplot for Income shows several outliers on the higher end.
- **Interpretation**:

- The distribution is right-skewed with a significant difference between mean and median, suggesting the presence of high-income outliers. The majority of individuals earn between $44,058.75 and $58,668, but some earn significantly more, affecting the mean.

6. Miles

- **Mean**: 103.19 miles
- **Median**: 94.0 miles
- **Difference**: 9.19 miles
- **Boxplot**:
  - The boxplot for Miles shows several outliers on the higher end (miles above the whisker).
- **Interpretation**:
  - The distribution is right-skewed, with the mean being higher than the median. Most individuals travel between 66 and 114.75 miles, but some travel significantly more, pulling the mean up.

General Insights

- **Outliers**:
  - Age, Income, and Miles show significant right skewness and have several outliers. These attributes have individuals with much higher values than the majority.
- **Symmetric Distributions**:
  - Education shows a symmetric distribution with no significant outliers. This suggests a uniform level of education among the individuals in the dataset.
- **Minor Skewness**:
  - Usage and Fitness have minor skewness with a few higher-end outliers. The majority of the data is clustered around the median, indicatli ghting the presence of outliers that might affect further analysis. ghting the presence of outliers that might affect further analysis.

Q3. Check if features like marital status, age have any effect on the product purchased (using countplot, histplots, boxplots etc)

```
aerofit['Age'].unique()
```

```
array([18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
34,
       35, 36, 37, 38, 39, 40, 41, 43, 44, 46, 47, 50, 45, 48, 42],
      dtype=int64)
```

```
age_bin = range(15,51, 5)
aerofit['age_group'] = pd.cut(aerofit['Age'], age_bin)
aerofit.head(5)
```

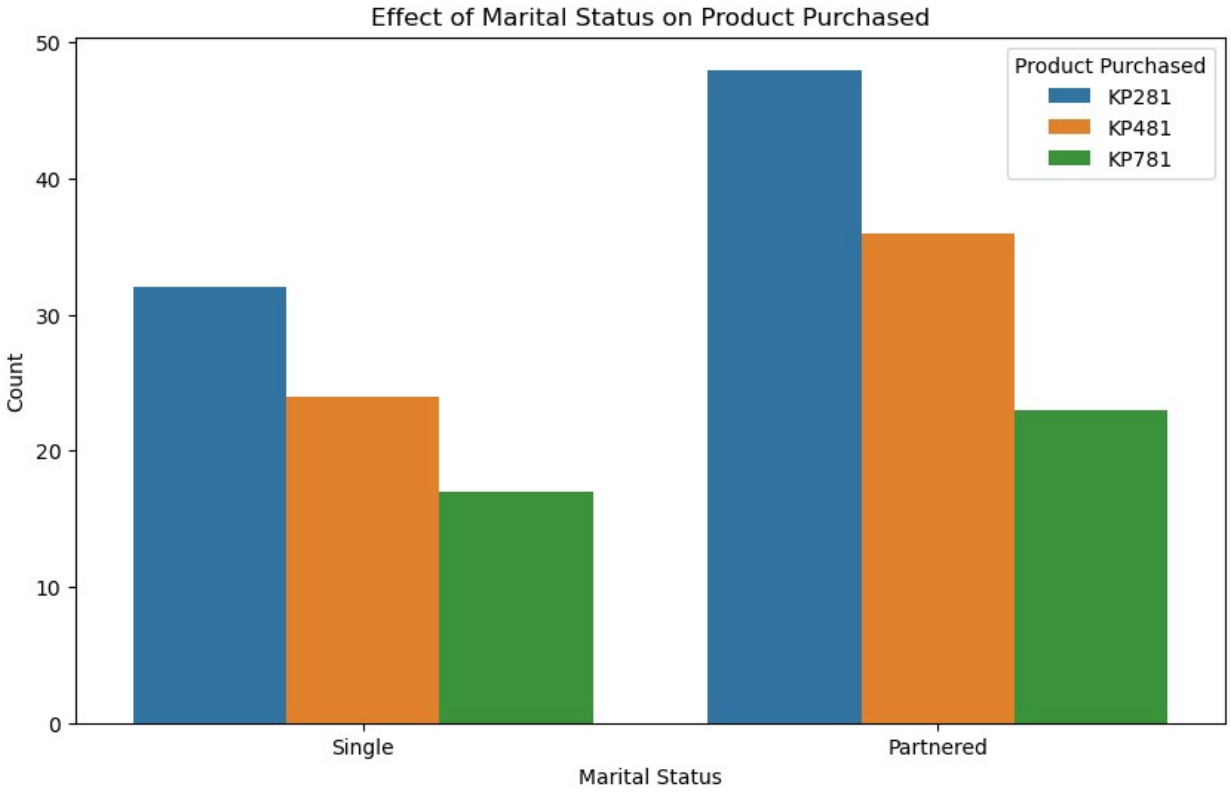
```
  Product  Age  Gender  Education MaritalStatus  Usage  Fitness
Income  \
0   KP281   18    Male         14        Single      3        4
29562
```

```
1   KP281   19    Male          15         Single      2         3
31836
2   KP281   19  Female          14     Partnered       4         3
30699
3   KP281   19    Male          12         Single      3         3
32973
4   KP281   20    Male          13     Partnered       4         2
35247
```

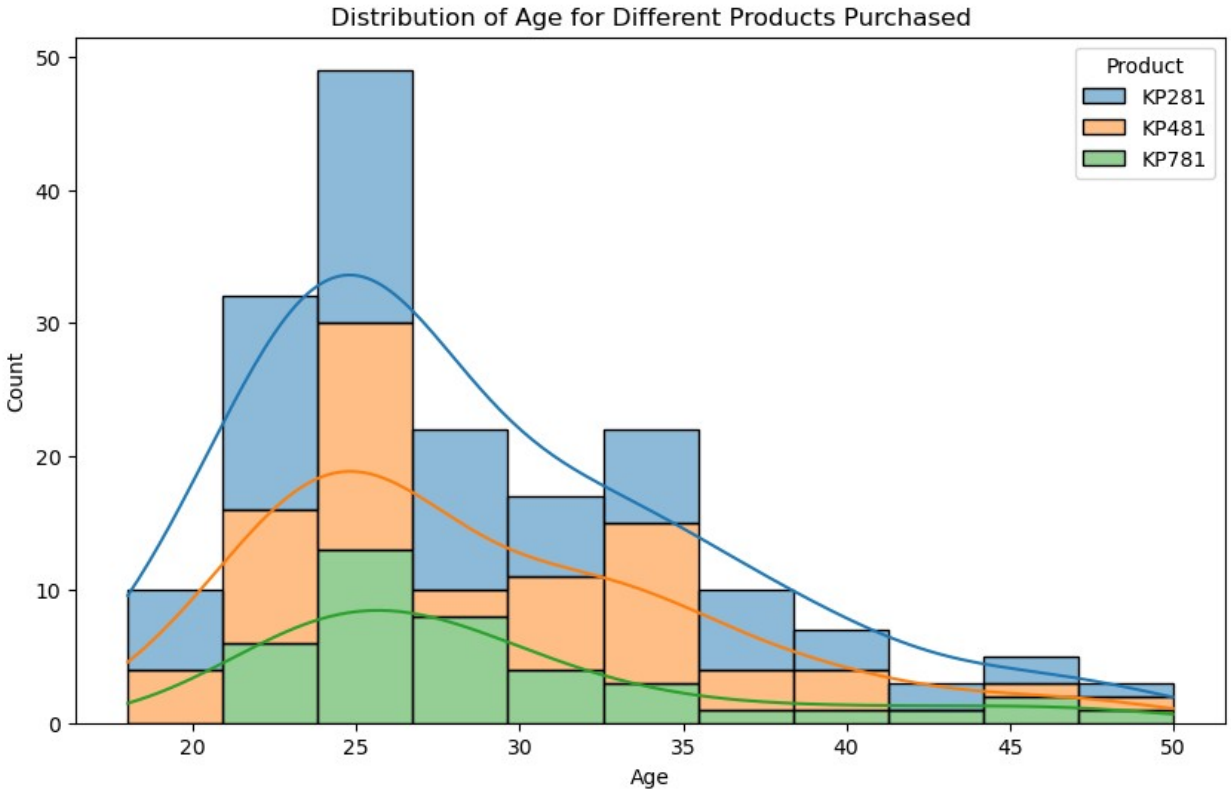
```
   Miles age_group
0    112  (15, 20]
1     75  (15, 20]
2     66  (15, 20]
3     85  (15, 20]
4     47  (15, 20]
```

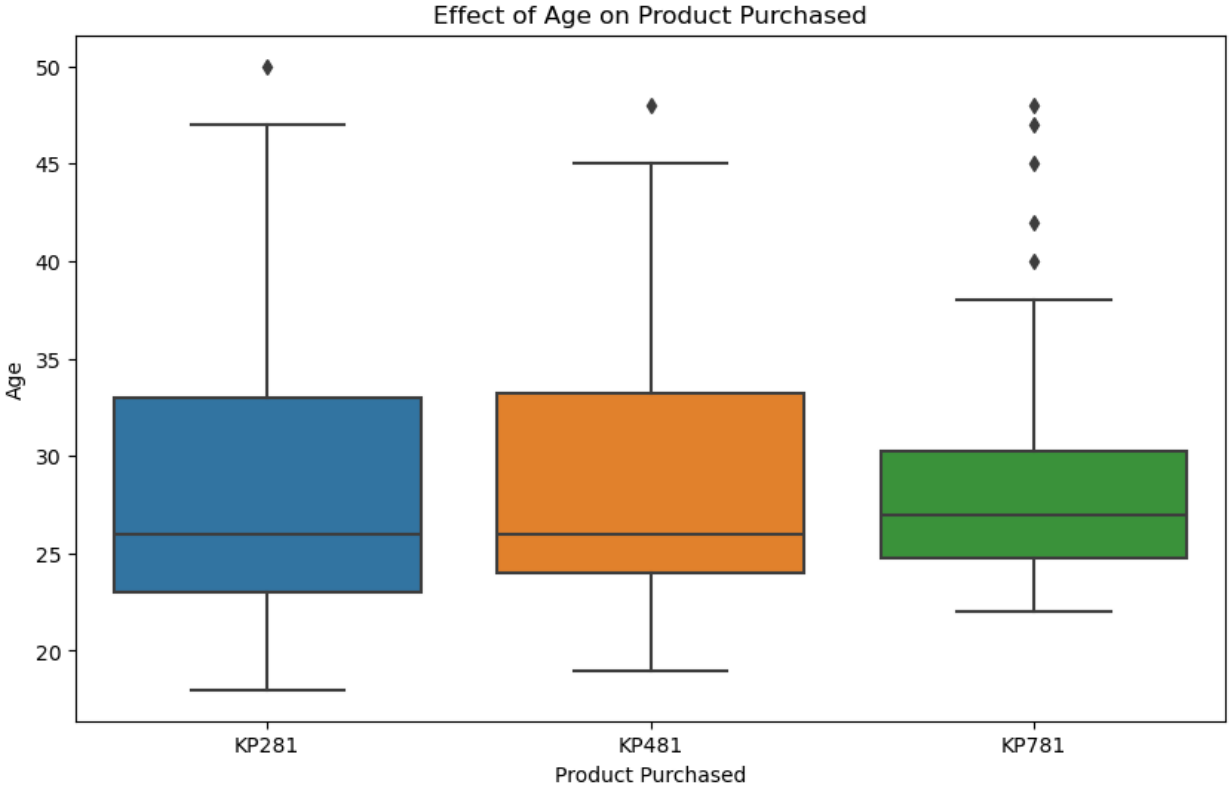
```
plt.figure(figsize=(10, 6))
sns.countplot(x='MaritalStatus', hue='Product', data=aerofit)
plt.title('Effect of Marital Status on Product Purchased')
plt.xlabel('Marital Status')
plt.ylabel('Count')
plt.legend(title='Product Purchased')
plt.show()

# Step 4: Visualize the effect of age on product purchased using
histplot
plt.figure(figsize=(10, 6))
sns.histplot(data=aerofit, x='Age', hue='Product', multiple='stack',
kde=True)
plt.title('Distribution of Age for Different Products Purchased')
plt.xlabel('Age')
plt.ylabel('Count')
plt.show()

# Step 5: Visualize the effect of age on product purchased using
boxplot
plt.figure(figsize=(10, 6))
sns.boxplot(x='Product', y='Age', data=aerofit)
plt.title('Effect of Age on Product Purchased')
plt.xlabel('Product Purchased')
plt.ylabel('Age')
plt.show()

# Step 6: Visualize the effect of age group on product purchased using
countplot
plt.figure(figsize=(10, 6))
sns.countplot(x='age_group', hue='Product', data=aerofit)
plt.title('Effect of Age Group on Product Purchased')
plt.xlabel('Age Group')
plt.ylabel('Count')
```

```
plt.legend(title='Product Purchased')
plt.show()
```

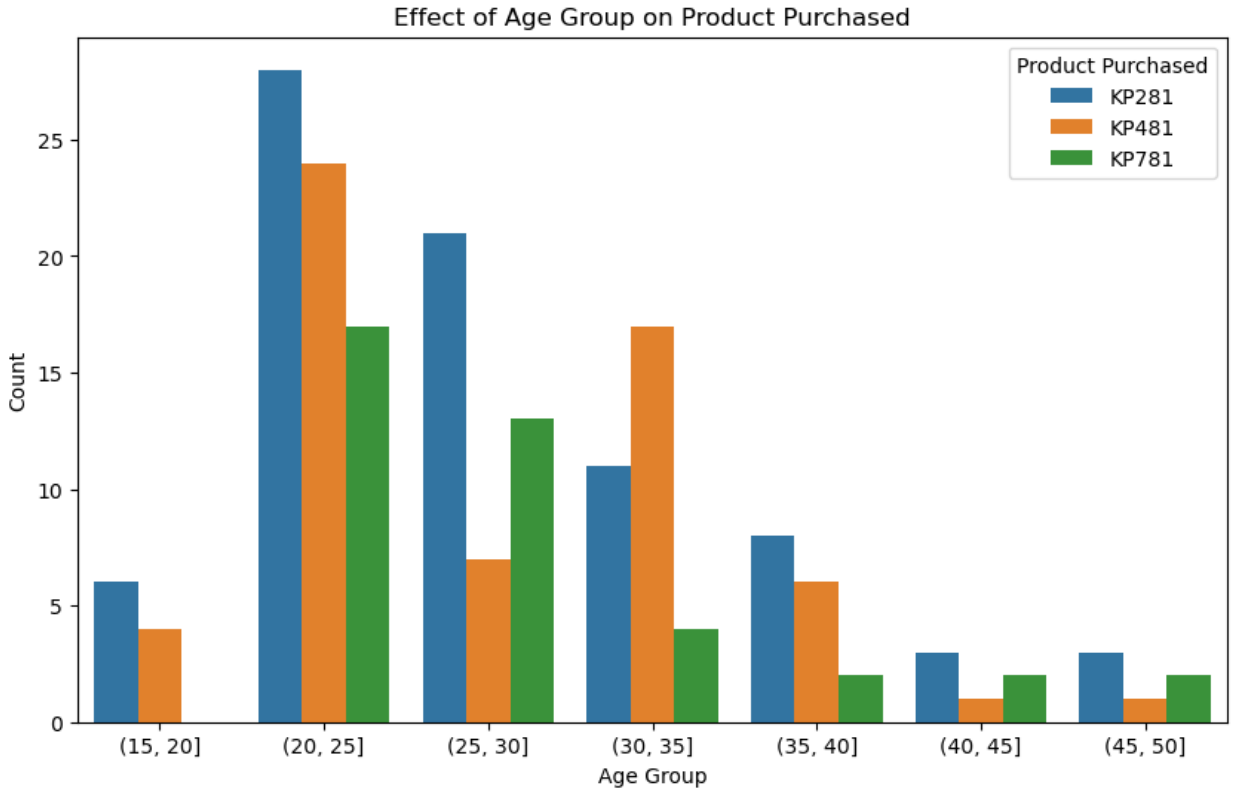



```
C:\Users\User\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```


Distribution of Age for Different Products Purchased
Product
KP281
KP481
KP781
Count
Age
0
10
20
30
40
50
20
25
30
35
40
45
50


Effect of Age on Product Purchased
Age
Product Purchased
KP281
KP481
KP781
20
25
30
35
40
45
50

```
C:\Users\User\anaconda3\Lib\site-packages\seaborn\categorical.py:641:
FutureWarning: The default of observed=False is deprecated and will be
changed to True in a future version of pandas. Pass observed=False to
retain current behavior or observed=True to adopt the future default
and silence this warning.
  grouped_vals = vals.groupby(grouper)
C:\Users\User\anaconda3\Lib\site-packages\seaborn\categorical.py:641:
FutureWarning: The default of observed=False is deprecated and will be
changed to True in a future version of pandas. Pass observed=False to
retain current behavior or observed=True to adopt the future default
and silence this warning.
  grouped_vals = vals.groupby(grouper)
```



Effect of Marital Status on Product Purchased

- **Observation**:
  - The count plot shows the distribution of products purchased by different marital statuses.
  - Single individuals have a higher frequency of purchasing the product compared to partnered individuals.
  - This suggests that marital status may influence the type of product purchased, with single individuals being more likely to buy certain products.

Effect of Age on Product Purchased

- **Observation**:

  - The histogram displays the age distribution for each product purchased, showing how age varies across products.
  - The age distribution indicates that younger individuals (around ages 18-25) tend to purchase certain products more frequently than older age groups.
  - The KDE (Kernel Density Estimate) lines suggest that certain products are more popular within specific age ranges.
- **Box Plot Analysis**:
  - The box plot illustrates the central tendency and spread of age for different products purchased.
  - The median age varies across different products, indicating age-specific preferences.
  - Some products have a wider interquartile range (IQR), suggesting a broader age appeal, while others have a narrower IQR, indicating a more targeted age group.

Effect of Age Group on Product Purchased

- **Observation**:
  - The count plot shows the distribution of products purchased by different age groups.
  - There are significant differences in the counts of products purchased across different age groups.
  - This suggests that the age group has a substantial impact on the choice of product, with certain age groups favoring specific products more than others.

Q4. Representing the marginal probability like - what percent of customers have purchased KP281, KP481, or KP781 in a table (can use pandas.crosstab here)

```
pd.crosstab(aerofit['Product'], aerofit['age_group'], margins =True)
```

```
age_group  (15, 20]  (20, 25]  (25, 30]  (30, 35]  (35, 40]  (40, 45]
\
Product

KP281              6        28        21        11         8         3

KP481              4        24         7        17         6         1

KP781              0        17        13         4         2         2

All               10        69        41        32        16         6


age_group  (45, 50]  All
Product
KP281              3   80
KP481              1   60
KP781              2   40
All                6  180
```

Most of the people purchasing the products are of the age group of 20-15, and as the age_group increases we can see a decrease in number. However in 45-50 years we see a slight rise than that of 40-45 age_group.

Around 38%(63/167) are in the age category of 20-25. And among that age category KP781 has more percentage of people (13/29 = 45%) than that of KP281 - 34% and KP481 - 34%

```
aerofit_gender = aerofit['Gender'].value_counts()
aerofit_gender
```

```
Gender
Male      104
Female     76
Name: count, dtype: int64
```

```
pd.crosstab(aerofit['Product'], aerofit['Gender'], margins = True)
```

Gender	Female	Male	All
Product			
KP281	40	40	80
KP481	29	31	60
KP781	7	33	40
All	76	104	180

For KP281 and KP481 the percentage of Females using the products are almost equal comparing to Males. However for KP781 around 90% of the people using the products are Male

Q6. With all the above steps you can answer questions like: What is the probability of a male customer buying a KP781 treadmill?

15%(26/167) male customer buying KP781. However around 90%(26/29) amoung the people purchasing KP781 are males.

```
aerofit_edu = aerofit['Education'].value_counts()
aerofit_edu
```

```
Education
16    85
14    55
18    23
15     5
13     5
12     3
21     3
20     1
Name: count, dtype: int64
```

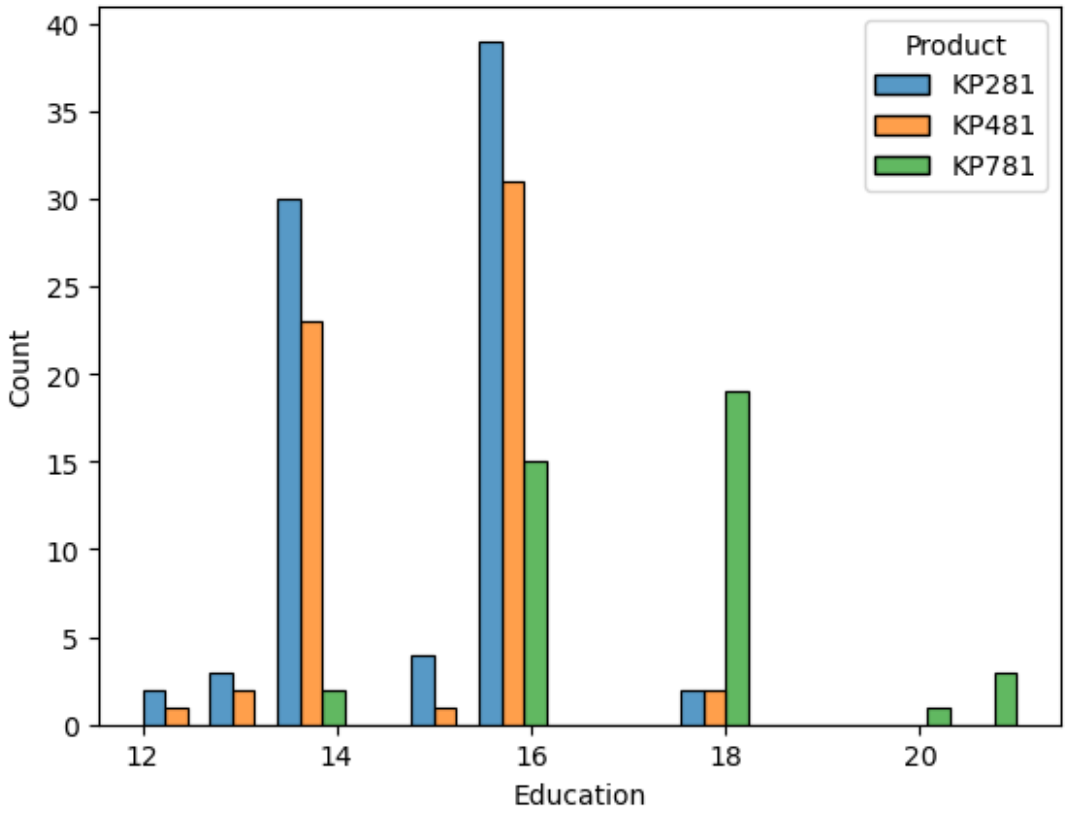
```
pd.crosstab(aerofit['Product'], aerofit['Education'], margins = True)
```

```
Education  12  13  14  15  16  18  20  21  All
Product
KP281       2   3  30   4  39   2   0   0   80
KP481       1   2  23   1  31   2   0   0   60
KP781       0   0   2   0  15  19   1   3   40
All         3   5  55   5  85  23   1   3  180
```

Around 80% of the people have done 16 years of Education, out of which around 50% (38/79) are KP281 users Around 50%(14/29) people using KP281 are having 18years of Education, and among this Education categories 77%(14/18) people are using KP781.

```
sns.histplot(data=aerofit, x='Education', hue='Product', multiple =
'dodge')
plt.show()
```

```
C:\Users\User\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```




```
aerofit_material = aerofit['MaritalStatus'].value_counts()
aerofit_material
```

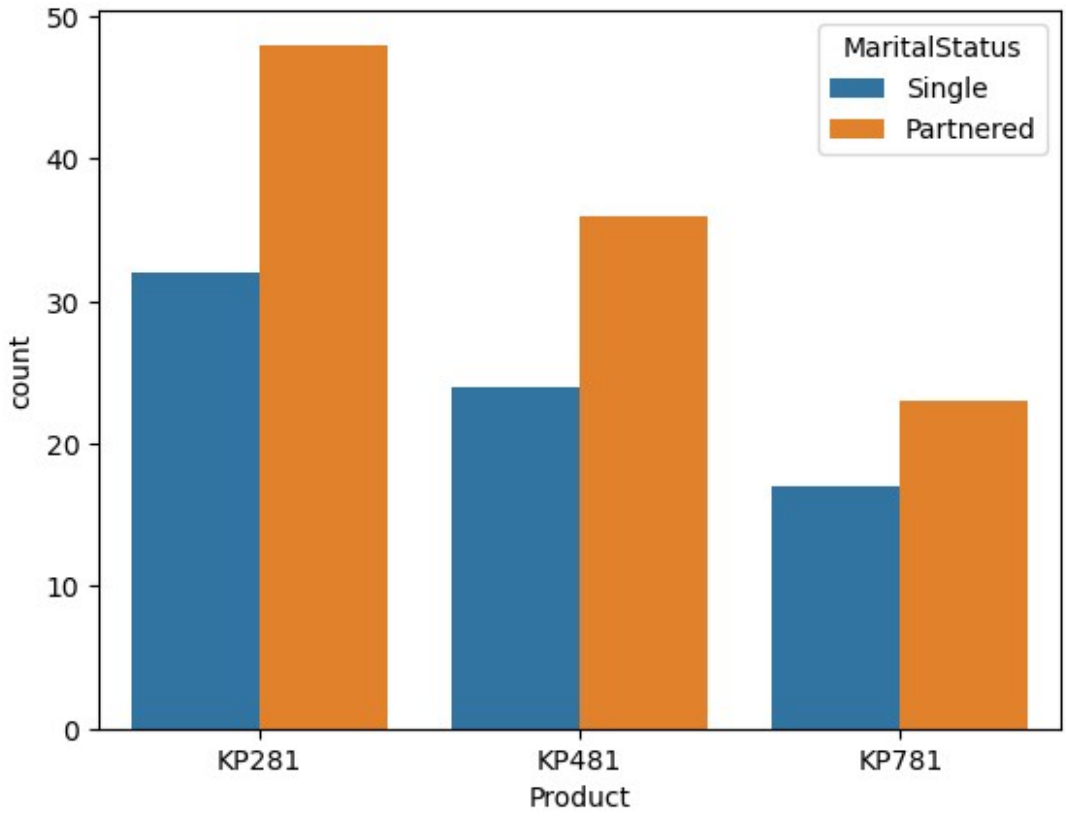
```
MaritalStatus
Partnered    107
Single        73
Name: count, dtype: int64
```

```
pd.crosstab(aerofit['Product'], aerofit['MaritalStatus'], margins =
True)
```

MaritalStatus	Partnered	Single	All
Product			
KP281	48	32	80
KP481	36	24	60
KP781	23	17	40
All	107	73	180

There are more maried people coming for fitness than single. The gap between the Single and Partnered counple are decreasing as we move from KP281 to KP481 and the KP781.

```
sns.countplot(data=aerofit, x='Product', hue='MaritalStatus',
dodge=True)
plt.show()
```



```
aerofit_usage = aerofit['Usage'].value_counts()
aerofit_usage
```

```
Usage
3    69
4    52
2    33
5    17
6     7
7     2
Name: count, dtype: int64
```

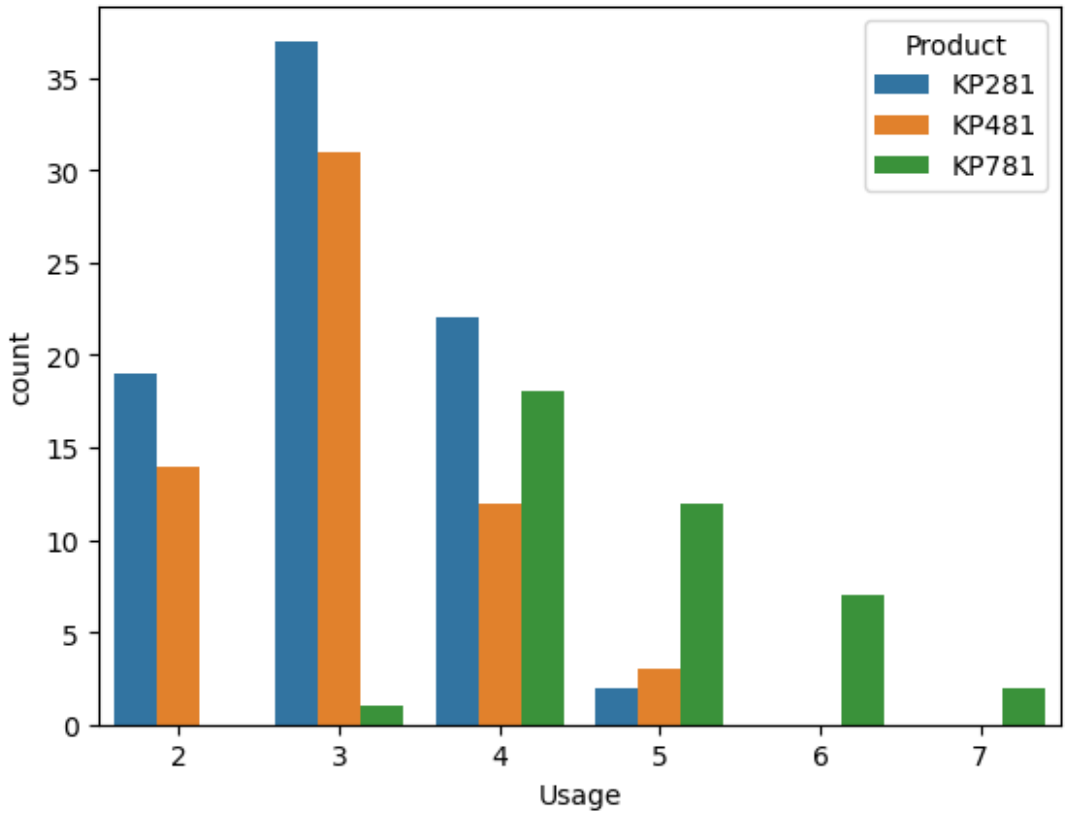
```
pd.crosstab(aerofit['Product'], aerofit['Usage'], margins = True)
```

Usage	2	3	4	5	6	7	All
Product							
KP281	19	37	22	2	0	0	80
KP481	14	31	12	3	0	0	60
KP781	0	1	18	12	7	2	40
All	33	69	52	17	7	2	180

Around 61%((33+69)/167) people plan to use the threadmills on an average in a week and around 90% people uses upto 4 times. Among the people using KP481 52%(31/59) uses it average 3 times in a week. Among the people using the threadmills 5 and above times a week 81% (9+3+1)/(12+3+1) of them uses KP781

```
sns.countplot(data=aerofit, x='Usage', hue='Product', dodge = True)
```

```
<Axes: xlabel='Usage', ylabel='count'>
```




```
aerofit_fitness = aerofit['Fitness'].value_counts()
aerofit_fitness
```

```
Fitness
3    97
5    31
2    26
4    24
1     2
Name: count, dtype: int64
```

```
pd.crosstab(aerofit['Product'], aerofit['Fitness'], margins = True)
```

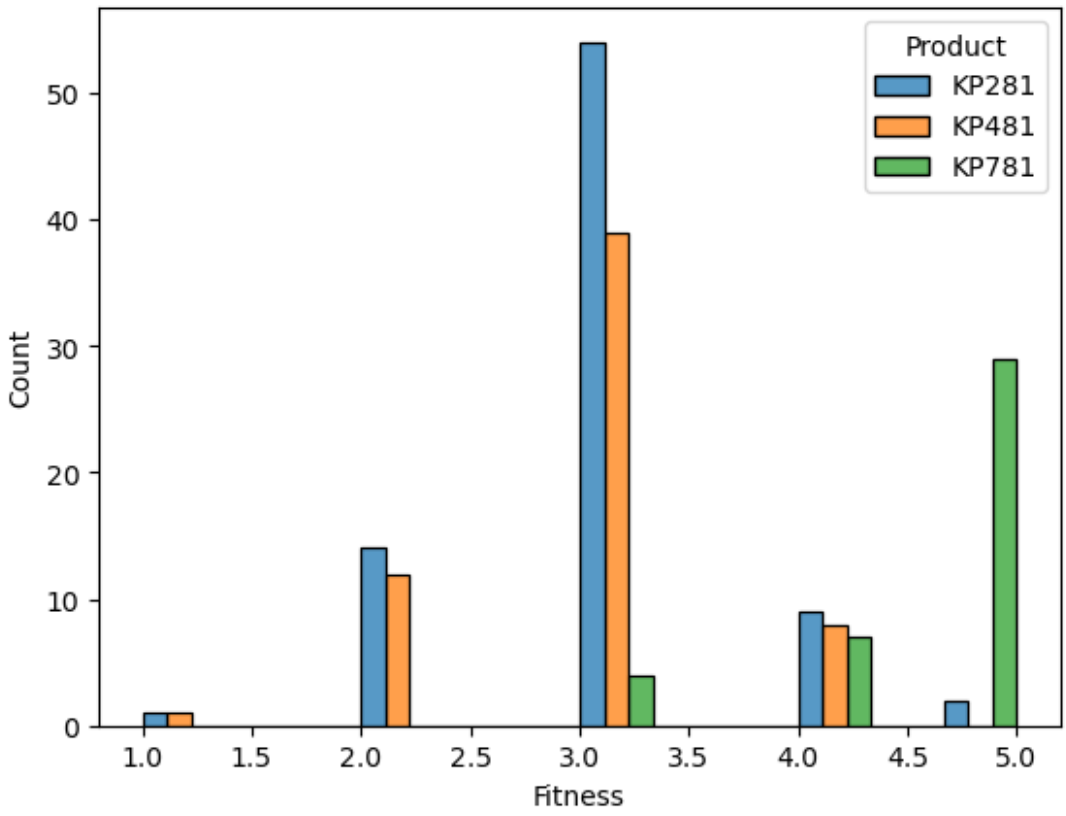
Fitness	1	2	3	4	5	All
Product						
KP281	1	14	54	9	2	80
KP481	1	12	39	8	0	60
KP781	0	0	4	7	29	40
All	2	26	97	24	31	180

More fitness rating of 4 and 5 are given by the users KP781 Though more people are using KP281, however the advanced model KP781 users are more satisified with better self rated fitness

```
sns.histplot(data=aerofit, x='Fitness', hue='Product', multiple =
'dodge')
```

```
C:\Users\User\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

```
<Axes: xlabel='Fitness', ylabel='Count'>
```




```
aerofit_income = aerofit['Income'].value_counts().sort_index(ascending
= False)
aerofit_income = pd.DataFrame(aerofit_income)
aerofit_income.shape
```

```
(62, 1)
```

```
income_bins = range(25000, 110000, 10000)
aerofit['income_group'] = pd.cut(aerofit['Income'], income_bins, right
= True)
aerofit.head()
```

```
  Product  Age  Gender  Education MaritalStatus  Usage  Fitness
Income  \
```

```
0   KP281   18    Male          14         Single      3         4
29562
1   KP281   19    Male          15         Single      2         3
31836
2   KP281   19  Female          14      Partnered      4         3
30699
3   KP281   19    Male          12         Single      3         3
32973
4   KP281   20    Male          13      Partnered      4         2
35247

   Miles age_group    income_group
0    112  (15, 20]  (25000, 35000]
1     75  (15, 20]  (25000, 35000]
2     66  (15, 20]  (25000, 35000]
3     85  (15, 20]  (25000, 35000]
4     47  (15, 20]  (35000, 45000]
```

```
pd.crosstab(aerofit['Product'], aerofit['income_group'], margins
=True)
```

```
income_group  (25000, 35000]  (35000, 45000]  (45000, 55000]  (55000,
65000]  \
Product

KP281                      8              26              35
9
KP481                      6               9              33
10
KP781                      0               0               9
7
All                       14              35              77
26

income_group  (65000, 75000]  (75000, 85000]  (85000, 95000]  (95000,
105000]  \
Product

KP281                      2               0               0
0
KP481                      2               0               0
0
KP781                      3               4              11
6
All                        7               4              11
6

income_group  All
Product
KP281          80
```

```
KP481          60
KP781          40
All           180
```

Insights

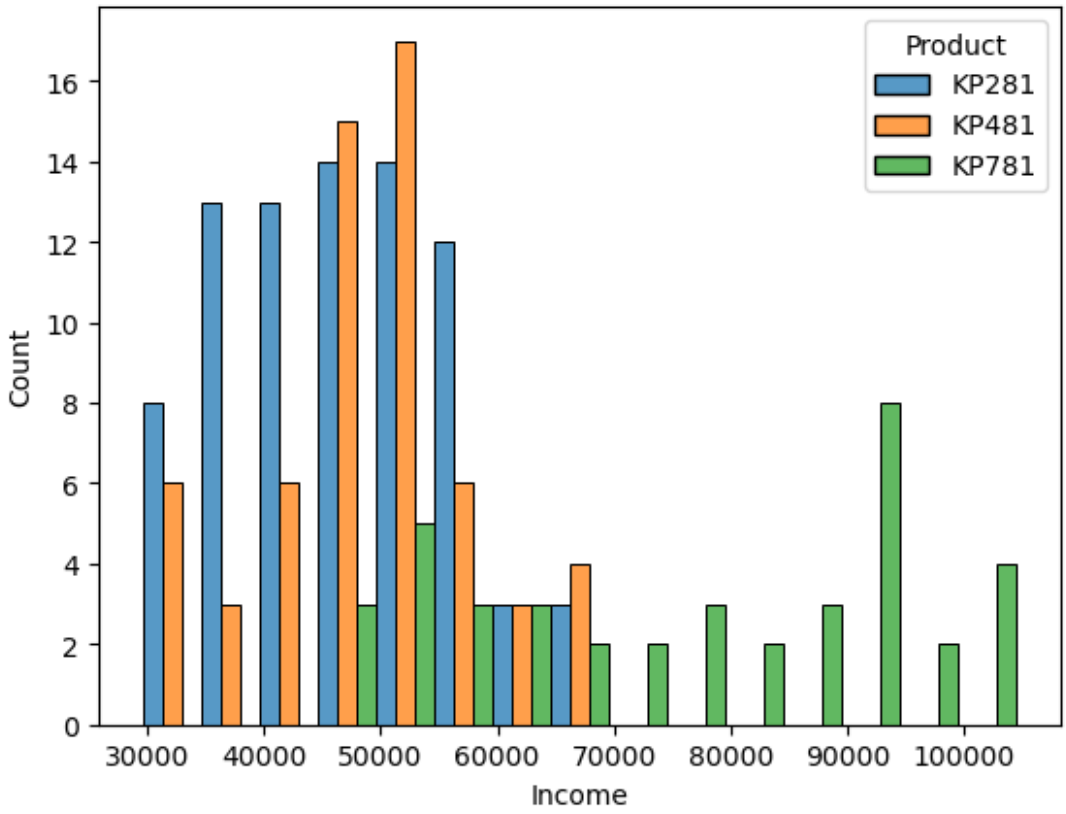
Effect of Income on Product Purchased

- **Observation**:
  - Most of the lower-income group people opt for KP281, while it is seen that most of them don't use KP781.
  - People of the higher income group prefer KP781.
  - More than 38% (6+5)/29 of people using KP781 are in the income group of more than 85,000.
  - Around 88% of the KP281 users are in the income category of 35,000-55,000.

```
sns.histplot(data=aerofit, x='Income', hue='Product', multiple =
'dodge')
```

```
C:\Users\User\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

```
<Axes: xlabel='Income', ylabel='Count'>
```




```
aerofit['Miles'].unique()
```

```
array([112,  75,  66,  85,  47, 141, 103,  94, 113,  38, 188,  56,
132,
       169,  64,  53, 106,  95, 212,  42, 127,  74, 170,  21, 120,
200,
       140, 100,  80, 160, 180, 240, 150, 300, 280, 260, 360],
dtype=int64)
```

```
miles_bins = range(0, 201, 25)
aerofit['miles_group'] = pd.cut(aerofit['Miles'], miles_bins, right =
True)
aerofit.head()
```

```
  Product  Age  Gender  Education MaritalStatus  Usage  Fitness
Income  \
0   KP281   18    Male         14        Single      3        4
29562
1   KP281   19    Male         15        Single      2        3
31836
2   KP281   19  Female         14     Partnered      4        3
30699
3   KP281   19    Male         12        Single      3        3
32973
4   KP281   20    Male         13     Partnered      4        2
```

```
35247
```

```
   Miles age_group    income_group miles_group
0    112  (15, 20]  (25000, 35000]  (100, 125]
1     75  (15, 20]  (25000, 35000]    (50, 75]
2     66  (15, 20]  (25000, 35000]    (50, 75]
3     85  (15, 20]  (25000, 35000]   (75, 100]
4     47  (15, 20]  (35000, 45000]    (25, 50]
```

```
pd.crosstab(aerofit['Product'], aerofit['miles_group'], margins =True)
```

```
miles_group  (0, 25]  (25, 50]  (50, 75]  (75, 100]  (100, 125]  (125,
150]  \
Product

KP281              0        12        26         24           12
4
KP481              1         4        16         23            8
5
KP781              0         0         0          8            4
5
All                1        16        42         55           24
14

miles_group  (150, 175]  (175, 200]  All
Product
KP281                1           1   80
KP481                2           0   59
KP781                6          12   35
All                  9          13  174
```

Insights

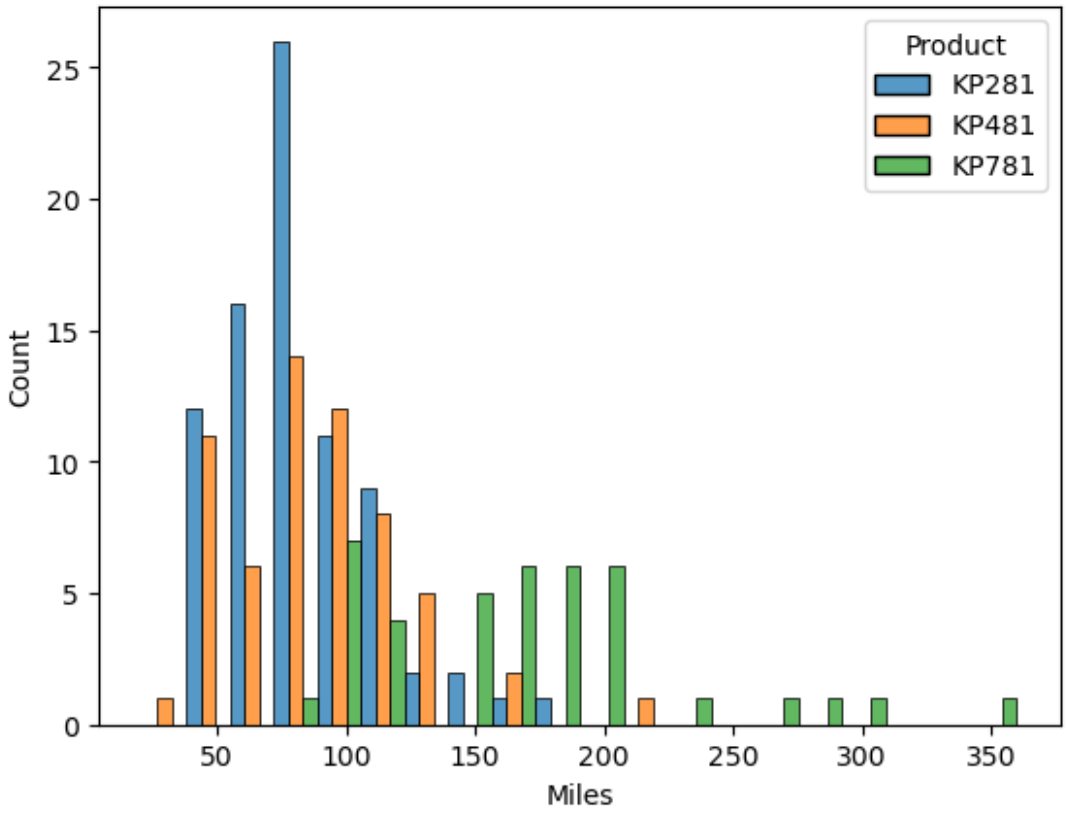
Effect of Miles Walked on Product Purchased

- **Observation**:
  - More than 41% ((6+6)/29) of people who use KP781 are able to walk more than 150 miles, and more than 72% walk more than 100 miles.
  - However, most of the people using KP281 and KP481 are able to walk only 50-100 miles.

```
sns.histplot(data=aerofit, x='Miles', hue='Product', multiple =
'dodge')
```

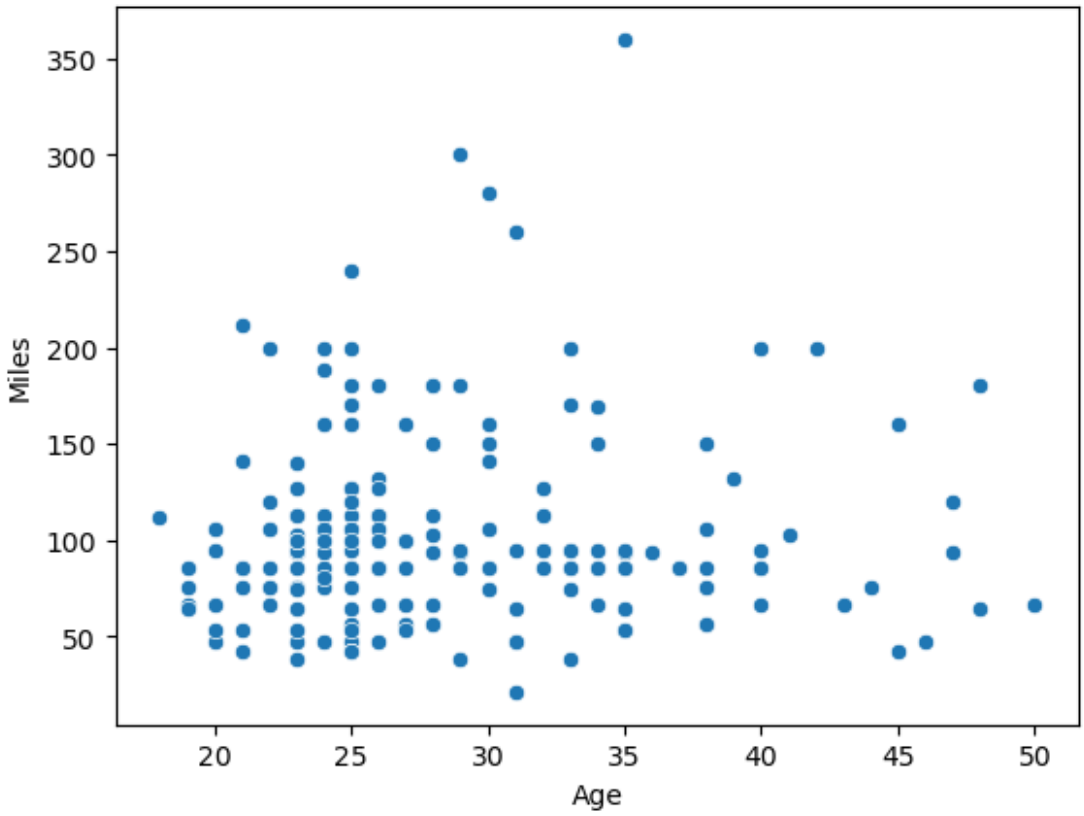
```
C:\Users\User\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

```
<Axes: xlabel='Miles', ylabel='Count'>
```



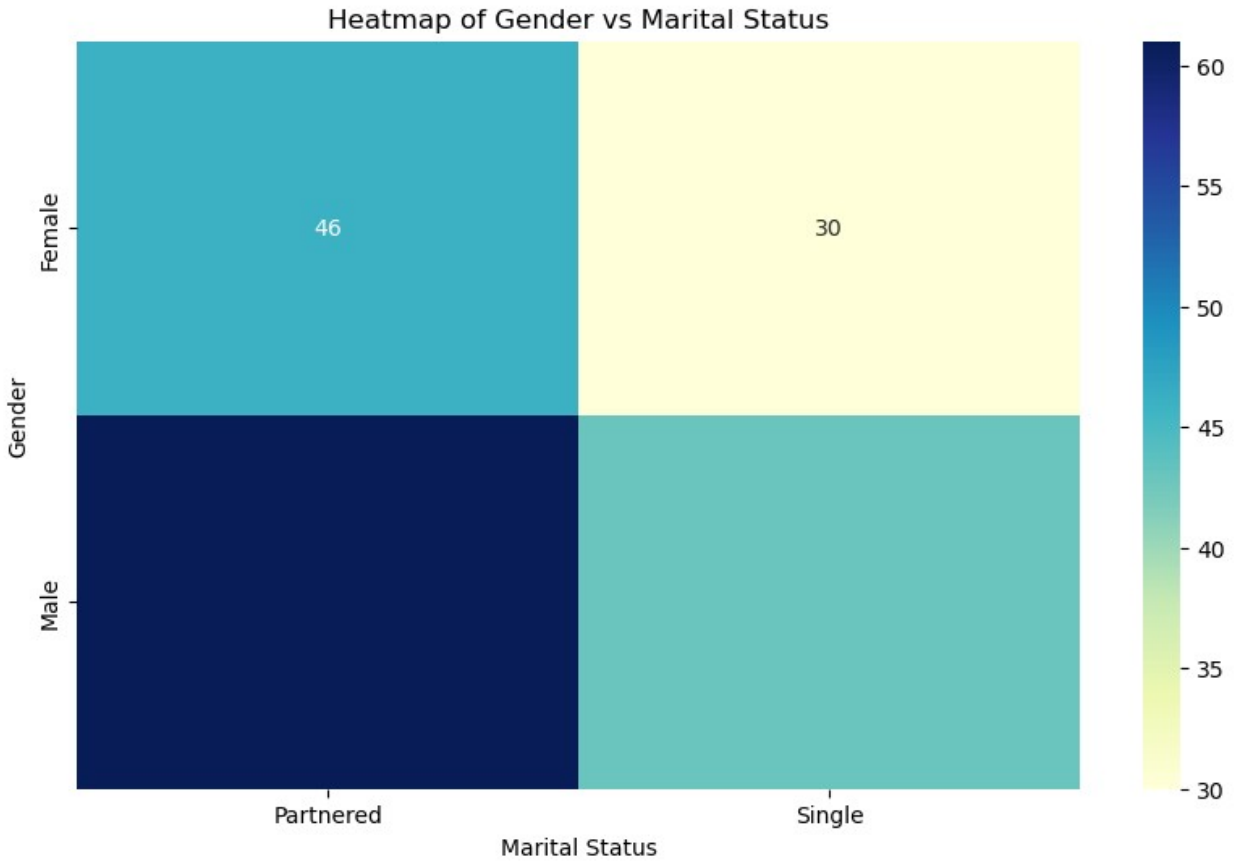

```
sns.scatterplot(data = aerofit, x = 'Age', y = 'Miles')
```

```
<Axes: xlabel='Age', ylabel='Miles'>
```



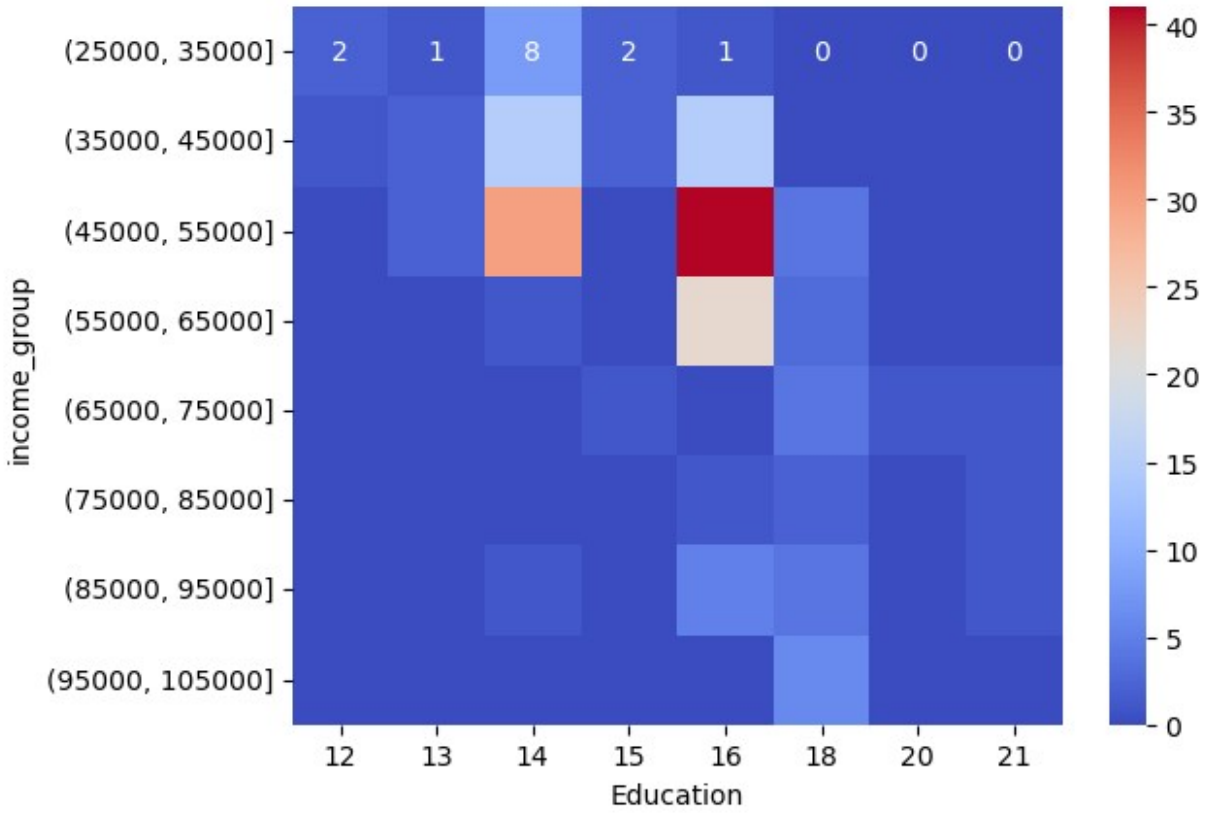
As the Age progress the number of peoples walking more miles decreses.

```
gender_Maritalstatus = pd.crosstab(aerofit['Gender'],
aerofit['MaritalStatus'])

plt.figure(figsize=(10, 6))
sns.heatmap(gender_Maritalstatus, annot=True, cmap='YlGnBu', fmt='d')
plt.title('Heatmap of Gender vs Marital Status')
plt.xlabel('Marital Status')
plt.ylabel('Gender')
plt.show()
```




```
inc_edu = aerofit[['income_group', 'Education']]
inc_edu = pd.crosstab(inc_edu['income_group'], inc_edu['Education'])
sns.heatmap(inc_edu, annot = True, cmap = 'coolwarm' )
```

```
<Axes: xlabel='Education', ylabel='income_group'>
```



Insights

Relationship between Education and Income

- **Observation**:
  - Most people belong to 14 & 16 years of education, with an annual income of around 45-55 thousand dollars.
  - As the education level increases, the income also increases.
  - However, people with the highest income group (i.e., 95,000-105,000 dollars) have 18 years of education.

Final Insights and Recommendations

Product Suitability and Target Audience

- **KP781**:
  - Suitable for people who are more interested in fitness.
  - Can be considered a high-end model.
  - Recommendation: Market KP781 to fitness enthusiasts and higher-income groups.
- **KP281**:
  - Considered a starter level product.
  - Suitable for individuals who are not heavily into fitness or may belong to lower income groups.

    - Recommendation: Promote KP281 as an entry-level product for beginners or those with budget constraints.
- **KP481**:
    - Positioned as a middle-level product.
    - Recommendation: Develop strategies to motivate KP281 users to upgrade to KP481, and KP481 users to upgrade to KP781.

Marital Status and Product Usage

- **Marital Status Insights**:
    - A significant number of married people are using treadmills.
    - Recommendation: Introduce special offers for couples to attract more married users and enhance customer loyalty.