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

About Aerofit

Aerofit is a leading brand in the field of fitness equipment. Aerofit provides a product range including machines such as treadmills, exercise bikes, gym equipment, and fitness accessories to cater to the needs of all categories of people.

Business Problem:

The market research team at AeroFit wants to identify the characteristics of the target audience for each type of treadmill offered by the company, to provide a better recommendation of the treadmills to the new customers. The team decides to investigate whether there are differences across the product with respect to customer characteristics.

Objectives:

1. Perform descriptive analytics to create a customer profile for each AeroFit treadmill product by developing appropriate tables and charts.
2. For each AeroFit treadmill product, construct two-way contingency tables and compute all conditional and marginal probabilities along with their insights/impact on the business.

Basic Analysis

```
df = pd.read_csv("Aerofit.csv")
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)
```

The data has 9 columns and 180 rows. The columns include: Product, Age, Gender, Education, Marital status, Usage, Fitness, Income and Miles.

There are 3 types of products i.e. KP281, KP481. KP781 • The KP281 is an entry-level treadmill that sells for $1,500.

• The KP481 is for mid-level runners that sell for $1,750.

• The KP781 treadmill is having advanced features that sell for $2,500.

We can see that KP781 is the most expensive treadmill that the company have. It is the top level product hence we can conclude higher the level, the more expensive the product will be as per this given information.

```
df.describe() #Statistical Analysis
```

	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

The mean **Age** of the customers is nearly around 28.7 years old.

The mean **Eduaction** of customers is 15 years with minimum of 12 years and maximum of 21 years.

The mean **Usage** of treadmill is about 3.4 times

The mean **fitness** rating of the customers is 3.3

The mean **Income** is around 53719, majority of people earn in the age bracket.

The customer on average runs/walks for around 103 **miles**.

```
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
```

The data type of the Product, Gender and Marital status column is object(String) and all other columns are of integer data type. There are no null values in the data set.

Unique Values and Value counts:

```
df["Product"].unique()
```

```
array(['KP281', 'KP481', 'KP781'], dtype=object)
```

```
df["Product"].value_counts()
```

```
KP281    80
KP481    60
KP781    40
Name: Product, dtype: int64
```

```
df["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])
```

```
df["Age"].nunique() #no. of unique ages
```

```
32
```

```
df["Gender"].value_counts() # Total number of males and females
```

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

```
df["Education"].unique()
```

```
array([14, 15, 12, 13, 16, 18, 20, 21])
```

```
df["Education"].nunique() #no. of unique years in education
```

```
8
```

```
df["MaritalStatus"].unique()
```

```
array(['Single', 'Partnered'], dtype=object)
```

```
df["MaritalStatus"].value_counts() # Total number of people categorised in partnered and Single status
```

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

```
df["Usage"].unique()
```

```
array([3, 2, 4, 5, 6, 7])
```

```
df["Usage"].value_counts()
```

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

```
df["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])
```

```
df["Miles"].nunique()
```

```
37
```

```
df["Miles"].value_counts
```

```
<bound method IndexOpsMixin.value_counts of 0      112
1       75
2       66
3       85
4       47
      ...
175    200
176    200
177    160
178    120
179    180
Name: Miles, Length: 180, dtype: int64>
```

Summing up the basic numerical analysis:

Customers are purchasing KP281, the most given is is the cheapest one.

People are using Treadmill for atleast 3 days in a week

People with partners prefer buying Treadmill as opposed to people who are single.

Categoring the Fitness Rating into Descriptive categories/categorical variable:

Made a new columan to guage the interpreation of the fitness score and what category an individual belongs to with 1 being poor shape and 5 being in excellent shape.

```
df_1 = df
df_1['Fitness_category'] = df.Fitness
df_1.head()
```

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

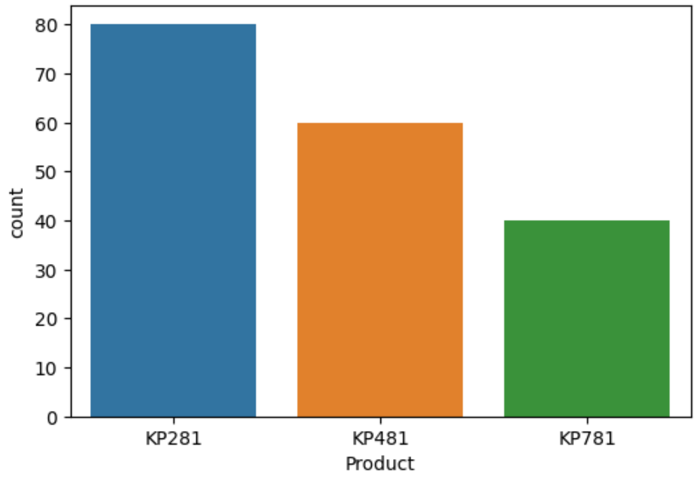
```
df_1["Fitness_category"].replace({1:"Poor Shape",2:"Bad Shape",3:"Average Shape",4:"Good Shape",5:"Excellent Shape"},inplace=True)
df_1.head()
```

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

Univariate and Bivariate Analysis

```
plt.figure(figsize=(6, 4))
sns.countplot(data=df,x="Product")
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

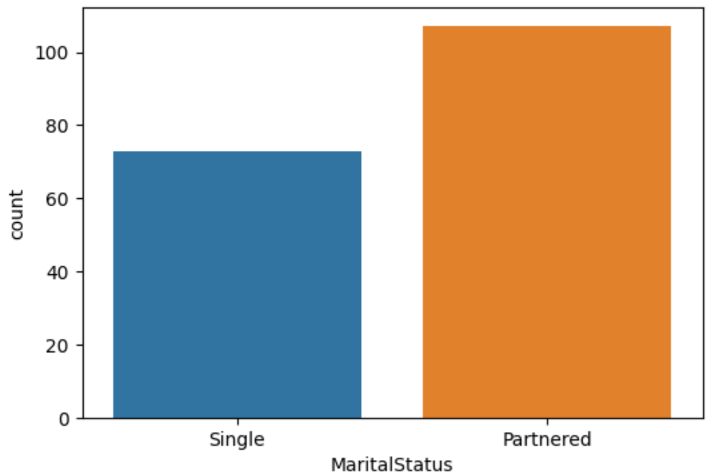


Product analysis using the count plot.

We can say via this that KP281 is the highest purchased treadmill and with others being the 2nd highest and the lowest.

```
plt.figure(figsize=(6, 4))
sns.countplot(data=df,x="MaritalStatus")
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```



MaritalStatus analysis using the count plot.

We can see that Married/partnered people tend to buy more treadmill than the single people

```
plt.figure(figsize=(6, 4))
sns.countplot(data=df,x="Gender",palette='hls')
plt.show
```

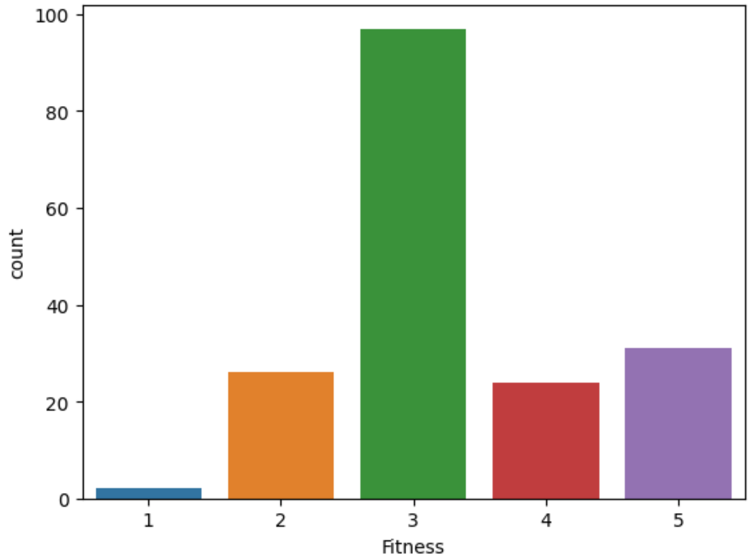
```
<function matplotlib.pyplot.show(close=None, block=None)>
```

Gender analysis using the count plot:

We can see that male customers are more interested in buying the treadmill as compared to the female customers.

```
sns.countplot(data=df, x="Fitness", palette='tab10')
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```



Fitness analysis using the count plot:

Most of the customers(about 90%) have rated their fitness rating as average i.e 3 which describes they being in average shape. About 35% feel they are in excellent shape with teh rating of 5

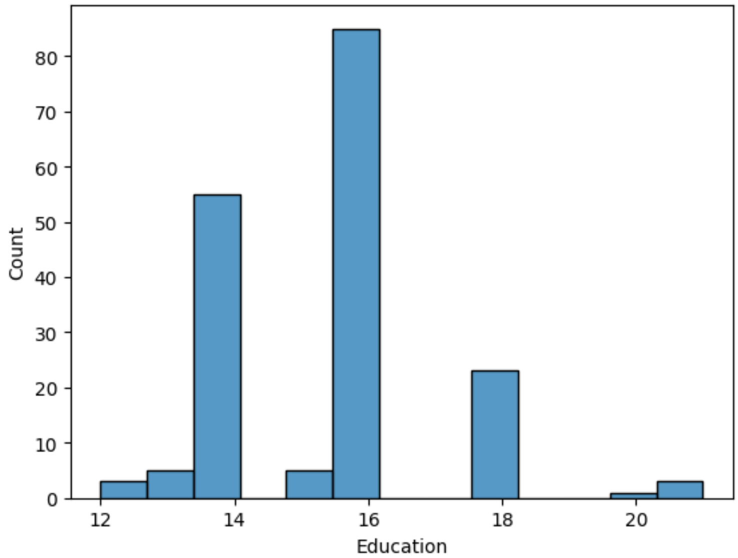
```
sns.histplot(df.Income,kde=True)
plt.show()
```

Income Analysis using Histogram/density analyis

Majority of people who have purchased the product has their income between 40K and 60K;

```
sns.histplot(data=df,x="Education")
```

<Axes: xlabel='Education', ylabel='Count'>

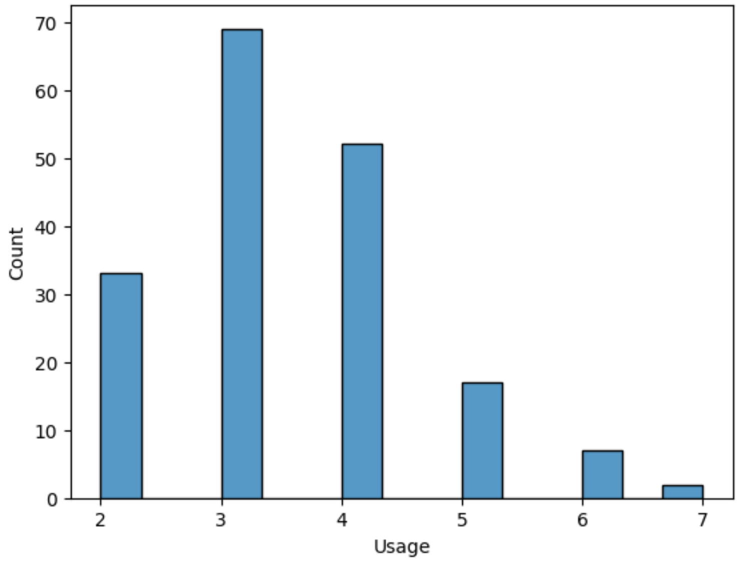


Education analysis using histogram

We can see that majority of customers have 16 as their education and customers have 20 as the least education.

```
sns.histplot(data=df,x="Usage")
```

<Axes: xlabel='Usage', ylabel='Count'>

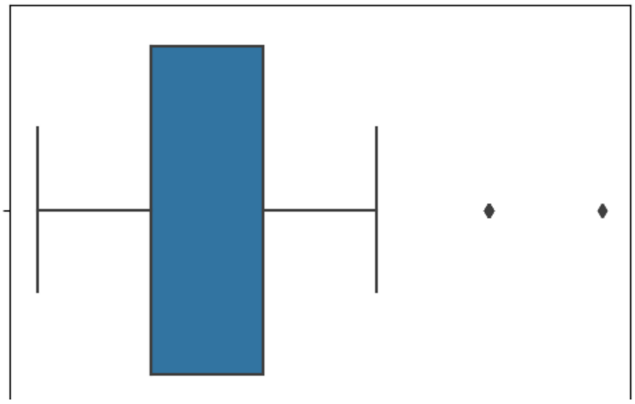


Usage analysis using histogram:

3 days per week is the most commonly used time among all the customers, followed by 4 days and 2 days.

Box plots for Categorical Analysis

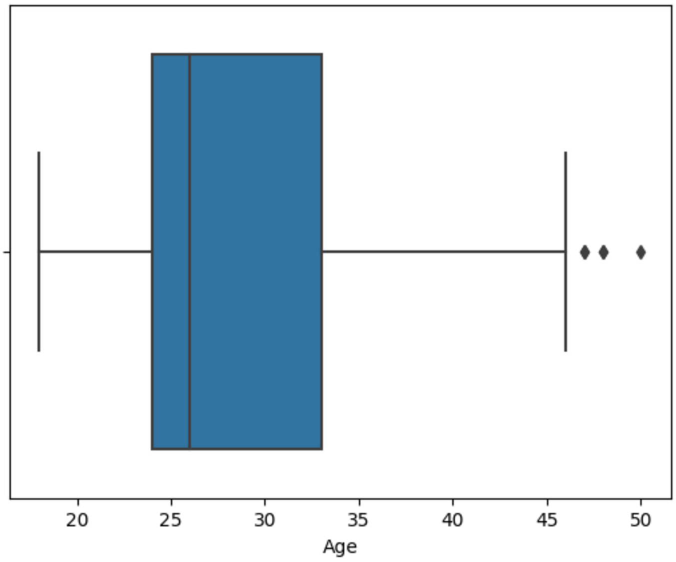
```
plt.figure(figsize=(6,4))
sns.boxplot(data=df, x="Usage")
plt.show()
```



Usage Analysis:

We can see that 3 to 4 days are the most common number of days for the users. Very few customers prefer 6 to 7 days(they can be our potential otliers.)

```
sns.boxplot(data=df,x="Age")
plt.show()
```



Age Analysis:

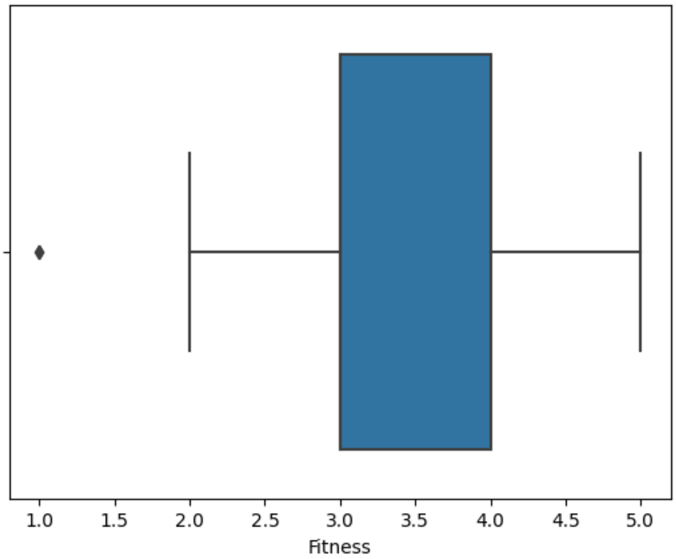
The customers in the age group of 23 to 34 have preferred buying the product more than the other age groups. In the age group of 45 to 50+ there are only a few customers that would prefer buying the product.

```
sns.boxplot(data=df,x="Income")
plt.show()
```

Income Analysis:

We can see that most customers are earning an income of 40K to 60K and they are the one's who tend to buy the product more than any other income groups. Only a very few customers earn above 80K, they can be considered outliers here.

```
sns.boxplot(data=df,x="Fitness")
plt.show()
```



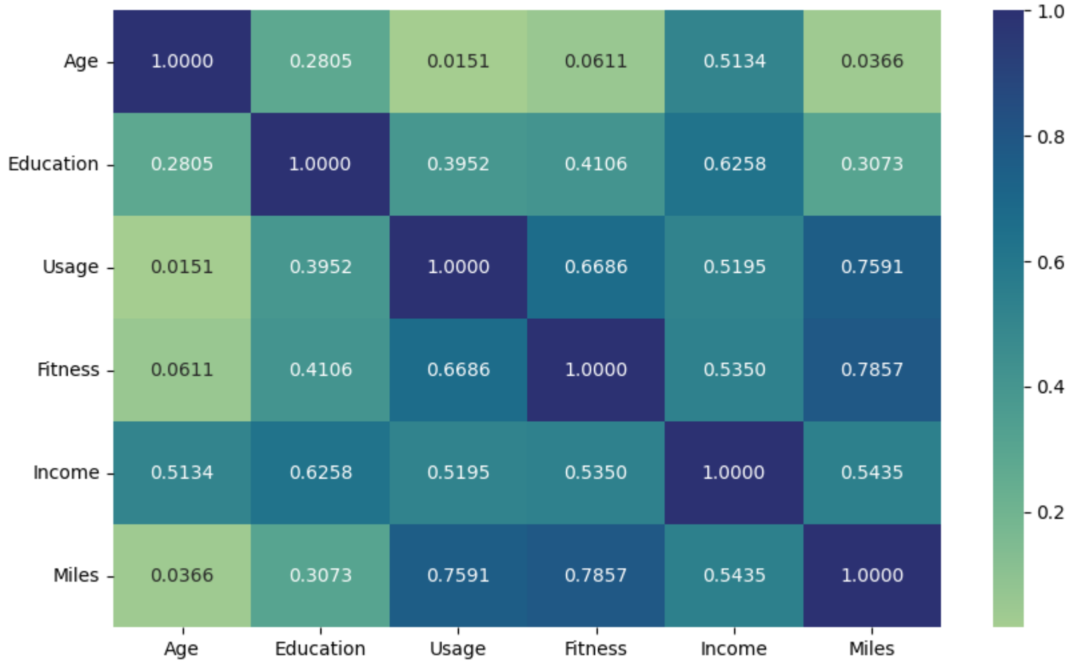
Fitness Analysis

Majority of customers have rated their fitness rating between 3 to 4 which implies being in average and good shape. Only a few customers have rated themselves as 1 i.e. being in poor shape.

Correlation using Pairplots and Heatmaps

```
plt.figure(figsize=(10,6))
ax = sns.heatmap(df.corr(),annot=True,fmt='.4f',cmap='crest')
plt.yticks(rotation=0)
plt.show()
```

```
<ipython-input-130-cab87127048b>:2: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future v
  ax = sns.heatmap(df.corr(),annot=True,fmt='.4f',cmap='crest')
```

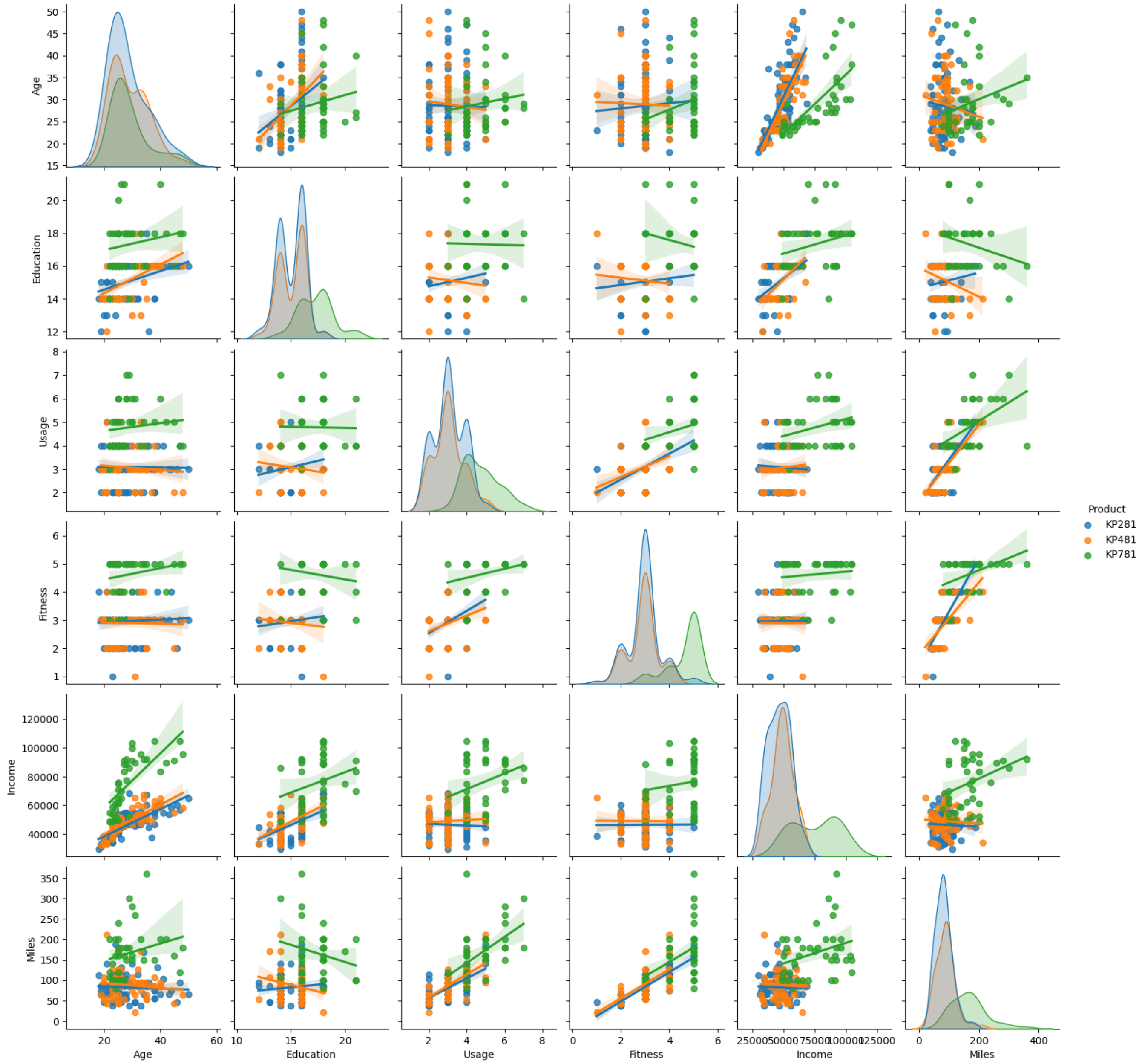


From the above heatmap, a linear correlation is found between the variables

1. Correlation between Age and Miles is 0.03: It suggests a very weak positive linear relationship between age and miles. As age increases, miles can slightly increase too.
2. Correlation between Education and Income is 0.62: It means that there is a strong positive relation between education and income. As education increases, the income increases too.
3. Correlation between Usage and Fitness is 0.66 which suggestes higher the usage of treadmill, higher will be the fitness level
4. Correlation between Fitness and Age is 0.06 suggests that there is almost no positive relation between fitness and age. If age increases the fitness levels tend increase only a little bit.
5. Correlation between Income and Usage is 0.51 suggests that higher the income, higher will be the usage
6. Correlation between Miles and Age is 0.03 suggests a weak positive linear relationship.

```
plt.figure(figsize=(10,6))
sns.pairplot(df,hue="Product", kind="reg")
plt.show()
```

```
<Figure size 1000x600 with 0 Axes>
```



The pair plot above helped us in summarising the data and it is showing us the pairwise realtionship between the variables. It is showing us the exact correlation that we found out using the heatmaps. Some of the variable have liner positive realtionship and some have negative relationship and some have mild to no relation. The data points where there is a lot of skewness, we can say that majority of customers are from that group.

There are certain points that follow no relation and are scattered without any regression line, those pointers are the potential outliers.

Overall the conclusion that we form using the Heatmaps are in line with the coclusion/correlation we have here as well.

Bivariate Analysis

Average of different variables when compared to each product:

```
df.groupby("Product") ["Usage"].mean()
```

```
Product
KP281    3.087500
KP481    3.066667
KP781    4.775000
Name: Usage, dtype: float64
```

```
df.groupby("Product")["Age"].mean()
```

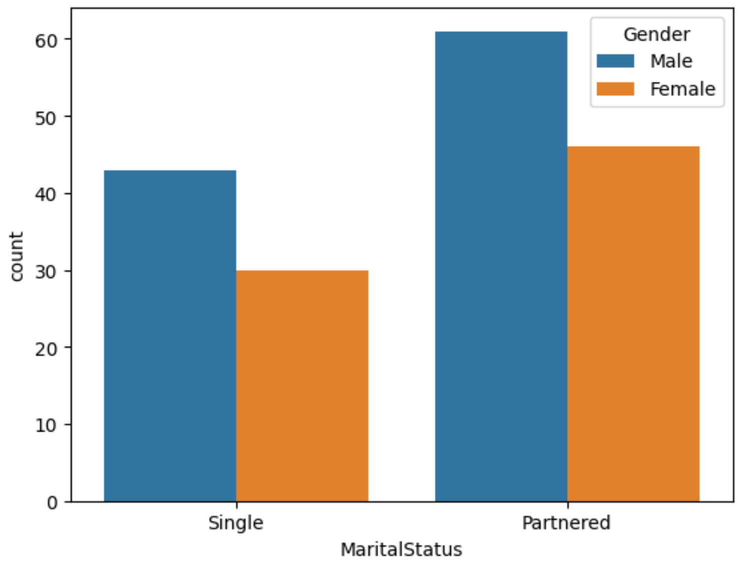
```
Product
KP281    28.55
KP481    28.90
KP781    29.10
Name: Age, dtype: float64
```

```
df.groupby("Product")["Education"].mean()
```

```
Product
KP281    15.037500
KP481    15.116667
KP781    17.325000
Name: Education, dtype: float64
```

```
df.groupby("Product")["Fitness"].mean()
```

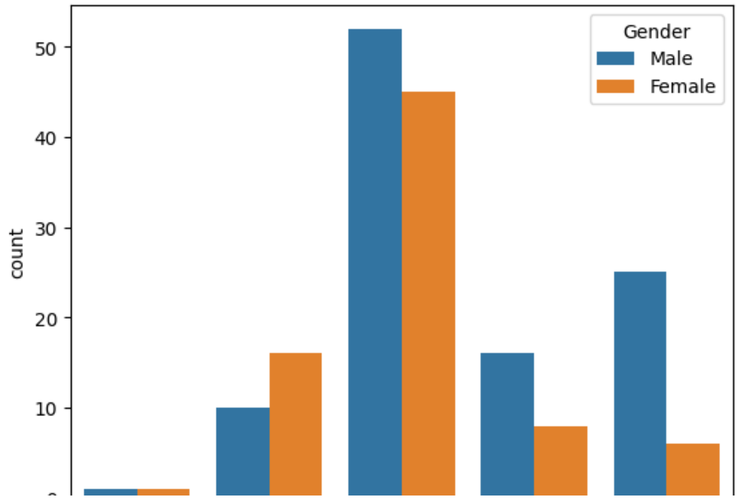
```
sns.countplot(data=df,x="MaritalStatus",hue="Gender")
plt.show()
#count of people based on the gender and marital status
```



We can see that, people with partners prefer to purcahse most of the brand's products as opposed to single people.

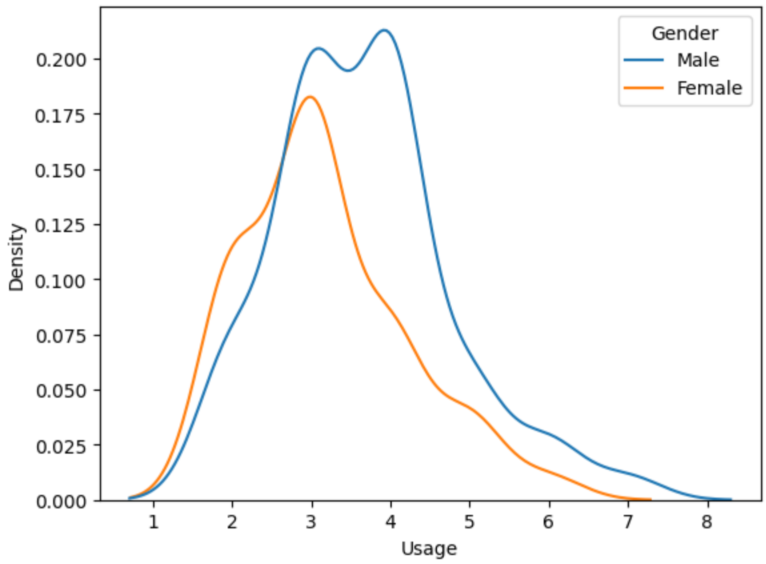
Male customers prefer buying more of the products when compared to female customers in both Single and Partnered maritalStatus.

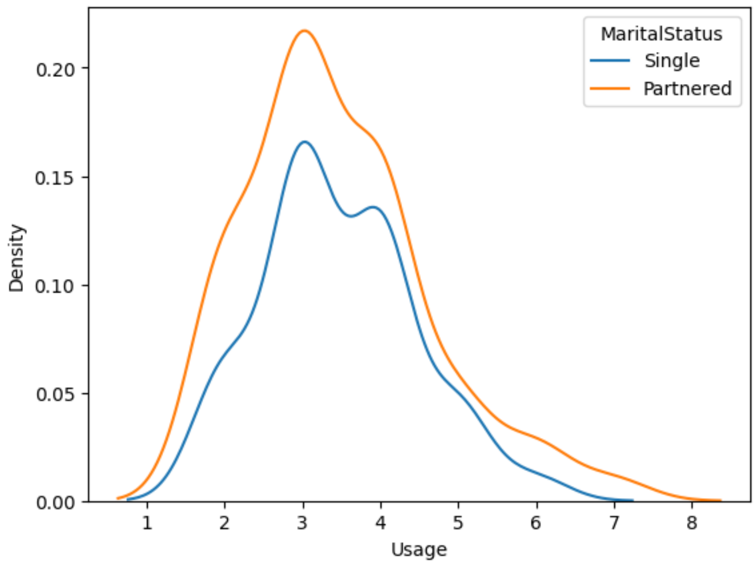
```
sns.countplot(data=df,x='Fitness',hue='Gender')
plt.show()
#count of fitness rating among both the genders
```



We can see the average fitness rating is 3 and males are in superior shape as compared to females. A significant number of males are in excellent shape and there are far less females.

```
sns.kdeplot(data=df,x='Usage',hue='Gender') # Product customer usage per week and gender comparision
plt.show()
sns.kdeplot(data=df,x='Usage',hue='MaritalStatus') # Product customer usage per week and Marital status comparision
plt.show()
```





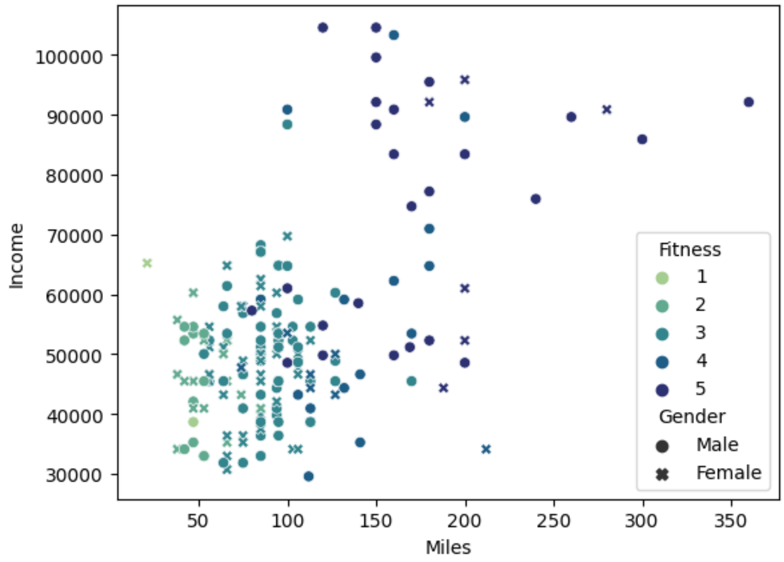
From the first kde plaot we can see that males tend to purchase and use the product more as compared to females. Females only use about 3 days per week on an average and then lack consistency thereafter.

From the second kde plot we can see that partnered custormers usage is higher than the single customers.

Scatter plot for comparing Fitness, Gender, Income and Miles

```
sns.scatterplot(x='Miles',y='Income',data=df,hue='Fitness',style='Gender',palette='crest')
```

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



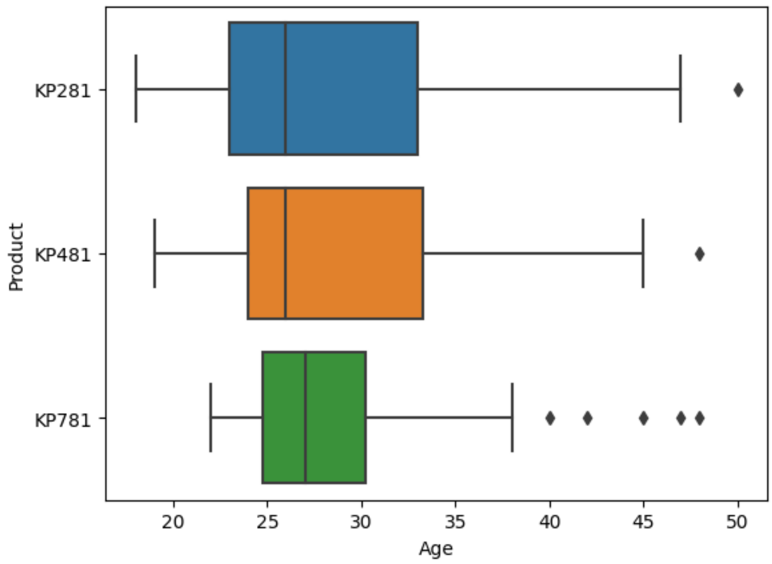
The above scatter plot provides an overall view of customers' income and how much they exercise in relation to their gender and fitness level.

The majority of customers maintain a fitness level ranging from 3 to 4. The data indicates that individuals who cover more miles tend to achieve higher fitness levels.

While a correlation between income and miles exists, it's noteworthy that only a small percentage of customers who earns a lot, run more miles.

Checking to see if different variables have impact on product purchased

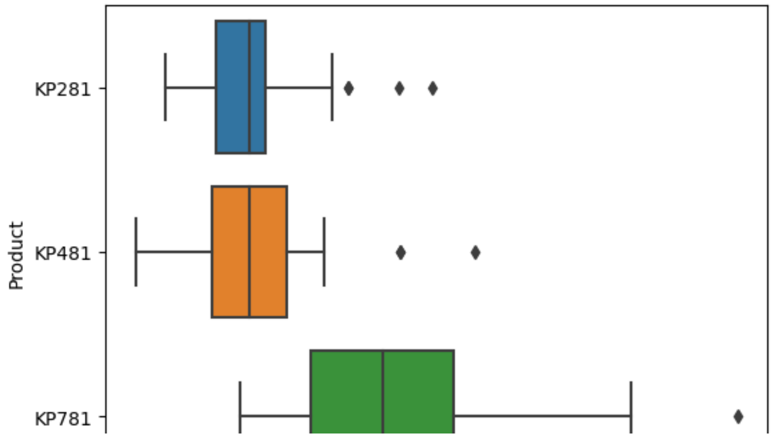
```
sns.boxplot(x='Age',y='Product',data=df) #Age and product purchased
plt.show()
```



Most customers of every prefers KP281 product.

Younger customer within the age bracket of 25-30 prefers to use KP781 and only a few customers with age above 40+ prefer KP781.

```
sns.boxplot(x='Miles',y='Product',data=df) #Miles and product purchased
plt.show()
```

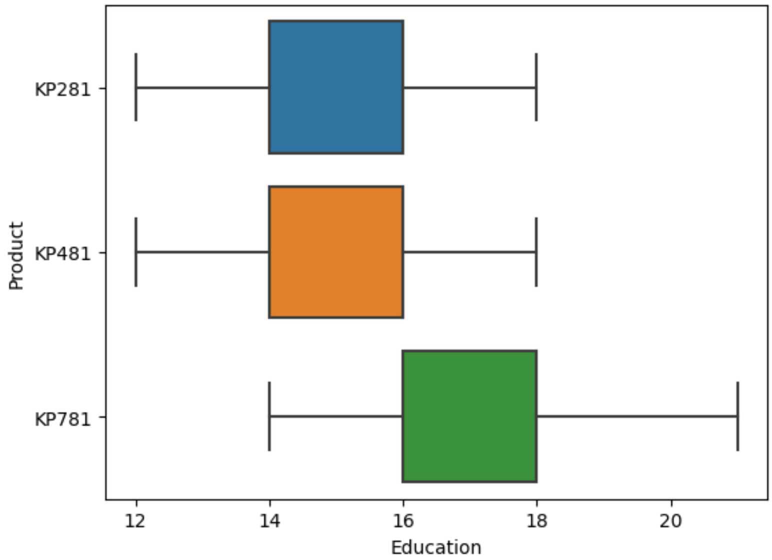


If customer covers over 120 miles per week by walking or running, the likelihood of purchasing the KP781 product increases.

For other two products, the customers had covered less distance than KP781

Miles

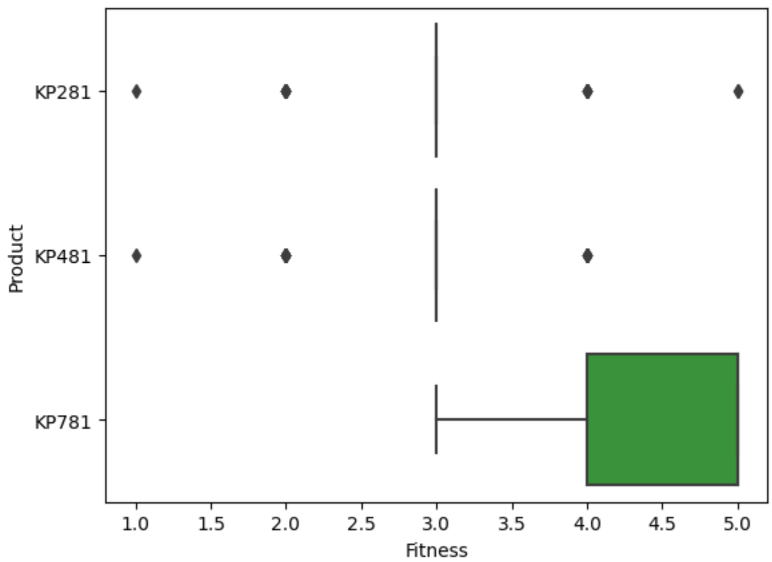
```
sns.boxplot(x='Education',y='Product',data=df) #Education and product purchased
plt.show()
```



Customers with higher education of 16 to 18 prefers using KP781.

Customers with education level of 14 to 16 have equal chances of purchasing both KP481 and KP281 equally.

```
sns.boxplot(x='Fitness',y='Product',data=df) #fitness and product purchased
plt.show()
```



If the customer is more fit i.e. in the range of 4 to 5, they will most likely be purcahsing the KP781.

KP481 and KP281 will be preferred by people with various fitness rating as it is scattered across all fitness levels.

Missing values and outliers detection

```
df.isnull().sum()
```

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

```
df.duplicated().sum()
```

```
0
```

We can see there are no missing/null values or duplicate values in the dataset.

Business Insights based on Non-Graphical and Visual Analysis

Marginal Probabilities

```
df.Product.value_counts(normalize=True)
```

```
KP281    0.444444
KP481    0.333333
KP781    0.222222
Name: Product, dtype: float64
```

We can see the probability of a person buying each product is stated above.The customer is most likely to but KP281

```
df.Gender.value_counts(normalize=True)
```

```
Male      0.577778
Female    0.422222
Name: Gender, dtype: float64
```

Males are more likely to buy the product rather than the females.

```
df.MaritalStatus.value_counts(normalize=True)
```

```
Partnered    0.594444
Single       0.405556
Name: MaritalStatus, dtype: float64
```

Customers with partners will prefer buying the product and single will prefer it far less.

```
df.Education.value_counts(normalize=True)
```

```
16    0.472222
14    0.305556
18    0.127778
15    0.027778
13    0.027778
12    0.016667
21    0.016667
20    0.005556
Name: Education, dtype: float64
```

Customers with higher education will prefer buying the product more than the customers with low eduaction level

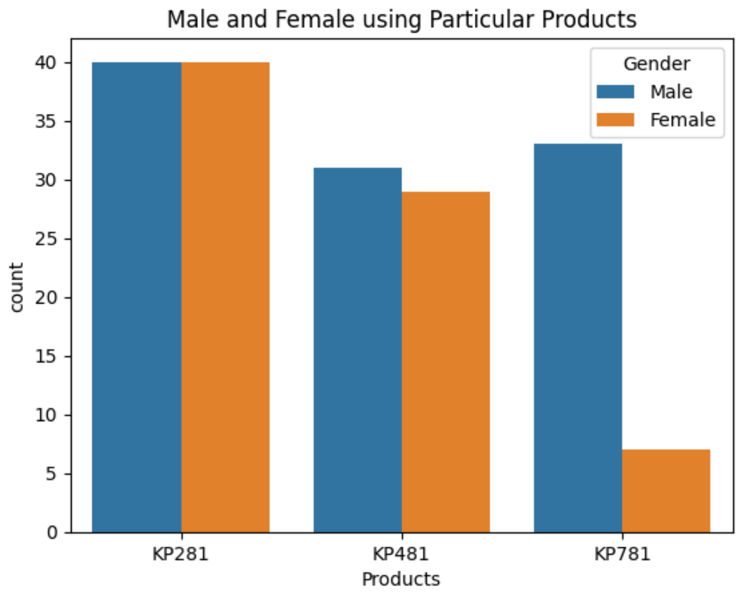
Conditional Probabilities

Probability of each product for both genders

```
def probability_of_gender(gender,df): #defining the function
    print(f"Prob P(KP781) for {gender}: {round(df['KP781'][gender]/df.loc[gender].sum(),3)}") #printing probability for the specified gen
    print(f"Prob P(KP481) for {gender}: {round(df['KP481'][gender]/df.loc[gender].sum(),3)}") #printing probability for the specified gen
    print(f"Prob P(KP281) for {gender}: {round(df['KP281'][gender]/df.loc[gender].sum(),3)}") #printing probability for the specified gen

df_temp = pd.crosstab(index=df['Gender'],columns=[df['Product']])
print("Prob of Male: ",round(df_temp.loc['Male'].sum()/len(df),3))
print("Prob of Female: ",round(df_temp.loc['Female'].sum()/len(df),3))
print()
gender_Probability('Male',df_temp)
print()
gender_Probability('Female',df_temp)
```

```
Prob of Male:  0.578
Prob of Female:  0.422

Prob P(KP781) for Male: 0.317
Prob P(KP481) for Male: 0.298
Prob P(KP281) for Male: 0.385

Prob P(KP781) for Female: 0.092
Prob P(KP481) for Female: 0.382
Prob P(KP281) for Female: 0.526
```

```
sns.countplot(x = "Product", data= df, hue = "Gender")
plt.xlabel("Products")
plt.title("Male and Female using Particular Products")
plt.show()
```



We can see via these conditional probabilities and the graph as well, that probablity of male buying any product is higher than that of females. Among both the genders the preferred proddduct is KP281. While male prefer KP781 as their second preference, females on the other hand prefers KP481 as their second choice.

Probability of each product for both marital status

```
def MaritalS(MaritalStatus,df):
    print(f"Prob P(KP781) for {MaritalStatus}: {round(df['KP781'][MaritalStatus]/df.loc[MaritalStatus].sum(),3)}")
    print(f"Prob P(KP481) for {MaritalStatus}: {round(df['KP481'][MaritalStatus]/df.loc[MaritalStatus].sum(),3)}")
    print(f"Prob P(KP281) for {MaritalStatus}: {round(df['KP281'][MaritalStatus]/df.loc[MaritalStatus].sum(),3)}")

df_temp = pd.crosstab(index=df['MaritalStatus'],columns=[df['Product']])
print("Prob of Single: ",round(df_temp.loc['Single'].sum()/len(df),3))
print("Prob of Partnered: ",round(df_temp.loc['Partnered'].sum()/len(df),3))
print()
MaritalS('Single',df_temp)
print()
MaritalS('Partnered',df_temp)
```

```
Prob of Single:  0.406
Prob of Partnered:  0.594
```

```
Prob P(KP781) for Single: 0.233
Prob P(KP481) for Single: 0.329
Prob P(KP281) for Single: 0.438

Prob P(KP781) for Partnered: 0.215
Prob P(KP481) for Partnered: 0.336
Prob P(KP281) for Partnered: 0.449
```

As we can see the probability of people with partners is more than the single one's. Both single and people with partners prefers KP281 as their first choice and KP481 as their second choice.

Two way contingency tables

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

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

```
np.round(pd.crosstab([df.Product],df.Gender, margins=True)/180*100,2)
```

Gender Product	Female	Male	All
KP281	22.22	22.22	44.44
KP481	16.11	17.22	33.33
KP781	3.89	18.33	22.22
All	42.22	57.78	100.00

Probability of Male customer buying the product is more than the probability of female customer buying any product.

```
np.round((pd.crosstab([df.Product],df.Gender,margins=True,normalize="columns"))*100,2)
```

Gender Product	Female	Male	All
KP281	52.63	38.46	44.44
KP481	38.16	29.81	33.33
KP781	9.21	31.73	22.22

KP281 is a preferable choice for female customers.

The likelihood of a female customer purchasing KP281 (52.63%) surpasses that of a male customer (38.46%).

KP481 is particularly recommended for female customers.

The probability of a male customer acquiring Product KP781 (31.73%) is considerably higher than that of a female customer (9.21%)

Insights and Recommendations:

KP281

- KP281, is the higest selling product and is the cheapest as well. The company should continue to do what is is doing to increase the sales of this one.
- Both male and females are equally likely to buy this product.
- Average distance covered in the model is 70 to 90 miles
- People use this model about 3-4 times in a week.
- Most of the younger customers prefer buying this product
- People with income between 40k to 59K prefer using this model, as this is relatively cheaper.

KP481

- This model is the second preference of the customers.
- Customers cover more miles with this model.
- We have more female customers for this model than male customers.
- Customers with an average income of 45-50K prefer using this model.
- People cover 75-100 miles when using this model
- Age range for this product is 24-34(mix of younger and people in late 30s)

KP781

- This model is the least preferred by all customers, as this is relatively more expensive nad advance level of product.
- Customers covers more than 120 to 200 miles per week with this model.
- This product is used about 3-4 times in a week.
- Single people buy this more than the married people
- Males prefer this product more than female does.
- Income of people using this product is 75K or higher
- People who trust aerofit brand tend to buy and invest in this product.
- Customers with higher education and higher income prefer to buy this model as compared to people with low income.

Overall Recommendations:

-
- Company can use the in-house fitness counsellor, who can talk to gyms, other direct customers and help them understandf the benefits etc to promote the sales.
- More advertising for KP481 and KP781 to the users who have been consistently using teh aerofit products.
- KP781, is an advanced product with more features and have a high pricing too. The company should intend to sell this product to athletes or celebrity fitness trainer. They will be able to afford it and use it to its utmost core.
- KP781, should be suggested to females and the customers with an age group of abouve 40+
- KP281, should be promoted a low priced/budget treadmill, so that more and more customers will continue to buy those.
- Female customers prefer exercising less than male customers, we should run a marketing campaign with a strong female celebrity personality, which will influence female customers as well.
- The compnay should conduct a market research of why customers above 50 years are not preferring to buy any product. Telling them about the benefit of exercising will get more customers as well.