Business Case: Aerofit - Descriptive Statistics & Probability

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.
Problem Statement: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.

Importing Libraries, Loading the dataset and Basic Analysis

In [4]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.simplefilter(action='ignore', category=Warning)
```

...

In [5]:
```
df=pd.read_csv('aerofit.csv')
```

In [6]:
```
df.head()
```

Out[6]:

	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

In [314]:
```
df.shape
```

Out[314]:
```
(180, 9)
```

We have a dataset of 180 rows and 9 columns

In [7]:
```python
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 dataset does not have any missing values.

In [316]:
```python
df.describe(include="all").T
```

Out[316]:

	count	unique	top	freq	mean	std	min	25%	
Product	180	3	KP281	80	NaN	NaN	NaN	NaN	
Age	180.0	NaN	NaN	NaN	28.788889	6.943498	18.0	24.0	
Gender	180	2	Male	104	NaN	NaN	NaN	NaN	
Education	180.0	NaN	NaN	NaN	15.572222	1.617055	12.0	14.0	
MaritalStatus	180	2	Partnered	107	NaN	NaN	NaN	NaN	
Usage	180.0	NaN	NaN	NaN	3.455556	1.084797	2.0	3.0	
Fitness	180.0	NaN	NaN	NaN	3.311111	0.958869	1.0	3.0	
Income	180.0	NaN	NaN	NaN	53719.577778	16506.684226	29562.0	44058.75	50
Miles	180.0	NaN	NaN	NaN	103.194444	51.863605	21.0	66.0	

Observations:
1.KP281 is the most popular product with 80 number of records.
2.Mean age of the total customer base is 28.79 with maximum equal to 50 and minimum equal to 18.
3.Mean income of the total customer base is 53719.58 with maximum equal to 104581 and minimum 29562.
4.Average no of years of education is 15.57 years while minimum being 1.62 years and maximum being 21 years.

As per information provided, Product Portfolio is as follows:

1. The KP281 is an entry-level treadmill that sells for 1,500.
2. The KP481 is for mid-level runners that sell for 1,750.
3. The KP781 treadmill is having advanced features that sell for 2,500.

In [317]:
```python
df['Product'].value_counts()
```

Out[317]:
```
KP281    80
KP481    60
KP781    40
Name: Product, dtype: int64
```

Observations:

- KP281 is the cheapest and most selling treadmill of Aerofit.
- KP481 is the middle range product with 60 number of sales records.
- KP781, being the most costly, has lower number of sales.

In [318]:
```python
df['Gender'].value_counts()
```

Out[318]:
```
Male      104
Female     76
Name: Gender, dtype: int64
```

We have 104 male customers and 76 female customers.

In [319]:
```python
df['MaritalStatus'].value_counts()
```

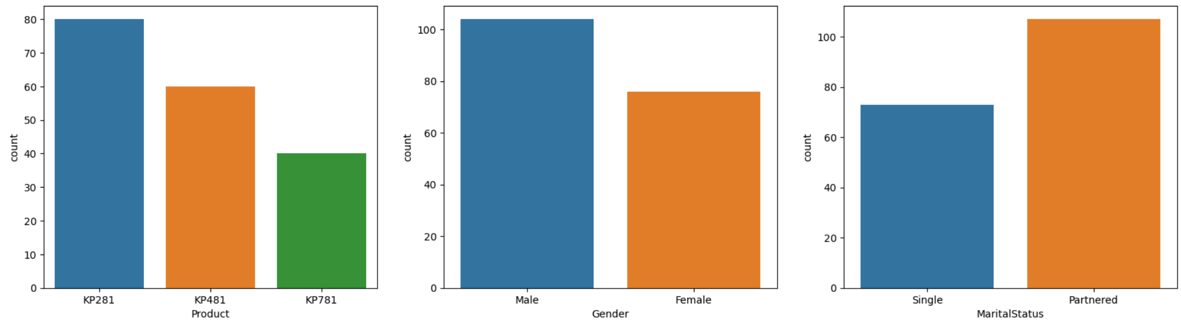
Out[319]:
```
Partnered    107
Single        73
Name: MaritalStatus, dtype: int64
```

We have 107 partnered and 73 single customers.

Univariate Analysis

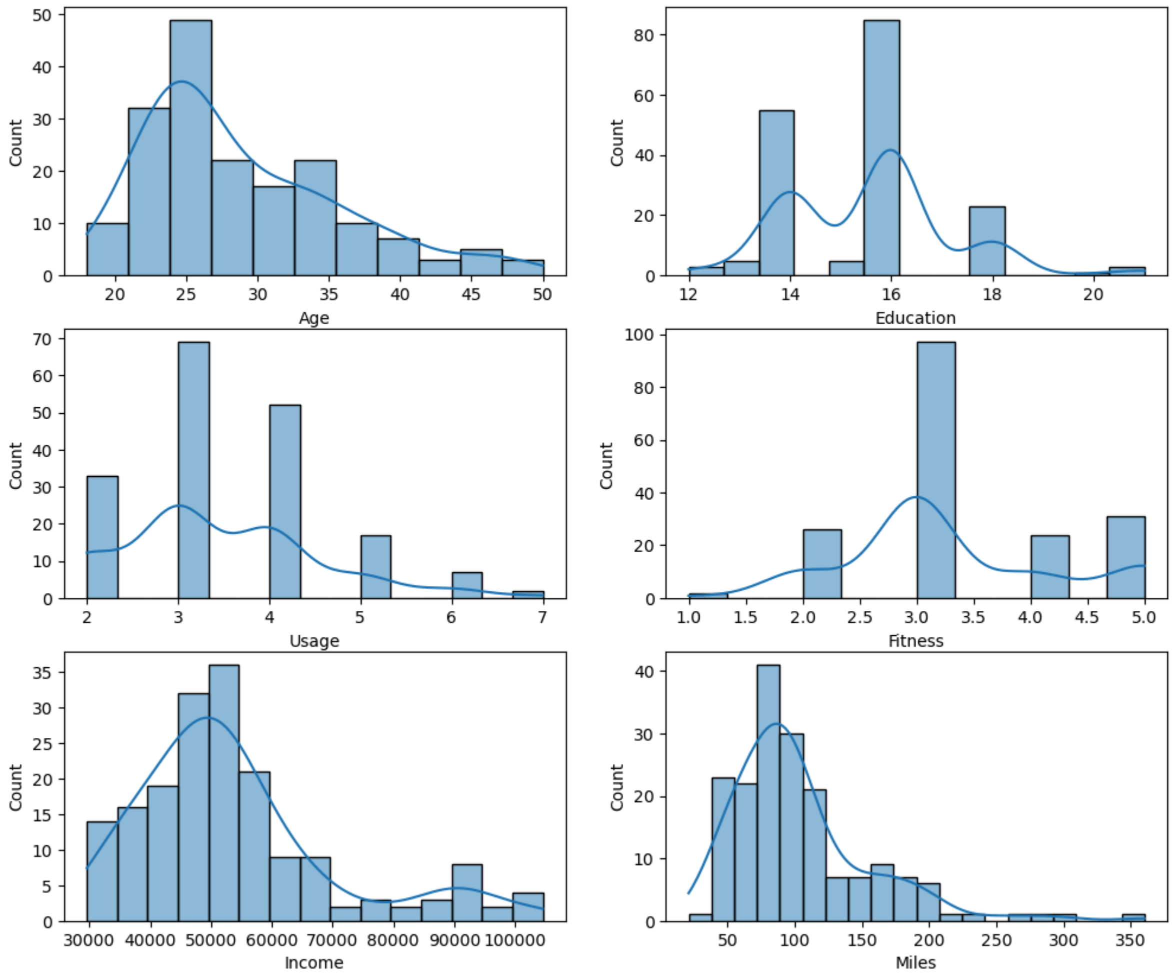
i. Categorical Variables

In [320]:
```python
fig,axis= plt.subplots(1,3,figsize=(20,5))
sns.countplot(data=df,x="Product",ax=axis[0])
sns.countplot(data=df,x="Gender",ax= axis[1])
sns.countplot(data=df,x="MaritalStatus",ax=axis[2])
plt.show()
```



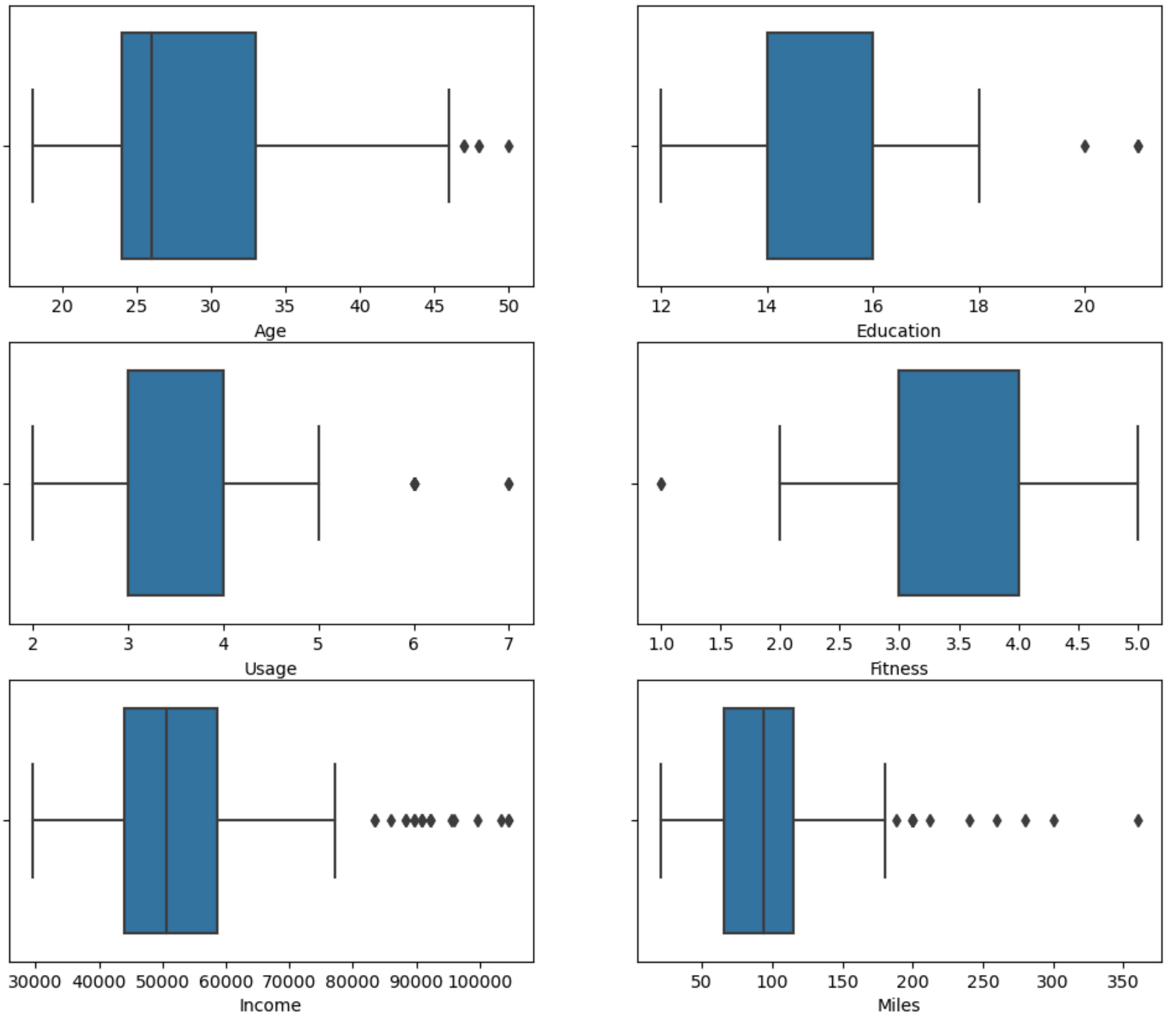
ii. Continous Variables

In [321]:
```
fig,axis= plt.subplots(3,2,figsize=(12,10))
sns.histplot(data=df,x="Age" , kde=True,ax=axis[0,0])
sns.histplot(data=df,x="Education",kde=True,ax= axis[0,1])
sns.histplot(data=df,x="Usage",kde=True,ax=axis[1,0])
sns.histplot(data=df,x="Fitness",kde=True,ax=axis[1,1])
sns.histplot(data=df,x="Income",kde=True,ax=axis[2,0])
sns.histplot(data=df,x="Miles",kde=True,ax=axis[2,1])
plt.show()
```



Outlier Detection

In [322]:
```
fig,axis= plt.subplots(3,2,figsize=(12,10))
sns.boxplot(x=df["Age"],ax=axis[0,0])
sns.boxplot(x=df["Education"],ax= axis[0,1])
sns.boxplot(x=df["Usage"],ax=axis[1,0])
sns.boxplot(x=df["Fitness"],ax=axis[1,1])
sns.boxplot(x=df["Income"],ax=axis[2,0])
sns.boxplot(x=df["Miles"],ax=axis[2,1])
plt.show()
```



In [323]:
```
df[df['Product']=='KP781']['Income'].quantile(0.25)
```

Out[323]: 58204.75

Income and Miles have more number of outliers.
Let us try to categorize two segments of customers based on income:

1. Low Income- Customers whose income is less than 58000
2. High Income- Customers whose income is greater than or equal to 58000

In [324]:
```
df['Income_Category'] = np.where(df['Income'] >= 58000 ,'High Income','Low Inc
```

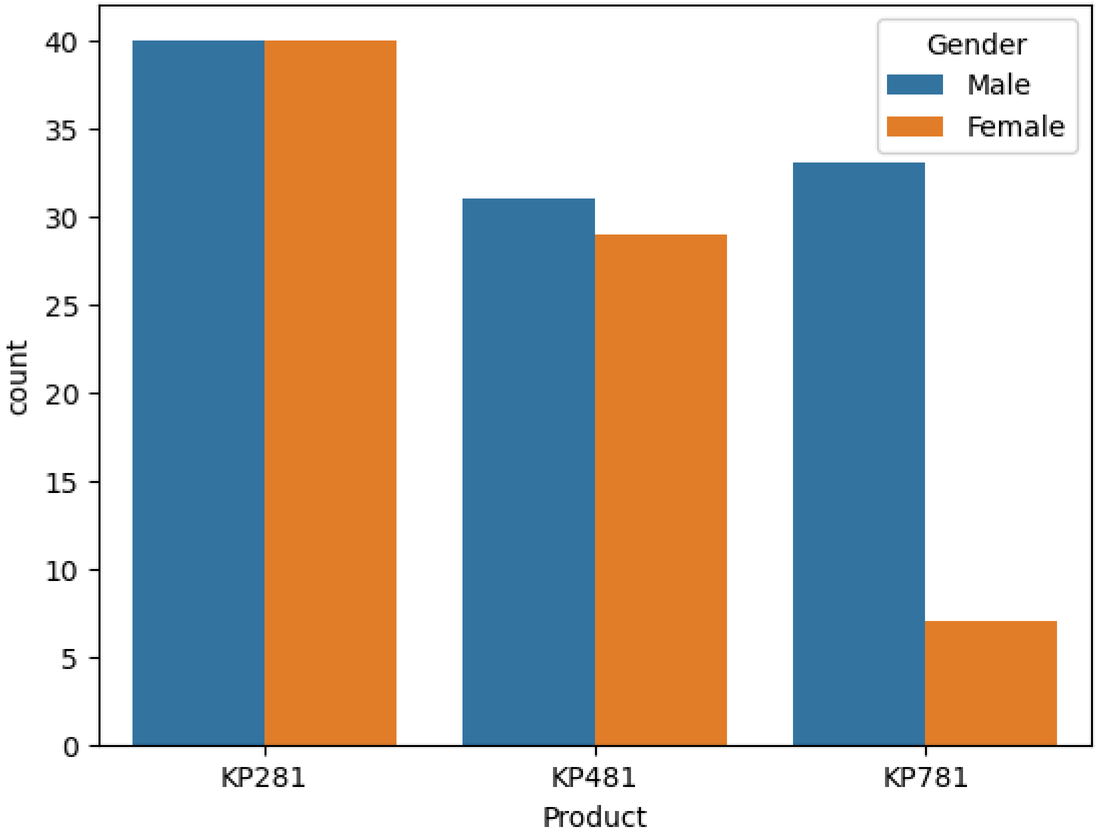
In [325]:
```
pd.crosstab(df['Income_Category'], df['Product'])
```

Out[325]:

Product / Income_Category	KP281	KP481	KP781
High Income	7	9	30
Low Income	73	51	10

Bivariate Analysis

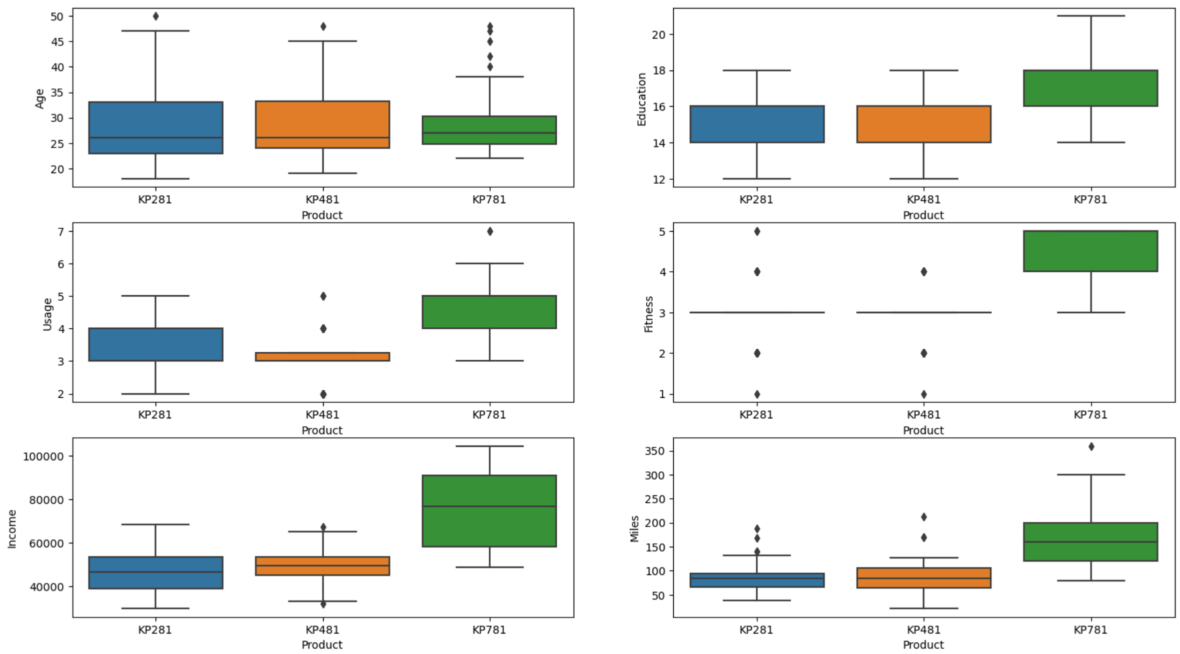
In [326]:
```
sns.countplot(data=df,x="Product",hue='Gender')
plt.show()
```



Observations:

1. KP281 has equal number of male and female customers.
2. KP481 has slight higher no of male customers as compared to female customers.
3. KP781 has significanly higbher number of male customers as compared to female customers.

In [327]:
```
fig,axis= plt.subplots(3,2,figsize=(18,10))
sns.boxplot(data=df,y='Age',x="Product",ax=axis[0,0])
sns.boxplot(data=df,y="Education",x="Product",ax=axis[0,1])
sns.boxplot(data=df,y="Usage",x="Product",ax=axis[1,0])
sns.boxplot(data=df,y="Fitness",x="Product",ax=axis[1,1])
sns.boxplot(data=df,y="Income",x="Product",ax=axis[2,0])
sns.boxplot(data=df,y="Miles",x="Product",ax=axis[2,1])
plt.show()
```



A very interesting observation is found in the graph of miles vs product. After a certain point, customers are more likely to purchase KP781.

In [328]:
```
df[df['Product']=='KP781']['Miles'].quantile(0.25)
```

Out[328]: 120.0

Let us try to categorize two segments of customers based on average miles per week:

1. Miles_below_120 - Customers whose expects to run each week less than or equal to 120 miles
2. Miles_above_120 - Customers whose expects to run each week more than 120 miles

In [329]:
```
df['Miles_Category'] = np.where(df['Miles'] >=120 ,'Miles_above_120','Miles_be
```

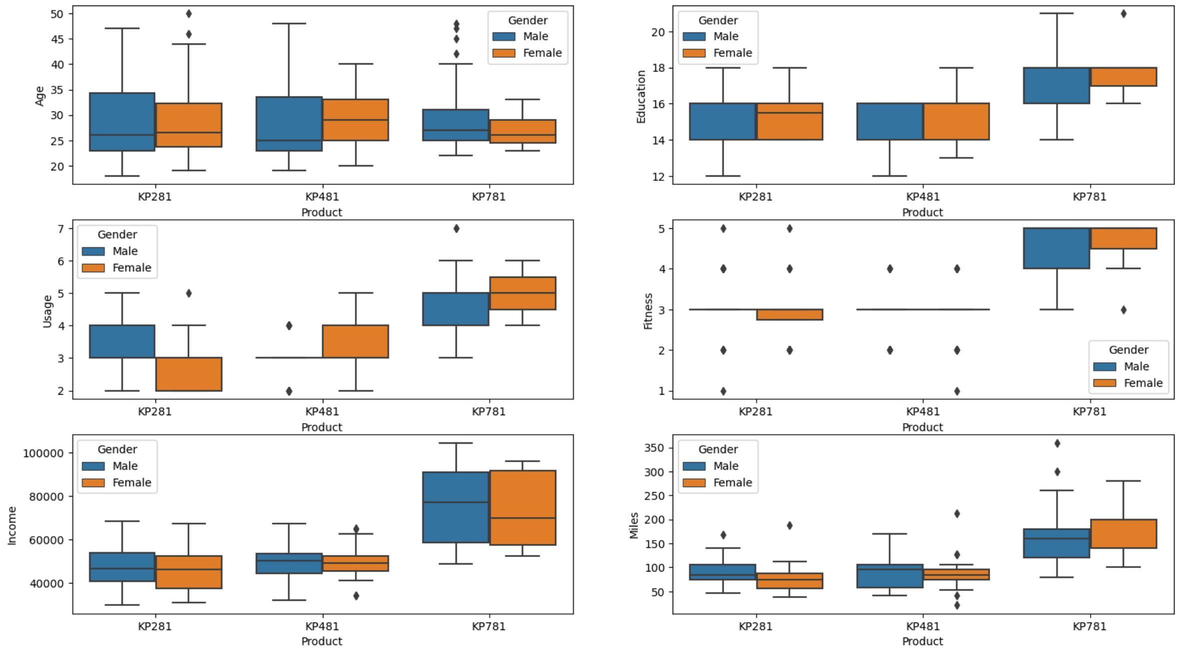
In [330]:
```
pd.crosstab(df['Miles_Category'], df['Product'])
```

Out[330]:

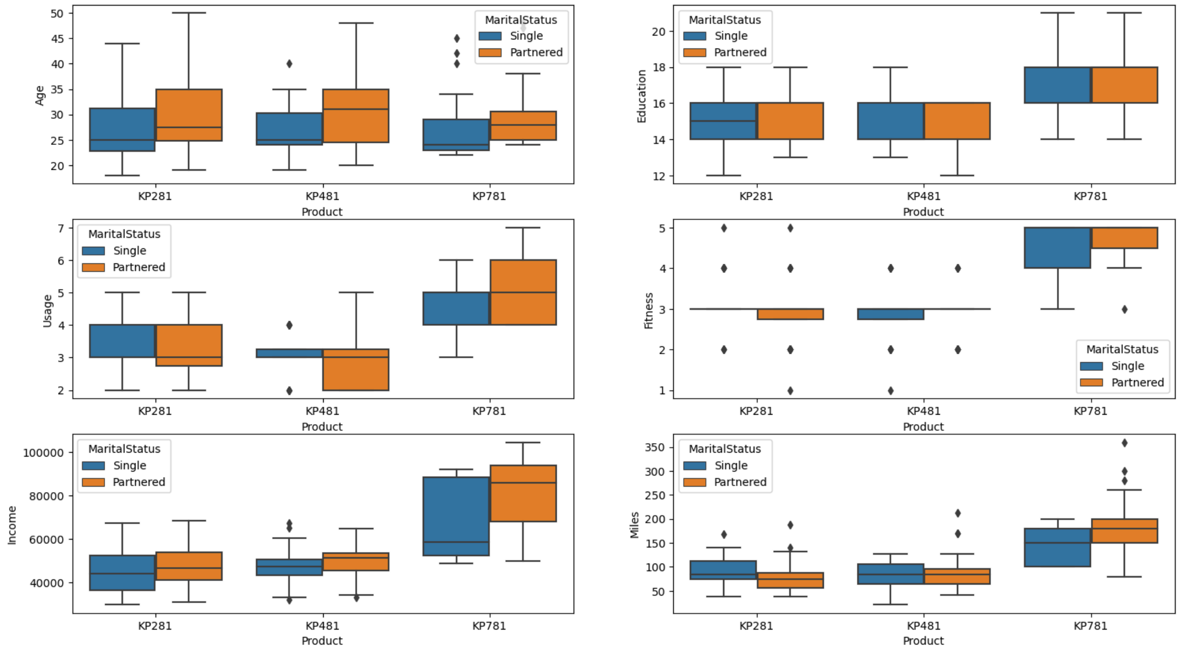
Product	KP281	KP481	KP781
Miles_Category			
Miles_above_120	6	8	31
Miles_below_120	74	52	9

Multivariate Analysis

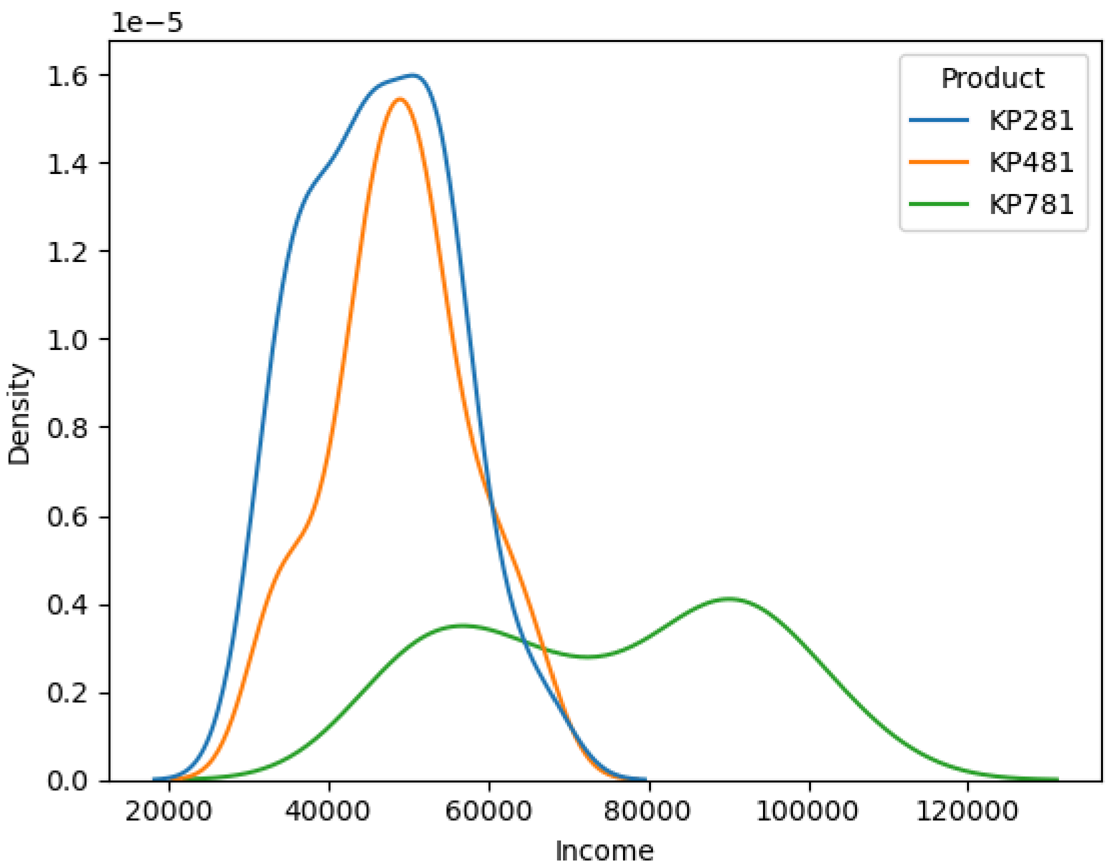
In [331]:
```
fig,axis= plt.subplots(3,2,figsize=(18,10))
sns.boxplot(data=df,y="Age",x="Product",hue="Gender",ax=axis[0,0])
sns.boxplot(data=df,y="Education",x="Product",hue="Gender",ax=axis[0,1])
sns.boxplot(data=df,y="Usage",x="Product",hue="Gender",ax=axis[1,0])
sns.boxplot(data=df,y="Fitness",x="Product",hue="Gender",ax=axis[1,1])
sns.boxplot(data=df,y="Income",x="Product",hue="Gender",ax=axis[2,0])
sns.boxplot(data=df,y="Miles",x="Product",hue="Gender",ax=axis[2,1])
plt.show()
```



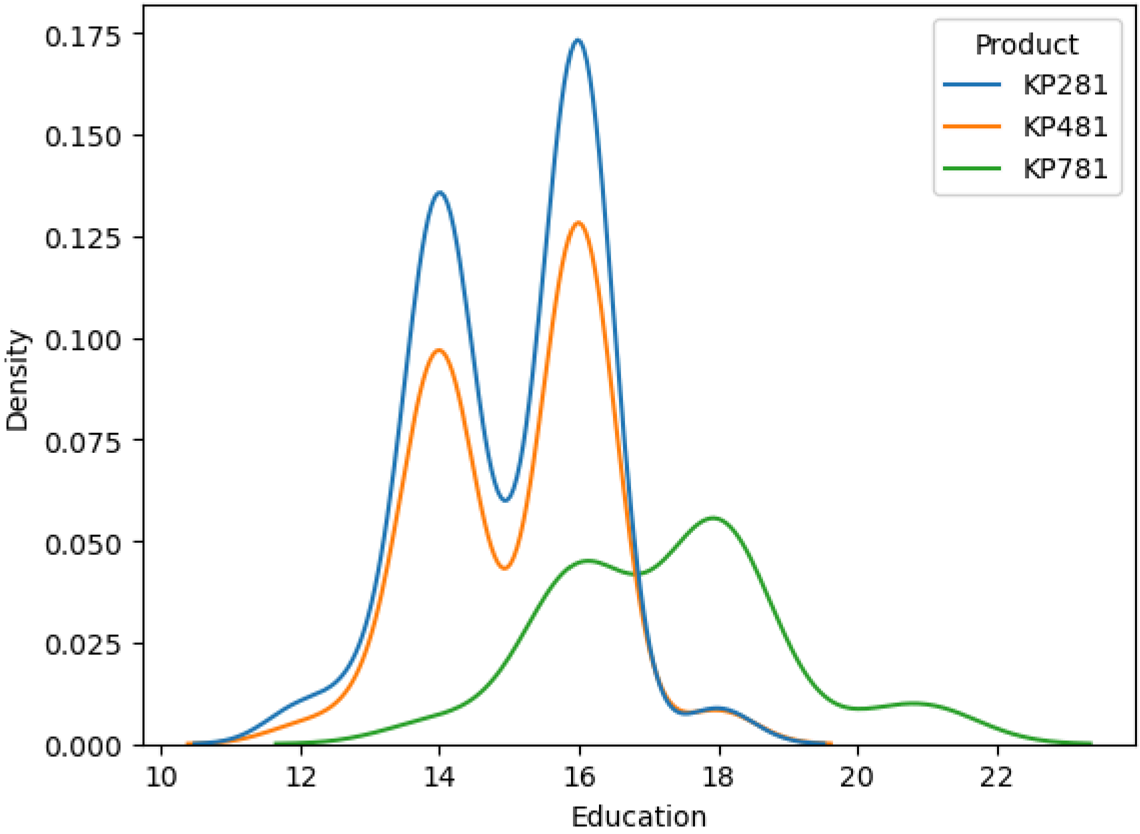
In [332]:
```python
fig,axis= plt.subplots(3,2,figsize=(18,10))
sns.boxplot(data=df,y="Age",x="Product",hue="MaritalStatus",ax=axis[0,0])
sns.boxplot(data=df,y="Education",x="Product",hue="MaritalStatus",ax=axis[0,1]
sns.boxplot(data=df,y="Usage",x="Product",hue="MaritalStatus",ax=axis[1,0])
sns.boxplot(data=df,y="Fitness",x="Product",hue="MaritalStatus",ax=axis[1,1])
sns.boxplot(data=df,y="Income",x="Product",hue="MaritalStatus",ax=axis[2,0])
sns.boxplot(data=df,y="Miles",x="Product",hue="MaritalStatus",ax=axis[2,1])
plt.show()
```



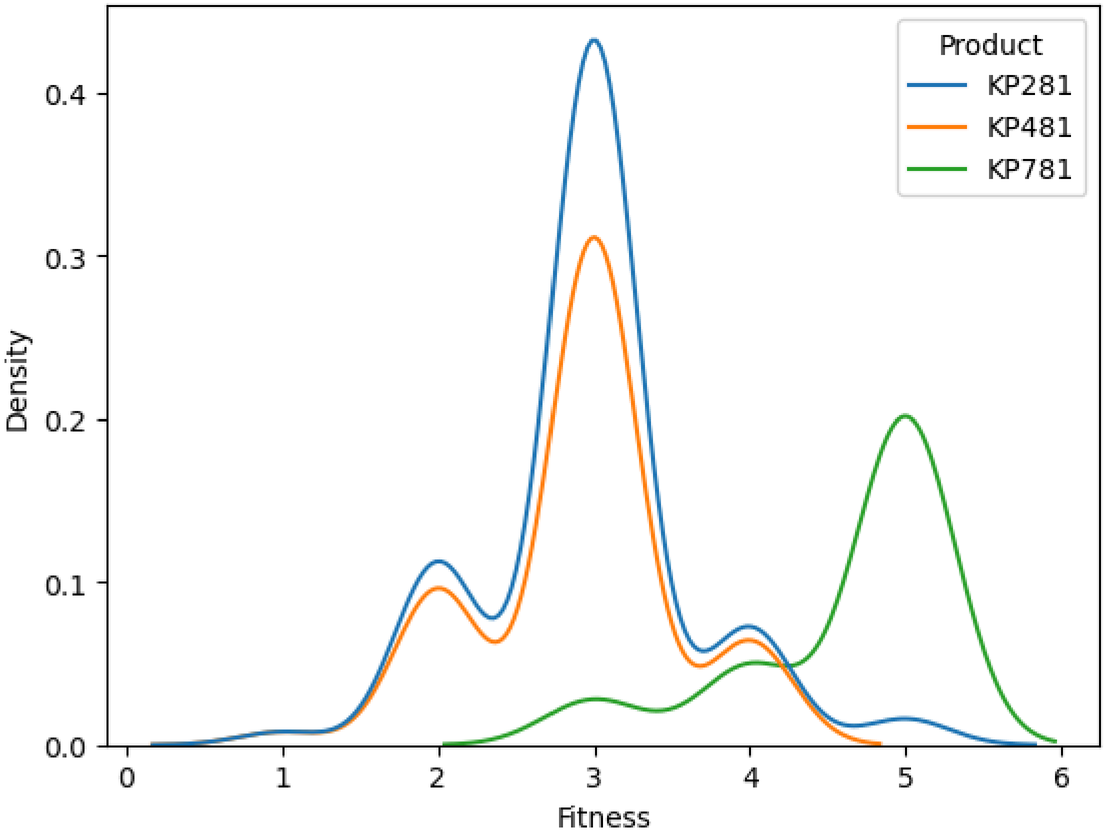
In [349]:
```python
sns.kdeplot(data=df,x="Income",hue="Product")
plt.show()
```



In [333]:
```
sns.kdeplot(data=df,x="Education",hue="Product")
plt.show()
```

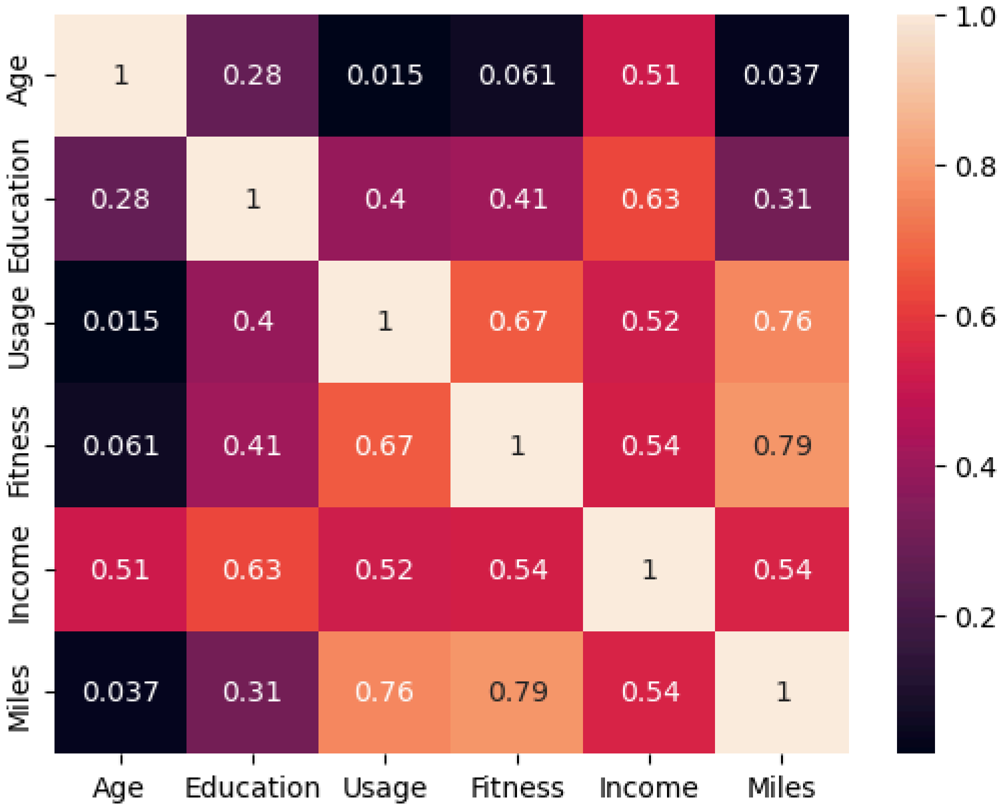


In [334]:
```
sns.kdeplot(data=df,x="Fitness",hue="Product")
plt.show()
```



In [351]:
```
sns.heatmap(df[['Age','Education','Usage','Fitness','Income','Miles']].corr(),
```

Out[351]:
```
<AxesSubplot:>
```



Marginal and Conditional Probability

In [335]:
```
df['Product'].value_counts(normalize=True)
```

Out[335]:
```
KP281    0.444444
KP481    0.333333
KP781    0.222222
Name: Product, dtype: float64
```

Marginal Probability for each product is: KP281 = 0.44, KP481 = 0.33, KP781 = 0.22

In [336]:
```
pd.crosstab(df['Gender'],df['Product'], margins=True,normalize=True)
```

Out[336]:

Product	KP281	KP481	KP781	All
Gender				
Female	0.222222	0.161111	0.038889	0.422222
Male	0.222222	0.172222	0.183333	0.577778
All	0.444444	0.333333	0.222222	1.000000

Observations:

- Marginal Probability of each Gender is: Male=0.58 and Female=0.42
- Percentage of a Male for KP281 is equal i.e. 22.22 % to percentage of female.

- For KP481, percentage of Male customers are 1.1% greater than percentage of Female customers
- For KP781, percentage of Male customers are 15.55% greater than percentage of Female

In [337]:
```
pd.crosstab(df['MaritalStatus'],df['Product'], margins=True,normalize=True)
```

Out[337]:

Product	KP281	KP481	KP781	All
MaritalStatus				
Partnered	0.266667	0.200000	0.127778	0.594444
Single	0.177778	0.133333	0.094444	0.405556
All	0.444444	0.333333	0.222222	1.000000

In [338]:
```
pd.crosstab(df['Income_Category'],df['Gender'], margins=True,normalize=True)
```

Out[338]:

Gender	Female	Male	All
Income_Category			
High Income	0.072222	0.183333	0.255556
Low Income	0.350000	0.394444	0.744444
All	0.422222	0.577778	1.000000

Observations:

- Marginal Probability of each Gender is: Male=57.78 % and Female=42.22%
- For all three products, percentage of partenered customers is greater than that of single customers.
- For KP281 is equal i.e. 22.22 % to percentage of female.
- For KP481, percentage of Male customers are 1.1% greater than percentage of Female customers
- For KP781, percentage of Male customers are 15.55% greater than percentage of Female customers.

In [339]:
```
pd.crosstab(df['Income_Category'],df['Product'], margins=True,normalize=True)
```

Out[339]:

Product	KP281	KP481	KP781	All
Income_Category				
High Income	0.038889	0.050000	0.166667	0.255556
Low Income	0.405556	0.283333	0.055556	0.744444
All	0.444444	0.333333	0.222222	1.000000

In [340]:
```python
pd.crosstab([df['Income_Category'],df['Gender']],df['Product'], margins=True)
```

Out[340]:

Income_Category	Gender	KP281	KP481	KP781	All
High Income	Female	4	4	5	13
	Male	3	5	25	33
Low Income	Female	36	25	2	63
	Male	37	26	8	71
All		80	60	40	180

In [341]:
```python
pd.crosstab([df['MaritalStatus'],df['Gender']],df['Product'], margins=True,nor
```

Out[341]:

MaritalStatus	Gender	KP281	KP481	KP781	All
Partnered	Female	0.150000	0.083333	0.022222	0.255556
	Male	0.116667	0.116667	0.105556	0.338889
Single	Female	0.072222	0.077778	0.016667	0.166667
	Male	0.105556	0.055556	0.077778	0.238889
All		0.444444	0.333333	0.222222	1.000000

Observations:

- Marginal Probability of each Gender is: Male=57.78 % and Female=42.22%
- For all three products, percentage of partenered customers is greater than that of single customers.
- For KP281 is equal i.e. 22.22 % to percentage of female.
- For KP481, percentage of Male customers are 1.1% greater than percentage of Female customers
- For KP781, percentage of Male customers are 15.55% greater than percentage of Female customers.

In [342]:
```
pd.crosstab(df['Education'], df['Product'], margins=True,normalize=True)
```

Out[342]:

Product Education	KP281	KP481	KP781	All
12	0.011111	0.005556	0.000000	0.016667
13	0.016667	0.011111	0.000000	0.027778
14	0.166667	0.127778	0.011111	0.305556
15	0.022222	0.005556	0.000000	0.027778
16	0.216667	0.172222	0.083333	0.472222
18	0.011111	0.011111	0.105556	0.127778
20	0.000000	0.000000	0.005556	0.005556
21	0.000000	0.000000	0.016667	0.016667
All	0.444444	0.333333	0.222222	1.000000

In [343]:
```
pd.crosstab(df['Fitness'], df['Product'], margins=True,normalize=True)
```

Out[343]:

Product Fitness	KP281	KP481	KP781	All
1	0.005556	0.005556	0.000000	0.011111
2	0.077778	0.066667	0.000000	0.144444
3	0.300000	0.216667	0.022222	0.538889
4	0.050000	0.044444	0.038889	0.133333
5	0.011111	0.000000	0.161111	0.172222
All	0.444444	0.333333	0.222222	1.000000

In [344]:
```
pd.crosstab(df['Usage'], df['Product'], margins=True,normalize=True)
```

Out[344]:

Product Usage	KP281	KP481	KP781	All
2	0.105556	0.077778	0.000000	0.183333
3	0.205556	0.172222	0.005556	0.383333
4	0.122222	0.066667	0.100000	0.288889
5	0.011111	0.016667	0.066667	0.094444
6	0.000000	0.000000	0.038889	0.038889
7	0.000000	0.000000	0.011111	0.011111
All	0.444444	0.333333	0.222222	1.000000

In [345]:
```
pd.crosstab([df['Usage'],df['Gender']], df['Product'], margins=True,normalize=
```

Out[345]:

Usage	Gender	KP281	KP481	KP781	All
2	Female	0.072222	0.038889	0.000000	0.111111
	Male	0.033333	0.038889	0.000000	0.072222
3	Female	0.105556	0.077778	0.000000	0.183333
	Male	0.100000	0.094444	0.005556	0.200000
4	Female	0.038889	0.027778	0.011111	0.077778
	Male	0.083333	0.038889	0.088889	0.211111
5	Female	0.005556	0.016667	0.016667	0.038889
	Male	0.005556	0.000000	0.050000	0.055556
6	Female	0.000000	0.000000	0.011111	0.011111
	Male	0.000000	0.000000	0.027778	0.027778
7	Male	0.000000	0.000000	0.011111	0.011111
All		0.444444	0.333333	0.222222	1.000000

In [346]:
```
pd.crosstab(df['Miles_Category'], df['Product'], margins=True,normalize=True)
```

Out[346]:

Miles_Category	KP281	KP481	KP781	All
Miles_above_120	0.033333	0.044444	0.172222	0.25
Miles_below_120	0.411111	0.288889	0.050000	0.75
All	0.444444	0.333333	0.222222	1.00

In [347]:
```
pd.crosstab([df['Usage'],df['Gender'],df['Miles_Category']],df['Product'], mar
```

Out[347]:

Usage	Gender	Miles_Category	KP281	KP481	KP781	All
2	Female	Miles_below_120	0.072222	0.038889	0.000000	0.111111
	Male	Miles_below_120	0.033333	0.038889	0.000000	0.072222
3	Female	Miles_above_120	0.000000	0.005556	0.000000	0.005556
		Miles_below_120	0.105556	0.072222	0.000000	0.177778
	Male	Miles_above_120	0.000000	0.000000	0.005556	0.005556
		Miles_below_120	0.100000	0.094444	0.000000	0.194444
4	Female	Miles_above_120	0.000000	0.005556	0.005556	0.011111
		Miles_below_120	0.038889	0.022222	0.005556	0.066667
	Male	Miles_above_120	0.022222	0.027778	0.050000	0.100000
		Miles_below_120	0.061111	0.011111	0.038889	0.111111
5	Female	Miles_above_120	0.005556	0.005556	0.011111	0.022222
		Miles_below_120	0.000000	0.011111	0.005556	0.016667
	Male	Miles_above_120	0.005556	0.000000	0.050000	0.055556
6	Female	Miles_above_120	0.000000	0.000000	0.011111	0.011111
	Male	Miles_above_120	0.000000	0.000000	0.027778	0.027778
7	Male	Miles_above_120	0.000000	0.000000	0.011111	0.011111
All			0.444444	0.333333	0.222222	1.000000

In [348]:
```
pd.crosstab([df['Fitness'],df['Gender']],df['Product'], margins=True,normalize
```

Out[348]:

Fitness	Gender	KP281	KP481	KP781	All
1	Female	0.000000	0.005556	0.000000	0.005556
	Male	0.005556	0.000000	0.000000	0.005556
2	Female	0.055556	0.033333	0.000000	0.088889
	Male	0.022222	0.033333	0.000000	0.055556
3	Female	0.144444	0.100000	0.005556	0.250000
	Male	0.155556	0.116667	0.016667	0.288889
4	Female	0.016667	0.022222	0.005556	0.044444
	Male	0.033333	0.022222	0.033333	0.088889
5	Female	0.005556	0.000000	0.027778	0.033333
	Male	0.005556	0.000000	0.133333	0.138889
All		0.444444	0.333333	0.222222	1.000000

Customer Profiling for KP281:

- Females with usage=2 and income less than 58k dollars

- Females with usage=3
- Partenered females are more keen to buy KP281
- Males with usage of 3 or 4 times a week
- Customers having income less than 58k dollars(lower income category)
- Customers who are planning to run less than 120 miles per week
- Males having fitness equal to 3 are more likely to buy KP281
- Customers having education of 14-16 years

Customer Profiling for KP481:

- Females with usage<=3
- Partenered Males are more keen to buy KP481 as compared to single males
- Males with usage of 3 or 4 times a week
- Customers having income less than 58k dollars(lower income category)
- Customers who are planning to run less than 120 miles per week
- Males having fitness equal to 3 are more likely to buy KP481
- Customers having education of 14-16 years

Customer Profiling for KP781:

- Customers having income having 58k dollars or more (high income group)
- Customers who are planning to run 120 miles or more per week
- Customer who rate their fitness equal to 5 are more likely to buy KP781
- Customers having usage for 4 or more times a week are more likely to buy KP781
- Males are more likely to buy KP781 as compared to females
- Couples with income more than 58k are more likely to buy as compared to single customers
- Customers who has education of more than 16 years are more likely to buy KP781 which shows its correlation with high income customers

Recommendations:

- KP781 should be marketed as a premium model and it should be promoted to high salaried people and heavy usage customers like sportsperson.
- Buyers of KP281 and KP481 have similar characteristics and their prices are also similar, so Aerofit can sell KP481 by promoting KP481 as a better deal to prospective buyers as KP481 has more enhanced features with almost similar price to KP281,generating more revenue.