Business Case: Aerofit - Descriptive Statistics & Probability

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 new customers. The team decides to investigate whether there are differences across the product with respect to customer characteristics.

● Perform descriptive analytics to create a customer profile for each AeroFit treadmill product by developing appropriate tables and charts.

● Construct two-way contingency tables for each AeroFit treadmill product and compute all conditional and marginal probabilities and their insights/impact on the business.

Dataset

The company collected the data on individuals who purchased a treadmill from the AeroFit stores during the prior three months. The dataset has the following features:

- Product Purchased: KP281, KP481, or KP781
- Age: In years
- Gender: Male/Female
- Education: In years
- MaritalStatus: Single or partnered
- Usage: The average number of times the customer plans to use the treadmill each week.
- Income: Annual income (in $)
- Fitness: Self-rated fitness on a 1-to-5 scale, where 1 is the poor shape and 5 is the excellent shape.
- Miles: The average number of miles the customer expects to walk/run each week

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
import copy
```

```
df = pd.read_csv('https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/125/original/aerofit_treadmill.csv?1639992749')
df
```

	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
...	...	...	...	...	...	...	...	...	...
175	KP781	40	Male	21	Single	6	5	83416	200
176	KP781	42	Male	18	Single	5	4	89641	200
177	KP781	45	Male	16	Single	5	5	90886	160
178	KP781	47	Male	18	Partnered	4	5	104581	120
179	KP781	48	Male	18	Partnered	4	5	95508	180

180 rows × 9 columns

Exploratory Data

```
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
```

Insights

- Data has total 180 rows and 9 columns
- Data type present of columns are 'object' and 'int64'
- 'Non-Null Count' of all column is equal to number of rows that means there is no missing values in data

```
df.nunique()
```

```
Product           3
Age              32
Gender            2
Education         8
MaritalStatus     2
Usage             6
Fitness           5
Income           62
Miles            37
dtype: int64
```

Insights

There is no column present in data with all the unique values

Statistical Summary

```
df.describe(include="all")
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
count	180	180.000000	180	180.000000	180	180.000000	180.000000	180.000000	180.000000
unique	3	NaN	2	NaN	2	NaN	NaN	NaN	NaN
top	KP281	NaN	Male	NaN	Partnered	NaN	NaN	NaN	NaN
freq	80	NaN	104	NaN	107	NaN	NaN	NaN	NaN
mean	NaN	28.788889	NaN	15.572222	NaN	3.455556	3.311111	53719.577778	103.194444
std	NaN	6.943498	NaN	1.617055	NaN	1.084797	0.958869	16506.684226	51.863605
min	NaN	18.000000	NaN	12.000000	NaN	2.000000	1.000000	29562.000000	21.000000
25%	NaN	24.000000	NaN	14.000000	NaN	3.000000	3.000000	44058.750000	66.000000
50%	NaN	26.000000	NaN	16.000000	NaN	3.000000	3.000000	50596.500000	94.000000
75%	NaN	33.000000	NaN	16.000000	NaN	4.000000	4.000000	58668.000000	114.750000
max	NaN	50.000000	NaN	21.000000	NaN	7.000000	5.000000	104581.000000	360.000000

Insights

- The product model KP281 has highest sales performance among the three products.
- all model of Products are more popolar among males then compare to females.

- Based on the data higher number of buyers were Married compare to single.
- The age range of customers spans from 18 to 50 year, with an average age of 29 years.
- Customer education levels vary between 12 and 21 years, with an average education duration of 16 years.
- Customers intend to utilize the product anywhere from 2 to 7 times per week, with an average usage frequency of 3 times per week.
- On average, customers have rated their fitness at 3 on a 5-point scale, reflecting a moderate level of fitness.
- The annual income of customers falls within the range of USD 30,000 to USD 100,000, with an average income of approximately USD 54,000.
- Customers' weekly running goals range from 21 to 360 miles, with an average target of 103 miles per week.

Handing Null-values and Outliers

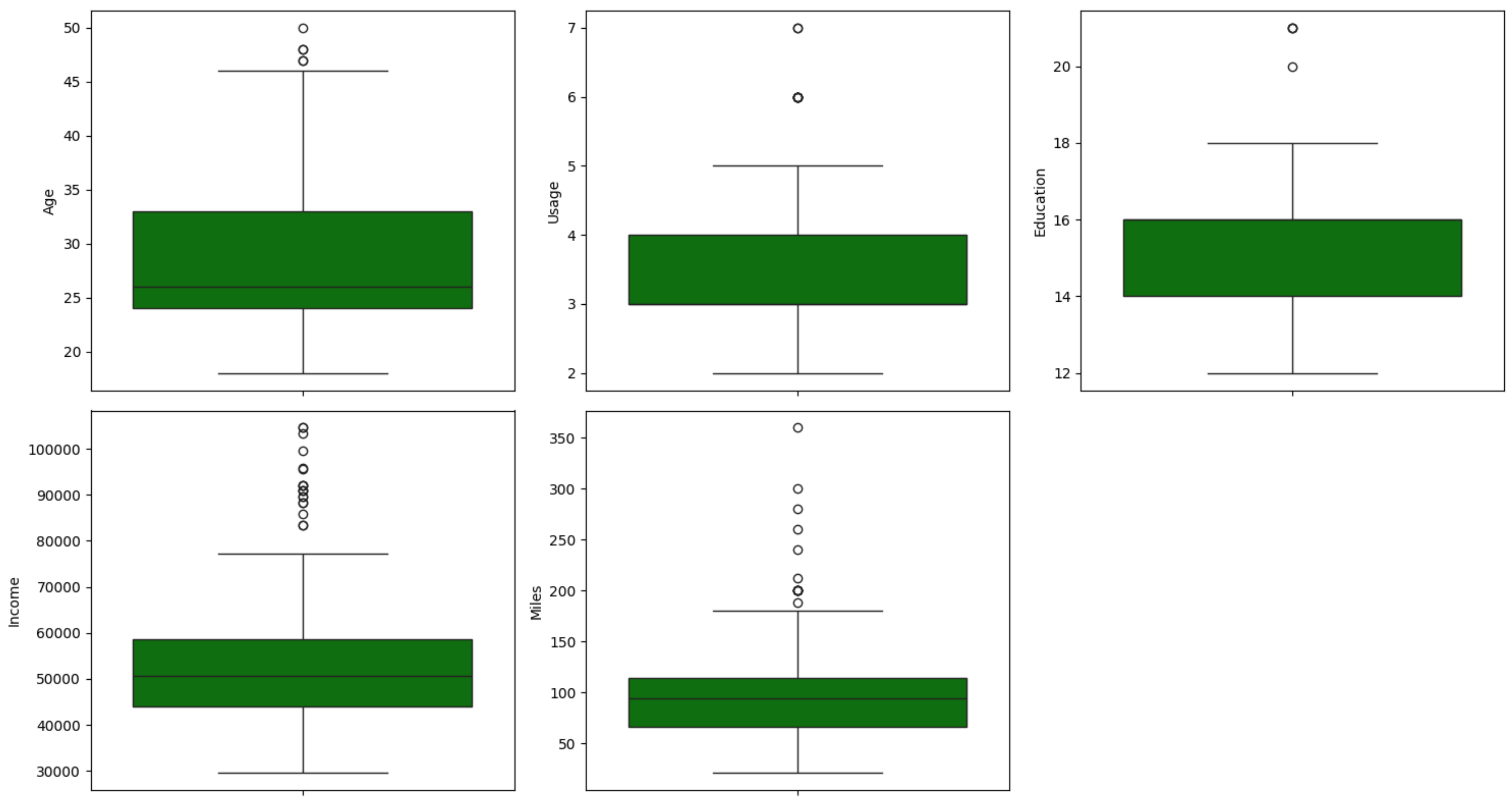
```
df.isna().sum()
```

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

Insights

no Null values in data

```
plt.figure(figsize=(15,8))

# Box Plot for Age
plt.subplot(2,3,1)
sns.boxplot(df['Age'],color='green')

# Box Plot for Usage
plt.subplot(2,3,2)
sns.boxplot(df['Usage'],color='green')

#Box Plot for Fitness
plt.subplot(2,3,3)
sns.boxplot(df['Education'],color='green')

#Box Plot for Income
plt.subplot(2,3,4)
sns.boxplot(df['Income'],color='green')

#Box Plot for Miles
plt.subplot(2,3,5)
sns.boxplot(df['Miles'],color='green')

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



Insights

Based on this graphical representation, it is evident that both 'Income' and 'Miles' have a huge number of outliers compare to 'age', 'usage' and 'education' years.

```
#Remove/clip the data between the 5 percentile and 95 percentile
def detect_outleirs(new_df,var):
    # Calculate the IQR for the variable
    Q1 = np.percentile(new_df, 25)
    Q3 = np.percentile(new_df, 75)
    Q2 = np.percentile(new_df, 50)
    IQR = Q3 - Q1

    # Define the outlier thresholds
    lower_threshold = Q1 - 1.5 * IQR
    upper_threshold = Q3 + 1.5 * IQR

    # Find the outliers for the variable
    outliers = new_df[(new_df< lower_threshold) | (new_df> upper_threshold)]

    # Calculate the percentage of outliers
    #outlier_percentage = round(len(outliers) / len(df) * 100, 2 )

    # Output the percentage of outliers
    print(f'{var}-->')
    print("Q1: ",Q1)
    print("Q3: ",Q3)
    print("Median: ",Q2)
    print(f"Inner Quartile Range for {var}: {Q3}-{Q1}")
    print(f"{var} Outlier count : {len(outliers)}")
    #print(f"Percentage of outliers for {var}: {outlier_percentage}% \n")
    return np.clip(new_df, np.percentile(new_df, 5), np.percentile(new_df,95))
```

```
clipped_age= (detect_outleirs(df['Age'],df.columns[1]))
```

```
Age-->
Q1:  24.0
Q3:  33.0
Median:  26.0
Inner Quartile Range for Age: 33.0-24.0
Age Outlier count : 5
```

```
clipped_usage= (detect_outleirs(df['Usage'],df.columns[5]))
```

```
Usage-->
Q1:  3.0
Q3:  4.0
Median:  3.0
Inner Quartile Range for Usage: 4.0-3.0
Usage Outlier count : 9
```

```
clipped_education = (detect_outleirs(df['Education'],df.columns[3]))
```

```
Education-->
Q1:  14.0
Q3:  16.0
Median:  16.0
Inner Quartile Range for Education: 16.0-14.0
Education Outlier count : 4
```

```
clipped_income= (detect_outleirs(df['Income'],df.columns[7]))
```

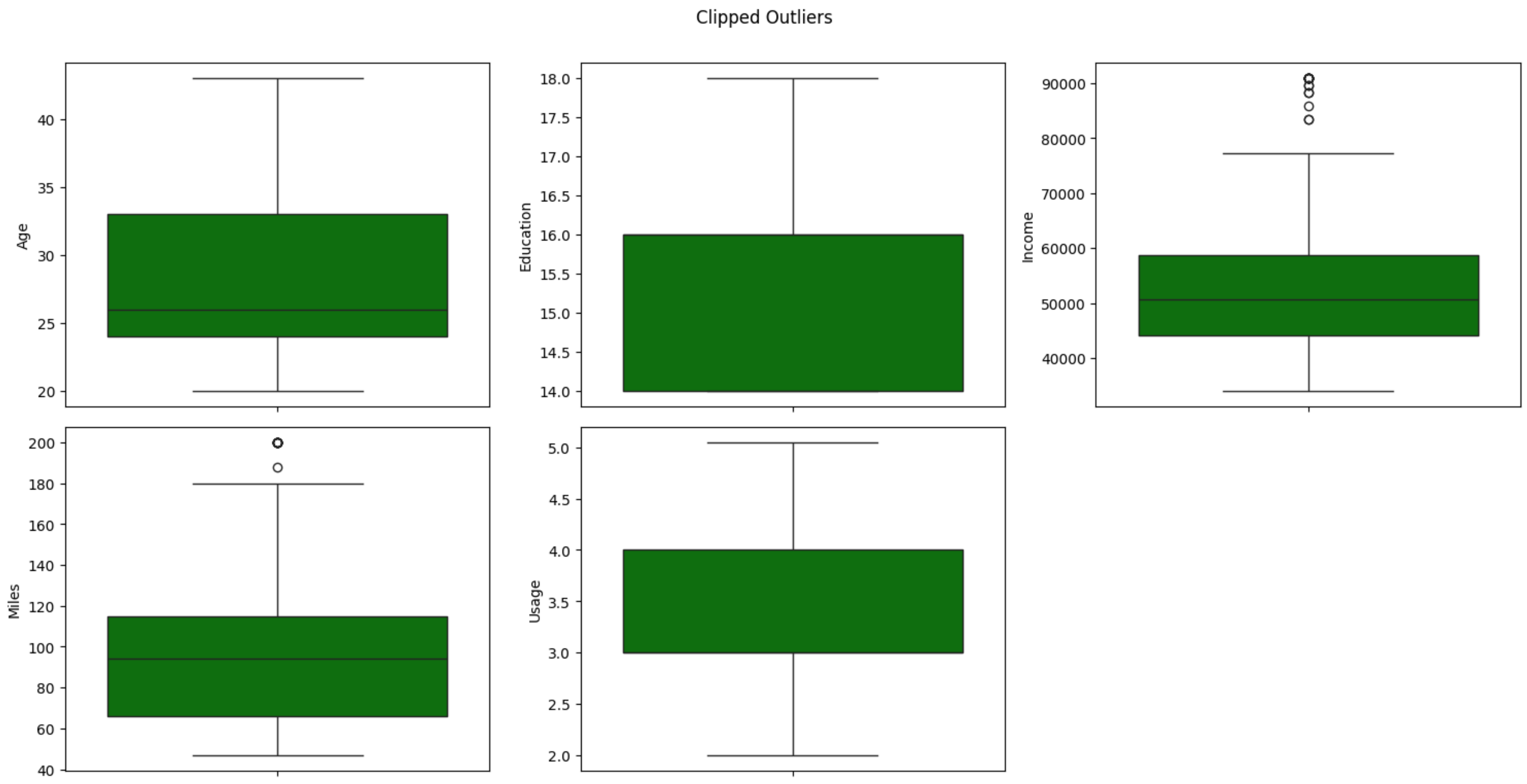
```
Income-->
Q1:  44058.75
```

```
Q3:  58668.0
Median:  50596.5
Inner Quartile Range for Income: 58668.0-44058.75
Income Outlier count : 19
```

```
clipped_miles= (detect_outleirs(df['Miles'],df.columns[8]))
```

```
Miles-->
Q1:  66.0
Q3:  114.75
Median:  94.0
Inner Quartile Range for Miles: 114.75-66.0
Miles Outlier count : 13
```

```
fig=plt.figure(figsize=(15,8))
fig.suptitle("\nClipped Outliers\n")

plt.subplot(2,3,1)
sns.boxplot(data=df,y=clipped_age,color='green')

plt.subplot(2,3,2)
sns.boxplot(data=df,y=clipped_education,color='green')

plt.subplot(2,3,3)
sns.boxplot(data=df,y=clipped_income,color='green')

plt.subplot(2,3,4)
sns.boxplot(data=df,y=clipped_miles,color='green')

plt.subplot(2,3,5)
sns.boxplot(data=df,y=clipped_usage,color='green')

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



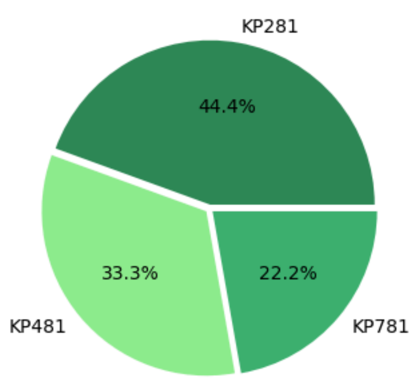
Univariate Analysis & Bivariate Analysis

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

Product	count
KP281	80
KP481	60
KP781	40

```
fig = plt.figure(figsize = (16,4))
fig.suptitle('Product Sales Distribution',fontweight = 'bold')
gs = fig.add_gridspec(2,2)
ax0 = fig.add_subplot(gs[:,0])
color_1 = ['seagreen','lightgreen','mediumseagreen']
graph = plt.pie(df.Product.value_counts(),explode=(0.025,0.025,0.025), labels=df.Product.value_counts().index,autopct='%1.1f%%',colors=color_1 )
ax0.axis('off')
ax1 = fig.add_subplot(gs[:,1])
product_portfolio = [['KP281','$1500','$120k'],['KP481','$1750','$105k'],['KP781','$2500','$100k']]
color_2 = [['seagreen','seagreen','seagreen'],['lightgreen','lightgreen','lightgreen'],['seagreen','seagreen','seagreen']]

table = ax1.table(cellText = product_portfolio, cellColours=color_2, cellLoc='center',colLabels =['Product','Price','Sales'],
                  colLoc = 'center',bbox =[0, 0, 1, 1])
ax1.axis('off')
table.set_fontsize(13)
plt.show()
```

Product Sales Distribution



Product	Price	Sales
KP281	$1500	$120k
KP481	$1750	$105k
KP781	$2500	$100k

Insight

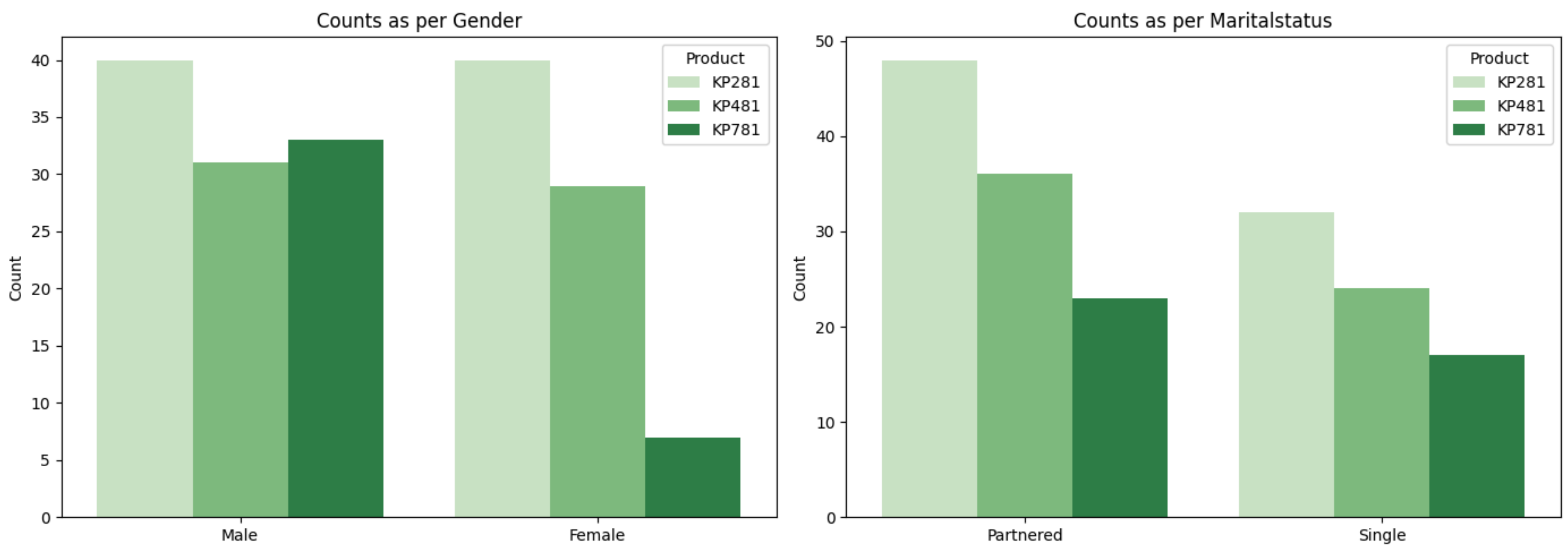
- The KP281 treadmill model, has the highest number of units sold.
- All three models have nearly equal contributions in terms of generating sales revenue

Double-click (or enter) to edit

```
categorical_columns= ['Gender', 'MaritalStatus']
 #a) Non-graphical analysis: Value counts for each categorical variable
for column in categorical_columns:
    print(f"{df.groupby([column])['Product'].value_counts()}\n")
```

```
Gender  Product
Female  KP281      40
        KP481      29
        KP781       7
Male    KP281      40
        KP781      33
        KP481      31
Name: count, dtype: int64

MaritalStatus  Product
Partnered      KP281      48
               KP481      36
               KP781      23
Single         KP281      32
               KP481      24
               KP781      17
Name: count, dtype: int64
```

```
fig,axes = plt.subplots(1,2,figsize=(14,5))
for i, column in enumerate(categorical_columns):
    order = df[column].value_counts().index[:10]
    sns.countplot(x=column, data=df, order=order, ax=axes[i], hue='Product',palette='Greens')
    axes[i].set_title(f'Counts as per {column.capitalize()}')
    axes[i].set_xlabel('')
    axes[i].set_ylabel('Count')
    axes[i].tick_params(axis='y',labelsize=10)
    axes[i].tick_params(axis='x',labelsize=10)
plt.tight_layout()
plt.show()
```



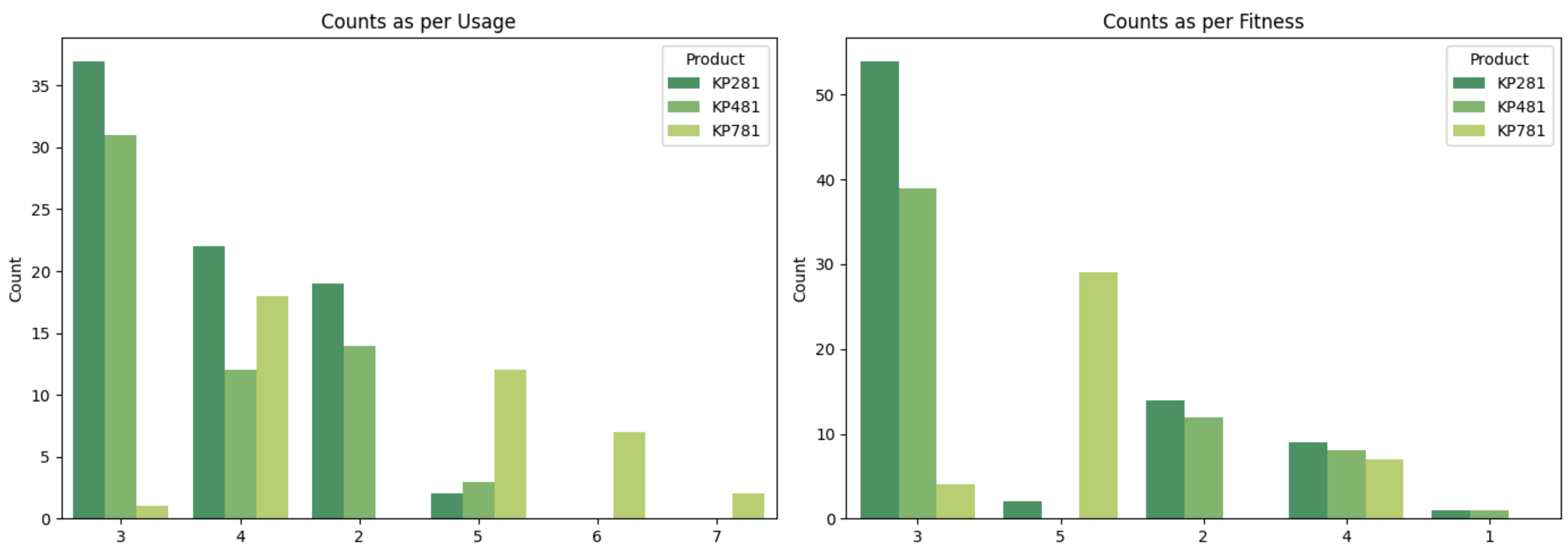
Insights

model KP281 and model KP481 is is equally popolar among both the gender , however model KP781 is not prefered by female

```
categorical_columns1= ['Usage', 'Fitness']
 #a) Non-graphical analysis: Value counts for each categorical variable
for column in categorical_columns:
    print(f"{df.groupby([column])['Product'].value_counts().to_frame()}\n")
```

```
                count
Gender Product
Female KP281       40
       KP481       29
       KP781        7
Male   KP281       40
       KP781       33
       KP481       31

                     count
MaritalStatus Product
Partnered     KP281       48
              KP481       36
              KP781       23
Single        KP281       32
              KP481       24
              KP781       17
```

```
fig,axes = plt.subplots(1,2,figsize=(14,5))
for i, column in enumerate(categorical_columns1):
    order = df[column].value_counts().index[:10]
    sns.countplot(x=column, data=df, order=order, ax=axes[i], hue='Product',palette='summer')
    axes[i].set_title(f'Counts as per {column.capitalize()}')
    axes[i].set_xlabel('')
    axes[i].set_ylabel('Count')
    axes[i].tick_params(axis='y',labelsize=10)
    axes[i].tick_params(axis='x',labelsize=10)
plt.tight_layout()
plt.show()
```



Insights

- Almost 85% of the customers plan to use the treadmill for 2 to 4 times a week and only 15% using 5 times and above each week
- large number of the customers have self-evaluated their fitness at a level 3 on a scale of 1 to 5 in entry-level model KP281. whereas in has advanced featured model KP781 highest number of customer self-evaluated level 5.

```
age_df= df.groupby(['Age'])['Product'].value_counts().reset_index()
age_df.head()
```

	Age	Product	count
0	18	KP281	1
1	19	KP281	3
2	19	KP481	1
3	20	KP481	3
4	20	KP281	2

Next steps: Generate code with age_df View recommended plots

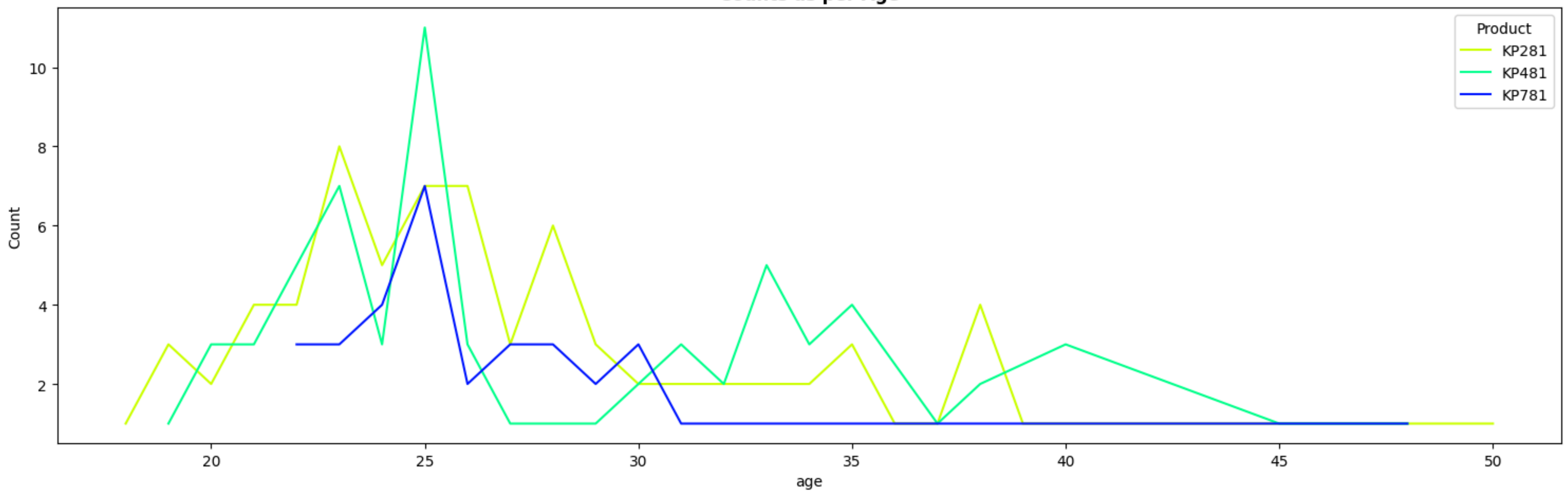
```
education_df=df.groupby(['Education'])['Product'].value_counts().reset_index()
education_df.head()
```

	Education	Product	count
0	12	KP281	2
1	12	KP481	1
2	13	KP281	3
3	13	KP481	2
4	14	KP281	30

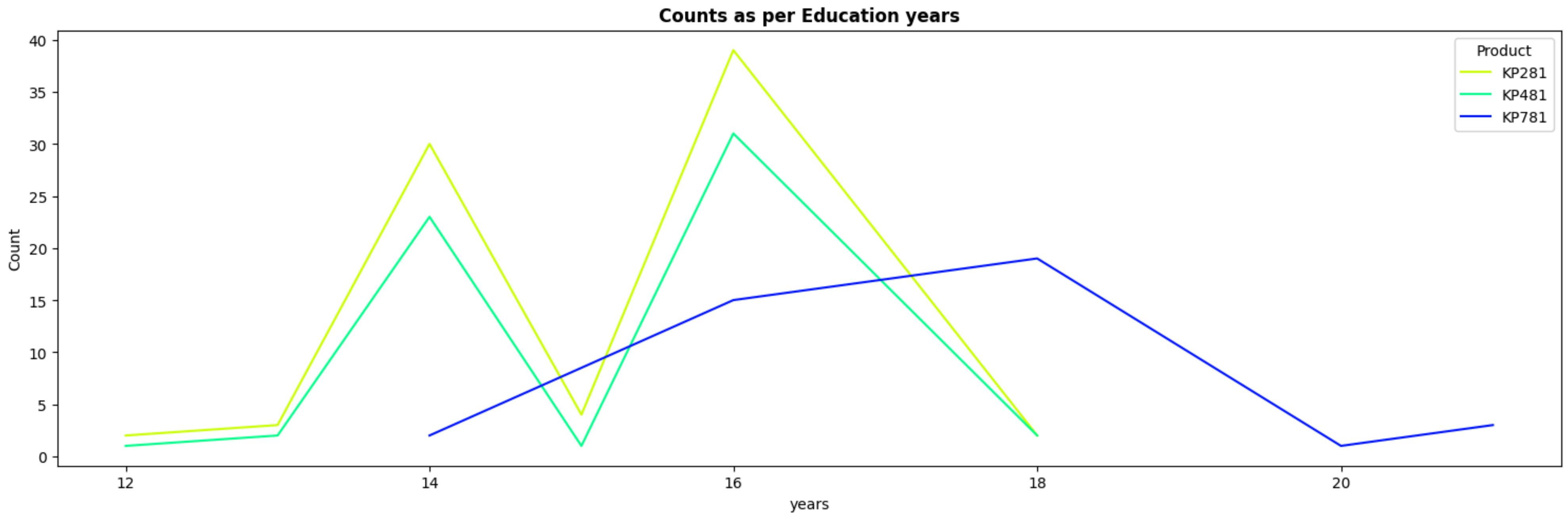
Next steps: Generate code with education_df View recommended plots

```
fig = plt.figure(figsize = (15,10))
gs = fig.add_gridspec(2,1)
ax0 = fig.add_subplot(gs[0,:])
sns.lineplot(x='Age',y='count',data= age_df, hue='Product',ax = ax0,palette='gist_rainbow')
ax0.set_title('Counts as per Age',fontweight = 'bold')
plt.xlabel('age')
plt.ylabel('Count')
plt.tick_params(axis='y',labelsize=10)
plt.tick_params(axis='x',labelsize=10)
plt.tight_layout()
ax1 = fig.add_subplot(gs[1,:])
sns.lineplot(x='Education',y='count',data= education_df, hue='Product',ax = ax1,palette='gist_rainbow')
ax1.set_title('Counts as per Education years',fontweight = 'bold')
plt.xlabel('years')
plt.ylabel('Count')
plt.tick_params(axis='y',labelsize=10)
plt.tick_params(axis='x',labelsize=10)
plt.tight_layout()
plt.show()
```

Counts as per Age



Counts as per Education years



Insights

- All type of models are purchased mostly by people between age group of 20-30.
- customers with age above 30 does not show any interest in purchasing advanced featured model KP781

Representing the Probability Marginal & Conditional Probability

```
marginal_probability_crosstab = df['Product'].value_counts(normalize = True ).round(2)
marginal_probability_crosstab.to_frame()
```

	proportion
Product	
KP281	0.44
KP481	0.33
KP781	0.22

Insights

The Marginal Probability of a treadmill Model KP281 entry-level treadmill being purchased is 44%.

The Marginal Probability of a treadmill Model KP281 mid-level runners treadmill being purchased is 33%.

The Marginal Probability of a treadmill Model KP281 advanced featured treadmill being purchased is 22%.

```
#binning the age values into categories
bin_range1 = [17,25,35,45,float('inf')]
bin_labels1 = ['Young Adults', 'Adults', 'Middle Aged Adults', 'Elder']
df['age_group'] = pd.cut(df['Age'],bins = bin_range1,labels = bin_labels1)
```

```
#Probability of product purchase w.r.t. gender
pd.crosstab(index =df['Product'],columns = df['Gender'],margins = True,normalize = True ).round(2)
```

Gender	Female	Male	All
Product			
KP281	0.22	0.22	0.44
KP481	0.16	0.17	0.33
KP781	0.04	0.18	0.22
All	0.42	0.58	1.00

Insights

The Marginal Probability of a treadmill being purchased by a female is 42%.

The conditional probability of purchasing the treadmill model given that the customer is female is:

For Treadmill model KP281 - 22%

For Treadmill model KP481 - 16%

For Treadmill model KP781 - 4%

The Marginal Probability of a treadmill being purchased by a male is 58%.

The conditional probability of purchasing the treadmill model given that the customer is male is:

For Treadmill model KP281 - 22%

For Treadmill model KP481 - 17%

For Treadmill model KP781 - 18%

```
#binning the age values into categories
bin_range1 = [17,25,35,45,float('inf')]
bin_labels1 = ['Young Adults', 'Adults', 'Middle Aged Adults', 'Elder']

df['age_group'] = pd.cut(df['Age'],bins = bin_range1,labels = bin_labels1)
```

```
#Probability of product purchase w.r.t. Age
pd.crosstab(index =df['Product'],columns = df['age_group'],margins = True,normalize = True ).round(2)
```

age_group	Young Adults	Adults	Middle Aged Adults	Elder	All
Product					
KP281	0.19	0.18	0.06	0.02	0.44
KP481	0.16	0.13	0.04	0.01	0.33
KP781	0.09	0.09	0.02	0.01	0.22
All	0.44	0.41	0.12	0.03	1.00

Insights

The Marginal Probability of a treadmill being purchased by a Young Adult(18-25) is 44%.

The conditional probability of purchasing the treadmill model given that the customer is Young Adult is:

For Treadmill model KP281 - 19%

For Treadmill model KP481 - 16%

For Treadmill model KP781 - 9%

The Marginal Probability of a treadmill being purchased by a Adult(26-35) is 41%.

The conditional probability of purchasing the treadmill model given that the customer is Adult is:

For Treadmill model KP281 - 18%

For Treadmill model KP481 - 13%

For Treadmill model KP781 - 9%

The Marginal Probability of a treadmill being purchased by a Middle Aged(36-45) is 12%.

The Marginal Probability of a treadmill being purchased by a Elder(Above 45) is only 3%.

```
#binning the education values into categories
bin_range2 = [0,12,15,float('inf')]
bin_labels2 = ['Primary Education', 'Secondary Education', 'Higher Education']
df['edu_group'] = pd.cut(df['Education'],bins = bin_range2,labels = bin_labels2)
```

```
#Probability of product purchase w.r.t. Education level
pd.crosstab(index =df['Product'],columns = df['edu_group'],margins = True,normalize = True ).round(2)
```

edu_group Product	Primary Education	Secondary Education	Higher Education	All
KP281	0.01	0.21	0.23	0.44
KP481	0.01	0.14	0.18	0.33
KP781	0.00	0.01	0.21	0.22
All	0.02	0.36	0.62	1.00

Insights

The Marginal Probability of a treadmill being purchased by a customer with Higher Education(Above 15 Years) is 62%.

The conditional probability of purchasing the treadmill model given that the customer has Higher Education is:

For Treadmill model KP281 - 23%

For Treadmill model KP481 - 18%

For Treadmill model KP781 - 21%

The Marginal Probability of a treadmill being purchased by a customer with Secondary Education(13-15 yrs) is 36%.

The conditional probability of purchasing the treadmill model given that the customer has Secondary Education is:

For Treadmill model KP281 - 21%

For Treadmill model KP481 - 14%

For Treadmill model KP781 - 1%

The Marginal Probability of a treadmill being purchased by a customer with Primary Education(0 to 12 yrs) is only 2%.

```
#binning the income values into categories
bin_range3 = [0,40000,60000,80000,float('inf')]
bin_labels3 = ['Low Income','Moderate Income','High Income','Very High Income']

df['income_group'] = pd.cut(df['Income'],bins = bin_range3,labels = bin_labels3)
```

```
#Probability of product purchase w.r.t. Income
pd.crosstab(index =df['Product'],columns = df['income_group'],margins = True,normalize = True ).round(2)
```

income_group Product	Low Income	Moderate Income	High Income	Very High Income	All
KP281	0.13	0.28	0.03	0.00	0.44
KP481	0.05	0.24	0.04	0.00	0.33
KP781	0.00	0.06	0.06	0.11	0.22
All	0.18	0.59	0.13	0.11	1.00

Insights

The Marginal Probability of a treadmill being purchased by a customer with Low Income(<40k) is 18%.

The conditional probability of purchasing the treadmill model given that the customer has Low Income is :

For Treadmill model KP281 - 13%

For Treadmill model KP481 - 5%

For Treadmill model KP781 - 0%

The Marginal Probability of a treadmill being purchased by a customer with Moderate Income(40k - 60k) is 59%.

The conditional probability of purchasing the treadmill model given that the customer has Moderate Income is :

For Treadmill model KP281 - 28%

For Treadmill model KP481 - 24%

For Treadmill model KP781 - 6%

The Marginal Probability of a treadmill being purchased by a customer with High Income(60k - 80k) is 13%

The conditional probability of purchasing the treadmill model given that the customer has High Income is :

For Treadmill model KP281 - 3%

For Treadmill model KP481 - 4%

For Treadmill model KP781 - 6%

The Marginal Probability of a treadmill being purchased by a customer with Very High Income(>80k) is 11%

The conditional probability of purchasing the treadmill model given that the customer has High Income is :

For Treadmill model KP281 - 0%

For Treadmill model KP481 - 0%

For Treadmill model KP781 - 11%

```
#Probability of product purchase w.r.t. Marital Status
pd.crosstab(index =df['Product'],columns = df['MaritalStatus'],margins = True,normalize = True ).round(2)
```

MaritalStatus Product	Partnered	Single	All
KP281	0.27	0.18	0.44
KP481	0.20	0.13	0.33
KP781	0.13	0.09	0.22
All	0.59	0.41	1.00

Insights

The Marginal Probability of a treadmill being purchased by a Married Customer is 59%.

The conditional probability of purchasing the treadmill model given that the customer is Married is:

For Treadmill model KP281 - 27%

For Treadmill model KP481 - 20%

For Treadmill model KP781 - 13%

The Marginal Probability of a treadmill being purchased by a Unmarried Customer is 41%.

The conditional probability of purchasing the treadmill model given that the customer is Unmarried is:

For Treadmill model KP281 - 18%

For Treadmill model KP481 - 13%

```
#Probability of product purchase w.r.t. Weekly Usage
pd.crosstab(index =df['Product'],columns = df['Usage'],margins = True,normalize = True ).round(2)
```

Usage Product	2	3	4	5	6	7	All
KP281	0.11	0.21	0.12	0.01	0.00	0.00	0.44
KP481	0.08	0.17	0.07	0.02	0.00	0.00	0.33
KP781	0.00	0.01	0.10	0.07	0.04	0.01	0.22
All	0.18	0.38	0.29	0.09	0.04	0.01	1.00

Insights

The Marginal Probability of a treadmill being purchased by a customer with Usage 3 per week is 38%.

The conditional probability of purchasing the treadmill model given that the customer has Usage 3 per week is:

For Treadmill model KP281 - 21%

For Treadmill model KP481 - 17%

For Treadmill model KP781 - 1%

The Marginal Probability of a treadmill being purchased by a customer with Usage 4 per week is 29%.

The conditional probability of purchasing the treadmill model given that the customer has Usage 4 per week is:

For Treadmill model KP281 - 12%

For Treadmill model KP481 - 7%

For Treadmill model KP781 - 10%

The Marginal Probability of a treadmill being purchased by a customer with Usage 2 per week is 18%

The conditional probability of purchasing the treadmill model given that the customer has Usage 2 per week is:

For Treadmill model KP281 - 11%

For Treadmill model KP481 - 8%

For Treadmill model KP781 - 0%

```
#Probability of product purchase w.r.t. Customer Fitness
pd.crosstab(index =df['Product'],columns = df['Fitness'],margins = True,normalize = True ).round(2)
```

Fitness Product	1	2	3	4	5	All
KP281	0.01	0.08	0.30	0.05	0.01	0.44
KP481	0.01	0.07	0.22	0.04	0.00	0.33
KP781	0.00	0.00	0.02	0.04	0.16	0.22
All	0.01	0.14	0.54	0.13	0.17	1.00

Insights

The Marginal Probability of a treadmill being purchased by a customer with Average(3) Fitness is 54%.

The conditional probability of purchasing the treadmill model given that the customer has Average Fitness is:

For Treadmill model KP281 - 30%

For Treadmill model KP481 - 22%

For Treadmill model KP781 - 2%

The Marginal Probability of a treadmill being purchased by a customer with Fitness of 2,4,5 is almost 15%.

The Marginal Probability of a treadmill being purchased by a customer with very low(1) Fitness is only 1%.

```
#binning the miles values into categories
bin_range4 = [0,50,100,200,float('inf')]
bin_labels4 = ['Light Activity', 'Moderate Activity', 'Active Lifestyle', 'Fitness Enthusiast ']

df['miles_group'] = pd.cut(df['Miles'],bins = bin_range4,labels = bin_labels4)
```

```
# Probability of product purchase w.r.t. weekly mileage
pd.crosstab(index =df['Product'],columns = df['miles_group'],margins = True,normalize = True ).round(2)
```

miles_group Product	Light Activity	Moderate Activity	Active Lifestyle	Fitness Enthusiast	All
KP281	0.07	0.28	0.10	0.00	0.44
KP481	0.03	0.22	0.08	0.01	0.33
KP781	0.00	0.04	0.15	0.03	0.22
All	0.09	0.54	0.33	0.03	1.00

Insights

The Marginal Probability of a treadmill being purchased by a customer with lifestyle of Light Activity(0 to 50 miles/week) is 9%.

The conditional probability of purchasing the treadmill model given that the customer has Light Activity Lifestyle is :

For Treadmill model KP281 - 7%

For Treadmill model KP481 - 3%

For Treadmill model KP781 - 0%

The Marginal Probability of a treadmill being purchased by a customer with lifestyle of Moderate Activity(51 to 100 miles/week) is 54%.

The conditional probability of purchasing the treadmill model given that the customer with lifestyle of Moderate Activity is:

For Treadmill model KP281 - 28%

For Treadmill model KP481 - 22%

For Treadmill model KP781 - 4%

The Marginal Probability of a treadmill being purchased by a customer has Active Lifestyle(100 to 200 miles/week) is 33%.

The conditional probability of purchasing the treadmill model given that the customer has Active Lifestyle is :

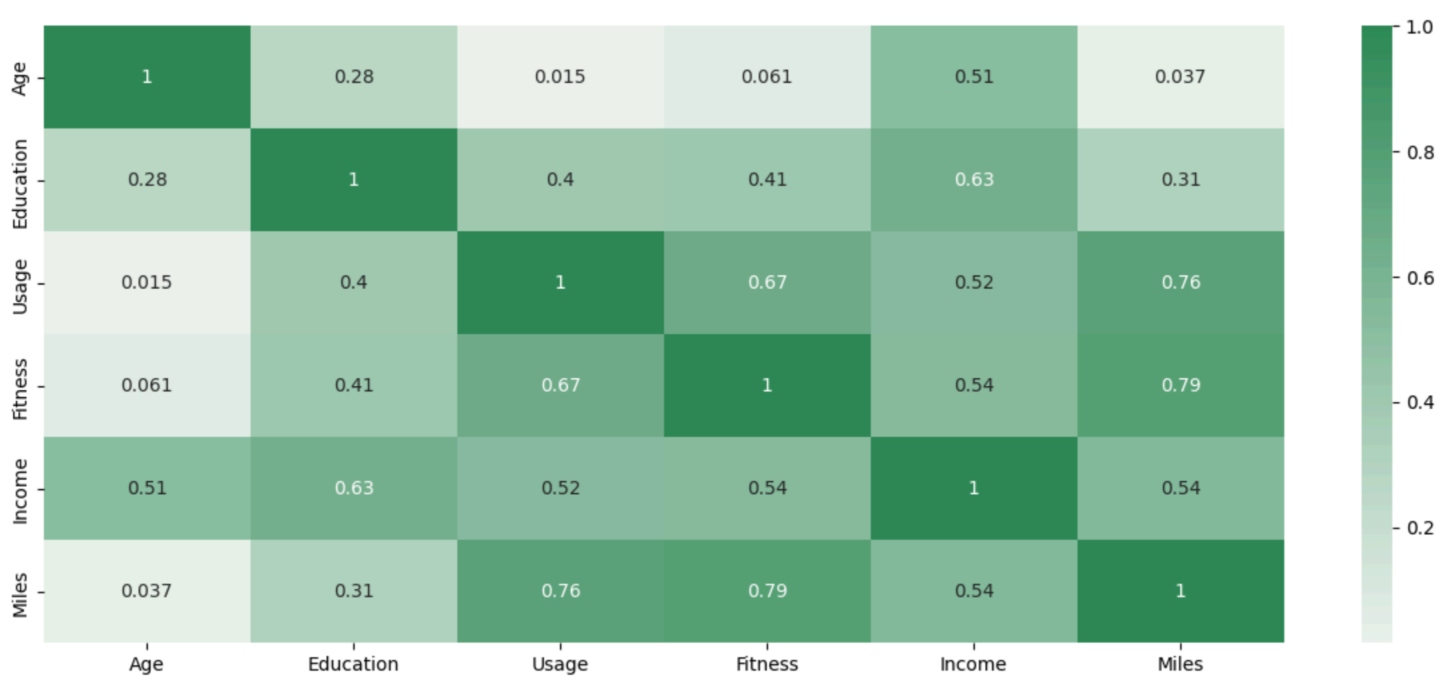
For Treadmill model KP281 - 10%

For Treadmill model KP481 - 8%

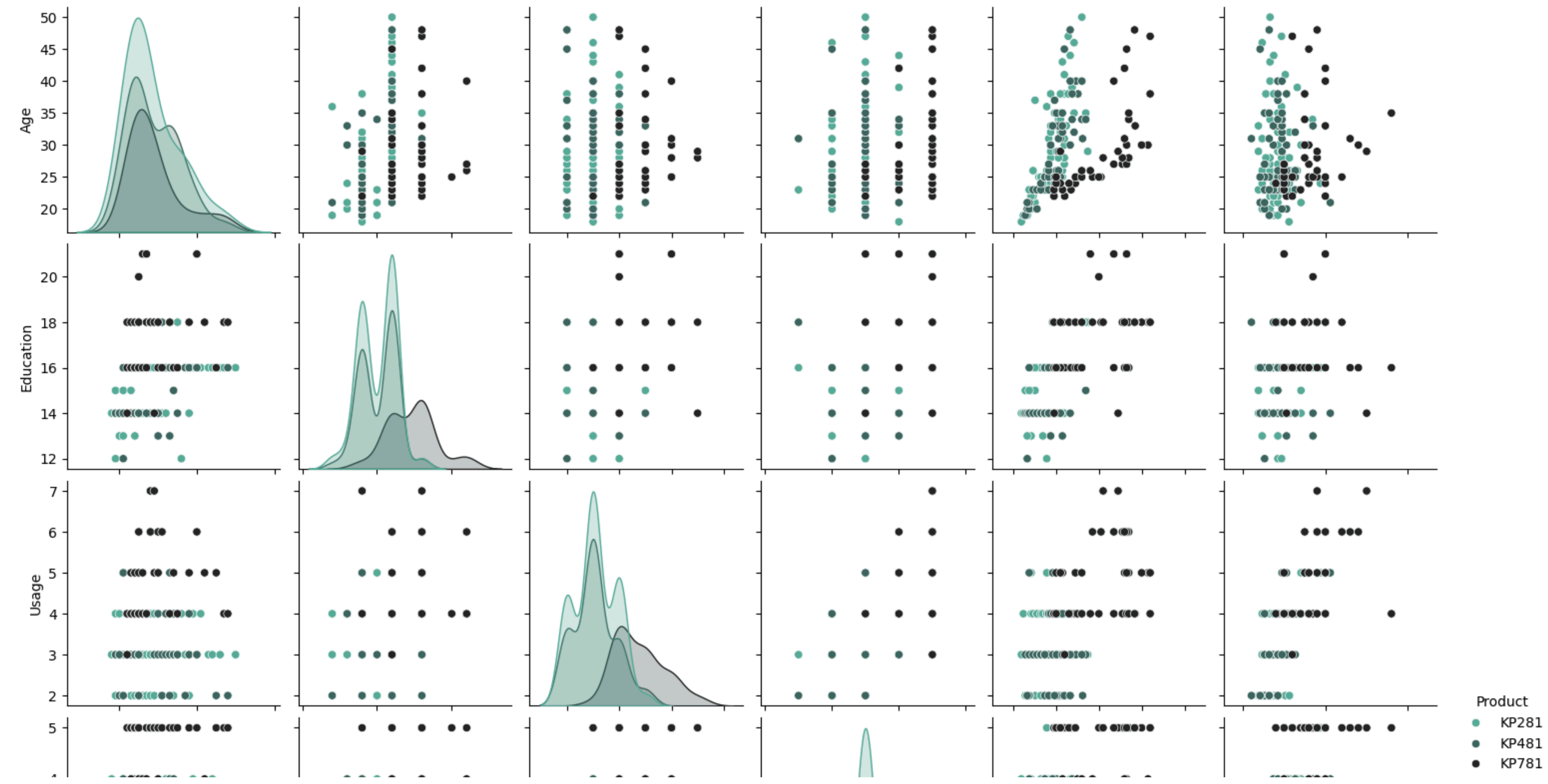
For Treadmill model KP781 - 15%

The Marginal Probability of a treadmill being purchased by a customer who is Fitness Enthusiast(>200 miles/week) is 3% only

```
corr_mat = df.corr(method='pearson', numeric_only = True)

plt.figure(figsize=(15,6))
colr=sns.light_palette("seagreen", as_cmap=True)
sns.heatmap(corr_mat,annot = True, cmap=colr)

plt.show()
```




```
sns.pairplot(df, hue ='Product', palette="dark:#5A9_r",)
plt.show()
```



Customer Profiling

1. Customer Profile for KP281 Treadmill

- Age of customer mainly between 18 to 35 years with few between 35 to 50 years
- Easily affordable entry level product, which is also the maximum selling product.
- Single female & Partnered male customers bought this product more than single male customers.
- Younger to Elder beginner level customers prefer this product.
- Education level of customer 13 years and above
- Annual Income of customer below $60,000
- The average planned usage is 3 times per week.
- Self-rated Fitness Scale lies between 2 to 4
- Weekly Running Mileage is 50 to 100 miles

2. Customer Profile for KP481 Treadmill

- Age of customer mainly between 18 to 35 years with few between 35 to 50 years
- Fitness Level of this product users varies from Bad to Average Shape depending on their usage.
- More Female customers prefer this product than males.
- More Partnered customers prefer this product.
- Education level of customer 13 years and above
- Annual Income of customer between $ 40,000 to 80,000
- The average planned usage is 3 times per week.
- Self-rated Fitness Scale lies between 2 to 4
- Weekly Running Mileage is 50 to 200 miles

3. Customer Profile for KP781 Treadmill

- Age of customer between 18 to 35 years
- Due to the High Price & being the advanced type, customer prefers less of this product.
- More Male customers prefer this product than Females.
- Education level of customer 15 years and above
- Annual Income of customer $ 80,000 and above
- Customers use 4 to 5 times a week at least.
- Self-rated Fitness Scale lies between 3 to 5
- Weekly Running Mileage is 100 miles and above
- Customers who have more experience with previous aerofit products tend to buy this product

Recommendations

- Focus on targeting customers with higher fitness levels by promoting the benefits of using fitness equipment regularly. Emphasize how regular usage can contribute to improving fitness and overall health.
- Female who prefer exercising equipments are very low here. Hence, we should run a marketing campaign on to encourage women to exercise more
- KP281 & KP481 treadmills are preferred by the customers whose annual income lies in the range of 39K - 53K Dollars. These models should promoted as budget treadmills.
- Enhance the marketing strategy for KP781 by associating it with renowned athletes , leveraging their achievements for better outreach.
- Conduct research to expand the customer base beyond 50 years of age. Offer basic treadmill models (KP281/KP481) as suitable options for beginners in this age group.
- For the model KP781 , consider providing flexible payment plans that allow customers to spread the cost over several months. This can make the treadmill more accessible to customers with varying budgets.
- Provide customer support and recommend users to upgrade from lower versions to next level versions after consistent usages.
- The KP781 model exhibits a significant sales disparity in terms of gender, very low of total sales attributed to female customers compare to male. To enhance this metric, it is recommended to implement targeted strategies such as offering special promotions on and trials exclusively designed for the female customers.