`Submitted by Priya Kushwaha`

Business Case: Aerofit - Descriptive Statistics & Probability

Table of Contents-Business Case: Aerofit - Descriptive Statistics & Probability

1. Import the dataset and do usual data analysis steps like checking the structure & characteristics of the dataset
2. Detect Outliers (using boxplot, “describe” method by checking the difference between mean and median)
3. Check if features like marital status, age have any effect on the product purchased (using countplot, histplots, boxplots etc)
4. Representing the marginal probability like - what percent of customers have purchased KP281, KP481, or KP781 in a table (can use pandas.crosstab here)
5. Check correlation among different factors using heat maps or pair plots.
6. With all the above steps you can answer questions like: What is the probability of a male customer buying a KP781 treadmill?
7. Customer Profiling - Categorization of users.
8. Probability- marginal, conditional probability.
9. Some recommendations and actionable insights, based on the inferences.

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.

Objective

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.

About Data

This tabuler dataset consist the data as 180 rows and 9 columns with detail such as Product, Age, Gender, Education, MaritalStatus, Usage, Fitness, Income, Miles.

Features of the data set-EDA

- **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

Exploratory Data Analysis

In [1]:
```
# Importing Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
#Loading the data set
df=pd.read_csv('aerofit_treadmill.txt')
```

 # Question 1:-Import the dataset and do usual data analysis steps like checking the structure & characteristics of the dataset

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

Out[ ]:

	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 [ ]:
```
df.tail()
```

Out[ ]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
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

In [ ]:
```
df.shape
```
Out[ ]:
```
(180, 9)
```
In [ ]:
```
df.size
```
Out[ ]:
```
1620
```
In [ ]:
```
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** From the above analysis it is clear that data has total 9 features.

Statistical Summary

In [ ]:
```
df.describe()
```
Out[ ]:

	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

In [3]:
```
df.describe(include='object')
```
Out[3]:

	Product	Gender	MaritalStatus
count	180	180	180
unique	3	2	2
top	KP281	Male	Partnered
freq	80	104	107

🔍 Insights

As per above analysis 3 unique products are available with quantity of 180 in which KP281 is contributing 44.5% , KP481 contributing 33.3% and KP781 contributing 22.2%. There are two gender in which male contributing 57.8%.

Duplicate detection

In [ ]:
```
df.isna().sum()
```

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

As per above analysis there are no missing value in any features.

Sanity check for columns

In [4]:
```
#Checking unique values for columns
for i in df.columns:
  print('Unique values in',i,'column are:-')
  print(df[i].unique())
  print('-'*70)
```

```
Unique values in Product column are:-
['KP281' 'KP481' 'KP781']
-----------------------------------------------------------------------
Unique values in Age column are:-
[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]
-----------------------------------------------------------------------
Unique values in Gender column are:-
['Male' 'Female']
-----------------------------------------------------------------------
Unique values in Education column are:-
[14 15 12 13 16 18 20 21]
-----------------------------------------------------------------------
Unique values in MaritalStatus column are:-
['Single' 'Partnered']
-----------------------------------------------------------------------
Unique values in Usage column are:-
[3 2 4 5 6 7]
-----------------------------------------------------------------------
Unique values in Fitness column are:-
[4 3 2 1 5]
-----------------------------------------------------------------------
Unique values in Income column are:-
[ 29562  31836  30699  32973  35247  37521  36384  38658  40932  34110
  39795  42069  44343  45480  46617  48891  53439  43206  52302  51165
  50028  54576  68220  55713  60261  67083  56850  59124  61398  57987
  64809  47754  65220  62535  48658  54781  48556  58516  53536  61006
  57271  52291  49801  62251  64741  70966  75946  74701  69721  83416
  88396  90886  92131  77191  52290  85906 103336  99601  89641  95866
 104581  95508]
-----------------------------------------------------------------------
Unique values in Miles column are:-
[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]
-----------------------------------------------------------------------
```

🔍 Insights

The dataset dose not contain any abnormal value.

Adding new columns for better analysis

- Creating new columns for categorise the value of columns Age,Education,Income and Miles for better visualization.

Age column

- Categorizing value of Age column in 4 different bucktes.

1. Young Adults: from 18-25
2. Adults: from 26-35
3. Middle Adults: from 36-45
4. Elders: from 46 above

Education column

- Categorizing value of education column in 3 different bucktes.

1. Primary Education: upto 12

2. Secondry Education: 13-15
3. Higher Education: 16 above

Income column

- Categorizing value of Income column in 4 different bucktes.

1. Low Income: upto 40000
2. Middle Income: 40000 - 60000
3. High Income: 60000 -80000
4. Very High Income: 80000 above

Miles column

- Categorizing value of Miles column in 4 different bucktes.

1. Light Activity: upto 50 miles
2. Moderate Activity : 51 - 100 miles
3. Active Lifestyle: 101- 200 miles
4. Fitness Enthusiast: above 200 miles

In [5]:
```
# Binning the age value into categories
bin_range1=[17, 25, 35, 45, float('inf')]
bin_labels1=['Young Adults','Adults','Middle Adults','Elders']

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

# Binning the education value into 3 categories
bin_range2=[0,12,15,float('inf')]
bin_labels2=['Primary Education', 'Secondry Education', 'Higher Education']

df['education_group']= pd.cut(df['Education'], bins=bin_range2, labels=bin_labels2)

# Binning the income value into 4 categories
bin_range3=[0, 40000, 60000, 80000, float('inf')]
bin_labels3=['Low Income', 'Middle Income', 'High Income', 'Very High Income']

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

# Binning the miles col into 4 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)
```

In [6]:
```
df.head(3)
```

Out[6]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles	age_group	education_group	income_group	miles_group
0	KP281	18	Male	14	Single	3	4	29562	112	Young Adults	Secondry Education	Low Income	Active Lifestyle
1	KP281	19	Male	15	Single	2	3	31836	75	Young Adults	Secondry Education	Low Income	Moderate Activity
2	KP281	19	Female	14	Partnered	4	3	30699	66	Young Adults	Secondry Education	Low Income	Moderate Activity

Non-Graphical Analysis

In [7]:
```
df.value_counts()
```

Out[7]:

```
Product  Age  Gender  Education  MaritalStatus  Usage  Fitness  Income  Miles  age_group     education_group     income_group      miles_group
KP281    18   Male    14         Single         3      4        29562   112    Young Adults  Secondry Education  Low Income        Active Lifestyle      1
KP481    30   Female  13         Single         4      3        46617   106    Adults        Secondry Education  Middle Income     Active Lifestyle      1
         31   Female  16         Partnered      2      3        51165   64     Adults        Higher Education    Middle Income     Moderate Activity     1
                      18         Single         2      1        65220   21     Adults        Higher Education    High Income       Light Activity        1
              Male    16         Partnered      3      3        52302   95     Adults        Higher Education    Middle Income     Moderate Activity     1
                                                                                                                                                     ..
KP281    34   Female  16         Single         2      2        52302   66     Adults        Higher Education    Middle Income     Moderate Activity     1
              Male    16         Single         4      5        51165   169    Adults        Higher Education    Middle Income     Active Lifestyle      1
         35   Female  16         Partnered      3      3        60261   94     Adults        Higher Education    High Income       Moderate Activity     1
                      18         Single         3      3        67083   85     Adults        Higher Education    High Income       Moderate Activity     1
KP781    48   Male    18         Partnered      4      5        95508   180    Elders        Higher Education    Very High Income  Active Lifestyle      1
Length: 180, dtype: int64
```

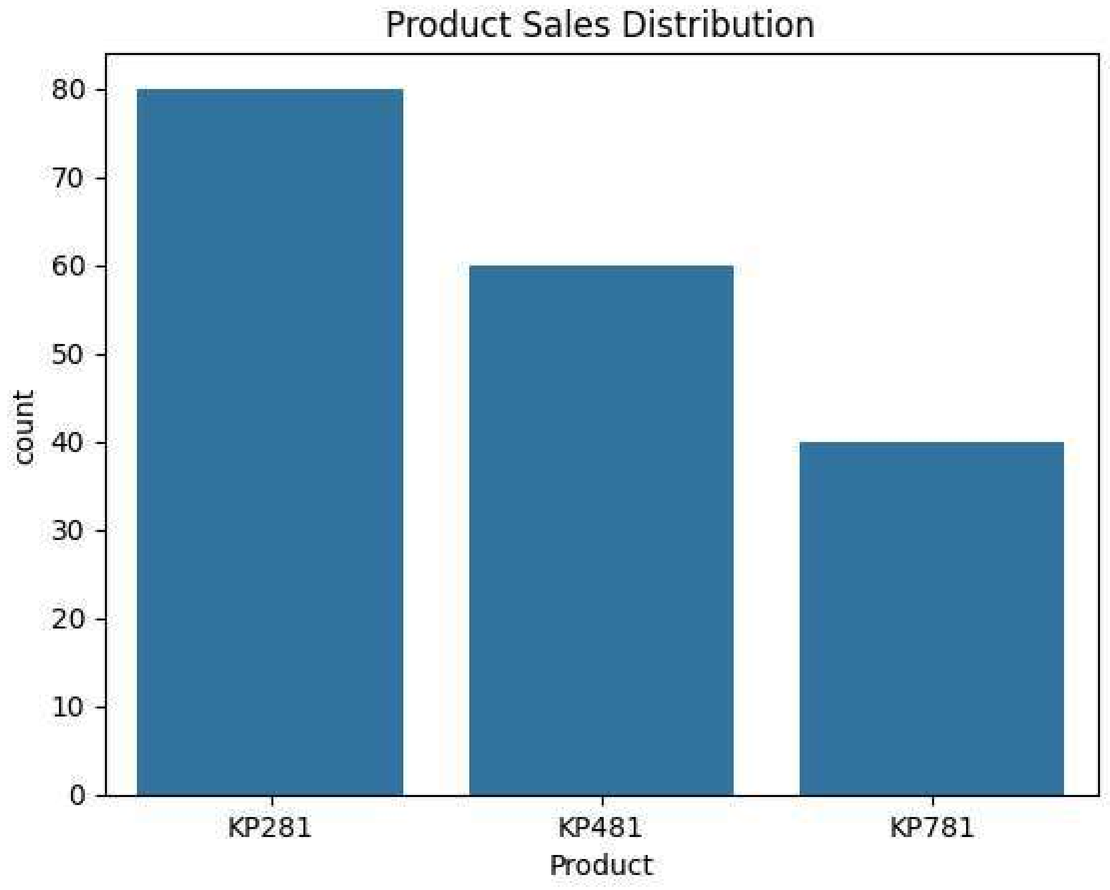
Univariate Analysis

Categorical Variables

- Product Sales Distribution

In [8]:

```
sns.countplot(data=df, x='Product')
plt.title('Product Sales Distribution')
plt.show()
```



 # Question 2:-Detect Outliers (using boxplot, “describe” method by checking the difference between mean and median)

Bivariate Analysis-Analysis of product type

In [9]:

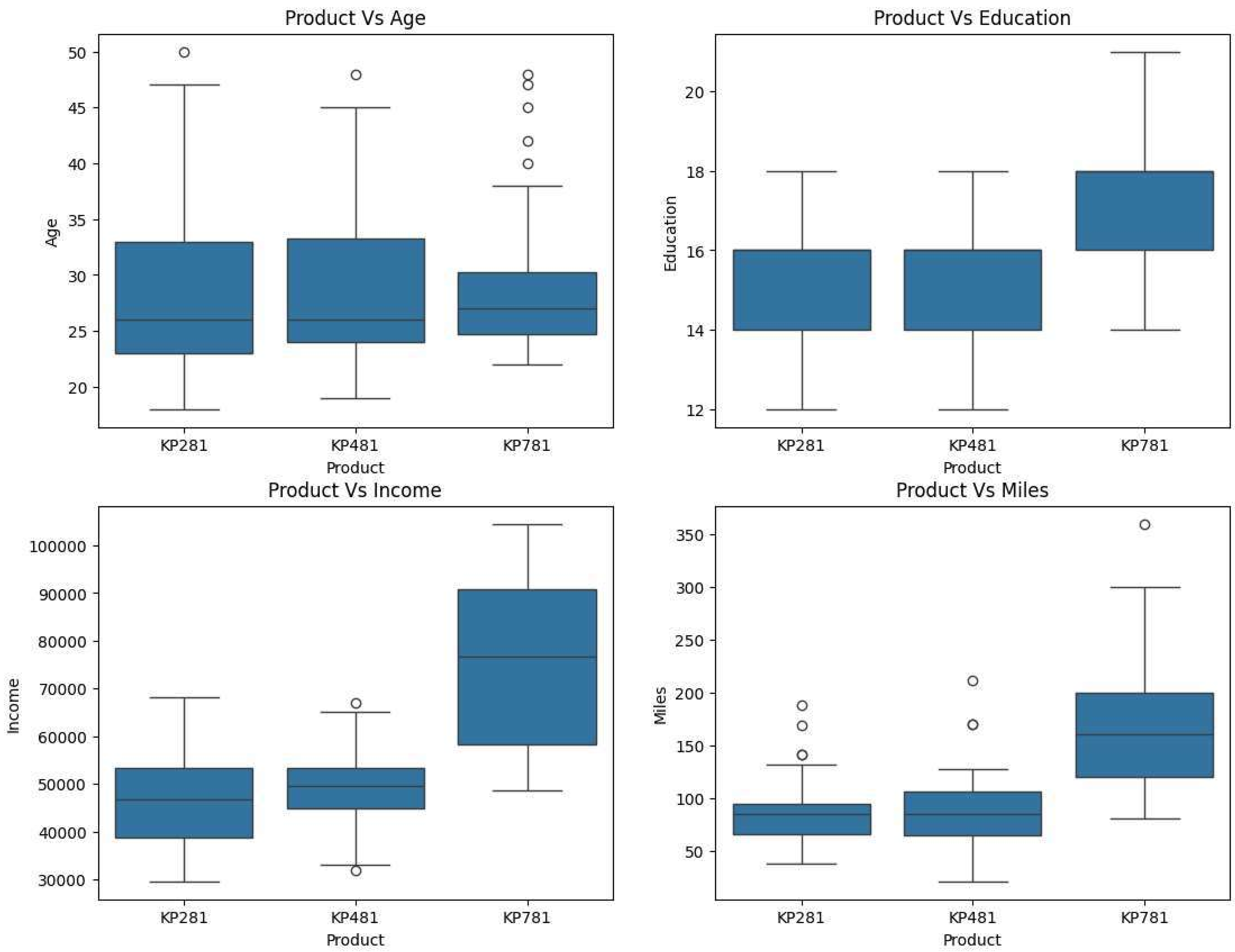
```
plt.figure(figsize=(13,10))
plt.subplot(2,2,1)
sns.boxplot(data=df,x='Product', y='Age')
plt.title('Product Vs Age')

plt.subplot(2,2,2)
```

```
sns.boxplot(data=df,x='Product', y='Education')
plt.title('Product Vs Education')

plt.subplot(2,2,3)
sns.boxplot(data=df,x='Product', y='Income')
plt.title('Product Vs Income')

plt.subplot(2,2,4)
sns.boxplot(data=df,x='Product', y='Miles')
plt.title('Product Vs Miles')
plt.show()
```



In [ ]:
```
df.head(100)
```

Out[ ]:

	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
...	...	...	...	...	...	...	...	...	...
95	KP481	24	Male	14	Single	3	4	48891	106
96	KP481	24	Female	16	Single	3	3	50028	106
97	KP481	25	Female	14	Partnered	2	3	45480	85
98	KP481	25	Female	14	Single	3	4	43206	127
99	KP481	25	Male	16	Partnered	2	2	52302	42

100 rows × 9 columns

🔍 Insights

The above analysis presented that treadmill KP781 is more demonding customer who possess **heigher eduaction, heigher income levels and intend to engage running activity 166 miles per week.**

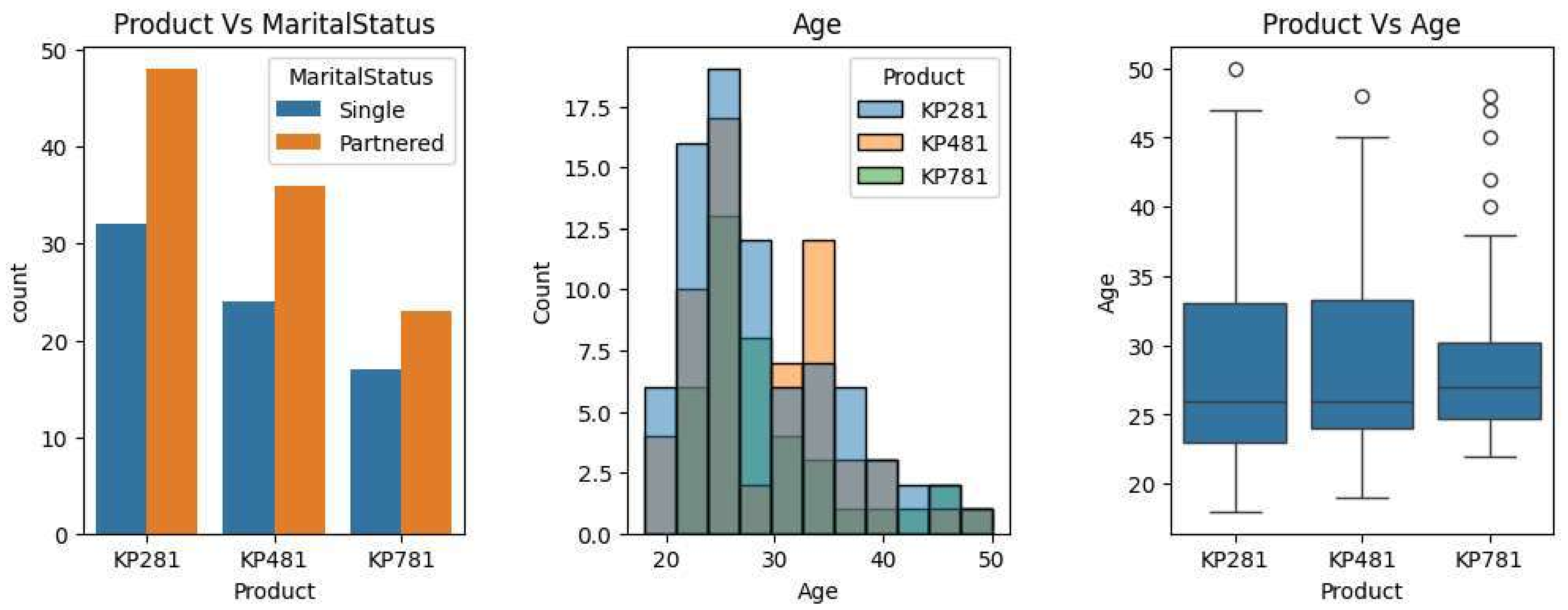
Question 3:-

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

In [10]:
```
plt.figure(figsize=(12,4))
plt.subplot(1,3,1)
plt.subplots_adjust(hspace=0.225, wspace=0.425)
sns.countplot(data=df, x='Product', hue='MaritalStatus')
plt.title('Product Vs MaritalStatus')

plt.subplot(1,3,2)
sns.histplot(data=df, x='Age', hue='Product')
plt.title('Age')

plt.subplot(1,3,3)
sns.boxplot(data=df, x='Product',y='Age')
plt.title('Product Vs Age')
plt.show()
```



🔍 Insights

MaritalStatus effect on Product The above analysis present that partnered customer highly preference for treadmill as compare to single.

Age effact on Product The above analysis clearly demonstrate uniform distribution of age groups across all the products.

Question 4 & 8:-

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

Computing Probability - Marginal, Conditional Probability

Probability of Product purchase w.r.t Age

In [11]:
```
pd.crosstab(index=df['Product'], columns=df['age_group'], margins=True, normalize=True).round(2)
```

Out[11]:

age_group	Young Adults	Adults	Middle Adults	Elders	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 probability of of a treadmill purchased by young adults **age_group (18-25) is 44%**

- **The conditional probability of Treadmill** given unique product of Young Adults is

1. Model KP281- 19%
2. Model KP481-16%
3. Model KP781-9%

- The probability of of a treadmill purchased by adults **age_group (26-35) is 41%**

- **The conditional probability of Treadmill** given unique product is

1. Model KP281- 18%
2. Model KP481-13%
3. Model KP781-9%

- The probability of of a treadmill purchased by adults **age_group (36-45) is 12%.**
- The probability of of a treadmill purchased by adults **age_group (46 abouve) is 3%**

Probability of Product purchase w.r.t Gender

In [12]:
```
pd.crosstab(index=df['Product'], columns=df['Gender'], margins=True, normalize=True).round(2)
```

Out[12]:

Gender Product	Female	Male	All
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 probability of treadmill pruchased by **Male customer is 58% which is heigher than Female 42%.**

- Model wise probability for the Male costomer and Female customes are-

Male customer probability model wise-

1. Model KP281- 22%
2. Model KP481-17%
3. Model KP781-18%

Female customer probability model wise-

1. Model KP281- 22%
2. Model KP481-16%
3. Model KP781-4%

Probability of Product purchase w.r.t Education

In [13]:
```
pd.crosstab(index=df['Product'], columns=df['education_group'], margins=True, normalize=True).round(2)
```

Out[13]:

education_group	Primary Education	Secondry Education	Higher Education	All
Product				
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 probability of treadmill purchased of **Higher Education group contributing 62%** follwed by **Secondrey Education group 36%** and **Primary Education group 2%**

Model wise probibility for Higher Education group is

1. Model KP281- 23%
2. Model KP481-18%
3. Model KP781-21%

Model wise probibility for Secondry Education group is

1. Model KP281- 21%
2. Model KP481-14%
3. Model KP781-1%

Model wise probibility for Primary Education group is

1. Model KP281- 1%
2. Model KP481-1%
3. Model KP781-00%

Probability of Product purchase w.r.t MaritalSatus

In [14]:
```
pd.crosstab(index=df['Product'], columns=df['MaritalStatus'], margins=True, normalize=True).round(2)
```

Out[14]:

MaritalStatus	Partnered	Single	All
Product			
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 probability of treadmill purchased by **partnered customer is heigher 59 %** as compare to **Single customer which is 41 %**

Model wise probibility for Partnered customer is

1. Model KP281- 27%
2. Model KP481-20%
3. Model KP781-13%

Model wise probibility for Single customer is

1. Model KP281- 18%
2. Model KP481-13%
3. Model KP781-9%

Probability of Product purchase w.r.t Income

In [15]:
```
pd.crosstab(index=df['Product'], columns=df['income_group'], margins=True, normalize=True).round(2)
```

Out[15]:

income_group Product	Low Income	Middle 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 probability of treadmill purchased by **Middle income is 59%** followed by **Low Income 18%, High income 13% and Very high income 11%**

Model wise probibility Middle income group is

1. Model KP281- 28%
2. Model KP481-24%
3. Model KP781-6%

Probability of Product purchase w.r.t Usage

In [16]:
```
pd.crosstab(index=df['Product'], columns=df['Usage'], margins=True, normalize=True).round(2)
```

Out[16]:

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 probability of treadmill purchased by customer **per week 3 usage is heigher 38% followed by per week 4 usage 29% per week 5 usage 9% per week 6 usage 9% per week 6 usage 4% and per week 7 usage 1%**

Model wise probibility of per week 3 usage is

1. Model KP281- 21%
2. Model KP481-17%
3. Model KP781-1%

Probability of Product purchase w.r.t Fitness

In [17]:
```
pd.crosstab(index=df['Product'], columns=df['Fitness'], margins=True, normalize=True).round(2)
```

Out[17]:

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 probability of treadmill purchased by customer **avgrage(3) fitness is 54% followed by avg(5) fitness17% ,avg(4) fitness 13%, avg(2) fitness 14% and avg(1) fitness 1%**

Model wise probibility of average(3) fitness

1. Model KP281- 30%
2. Model KP481-22%
3. Model KP781-2%

Probability of Product purchase w.r.t Miles

In [18]:
```
pd.crosstab(index=df['Product'], columns=df['miles_group'], margins=True, normalize=True).round(2)
```

Out[18]:

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 probability of treadmill purchased by customer **Moderate activity group is higher 54% followed by active lifestyle group 33% , light activity group 9% and fitness enthusiast 3%**

Model wise probibility of Moderate activity group are

1. Model KP281- 28%
2. Model KP481-22%
3. Model KP781-4%

Question 5:-

Check correlation among different factors using heat maps or pair plots.

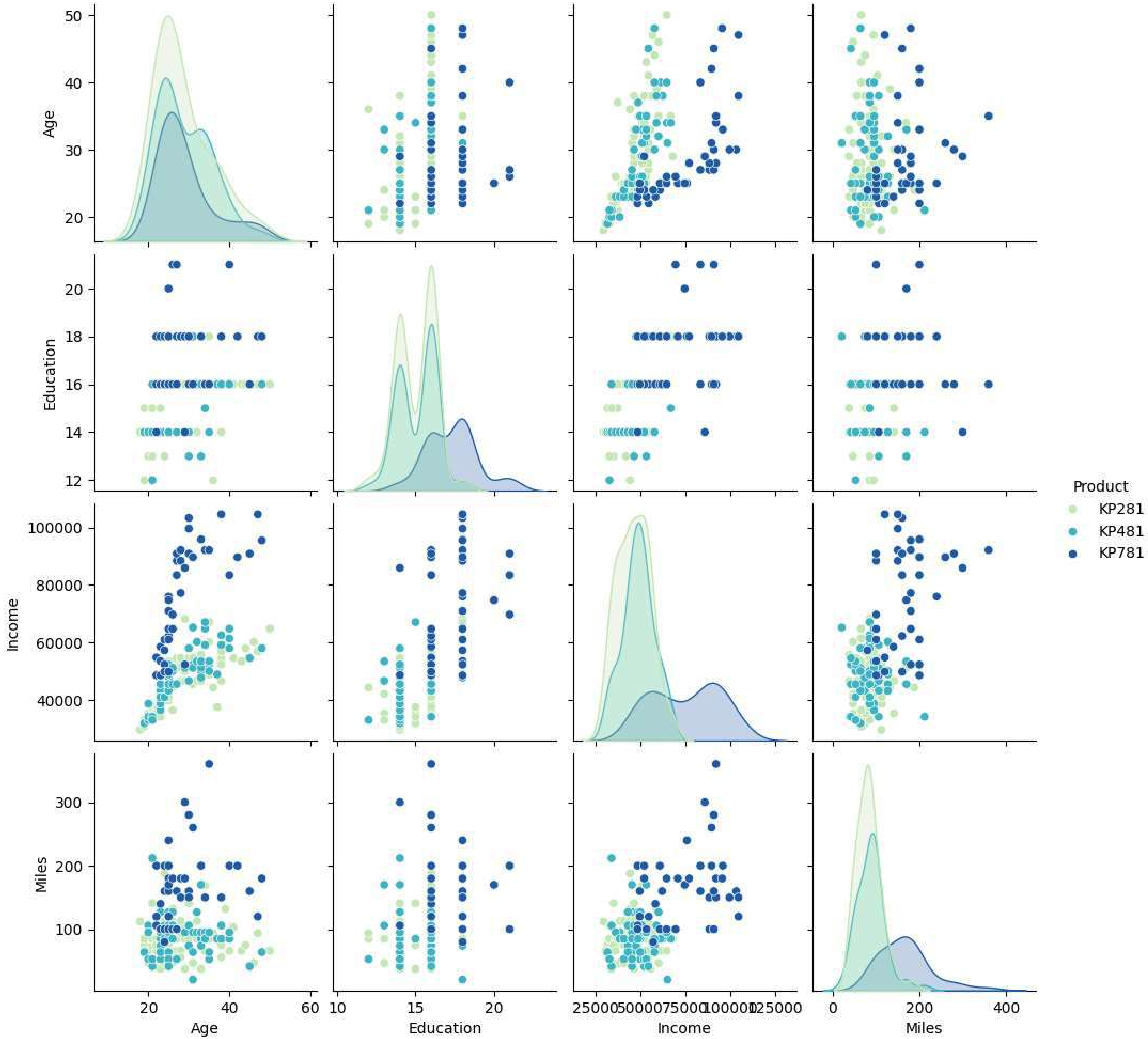
In [19]:
```
import copy
df_copy= copy.deepcopy(df)
df_copy.drop(columns=['age_group', 'income_group','education_group', 'miles_group','Fitness','Usage'], inplace=True)
df_copy.head(2)
```

Out[19]:

	Product	Age	Gender	Education	MaritalStatus	Income	Miles
0	KP281	18	Male	14	Single	29562	112
1	KP281	19	Male	15	Single	31836	75

In [20]:
```
#importing seaborn
import seaborn as sns
#pairplot
plt.figure(figsize=(15,5))
sns.pairplot(data=df_copy, hue='Product', palette='YlGnBu')
plt.show()
```

```
<Figure size 1500x500 with 0 Axes>
```



Heatmap

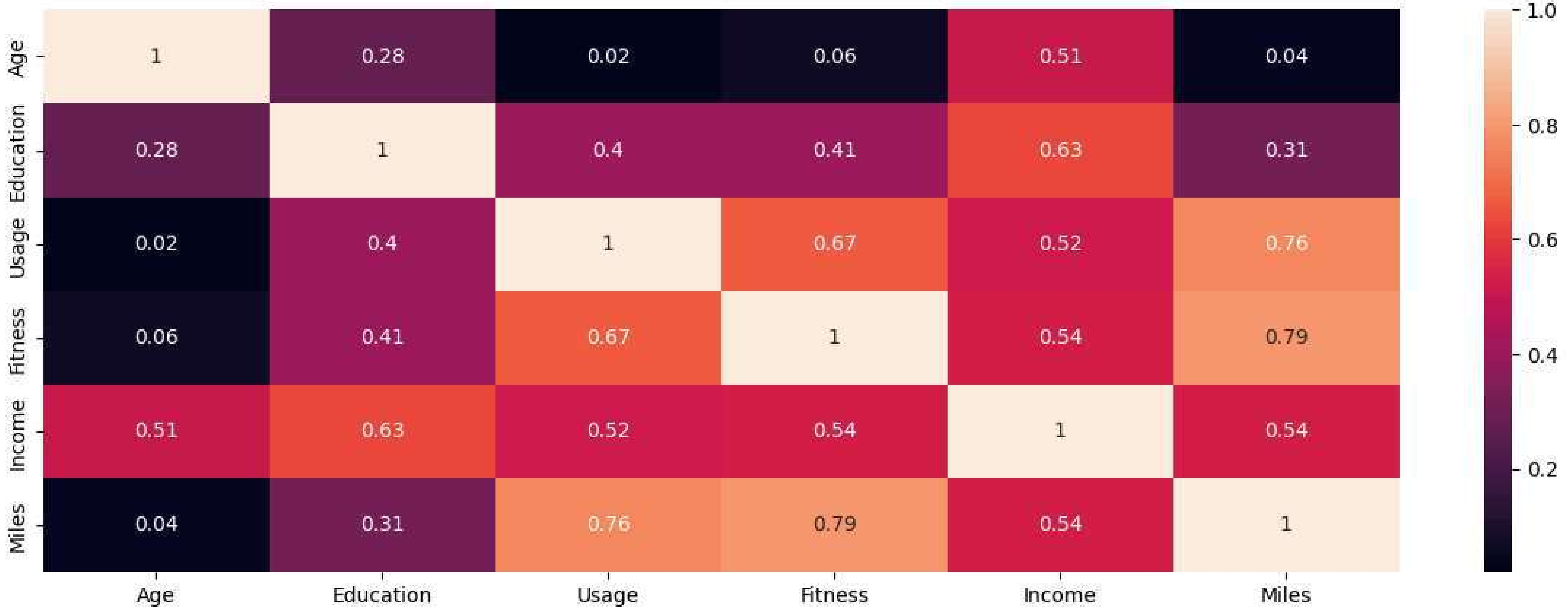
In [21]:
```
corr_df=df.corr()
corr_mat=np.round(corr_df,2)
corr_mat
```

```
<ipython-input-21-4bff3d5e6805>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this warning.
  corr_df=df.corr()
```

Out[21]:

	Age	Education	Usage	Fitness	Income	Miles
Age	1.00	0.28	0.02	0.06	0.51	0.04
Education	0.28	1.00	0.40	0.41	0.63	0.31
Usage	0.02	0.40	1.00	0.67	0.52	0.76
Fitness	0.06	0.41	0.67	1.00	0.54	0.79
Income	0.51	0.63	0.52	0.54	1.00	0.54
Miles	0.04	0.31	0.76	0.79	0.54	1.00

In [22]:
```
plt.figure(figsize=(15,5))
sns.heatmap(corr_mat, annot=True)
plt.show()
```



🔍 Insights

- As per **pairplot** analysis it shows that **income and age are highly correrated** and heatmap also showing storng correlation btween them.
- As per **heatmap** analysis **education and income are highly correlated** as education has significat correlation between fitness and usage.

- **Usage** is highly correlated with fitness and miles as more usage more fitness and miles

Question 6:-

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

In [23]:
```
pd.crosstab(index=df['Product'], columns=df['Gender'], margins=True, normalize=True).round(2)
```

Out[23]:

Gender Product	Female	Male	All
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 probability of product purchased by **male 58%** and **female 42%**

Condition Probability of purchased treadmill given that customer is male-

1. Model KP281-**22%**
2. Model KP481-**17%**
3. Model KP781-**18%**

Conditional Probability of purchased treadmill given that customer is female

1. Model KP281-**22%**
2. Model KP481-**16%**
3. Model KP781-**4%**

Question 7:-

Customer Profiling - Categorization of users.

Customer Profiling

Based on above analysis

- Probability of purchase of KP281=44%
- Probability of purchase of KP481=33%
- Probability of purchase of KP781=22%

Customer profiling for KP281 Treadmill

```
▪ Age of customer mainly between 18 to 35 and few from between 36 to 50.
▪ Education label for customer 13 and above
▪ Weekly usage per week 3 to 4 times
```

- Fitness scale 2 to 4
- Annual income range below USD 60000
- Weekly running miles 50 to 200 miles

Customer profiling for KP481 Treadmill

- Age of customer mainly between 18 to 35 and few from between 36 to 50.
- Education label for customer 13 and above
- Weekly usage per week 3 to 4 times
- Fitness scale 2 to 4
- Annual income range below USD 60000
- Weekly running miles 50 to 200 miles

Customer profiling for KP781 Treadmill

- Age of customer mainly between 18 to 35 and few from between 36 to 50.
- Education label for customer 13 and above
- Weekly usage per week 3 to 4 times
- Fitness scale 2 to 4
- Annual income range below USD 60000
- Weekly running miles 50 to 200 miles

Question 9:-

Some recommendations and actionable insights, based on the inferences.

8. Recommendations

Marketing Campaigns for KP781

- The KP781 model in terms of gender only 4 % female customers are purchaing it so there is scope in this model that we should keep more offarable price for this model and need to do more pramotions and need make some trail for customer to check it reliability.

Affordable Pricing and Payment Plans

- Given the target customer's age, education level, and income, it's important to offer the KP281 and KP481 Treadmill at an affordable price point. Additionally, 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. User-Friendly App Integration

User-Friendly App Integration

- Create a user-friendly app that syncs with the treadmill. This app could track users' weekly running mileage, provide real-time feedback on their progress, and offer personalized recommendations for workouts based on their fitness scale and goals.This can enhance the overall treadmill experience and keep users engaged.

In [25]:
```
df.to_csv('aerofit_treadmill.txt')
```