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 the 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. 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. 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 Product Portfolio:

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. What good looks like?

In [2]:

```
wget = "https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/125/original/aerofit_treadmill.csv?1639992749"
```

```
--2023-08-14 15:43:08--  http://=/ (http://=/)
Resolving = (=)... failed: Name or service not known.
wget: unable to resolve host address '='
--2023-08-14 15:43:08--  https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/125/original/aer
ofit_treadmill.csv?1639992749 (https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/125/origin
al/aerofit_treadmill.csv?1639992749)
Resolving d2beiqkhq929f0.cloudfront.net (d2beiqkhq929f0.cloudfront.net)... 2600:9000:245a:2200:11:1aff:4b00:
21, 2600:9000:245a:e00:11:1aff:4b00:21, 2600:9000:245a:4a00:11:1aff:4b00:21, ...
Connecting to d2beiqkhq929f0.cloudfront.net (d2beiqkhq929f0.cloudfront.net)|2600:9000:245a:2200:11:1aff:4b0
0:21|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7279 (7.1K) [text/plain]
Saving to: 'aerofit_treadmill.csv?1639992749'

aerofit_treadmill.c 100%[===================>]   7.11K  --.-KB/s    in 0s

2023-08-14 15:43:08 (1.05 GB/s) - 'aerofit_treadmill.csv?1639992749' saved [7279/7279]

FINISHED --2023-08-14 15:43:08--
Total wall clock time: 0.2s
Downloaded: 1 files, 7.1K in 0s (1.05 GB/s)
```

In [4]:

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

In [5]:

```
aerofit = pd.read_csv("aerofit_treadmill.csv?1639992749")
```

In [22]:

```
aerofit.shape
```

Out[22]:

```
(180, 9)
```

In [7]:

```
aerofit.describe()
```

Out[7]:

	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

No Null or missing values in any of the columns

In [8]:

```
aerofit.isna().sum()
```

Out[8]:

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

In [10]:

```
aerofit.dtypes
```

Out[10]:

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

Here we can see that, Product, Gender and Martial status are the 3 columns that have limited unique values and these columns can be put into categories

In [15]:

```
print("Product Categories ---> ", aerofit.Product.nunique())
print("Gender Categories ---> ",aerofit.Gender.nunique())
print("Marital Status Categories ---> ",aerofit.MaritalStatus.nunique())
```

```
Product Categories --->  3
Gender Categories --->  2
Marital Status Categories --->  2
```

In [16]:

```
categorical_columns = ["Product", "Gender", "MaritalStatus"]
aerofit[categorical_columns] = aerofit[categorical_columns].astype("category")
aerofit.dtypes
```

Out[16]:

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

Here we can see the total number of each type of treadmill purchased in the dataset

In [17]:

```
aerofit.Product.value_counts()
```

Out[17]:

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

total number of treadmills purchased by each gender in the dataset

In [27]:

```
aerofit.Gender.value_counts()
```

Out[27]:

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

In [ ]:

```
#### Total number of treadmills purchased by Single and Married people in the dataset
```

In [19]:

```
aerofit.MaritalStatus.value_counts()
```

Out[19]:

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

In [21]:

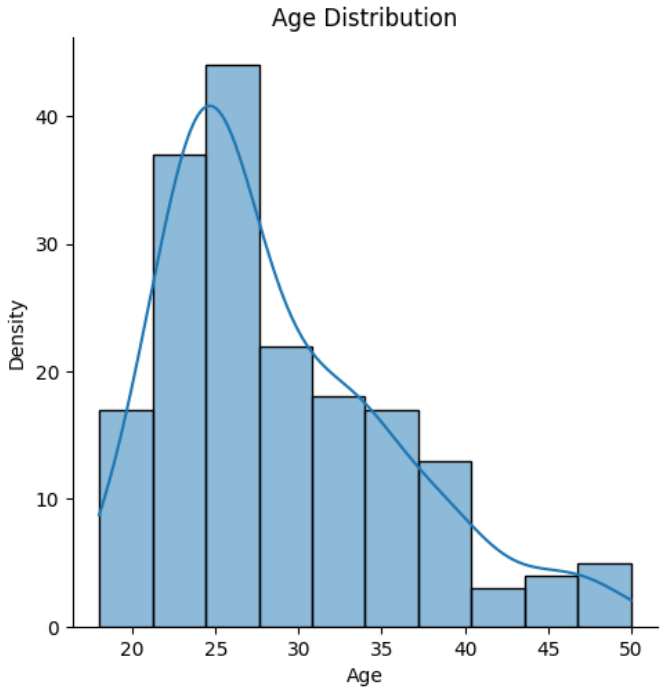
```
df = aerofit.copy()
```

(KDE)Kernel Density Estimate for Age of the people who bought aerofit treadmill

as you can see from the plot, maximum number of the people who bought the treadmill are of age 25 and less than 30

In [29]:

```
sns.displot(df["Age"], kde=True, bins=10)
plt.title("Age Distribution")
plt.xlabel("Age")
plt.ylabel("Density")
plt.show()
```

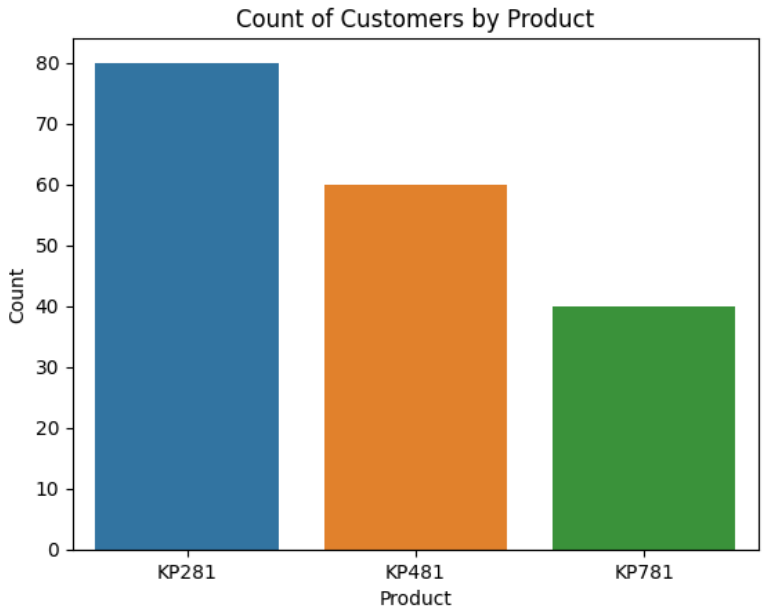


count for different treadmills

As we can see Aerofit's best selling treadmill is the "KP281" which is the cheapest and the most basic one

In [37]:

```
sns.countplot(x="Product", data=df)
plt.title("Count of Customers by Product")
plt.xlabel("Product")
plt.ylabel("Count")
plt.show()
```



Histogram for income,so we can see the income range of customers who buy the most treadmills from aerofit

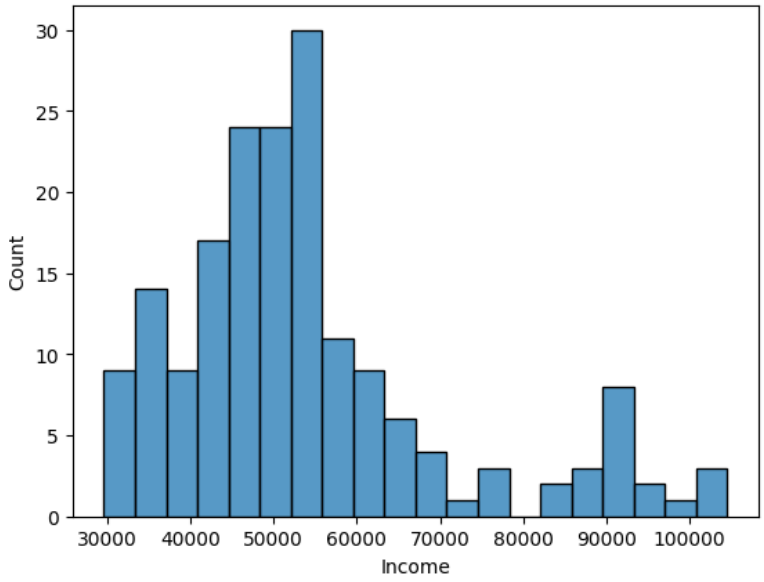
customers earning around 50,000 dollars a year are the ones that are buying treadmills the most

In [38]:

```
sns.histplot(df["Income"], bins = 20)
```

Out[38]:

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



In [39]:

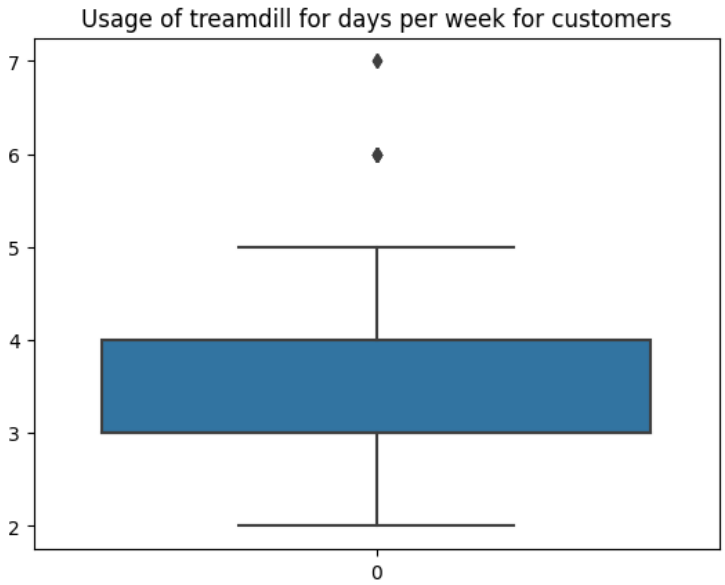
```
df.columns
```

Out[39]:

```
Index(['Product', 'Age', 'Gender', 'Education', 'MaritalStatus', 'Usage',
       'Fitness', 'Income', 'Miles'],
      dtype='object')
```

In [42]:

```
# we can create a box plot for the usage of treadmills
sns.boxplot(df["Usage"])
plt.title("Usage of treamdill for days per week for customers")
plt.show()
```



As you can see from the box plot above, there are little to no users who buy aerofit treadmill but don't use it for less than 3 days, Also there are some outliers who use it for 6 days and 7 days a week.

In [45]:

```
df.columns
```

Out[45]:

```
Index(['Product', 'Age', 'Gender', 'Education', 'MaritalStatus', 'Usage',
       'Fitness', 'Income', 'Miles'],
      dtype='object')
```

In [47]:

```
df.Education
```

Out[47]:

```
0      14
1      15
2      14
3      12
4      13
       ..
175    21
176    18
177    16
178    18
179    18
Name: Education, Length: 180, dtype: int64
```

Here we found a correlation that people with high fitness levels prefer to use KP781 Treadmill, as it is the most advanced and has alot of features, and advanced trainers want to challenge their fitness levels, hence they use the KP781 model

In [89]:

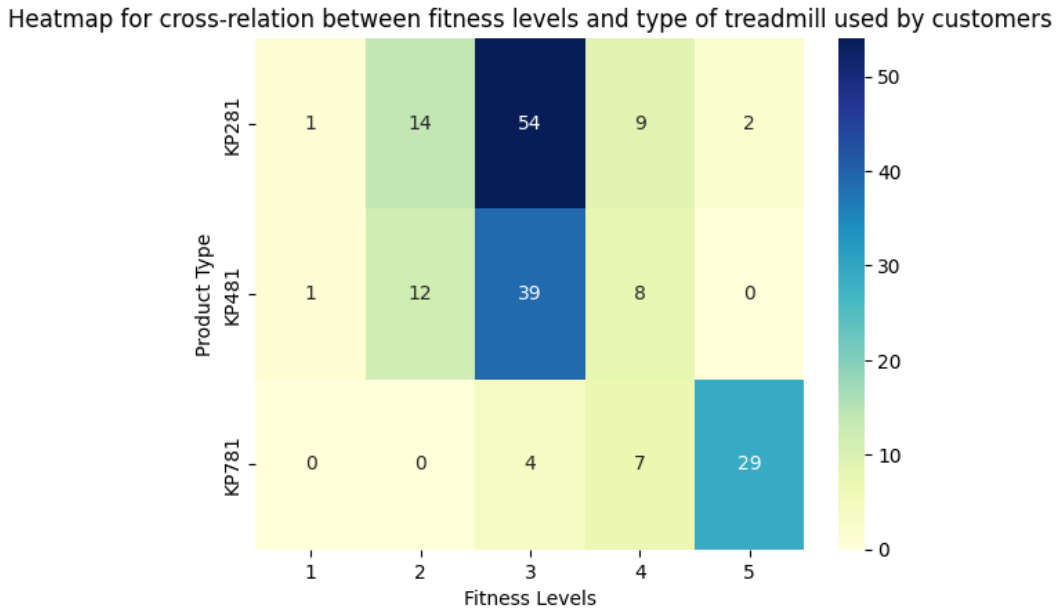
```
cross_tab = pd.crosstab(df["Product"], df["Fitness"])
cross_tab
```

Out[89]:

Fitness	1	2	3	4	5
Product					
KP281	1	14	54	9	2
KP481	1	12	39	8	0
KP781	0	0	4	7	29

In [57]:

```
sns.heatmap(cross_tab, annot=True, cmap="YlGnBu")
plt.title("Heatmap for cross-relation between fitness levels and type of treadmill used by customers")
plt.xlabel("Fitness Levels")
plt.ylabel("Product Type")
plt.show()
```



fitness goals influence on the choice of product

As we can see from the heatmap above customers with decent fitness levels which is 3, are more inclined to buy the budget friendly kp281.Whereas, the customers

as we can see from the cross-map below that the customers that use treadmill for more than 100 miles a week must be active and they prefer to buy the kp781 also customers using it less than 100 miles/week prefer to buy KP281

In [102]:

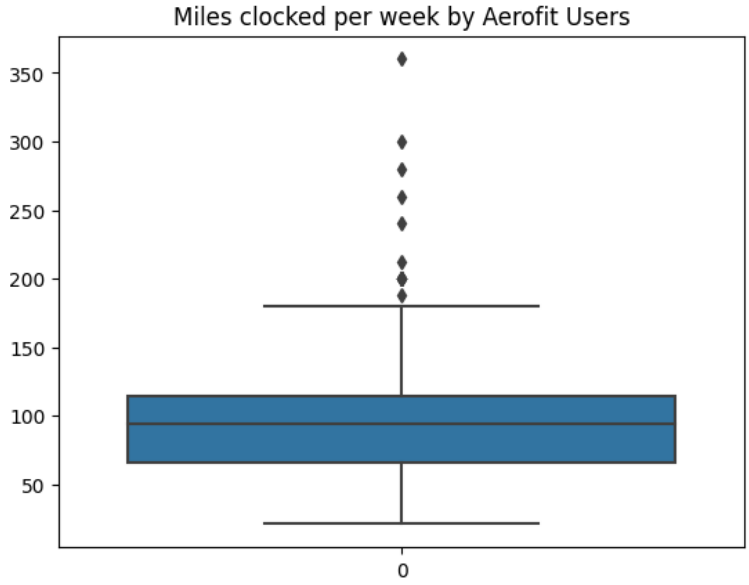
```
# if number of miles clocked on treadmill has any influence on the choice of product

miles_product = pd.crosstab(df["Miles"],df["Product"])
miles_product
```

Out[102]:

Product	KP281	KP481	KP781
Miles			
21	0	1	0
38	3	0	0
42	0	4	0
47	9	0	0
53	0	7	0
56	6	0	0
64	0	6	0
66	10	0	0
74	0	3	0
75	10	0	0
80	0	0	1
85	16	11	0
94	8	0	0
95	0	12	0
100	0	0	7
103	3	0	0
106	0	8	1
112	1	0	0
113	8	0	0
120	0	0	3
127	0	5	0
132	2	0	0
140	0	0	1
141	2	0	0
150	0	0	4
160	0	0	5
169	1	0	0
170	0	2	1
180	0	0	6
188	1	0	0
200	0	0	6
212	0	1	0
240	0	0	1
260	0	0	1
280	0	0	1
300	0	0	1
360	0	0	1

In [99]:

```
# lets use a box-plot to check how many professional atheletes use the treadmills for long distance running
sns.boxplot(df["Miles"])
plt.title("Miles clocked per week by Aerofit Users")
plt.show()
```



as we can see the outliers in the graph, the median and mean(avg) for miles is around (94-100Miles/week), and the customers going more than 200 miles are outliers who can be professional athletes, long-distance runners, marathon runners. as the maximum miles someone has clocked on the treadmill is 360 miles in a week.

In [100]:

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

Out[100]:

```
count    180.000000
mean     103.194444
std       51.863605
min       21.000000
25%       66.000000
50%       94.000000
75%      114.750000
max      360.000000
Name: Miles, dtype: float64
```

In [65]:

```
# does marital status mean the customers are using the treadmill more
family_usage = pd.crosstab(df["MaritalStatus"], df["Usage"])
family_usage
```

Out[65]:

Usage	2	3	4	5	6	7
MaritalStatus						
Partnered	22	40	29	9	5	2
Single	11	29	23	8	2	0

As we can see from the cross plot above, usage of treadmills is more in general, in Homes where customers have partners

What is the probability of a male customer buying a KP781 treadmill?

In [80]:

```
kp781 = df[df["Product"] == "KP781"]
```

In [84]:

```
kp781_stats = kp781.Gender.describe()
kp781_stats
```

Out[84]:

```
count       40
unique       2
top       Male
freq        33
Name: Gender, dtype: object
```

the probability of a male buying the kp781 treadmill is 82.5 % more than a female from the data that we have been provided

In [85]:

```
male = kp781_stats["freq"]
total = kp781_stats["count"]
print(male/total*100, "%")
```

```
82.5 %
```

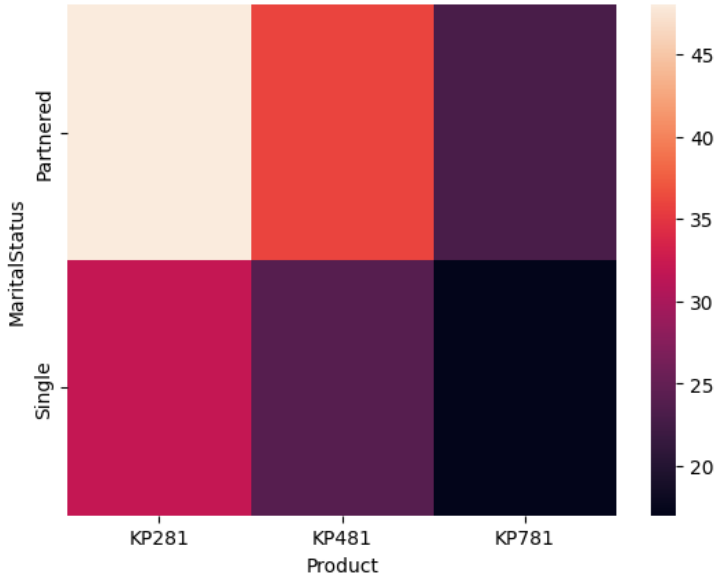
we can see that married customers prefer to buy the kp281 as it fits their budget

In [105]:

```
sns.heatmap(marital_product)
```

Out[105]:

```
<AxesSubplot:xlabel='Product', ylabel='MaritalStatus'>
```



In [108]:

```
# lets see if income influences the choice of product
df["Income"].describe()
```

Out[108]:

```
count       180.000000
mean      53719.577778
std       16506.684226
min       29562.000000
25%       44058.750000
50%       50596.500000
75%       58668.000000
max      104581.000000
Name: Income, dtype: float64
```

here we can see usually married customers dont have abnormal fitness levels, so they dont need the kp781, and neither do they clock more than 100 miles a week on treadmill, as we can check the 50% (median) of the usage, fitness and miles categories

In [111]:

```
df[df["MaritalStatus"] == "Partnered"].describe()
```

Out[111]:

	Age	Education	Usage	Fitness	Income	Miles
count	107.00000	107.000000	107.000000	107.000000	107.00000	107.000000
mean	29.88785	15.663551	3.448598	3.271028	55763.00000	104.289720
std	7.25067	1.523055	1.159324	0.967101	17499.65758	59.230762
min	19.00000	12.000000	2.000000	1.000000	30699.00000	38.000000
25%	25.00000	14.000000	3.000000	3.000000	45480.00000	66.000000
50%	28.00000	16.000000	3.000000	3.000000	52302.00000	85.000000
75%	34.50000	16.000000	4.000000	4.000000	61202.00000	120.000000
max	50.00000	21.000000	7.000000	5.000000	104581.00000	360.000000

provided that the customers use kp781, let's see what kind of customers use this treadmill

we can see from the stats below, that the average Miles ran/walked by KP781 users are 160 compared to less than 100 miles for kp281 users

In [113]:

```
kp_781 = df[df["Product"] == "KP781"]
kp_781.describe()
```

Out[113]:

	Age	Education	Usage	Fitness	Income	Miles
count	40.000000	40.000000	40.000000	40.000000	40.00000	40.000000
mean	29.100000	17.325000	4.775000	4.625000	75441.57500	166.900000
std	6.971738	1.639066	0.946993	0.667467	18505.83672	60.066544
min	22.000000	14.000000	3.000000	3.000000	48556.00000	80.000000
25%	24.750000	16.000000	4.000000	4.000000	58204.75000	120.000000
50%	27.000000	18.000000	5.000000	5.000000	76568.50000	160.000000
75%	30.250000	18.000000	5.000000	5.000000	90886.00000	200.000000
max	48.000000	21.000000	7.000000	5.000000	104581.00000	360.000000

In [114]:

```
# now lets see what kind of users use kp481
kp481 = df[df["Product"] == "KP481"]
kp481.describe()
```

Out[114]:

	Age	Education	Usage	Fitness	Income	Miles
count	60.000000	60.000000	60.000000	60.00000	60.000000	60.000000
mean	28.900000	15.116667	3.066667	2.90000	48973.650000	87.933333
std	6.645248	1.222552	0.799717	0.62977	8653.989388	33.263135
min	19.000000	12.000000	2.000000	1.00000	31836.000000	21.000000
25%	24.000000	14.000000	3.000000	3.00000	44911.500000	64.000000
50%	26.000000	16.000000	3.000000	3.00000	49459.500000	85.000000
75%	33.250000	16.000000	3.250000	3.00000	53439.000000	106.000000
max	48.000000	18.000000	5.000000	4.00000	67083.000000	212.000000

In [116]:

```
kp281 = df[df["Product"] == "KP281"]
kp281.describe()
```

Out[116]:

	Age	Education	Usage	Fitness	Income	Miles
count	80.000000	80.000000	80.000000	80.00000	80.00000	80.000000
mean	28.550000	15.037500	3.087500	2.96250	46418.02500	82.787500
std	7.221452	1.216383	0.782624	0.66454	9075.78319	28.874102
min	18.000000	12.000000	2.000000	1.00000	29562.00000	38.000000
25%	23.000000	14.000000	3.000000	3.00000	38658.00000	66.000000
50%	26.000000	16.000000	3.000000	3.00000	46617.00000	85.000000
75%	33.000000	16.000000	4.000000	3.00000	53439.00000	94.000000
max	50.000000	18.000000	5.000000	5.00000	68220.00000	188.000000

as we can see from the stats above, for both kp281 and kp481, we can see that the mean, median income of kp481 users is a little bit more than kp281 users

Actionable Insights & Recommendations

1. Married Customers tend to buy KP281 as they don't use it much.
2. There is slight Income difference between KP281 and KP481 Customers.
3. Customers with Fitness Level of 4 tend to buy KP281 and KP481 more, which is ironic as they need the KP781 to take their fitness to next level.
4. Customers who Miles ran/walked more than 100 miles/week tend to fall into good fitness levels and can be recommended the KP781 or KP481.
5. There are little to no users who buy aerofit treadmill but don't use it for less than 3 days, which shows those customers like to run on the treadmill.