Business Case

AEROFIT - DESCRIPTIVE STATISTICS & PROBABILITY

Akanksha Trivedi

Scaler Academy

Table of Contents

Introduction

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.

Purpose

As the market research team at Aerofit, we want 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.

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

Dataset characteristics

Importing data modules

```
import numpy as np # linear algebra
import pandas as pd # data processing,
import matplotlib.pyplot as plt
import seaborn as sns
```

Full data

```
>>> data_path = "aerofit_treadmill.csv"
>>> df = pd.read_csv(data_path)
>>> print(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
```

We have total 180 rows of data

Data columns

```
>>> 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
```

Total numbers of Columns is 9

Missing value

```
>>> print(df.isnull().any())
Product           False
Age               False
Gender            False
Education         False
MaritalStatus     False
Usage             False
Fitness           False
Income            False
Miles             False
dtype: bool
```

There is no such column that has missing data

Median/Mean/SD

```
>>> 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
```

Observation from characteristics

1. We have a total of 180 samples of data.
2. Each sample has 9 variables.
3. KP281 is a top selling product.
4. Age seems to be a very important factor, 25% to 75% of users are in 24-33 age groups. Mean age is 28.
5. Male users are more than female users.
6. Partnered users are more than single users.
7. Minimum number of time machine used is 2, 25% to 75% users using machine 3-4 times in a week
8. People rated themselves as fitness rating 3-4 contributes 25%-75% users.
9. Income/miles has very high SD value, we will have more insights once we outliners.

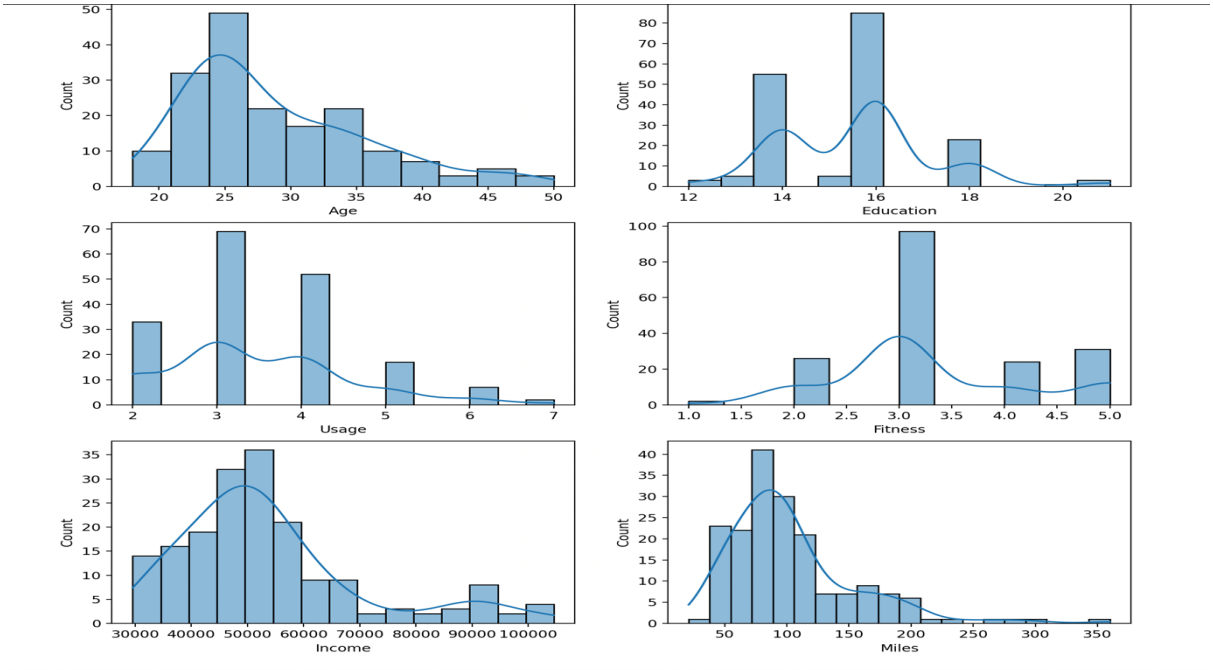
Univariate and Outliners

We have a total of 9 data columns Age , Gender, Usages , fitness , income and Miles. This section will see data distribution for these quantitative attributes.

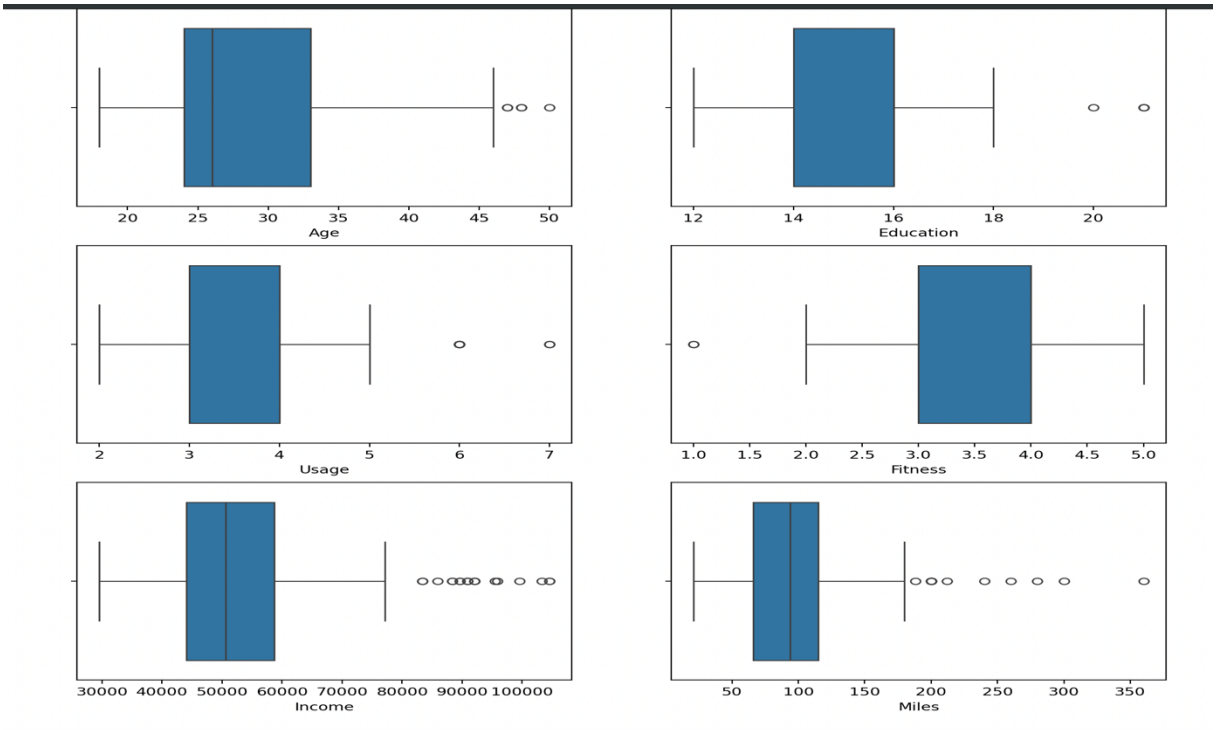
Queries

```
#fig, axis = plt.subplots(nrows=3, ncols=2, figsize=(12, 10))
#fig.subplots_adjust(top=1.0)

#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()

#fig, axis = plt.subplots(nrows=3, ncols=2, figsize=(12, 10))
#fig.subplots_adjust(top=1.0)
#sns.boxplot(data=df, x="Age", orient='h', ax=axis[0,0])
#sns.boxplot(data=df, x="Education", orient='h', ax=axis[0,1])
#sns.boxplot(data=df, x="Usage", orient='h', ax=axis[1,0])
#sns.boxplot(data=df, x="Fitness", orient='h', ax=axis[1,1])
#sns.boxplot(data=df, x="Income", orient='h', ax=axis[2,0])
#sns.boxplot(data=df, x="Miles", orient='h', ax=axis[2,1])
#plt.show()
```

Count

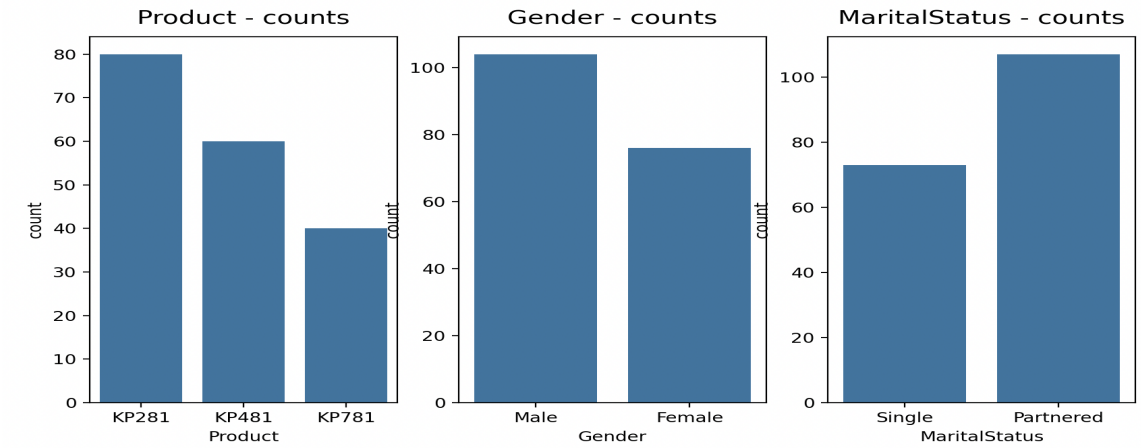


Outliners



Product, Gender and Marital Status are main candidate for univariate analysis

```
#fig, axs = plt.subplots(nrows=1, ncols=3, figsize=(10,5))
#sns.countplot(data=df, x='Product', ax=axs[0])
#sns.countplot(data=df, x='Gender', ax=axs[1])
#sns.countplot(data=df, x='MaritalStatus', ax=axs[2])
#axs[0].set_title("Product – counts", pad=10, fontsize=14)
#axs[1].set_title("Gender – counts", pad=10, fontsize=14)
#axs[2].set_title("MaritalStatus – counts", pad=10,
#fontsize=14)
#plt.show()
```



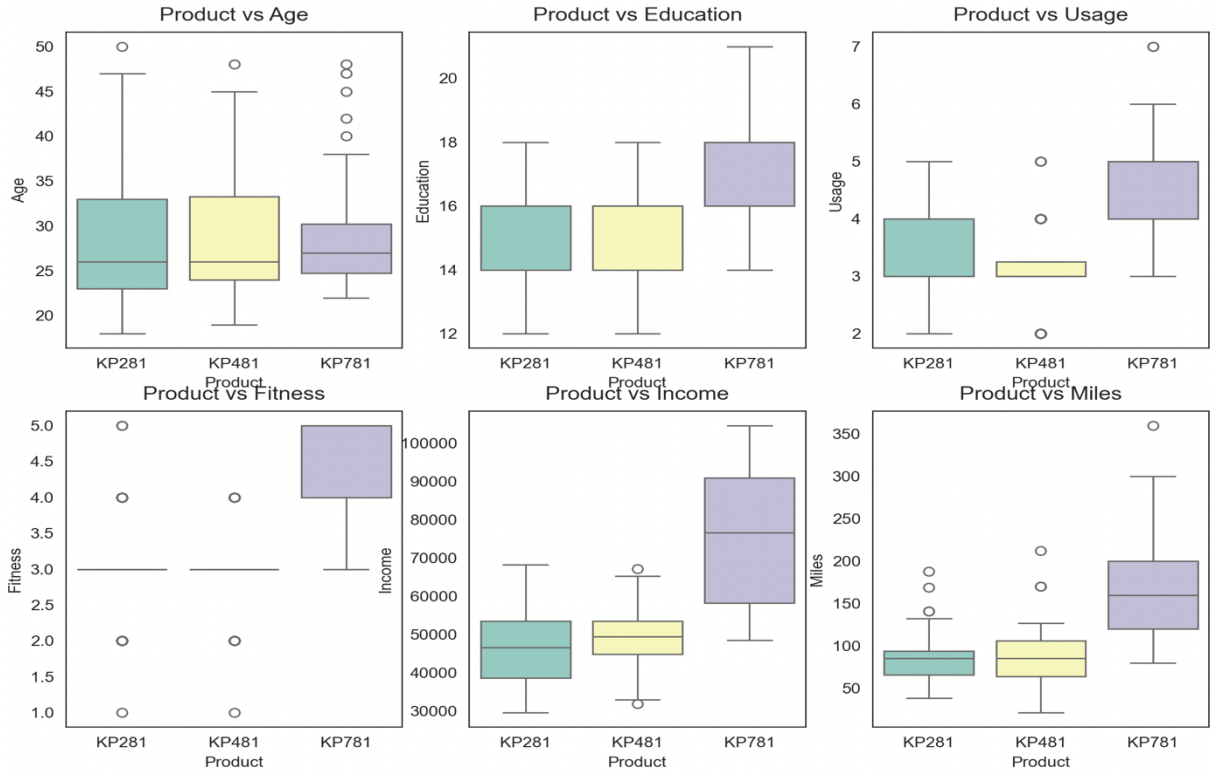
Observation:

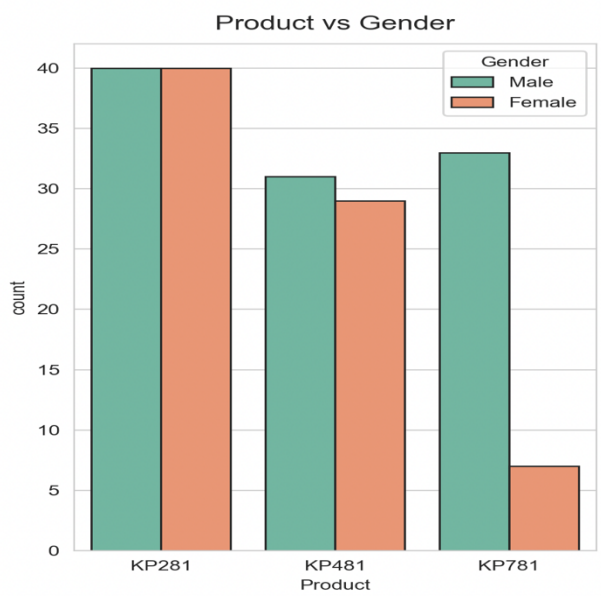
1. Around 25 is the most common age group buying the product.
2. 16 years of education are common education groups.
3. Product is used 3 to 4 times most of the users.
4. People rated themselves 3 are the most common user groups.
5. Income and Miles have very high numbers of outliers.
6. KP281 is top selling and KP781 is lowest, Percentage is 44.44, 33.33 and 22.22.

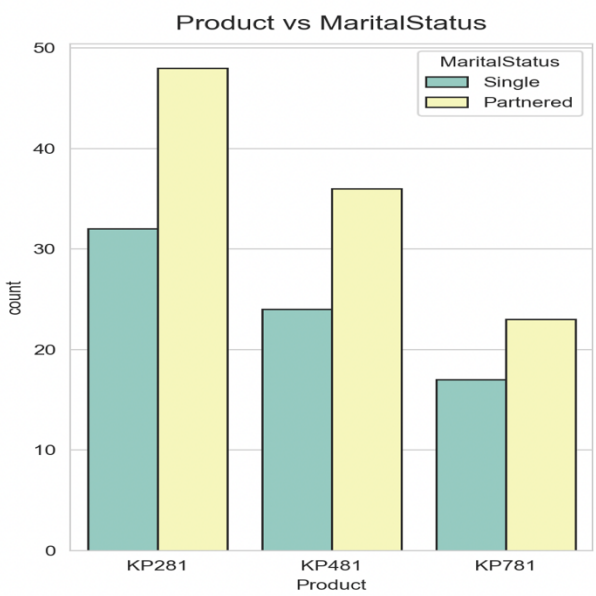
Bivariate Analysis

```
#sns.set_style(style='whitegrid')
#fig, axs = plt.subplots(nrows=1, ncols=2, figsize=(10, 6.5))
#sns.countplot(data=df, x='Product', hue='Gender', edgecolor="0.15", palette='Set2', ax=axs[0])
#sns.countplot(data=df, x='Product', hue='MaritalStatus', edgecolor="0.15", palette='Set3', ax=axs[1])
#axs[0].set_title("Product vs Gender", pad=10, fontsize=14)
#axs[1].set_title("Product vs MaritalStatus", pad=10, fontsize=14)
#plt.show()


#attrs = ['Age', 'Education', 'Usage', 'Fitness', 'Income', 'Miles']
#sns.set_style("white")
#fig, axs = plt.subplots(nrows=2, ncols=3, figsize=(12, 8))
#fig.subplots_adjust(top=0.95)
#count = 0
#for i in range(2):
#    for j in range(3):
#        sns.boxplot(data=df, x='Product', y=attrs[count], ax=axs[i,j], palette='Set3')
#        axs[i,j].set_title(f"Product vs {attrs[count]}", pad=8, fontsize=13)
#        count += 1
#plt.show()
```

Let's add each data for different product.







Observations:

1. For KP281, Gender does not seem to be effective, while more males are choosing KP781.
2. Marital status is not affecting product selection.
3. User group 25-35 is buying all 3 kinds of product but KP781 is purchased by higher age groups.
4. KP281 and KP481 have the same education group but KP781 is chosen by the higher education group.
5. KP281 and KP481 used by users who used the product 3-4 times , KP481 purchased by users who used machines less. Users who are more consistent likely purchased KP781.
6. Customers who considered themselves fit purchased KP781.
7. KP281 is purchased by customers having income 40k-50k , KP481 is purchased by the upper end of the same previous group. KP781 is used by customers having higher income.
8. Same as fitness level, Miles covered by customers having product KP781 is much more than the rest of two.

Marginal and Conditional probability

```
>>> df['Product'].value_counts(normalize=True)
Product
KP281    0.444444
KP481    0.333333
KP781    0.222222
Name: proportion, dtype: float64
```

```
def p_prod_given_gender(gender, print_marginal=False):
    if gender is not "Female" and gender is not "Male":
        return "Invalid gender value."
    df1 = pd.crosstab(index=df['Gender'], columns=[df['Product']])
    p_781 = df1['KP781'][gender] / df1.loc[gender].sum()
    p_481 = df1['KP481'][gender] / df1.loc[gender].sum()
    p_281 = df1['KP281'][gender] / df1.loc[gender].sum()
    if print_marginal:
        print(f"P(Male): {df1.loc['Male'].sum()/len(df):.2f}")
        print(f"P(Female): {df1.loc['Female'].sum()/len(df):.2f}\n")
        print(f"P(KP781/{gender}): {p_781:.2f}")
        print(f"P(KP481/{gender}): {p_481:.2f}")
        print(f"P(KP281/{gender}): {p_281:.2f}\n")


p_prod_given_gender('Male', True)
p_prod_given_gender('Female')
```

```
P(Male): 0.58
P(Female): 0.42

P(KP781/Male): 0.32
P(KP481/Male): 0.30
P(KP281/Male): 0.38
```

Customer Profiling and recommendations

Let's go each product's customer profile one by one

KP281

Used by age group 25-35
Being preferred by both male/female or Partnered/Single
Income group is 40k-50k

Recommendations

- This is a top selling product but needs to keep the price in check compared to other players in the market as this is a starting range and we are in a cost sensitive area here.

KP481

Used by age group 25-35
Being preferred by both male/female or Partnered/Single
Income group is 45k-55k
Same user group is KP481 except preferred by higher income groups

Recommendations

- Seems like feature wise this product does not offer much as compare to KP281 vs price difference, need to add more features here to make its own user group . Right now it is just a subpart of KP281 product's higher income group.
- Need more advertisement and push from sells that how it will offer better experience than KP281

KP781

Used by age group 25-35 but higher deviations on high age groups.
Has more male buyers
Preferred by people who are more health conscious.
Higher income group.
Higher education group.

Recommendations

- This is high end product used by fitness conscious and higher income group people
- There is user group with higher income and low fitness level which could be potential buyer