Background:

The data is for customers of the treadmill product(s) of a retail store called Cardio Good Fitness. It contains the following variables

Objective:

Preliminary Data Analysis. Explore the dataset and practice extracting basic observations about the data. The idea is for you to get comfortable working in Python.

1.Come up with a customer profile (characteristics of a customer) of the different products 2.Perform uni-variate and multi-variate analyses 3.Generate a set of insights and recommendations that will help the company in targeting new customers

Data:

1. Product - the model no. of the treadmill
2. Age - in no of years, of the customer
3. Gender - of the customer
4. Education - in no. of years, of the customer
5. Marital Status - of the customer
6. Usage - Avg. # times the customer wants to use the treadmill every week
7. Fitness - Self rated fitness score of the customer (5 - very fit, 1 - very unfit)
8. Income - of the customer
9. Miles- expected to run

Import the necessary libraries - pandas, numpy, seaborn, matplotlib.pyplot

In [2]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [3]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Read in the dataset

In [4]:
```
data = pd.read_csv('CardioGoodFitness.csv')
```

In [5]:
```
data.head() #display the first five rows
```

Out[5]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
0	TM195	18	Male	14	Single	3	4	29562	112
1	TM195	19	Male	15	Single	2	3	31836	75
2	TM195	19	Female	14	Partnered	4	3	30699	66
3	TM195	19	Male	12	Single	3	3	32973	85
4	TM195	20	Male	13	Partnered	4	2	35247	47

In [6]:
```
data.info() #checking the data types of each column
```

```
<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
```

In [7]:

```
data.shape  #Checking the shape of the data. We have 180 rows and 9 columns
```

Out[7]:

```
(180, 9)
```

In [8]:

```
data.isnull().sum() #Checking for the number of null values - no null values found
```

Out[8]:

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

Converting Objects into Categorical Variables

In [9]:

```
data['Product'] = data['Product'].astype('category')
data['Gender'] = data['Gender'].astype('category')
data['MaritalStatus'] = data['MaritalStatus'].astype('category') #converting each object values into categorical
```

In [10]:

```
data.info() #checking the data types again
```

```
<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    category
 1   Age            180 non-null    int64   
 2   Gender         180 non-null    category
 3   Education      180 non-null    int64   
 4   MaritalStatus  180 non-null    category
 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: category(3), int64(6)
memory usage: 9.4 KB
```

Viewing all statisitics of the data

In [11]:

```
statistic = data.describe(include = 'all')
print(statistic) #checking all the statistics of the data
```

```
       Product         Age Gender   Education MaritalStatus       Usage  \
count      180  180.000000    180  180.000000           180  180.000000   
unique       3         NaN      2         NaN             2         NaN   
top      TM195         NaN   Male         NaN     Partnered         NaN   
freq        80         NaN    104         NaN           107         NaN   
mean       NaN   28.788889    NaN   15.572222           NaN    3.455556   
std        NaN    6.943498    NaN    1.617055           NaN    1.084797   
min        NaN   18.000000    NaN   12.000000           NaN    2.000000   
25%        NaN   24.000000    NaN   14.000000           NaN    3.000000   
50%        NaN   26.000000    NaN   16.000000           NaN    3.000000   
75%        NaN   33.000000    NaN   16.000000           NaN    4.000000   
max        NaN   50.000000    NaN   21.000000           NaN    7.000000   

           Fitness        Income       Miles  
count   180.000000    180.000000  180.000000  
unique         NaN           NaN         NaN  
top            NaN           NaN         NaN  
freq           NaN           NaN         NaN  
mean      3.311111  53719.577778  103.194444  
std       0.958869  16506.684226   51.863605  
min       1.000000  29562.000000   21.000000  
25%       3.000000  44058.750000   66.000000  
```

```
50%       3.000000   50596.500000   94.000000
75%       4.000000   58668.000000  114.750000
max       5.000000  104581.000000  360.000000
```

Observations:

1. There are 3 unique products values(items)
2. Usage is categorized form values of 2 to 7, the 25% percentile is same as the mean.
3. Fitness is categorized from values of 1 to 5.
4. Every individual has atleast compelted 21 miles.
5. 104 data points for Male, 76 data points for Female
6. Unique values for Fitness ranges from 1, to 5 with a total of 5 unique values.
7. Unique values for Education ranges from 12 to 21 with a total of 8 unique values.

EDA

In [ ]:
```
def histogram_boxplot(feature):
    """ Boxplot and histogram combined
    feature: 1-d feature array
    """
    figure, (ax_box2, ax_hist2, ax_hist3) = plt.subplots(
        nrows = 1, ncols=3,# Number of rows of the subplot grid= 2
        figsize  = (20,5)) # creating the 2 subplots
    figure.tight_layout(pad = 7)
    sns.boxplot(x = feature,ax=ax_box2, color = '#4B8BBE', orient = 'v') # boxplot will be created
    sns.distplot(feature, kde=True, ax=ax_hist2, color = '#a9a38f') # For histogram
    sns.distplot(feature, kde= True, ax=ax_hist3, hist = False) #Making an outline of the histogram
    ax_hist2.axvline(np.mean(feature), color='r', linestyle='--') # Add mean to the histogram
    ax_hist2.axvline(np.median(feature), color='black', linestyle='-') # Add median to the histogram
    ax_hist3.axvline(np.mean(feature), color = 'black', linestyle = '--') #Adding mean to second histogram
    ax_hist3.axvline(np.median(feature), color='black', linestyle='-') #Adding median to second histogram
```

Observations on Age

In [ ]:
```
histogram_boxplot(data.Age) #boxplot, histogram with and wihtout bars, for age.
```

Observations:

1. Age has 3 outliers
2. The peak in the data seems to occur at 25.
3. The data has slight positive skeweness/Right skeweness.

Observations on Education

In [ ]:
```
histogram_boxplot(data.Education)
```

Observations:

1. Education has 2 outliers
2. The peak in the data seems to occur at 16 years of education.
3. The data suggest most individuals have education between 13 to 18 years.

Observations on Usage

In [ ]:
```
histogram_boxplot(data.Usage)
```

Observations:

1. Usage has 2 outliers
2. The peak in the data seems to occur at 3.
3. Most of the individuals lies between 2 to 4 level fitness.

Observations on Income

In [ ]:
```
histogram_boxplot(data.Income)
```

Observations:

Observations:

1. Income seems to have more than 10 outliers.
2. The peak in the data seems to occur at $45480.
3. The data has very slight positive skeweness/Right skeweness.

Observations on Fitness

In [ ]:
```
histogram_boxplot(data.Fitness)
```

Observations:

1. Fitness has 1 outlier.
2. The peak in the data seems to occur at level 3 fitness.

Observations on Miles

In [ ]:
```
histogram_boxplot(data.Miles)
```

Observations:

1. Miles has 8 outliers
2. The peak in the data seems to occur at 85 miles.
3. The data has slight positive skeweness/Right skeweness.
4. Most people tend to average around 103 miles.

Categorical Variables

In [ ]:
```
def bar_perc(plot, feature):
    '''
    plot
    feature: 1-d categorical feature array
    '''
    total = len(feature) # length of the column
    for p in ax.patches:
        percentage = '{:.1f}%'.format(100 * p.get_height()/total) # percentage of each class of the category
        x = p.get_x() + p.get_width() / 2 - 0.05 # width of the plot
        y = p.get_y() + p.get_height()           # hieght of the plot
        ax.annotate(percentage, (x, y), size = 12) # annotate the percantage
```

Observations on Gender

In [ ]:
```
plt.figure(figsize=(10,7))
ax = sns.countplot(data['Gender']) #count plot for Gender
plt.xlabel('Gender')
plt.ylabel('Count')
bar_perc(ax,data['Gender'])
```

Observations on Marital Status

In [ ]:
```
plt.figure(figsize=(10,7))
ax = sns.countplot(data['MaritalStatus'])
plt.xlabel('Marital Status')
plt.ylabel('Count')
bar_perc(ax,data['MaritalStatus'])
```

Observations on Product

In [ ]:
```
plt.figure(figsize=(10,7))
ax = sns.countplot(data['Product'])
plt.xlabel('Product')
plt.ylabel('Count')
bar_perc(ax,data['Product'])
```

Observations:

1. There are more Males than females, 57.8% to 42.2%.
2. Most of the participants are Partnered than single, 59.4% to 40.6%.

3. The most common product is TM195 with 44.4% while the least common product in TM798.

Bivariate Analysis

Correlation and Covariance

In [ ]:
```
data.corr() #shows the correlation
```

In [ ]:
```
data.cov() #shows the covariance
```

In [ ]:
```
plt.figure(figsize=(16,12))
sns.heatmap(data.corr(), annot=True, linewidths=.5, fmt= '.1f', center = 1 )  # heatmap
plt.show()
```

Observations:

1. Education has a high correlation with Income, which is to be expected as having higher education means that the income is higher.
2. Usage has high correlation between Fitness (0.66) and Miles (0.759) a person using the treadmills more are more likely to be Fit and run more miles.
3. Correlation does not imply causation.
4. There does not seem to be a relationship between Education and Fitness.

Bivariate Scatter Plots

In [ ]:
```
sns.pairplot(data = data, kind = 'reg', hue = 'Gender') #pairplot with gender as hue
```

Observation:

1. The data shows similar trend as observed with the heat map.
2. Both male and female show positive correlation with Age.
3. As age increases it shows a positive correlation for males, while there is a negative correlation for females.
4. As age increases there is a positive correlation with miles run for males and negative correlation for females.
5. Usage is high, among high income clients as shown for both male and female. As usage increases likely fitness level is high for both male and female of high income clients.
6. Higher educated clients are more likely to use the products are show positive correlation with both usage and fitness, in both male and female genders.

In [ ]:
```
sns.pairplot(data = data, kind = 'reg', hue = 'MaritalStatus') #pairplot with marital Status as hue
```

Observations:

1. The data shows similar trend as observed with the heat map.
2. Age and fitness level is showing a positive correlation among both partnered and single clients.
3. Higher education indicates higher Usage and Fitness level for both partnered and single clients.
4. Higher income clients show higher usage, fitness level and more miles run for both partnered and single clients.

Bivariate Analysis Bar Plots

In [ ]:
```
plt.figure(figsize=(10,5))  # setting the figure size
ax = sns.barplot(x='Gender', y = 'Usage',  data=data, palette='muted')  #barplot
```

Observation:

1. Males show an increase in usage as compared to women.

In [ ]:
```
plt.figure(figsize=(10,5))  # setting the figure size
ax = sns.barplot(x='Gender', y = 'Fitness',  data=data, palette='dark')
```

Observation:

1. Male have a higher fitness level compared to women, which indicates that males are more likely to use the treadmills.

In [ ]:
```
plt.figure(figsize=(10,5))  # setting the figure size
ax = sns.barplot(x='MaritalStatus', y = 'Fitness',  data=data, palette='muted')
```

In [ ]:
```
plt.figure(figsize=(10,5))  # setting the figure size
```

```
ax = sns.barplot(x='Education', y = 'Usage',  data=data, palette='muted')
```

Observation:

1. Both married and single clients show very similar level of fitness, which again indicates similar level of usuage of the product.

In [ ]:
```
plt.figure(figsize=(10,5))  # setting the figure size
ax = sns.barplot(x='Fitness', y = 'Miles',  data=data, palette='muted')
```

Observation:

1. As previously indicated by the heatmap, there is a strong correlation between Fitness level and Miles run. The higher the fitness of the clients, the more they are able to run.

In [ ]:
```
plt.figure(figsize=(10,5))  # setting the figure size
ax = sns.barplot(x='Fitness', y = 'Usage',  data=data, palette='muted')
```

Observation:

1. Again as previously indicated on the heatmap, level of Fitness strongly correlates with Usage, as the more a client uses the product the more their level of fitness.

In [ ]:
```
plt.figure(figsize=(10,5))  # setting the figure size
ax = sns.barplot(x='Product', y = 'Usage',  data=data, palette='muted')
```

Observation:

1. The clients are more likely to use the TM798 by a greater margin, while both TM195 and TM498 show similar levels of usage.

Multivariate Analysis

Bar Plots

In [ ]:
```
plt.figure(figsize=(25,10))
sns.barplot(data=data,x='Gender',y='Usage',hue='MaritalStatus')
plt.show()
```

Observation:

1. Partnered Females are more likely to use the product compared to single females.
2. Single Males are more likely to use the product compared to Partnered Females.

In [ ]:
```
plt.figure(figsize=(15,5))
sns.barplot(data=data,x='Age',y='Usage',hue='Product')
plt.show()
```

Observation:

1.TM789 is seen to be used more compared to the other two products, from the ages 19 to 40.

1. Usage is highest for TM798 around 28 to 29 year olds.

In [ ]:
```
plt.figure(figsize=(15,5))
sns.barplot(data=data,x='Education',y='Miles',hue='Gender')
plt.show()
```

Observation:

1. Males who are more educated more are more likely to run compared to their female counterpart.

In [ ]:
```
data.groupby(by=['Product'])['Miles'].sum().reset_index().sort_values(['Miles']).tail(10).plot(x='Product',
                                                                                                   y='Mil
                                                                                                   kind='
                                                                                                   figsiz

plt.ylabel('Miles') #bar plot showing miles per product in ascending order.
plt.show()
```

Observation:

1. Both TM195 and TM798 are equally likely for the customers to run more on.

Pointplots

.

In [ ]:
```
plt.figure(figsize=(15,5))

sns.pointplot(x="Product", y="Usage", hue = 'Gender',  data=data) #pointplots
plt.show()
```

Observation:

1. For the TM195, males are more likely to use the product.
2. For both the TM498 and TM798, Females are more likely to use the product.

In [ ]:
```
plt.figure(figsize=(15,5))

sns.pointplot(x="Product", y="Usage", hue = 'MaritalStatus',  data=data)
plt.show()
```

Observation:

1. For TM195, the usage is mostly by single customers.
2. While the TM798 is mostly used by partnered customers.

Lineplots

In [ ]:
```
sns.lineplot(x='Usage',y='Fitness', data=data, hue = 'Gender' ) #line plots
```

Observation:

1. As the Usage increases both Male and Female customers show increase in their Fitness Level
2. Males Show a higher increase rate as compared to Females when it comes to Usage and corresponding Fitness levels.

In [ ]:
```
sns.lineplot(x='Usage',y='Fitness', data=data, hue = 'MaritalStatus' )
```

Observation:

1. Among both single and partnered clients, both show a positive correlation between Usage and Fitness.

In [ ]:
```
sns.lineplot(x='Usage',y='Income', data=data, hue = 'Product' )
```

Observation:

1. There is a significant increase in Usage of TM796 product among higher income clients as compared to the other two products.
2. Both TM195 and TM498 see significant decrease in usage between 4 to 5.
3. Both TM195 and TM498 show increase in usage from 2 to 4.

In [ ]:
```
sns.lineplot(x='Usage',y='Income', data=data, hue = 'Gender' )
```

Observation:

1. There is a significant increase in usage among females with higher income.
2. Usage streadily increases among males as income increases.

In [ ]:
```
sns.lineplot(x='Fitness',y='Education', data=data, hue = 'Gender' )
```

Observation:

1. Higher education doesn't necessarily mean higher Fitness, which means low usage of the product.
2. However, there is an increase in fitness level for both female and male clients, who have 15 years to 17 years of experience.
3. There is a sharp decline in fitness level from 18 to 16 years of educational experience but increases later at a steady rate.

Conclusion and Recommendations

Conclusion:

We analyzed a dataset containing 180 entries with 9 columns including 3 categorical variables and 5 integer based variables. The data contained 3 unique product information of different type of treadmill products, and their associated variables pertaining to the purchase of the product which includes Age, Gender, Education, Marital Status, Usage of the product, Fitness level, Income and Miles ran in a week using the different products.

1. More males are more likely to use the product as compared to females.
2. More partnered clients bought the treadmills as compared to single clients, but both single and partnered clients were equally likely to use the products.
3. The TM798 was in higher popularity, followed by TM498 and TM195.
4. Education has a high correlation with Income, which is to be expected as having higher education means that the income is higher. Higher education doesn't necessarily mean higher Fitness, which means low usage of the product. However, there is an increase in fitness level for both female and male clients, who have 15 years to 17 years of experience.
5. Usage has high correlation between Fitness (0.66) and Miles (0.759) a person using the treadmills more are more likely to be Fit and run more miles.
6. Fitness level in males were higher than in females, due to their usage and extra miles run. Fitness level correlated positively with extra miles ran and the usage of the product, and both partnered and single clients were equally fit.
7. Partnered Females are more likely to use the product compared to single females. Single Males are more likely to use the product compared to Partnered Females. Males who are more educated are more likely to run compared to their female counterpart.
8. Even though both TM195 and TM798 the customers were likely to run more miles on, the usage is seen highest in TM195 followed, by TM798. TM195 is mostly used by single customers, while TM798 is mostly used by partnered customers. There is a significant increase in Usage of TM796 product among higher income clients as compared to the other two products.

Recommendation:

1. Targeting more males than females would be ideal as more male clients are present.
2. The TM798 is more likely to be used by higher income clients. TM798 show higher usage and popularity among customers, so this product will sell higher than TM498 and TM195.
3. Target more partnered clients as they are more likely to buy the products.
4. Target higher level fitness male clients as they are more likely to use the product.
5. When targeting females target partnered females as they are more likely to buy the product compared to single females.
6. Targeting clients who have educational experience from 15 years to 17 years will be optimal.
7. Higher income clients are more likely to use the product than low from 60000 above are more likely to use the products, so targeting the higher income group will be optimal.

Further Analysis using Profiling

Conducting further analysis using inbuilt function profiling. We can dig deeper and find any trends between the variables in bivariate analysis.

In [ ]:
```
from pandas_profiling import ProfileReport
# Use the original dataframe, so that original features are considered
prof = ProfileReport(data)
# to view report created by pandas profile
prof
```

In [ ]:
```
prof.to_file('output.html')
```