Business Case: Aerofit - Descriptive Statistics & Probability

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from wordcloud import WordCloud
import plotly.express as px
import warnings # Importing warnings to ignore warnings
warnings.filterwarnings("ignore")
```

Defining Problem Statement and Analysing basic metrics

Aerofit is a leading brand in the field of fitness equipment. Objective of the analysis is to help the market research team at AeroFit 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. Importing the dataset and performing usual data analysis steps:

```
aerofit = pd.read_csv("/content/Aerofit data set.csv")
aerofit
```

	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

```
aerofit.head()
```

	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

```
aerofit.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
```

From above we can see that there are 8 columns, 180 rows

```
aerofit.describe()
```

	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

Observations: **1.**There are 180 Rows and 9 Columns. **2.**Thereare no missing values in data. 3.Min and max age of the person is 18 & 50,mean is 28.7 and 75% of the persons have the age less thav equal to 33. **4.**Education: Mean Education is 15 with maximum as 21 and minimum as 12. **5.**Usage: Mean Usage per week is 3.4, with maximum as 7 and minimum as 2. **6.**Fitness: Average rating is 3.3 on a scale of 1 to 5. **7.**Miles: Average number of miles the customer walks is 103 with maximum distance travelled by most people is almost 115 and minimum is 21. **8.**Income (in $): Most customer earns around 58K annually, with maximum of 104K and minimum almost 30K

Non-Graphical Analysis: Value counts and unique attributes

```
aerofit.nunique()
```

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

```
aerofit['Product'].unique().tolist()
```

```
['KP281', 'KP481', 'KP781']
```

```
aerofit['Age'].unique()
```

```
array([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])
```

```
aerofit['Age'].value_counts()
```

```
25    25
23    18
24    12
26    12
28     9
35     8
33     8
30     7
38     7
21     7
22     7
27     7
31     6
34     6
29     6
20     5
40     5
32     4
19     4
```

```
48     2
37     2
45     2
47     2
46     1
50     1
18     1
44     1
43     1
41     1
39     1
36     1
42     1
Name: Age, dtype: int64
```

```
aerofit['Gender'].value_counts()
```

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

```
aerofit['Education'].unique().tolist()
```

```
[14, 15, 12, 13, 16, 18, 20, 21]
```

```
aerofit['Fitness'].value_counts()
```

```
3    97
5    31
2    26
4    24
1     2
Name: Fitness, dtype: int64
```

```
aerofit['Product'].value_counts()
```

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

```
aerofit['Usage'].value_counts()
```

```
3    69
4    52
2    33
5    17
6     7
7     2
Name: Usage, dtype: int64
```

```
aerofit['MaritalStatus'].value_counts()
```

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

Observation:

1) There are 3 unique products 'KP281', 'KP481', 'KP781', with 'KP281' being used by most of the users. 2) There are 32 unique ages. Showing large number of users in there 20's. 3) In the data provided, 104 entries are for Males and 76 for females. Showing males are more into fitness activity. 4) Years of education in the data set for users ranging from 12 to 21 years. Unique years of education as shown in this list [14, 15, 12, 13, 16, 18, 20, 21]. 5) For the self rated fitness data we can observe that fitness value ranges from 2 to 7, with large number of users rating themselves as 3 on fitness scale. 6) Users plan to use the trademill varies from 2 to 7 days a week with highest number of days planned as 3 and lowest as 7 days. 7) There are 107 partnered users and 73 singles.

Converting categorical attributes to 'category'

```
df_cat = aerofit
df_cat['Fitness_category'] = df_cat['Fitness'].astype('category')
df_cat.head()
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles	Fitness_cat
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	

```
Fitness_mapping = {
    1 : 'Poor Shape',
    2 : 'Bad Shape',
    3 : 'Average Shape',
    4 : 'Good Shape',
    5 : 'Excellent Shape'
}
df_cat['Fitness_category'] = df_cat['Fitness_category'].replace(Fitness_mapping)
df_cat.head()
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles	Fitness_cat
0	KP281	18	Male	14	Single	3	4	29562	112	Good
1	KP281	19	Male	15	Single	2	3	31836	75	Average
2	KP281	19	Female	14	Partnered	4	3	30699	66	Average
3	KP281	19	Male	12	Single	3	3	32973	85	Average
4	KP281	20	Male	13	Partnered	4	2	35247	47	Bad

Categorization of Fitness Rating:

1. Poor Shape
2. Bad Shape
3. Average Shape
4. Good Shape
5. Excellent Shape

Statistical Summary

```
sr = aerofit['Product'].value_counts()
sr.map(lambda val: round(val / sr.sum() * 100, 2))
```

```
KP281    44.44
KP481    33.33
KP781    22.22
Name: Product, dtype: float64
```

44.44% of customers bought KP281 product type 33.33% of customers bought KP481 product type 22.22% of customers bought KP781 product type

```
gender_counts = aerofit['Gender'].value_counts()
total_count = gender_counts.sum()
gender_percentage = gender_counts.map(lambda count: (count / total_count) * 100)
gender_percentage = gender_percentage.round(2)
gender_percentage
```

```
Male      57.78
Female    42.22
Name: Gender, dtype: float64
```

57.78% of users are Male and 42.22% users are Female

```
maritial_status_counts = aerofit['MaritalStatus'].value_counts()
total_count_mar = maritial_status_counts.sum()
maritial_status_percentage = maritial_status_counts.map(lambda count: (count / total_count_mar) * 100)
maritial_status_percentage = maritial_status_percentage.round(2)
maritial_status_percentage
```

```
Partnered    59.44
Single       40.56
Name: MaritalStatus, dtype: float64
```

59.44% of customers are Partnered 40.56% of customers are Single

```
fitness_counts = df_cat['Fitness_category'].value_counts()
total_fitness_counts = fitness_counts.sum()
fitness_cat_percentage = fitness_counts.map(lambda count: (count / total_fitness_counts) * 100)
fitness_cat_percentage = fitness_cat_percentage.round(2)
fitness_cat_percentage
```

```
Average Shape      53.89
Excellent Shape    17.22
Bad Shape          14.44
Good Shape         13.33
Poor Shape          1.11
Name: Fitness_category, dtype: float64
```

1) More than 53% of users have rated themselves as average in fitness (rated 3). 2) 14% of customers have rated their fitness less than average. 3) Over 17% of customers have peak fitness ratings.

Visual Analysis - Univariate & Bivariate

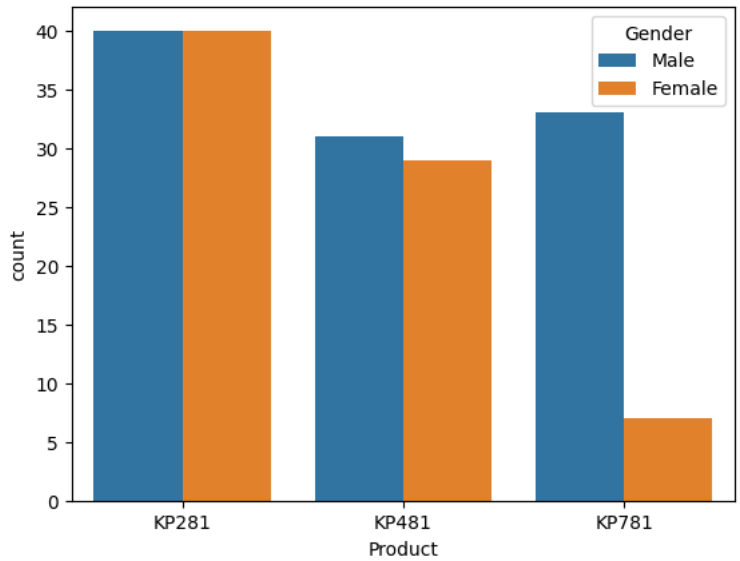
Univariate Analysis

1. Continuous variable(s): Distplot, countplot, histogram for univariate analysis

Understanding the distribution of the data for the quantitative attributes: Age, Education, Usage, Fitness, Income and Miles

```
sns.countplot(data=aerofit,x='Product', hue='Gender')
plt.show
```

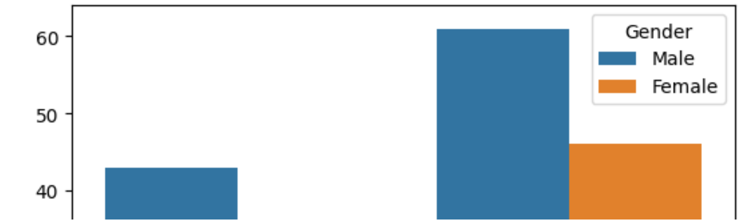
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



KP281 is the most commonly purchased product type, KP481 is the second most top product type purchased and KP781 is the least purchased product type

```
sns.countplot(data=aerofit,x='MaritalStatus', hue='Gender')
plt.show
```

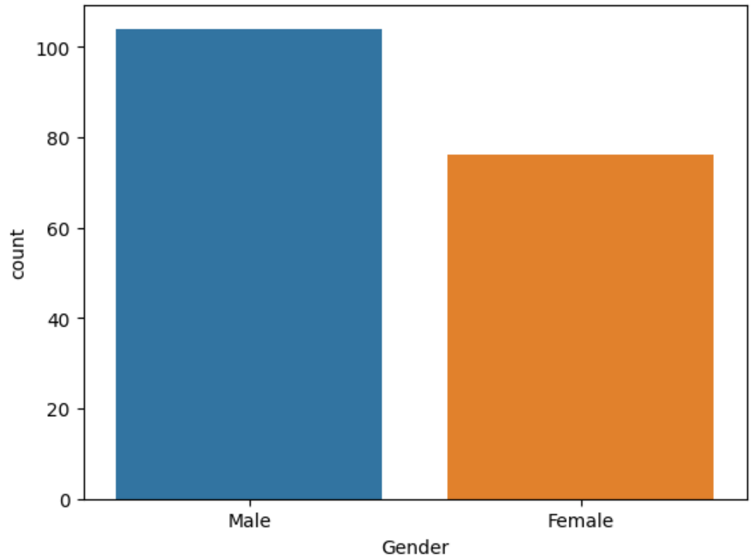
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



Most products purchased by couples/Married/Partnered customer category

```
total_miles = aerofit['Miles'].sum()
sns.countplot(data=aerofit,x='Gender')
plt.show
```

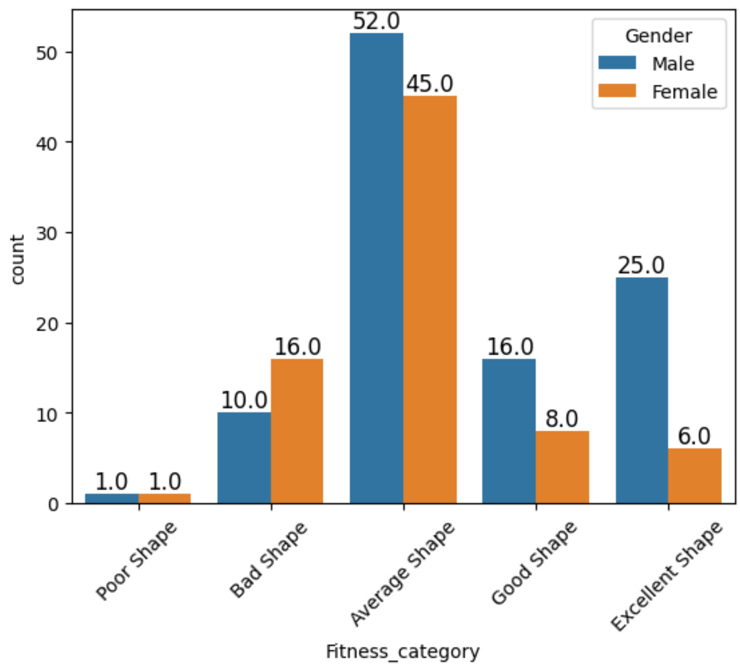
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



Most users are males. Most products purchased by Males, females are less interested in the product compared to Males

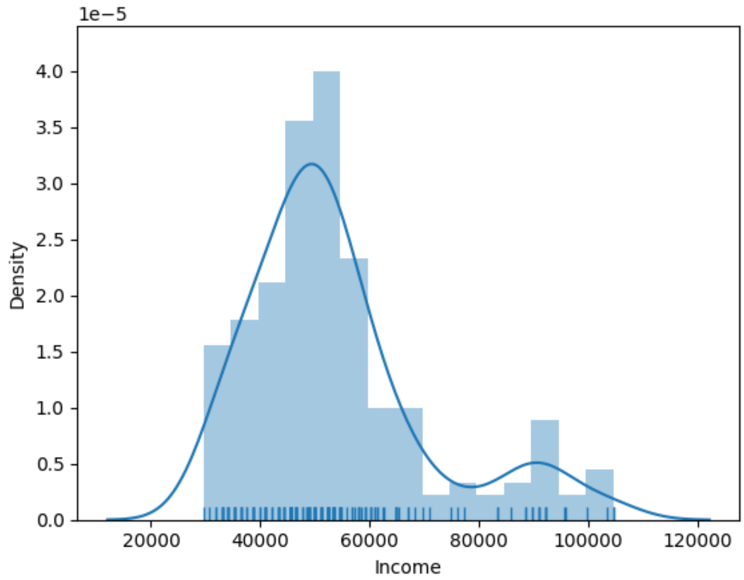
```
ax = sns.countplot(data=df_cat,x='Fitness_category', hue='Gender')
plt.xticks(rotation=45)
for p in ax.patches:
    ax.annotate(f'{p.get_height()}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='bottom', fontsize=12)
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```



More than 90 users have rated their physical fitness rating as Average, among these users 52 are males and 45 are females. Excellent shape is the second highest rating provided by the users to themselves having 25 males and 6 females.

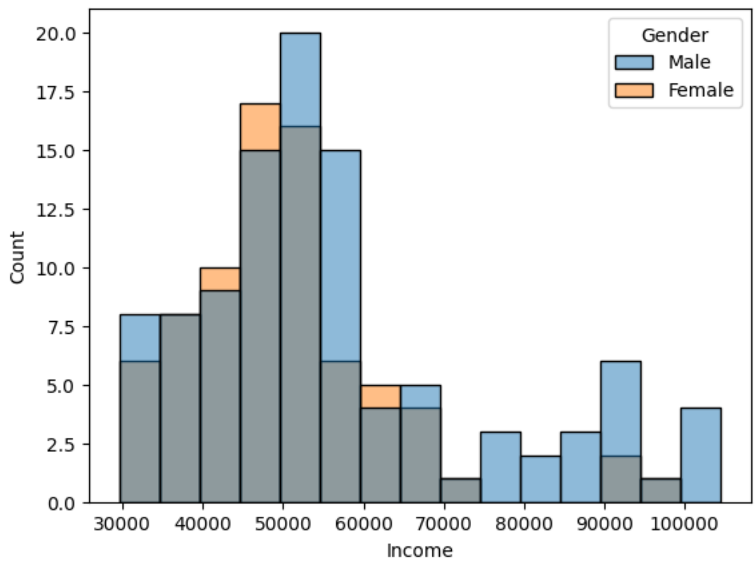
```
# Income Analysis - Distplot
sns.distplot(aerofit.Income,rug=True)
plt.show()
```



Most of customers who have purchased the product have a average income between 40K to 60K Average Income density is over 3.0

```
# Income Analysis - Histogram
sns.histplot(data=aerofit,x='Income', hue='Gender')
```

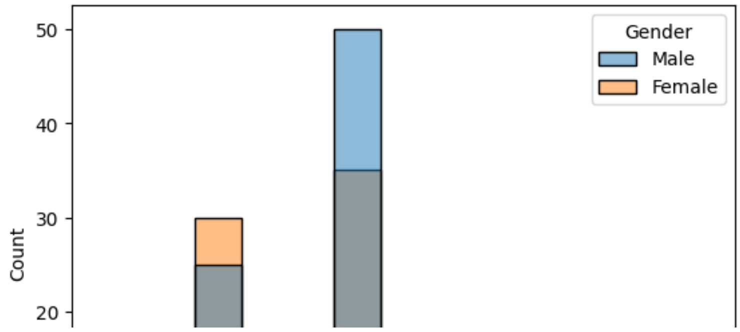
<Axes: xlabel='Income', ylabel='Count'>



More than 35 customers earn 50- 55K per year More than 30 customers earn 45-50K per year. More than 20 customers earn 55-60K per year.

```
# Education Analysis - Histogr
sns.histplot(data=aerofit,x='Education', hue='Gender')
```

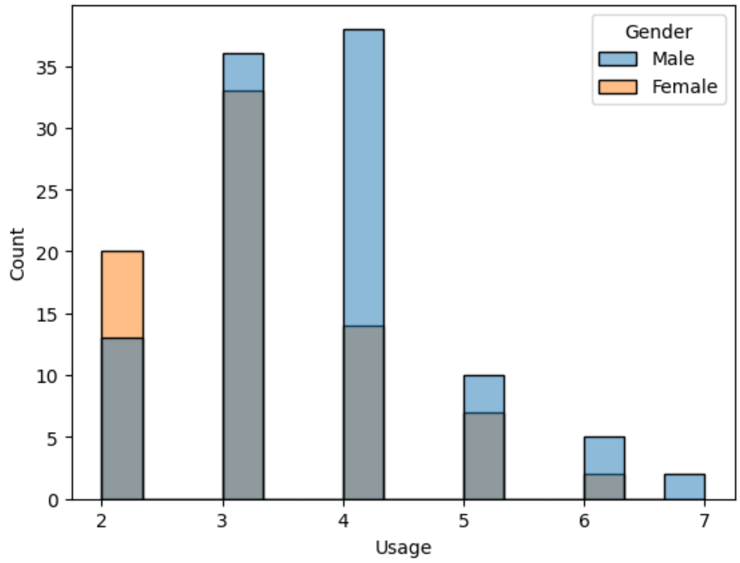
```
<Axes: xlabel='Education', ylabel='Count'>
```



Highest number of customers have 16 as their Education 14 is the second highest education among the customers 20 is the least education among the customers

```
# Usage Analysis - Histogram
sns.histplot(data=aerofit,x='Usage', hue='Gender')
```

```
<Axes: xlabel='Usage', ylabel='Count'>
```



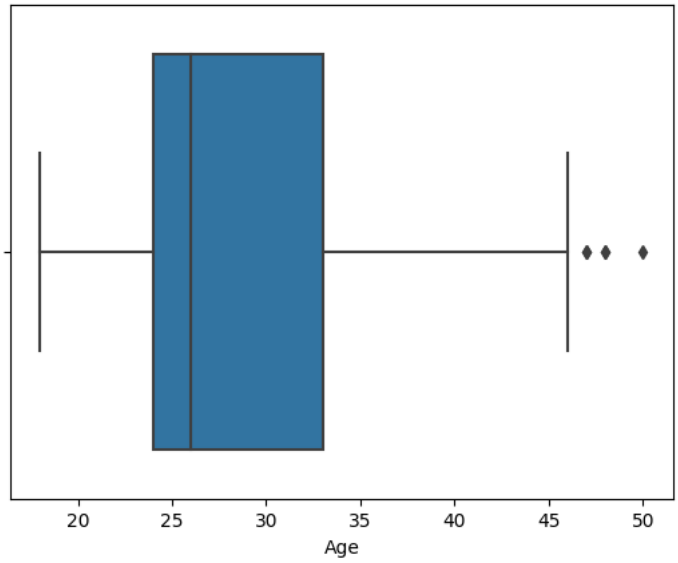
3 days per week is the most common usage among the customers 4 days and 2 days per week is the second and third highest usage among the customers Very few customers use product 7 days per week

2. For categorical variable(s): Boxplot

```
# Usage Analysis - Box plot
sns.boxplot(data=aerofit,x='Usage')
plt.show()
```

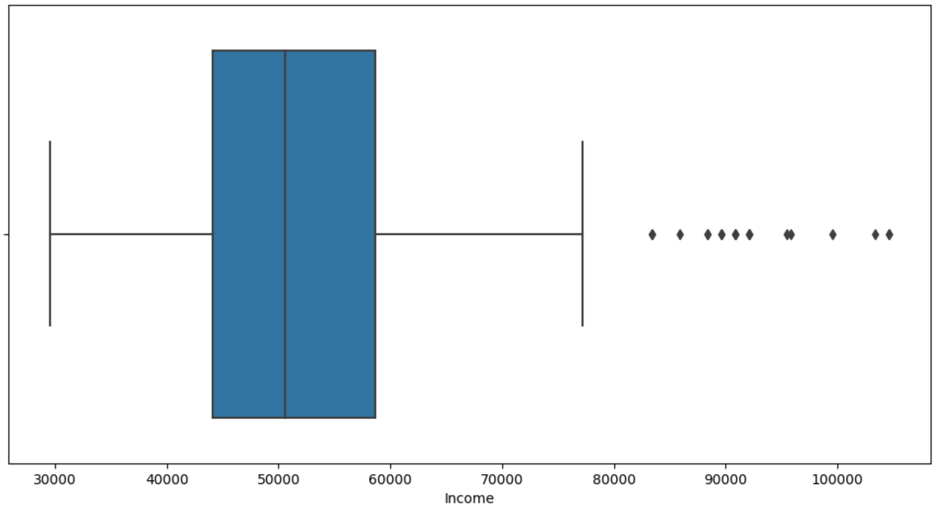
3 to 4 days is the most preferred usage days for customers 6 and 7 days per week is roughly the usage days for few customers (Outliers)

```
# Age Analysis - Box plot
sns.boxplot(data=aerofit,x='Age')
plt.show()
```



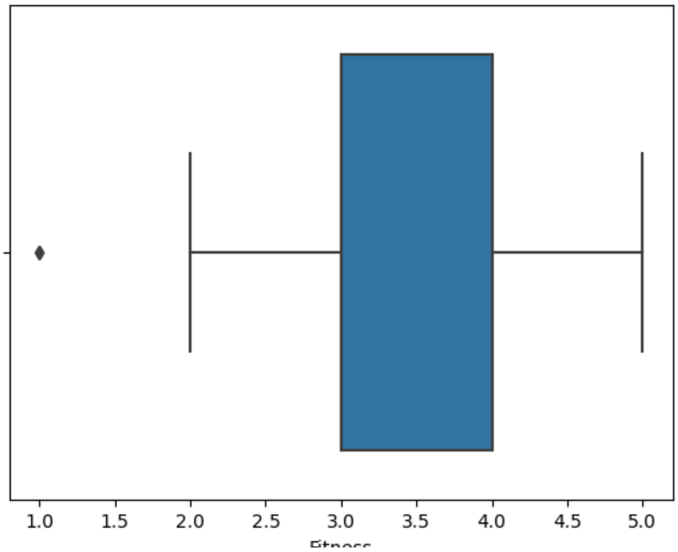
23 to 34 is the most common customer age group that has purchased the product Above 45 years old customers are very few compared to the young age group given in the dataset

```
# Income Analysis - Box plot
plt.figure(figsize=(12,6))
sns.boxplot(data=aerofit,x='Income')
plt.show()
```



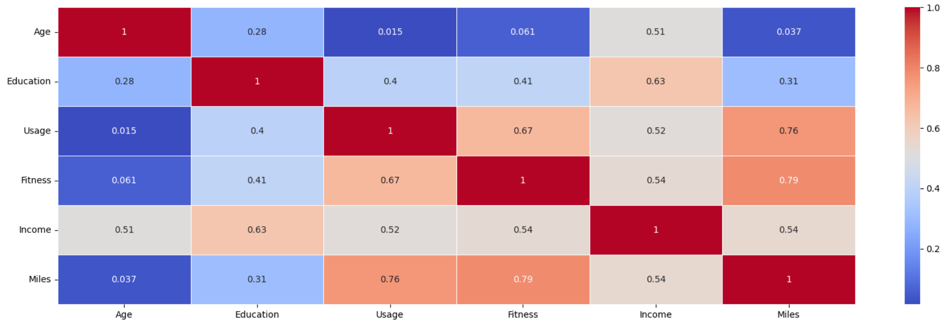
Few customers have income above 80K per annum(Outliers) Most customers earn from 45K to around 60K per annum

```
# Fitness Rating Analysis - Bo
sns.boxplot(data=aerofit,x='Fitness')
plt.show()
```



Couple of customers have rated their fitness rating as 1 - Poor Shape Most customers have rated fitness rating as 3.0 to 4.0

```
#Correlation HeatMap
plt.figure(figsize=(20,6))
ax = sns.heatmap(aerofit.corr(), annot=True, cmap='coolwarm', linewidths=0.5)
plt.yticks(rotation=0)
plt.show()
```



In the above heatmap linear relationship between data points is evaluated:

- Correlation between Age and Miles is 0.03.
- Correlation between Education and Income is 0.62.
- Correlation between Usage and Fitness is 0.66.
- Correlation between Fitness and Age is 0.06.
- Correlation between Income and Usage is 0.51.
- Correlation between Miles and Age is 0.03.

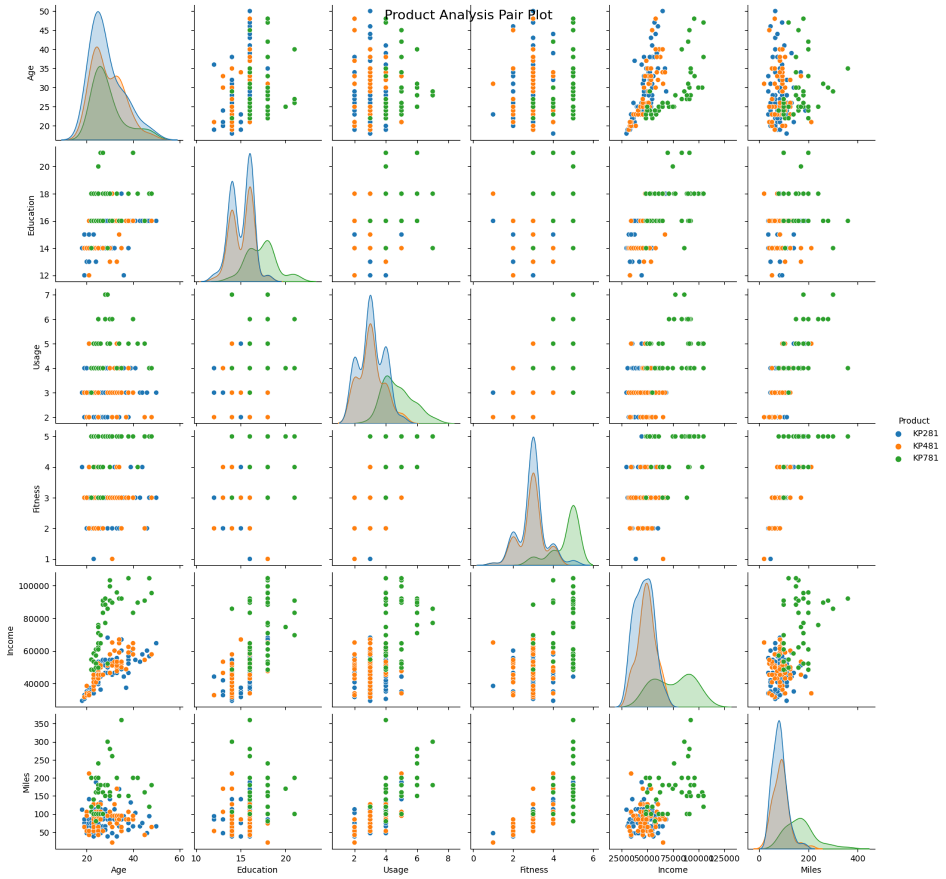
```
# Product Analysis - Pair Plot
plt.figure(figsize=(10, 6))  # Adjust the figure size if needed
sns.pairplot(aerofit, hue='Product', diag_kind='kde')

# Set the title for the pair plot
plt.suptitle('Product Analysis Pair Plot', fontsize=16)

# Display the plot
```

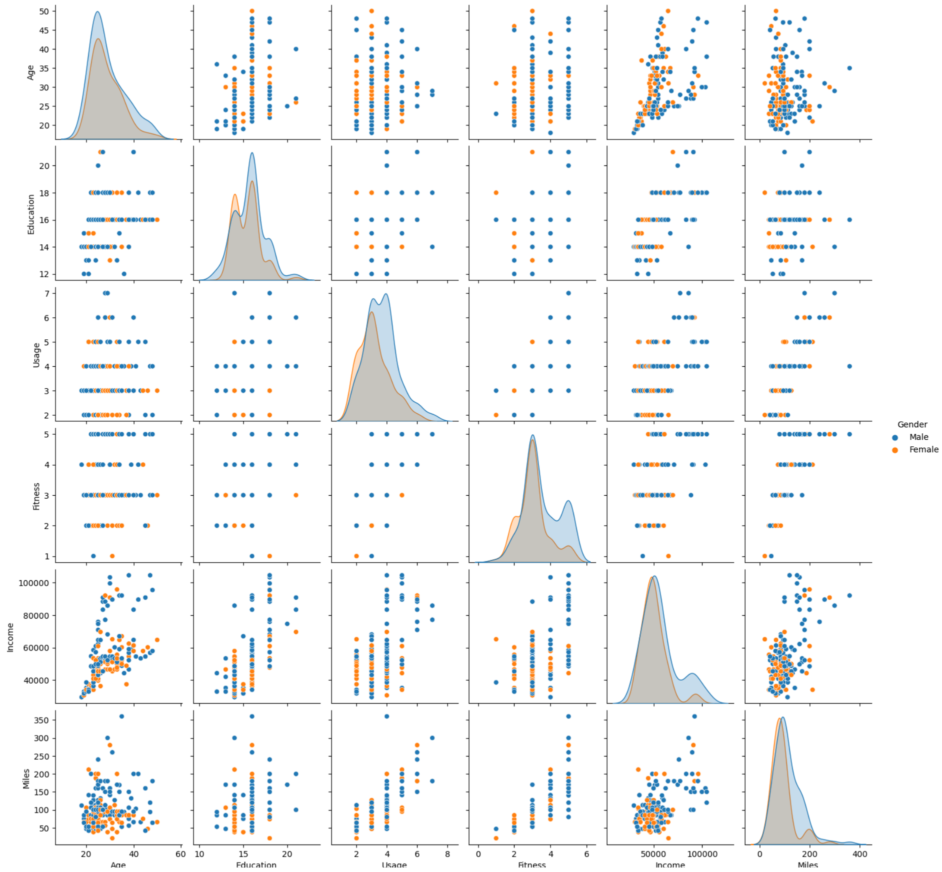
```
plt.show()
```

<Figure size 1000x600 with 0 Axes>



In the above pair plot the correlation with other attributes are pivotted around the marital status of the customer

```
# Gender Analysis - Pair Plot
sns.pairplot(aerofit,hue='Gender',kind='scatter')
plt.show()
```



Here the pair plot's correlation is same as the above mentioned heatmap

Bivariate Analysis

```
# Average usage of each produc
aerofit.groupby('Product')['Usage'].mean()
```

```
Product
KP281    3.087500
KP481    3.066667
KP781    4.775000
Name: Usage, dtype: float64
```

Mean usage for product KP281 is 3.08 Mean usage for product KP481 is 3.06 Mean usage for product KP781 is 4.77

```
# Average Age of customer usin
aerofit.groupby('Product')['Age'].mean()
```

```
Product
KP281    28.55
KP481    28.90
KP781    29.10
Name: Age, dtype: float64
```

Mean Age of the customer who purchased product KP281 is 28.55 Mean Age of the customer who purchased product KP481 is 28.90 Mean Age of the customer who purchased product KP781 is 29.10

```
# Average Education of custome
aerofit.groupby('Product')['Education'].mean()
```

```
Product
KP281    15.037500
KP481    15.116667
KP781    17.325000
Name: Education, dtype: float64
```

Mean Education qualification of the customer who purchased product KP281 is 15.03 Mean Education qualification of the customer who purchased product KP481 is 15.11 Mean Education qualification of the customer who purchased product KP781 is 17.32

```
# Average customer fitness rat
aerofit.groupby('Product')['Fitness'].mean()
```

```
Product
KP281    2.9625
KP481    2.9000
KP781    4.6250
Name: Fitness, dtype: float64
```

Customer fitness mean for product KP281 is 2.96 Customer fitness mean for product KP481 is 2.90 Customer fitness mean for product KP781 is 4.62

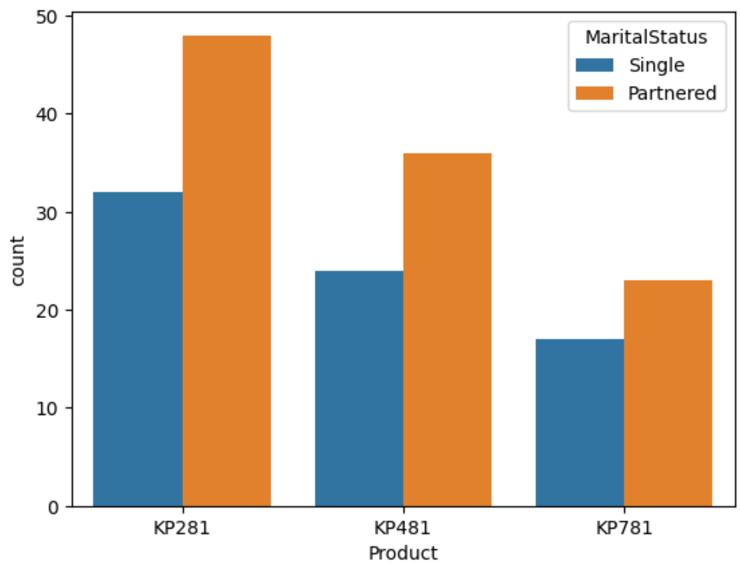
```
# attrs = ['Age', 'Education', 'Income', 'Miles', 'Product']
# sns.set_style("white")
# fig, axs = plt.subplots(nrows=2, ncols=3, figsize=(12, 8))
# fig.subplots_adjust(top=1.2)
# count = 0

# for i in range(2):
#     for j in range(3):
#         if count < len(attrs):
#             sns.boxplot(data=aerofit, x=attrs[count], ax=axs[i,j])
#             axs[i,j].set_title(f"Box Plot of {attrs[count]}")
#         count += 1

# # Adjust the space between subplots
# plt.tight_layout()

# # Display the plot
# plt.show()
```

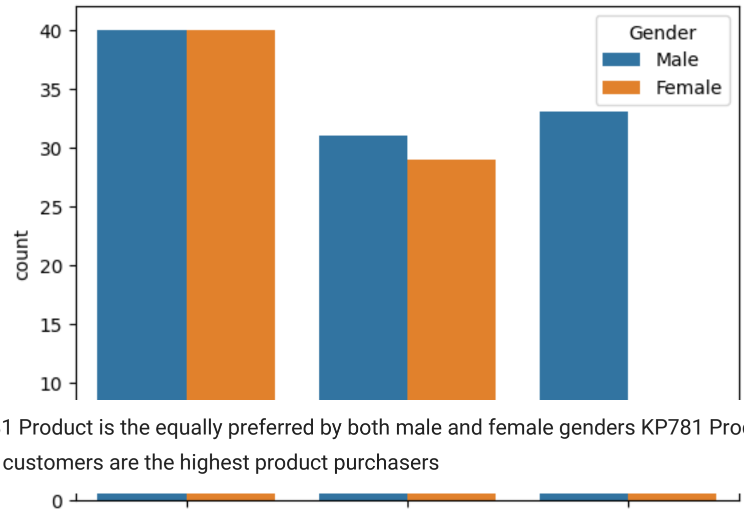
```
# Product purchased among Marr
sns.countplot(data=aerofit,x='Product', hue= 'MaritalStatus')
plt.show()
```



From the above countplot:

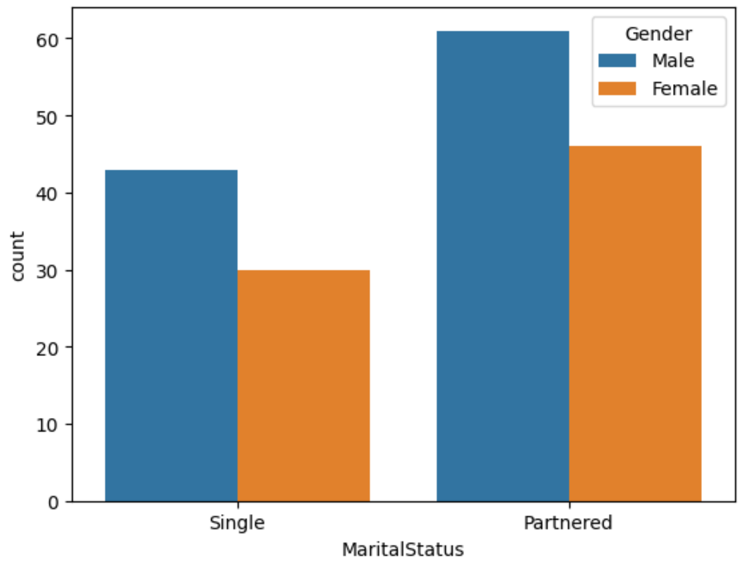
- KP281 is the most preferred product among customers
- KP481 is the second most preferred product among the customers
- Between Singles and Partnered, Partnered customers are the major product purchasers

```
# Product purchased among Male
sns.countplot(data=aerofit,x='Product', hue= 'Gender')
plt.show()
```



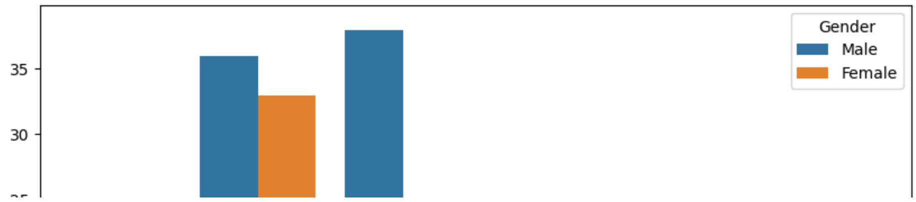
KP281 Product is the equally preferred by both male and female genders KP781 Product is mostly preferred among the Male customers Overall Male customers are the highest product purchasers

```
# Count among Gender and their
sns.countplot(data=aerofit,x='MaritalStatus', hue='Gender')
plt.show()
```



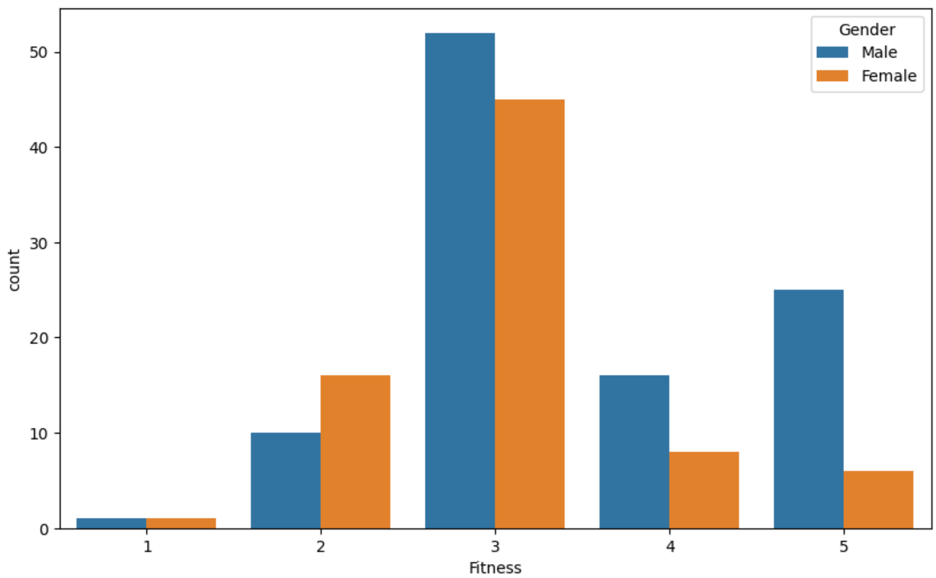
- Partnered customers are the most buyers of aerofit product
- Out of both Single and Partnered customers, Male customers are significantly high
- Female customers are considerably low compared to Male customers

```
# Purchased product usage amon
plt.figure(figsize=(10,6))
sns.countplot(data=aerofit,x='Usage', hue='Gender')
plt.show()
```



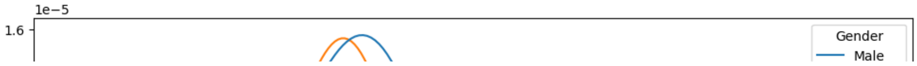
- Among Male and Female genders, Male's usage is 4 days per week
- Female customers mostly use 3 days per week
- Only few Male customers use 7 days per week whereas female customer's maximum usage is only 6 days per week

```
# Fitness rating among the cus
plt.figure(figsize=(10,6))
sns.countplot(data=aerofit,x='Fitness', hue='Gender')
plt.show()
```



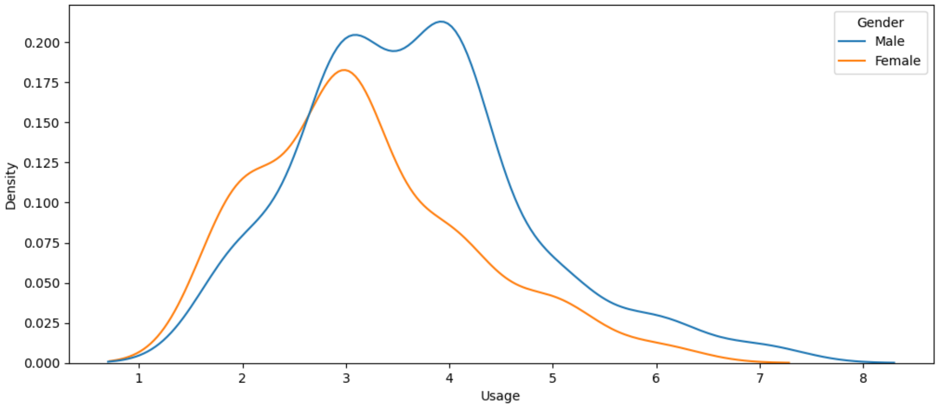
- Among the fitness rating both Male and Female most have rated as average
- Significant number of Male customers are at Excellent shape compared to Female customers

```
 #Product purchased Customers
plt.figure(figsize=(12,5))
sns.kdeplot(data=aerofit,x='Income', hue='Gender')
plt.show()
```



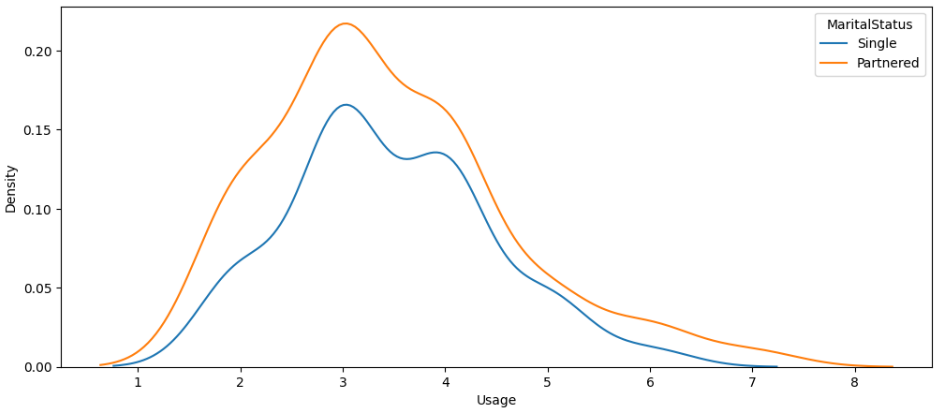
From the above diagram, we can conclude the spike from 40K to around 80K is the most common income per annum of the customers

```
# Product purchased Customers
plt.figure(figsize=(12,5))
sns.kdeplot(data=aerofit,x='Usage',hue= 'Gender')
plt.show()
```



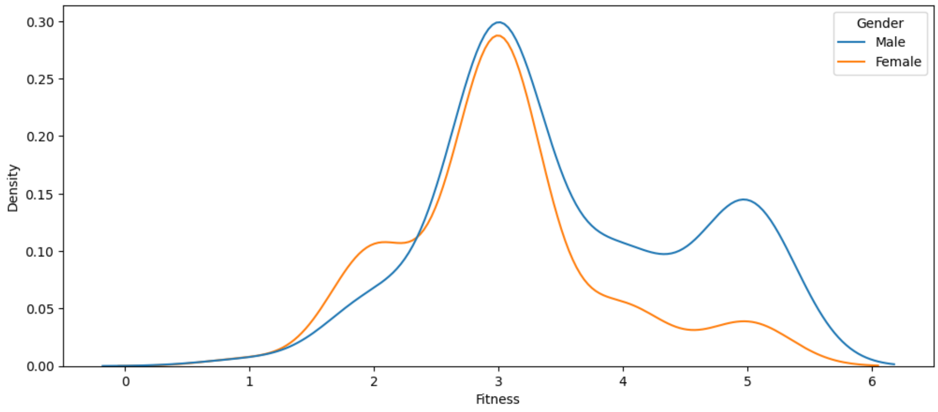
- Male customers usage is significantly higher the female customer
- Female customer's lack consistency after the 3 days per weeek

```
# Product purchased Customers
plt.figure(figsize=(12,5))
sns.kdeplot(data=aerofit,x='Usage',hue='MaritalStatus')
plt.show()
```



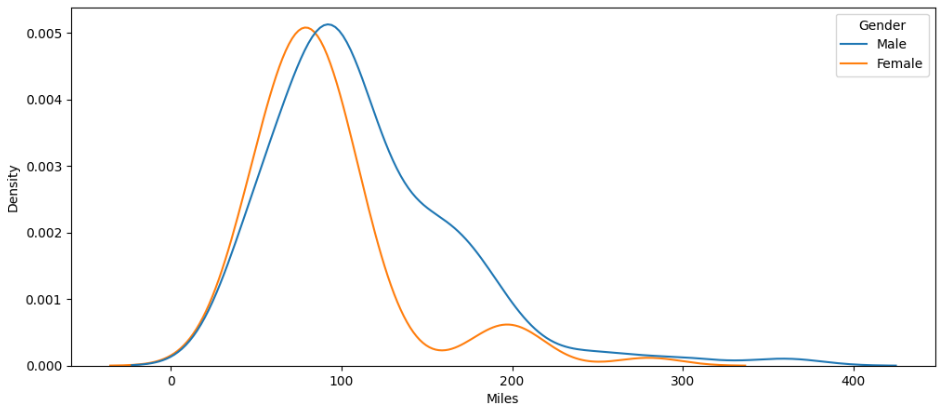
Partnered customers usage is higher than single customers Partnered customers also have greater consistency per week of 7 days per week than single customers

```
# Product purchased Customers
plt.figure(figsize=(12,5))
sns.kdeplot(data=df_cat,x='Fitness',hue='Gender')
plt.show()
```



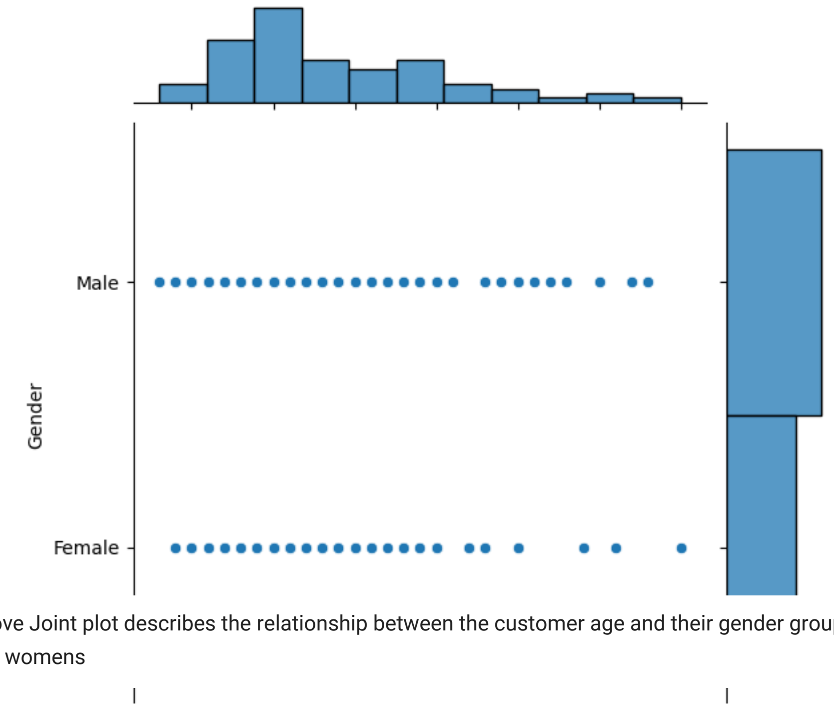
Overall male users have rated themselves in better shape as compared to female users

```
 #Distance covered by each Gen
plt.figure(figsize=(12,5))
sns.kdeplot(data=aerofit,x='Miles',hue='Gender')
plt.show()
```



Male customers have a consistent distance coverage than female customers Female customers have max distance covered as just over 300 miles

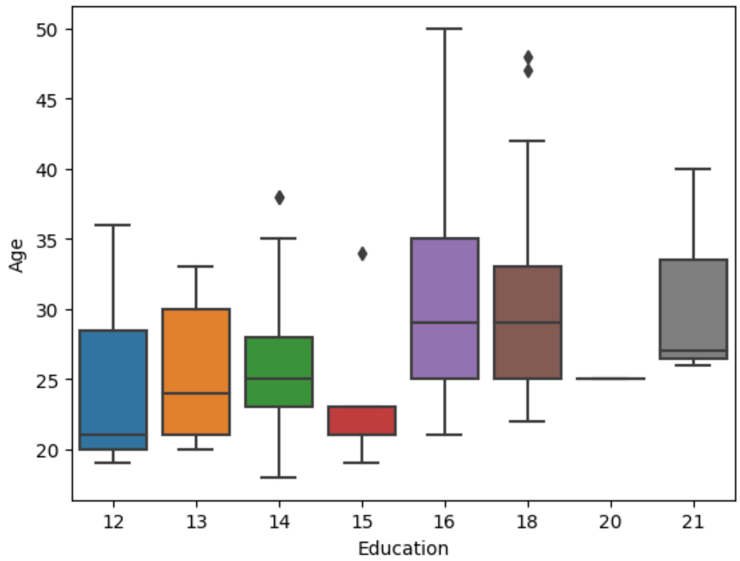
```
# Scatterplot for customers Ge
sns.jointplot(x='Age',y='Gender',data=aerofit)
plt.show()
```



Above Joint plot describes the relationship between the customer age and their gender grouping. Product is not familiar with older or middle age womens

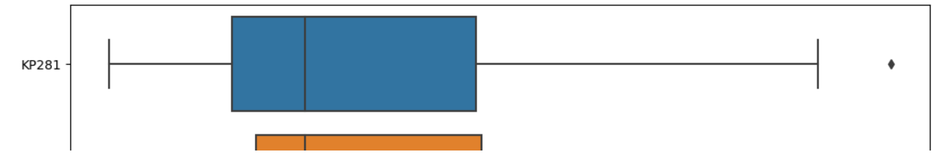
```
sns.boxplot(x='Education',y='Age',data=aerofit)
```

```
<Axes: xlabel='Education', ylabel='Age'>
```



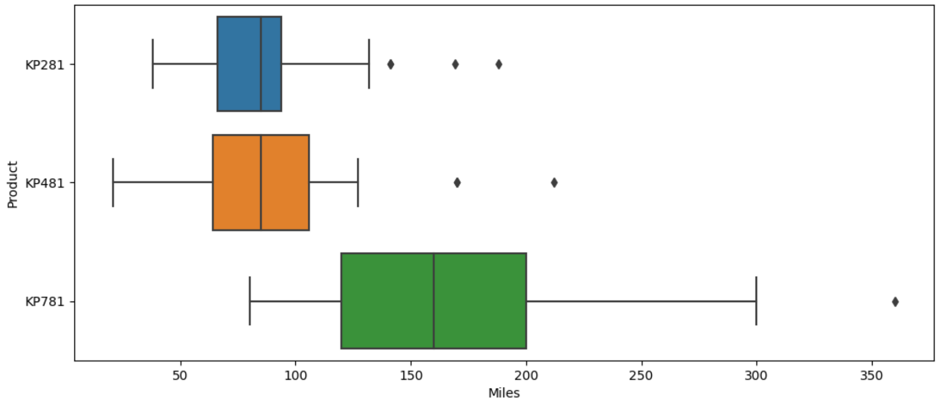
Above box plot shows Education data against Age of the customer

```
plt.figure(figsize=(12,5))
sns.boxplot(x='Age',y='Product', data=aerofit)
plt.show()
```



Roughly few customers with age above 40 use product KP781 Most of the customers are comfortable with KP281 product type KP481 is the second highest popular product among the younger side of the customer

```
# Miles with each product
plt.figure(figsize=(12,5))
sns.boxplot(x='Miles',y='Product', data=aerofit)
plt.show()
```

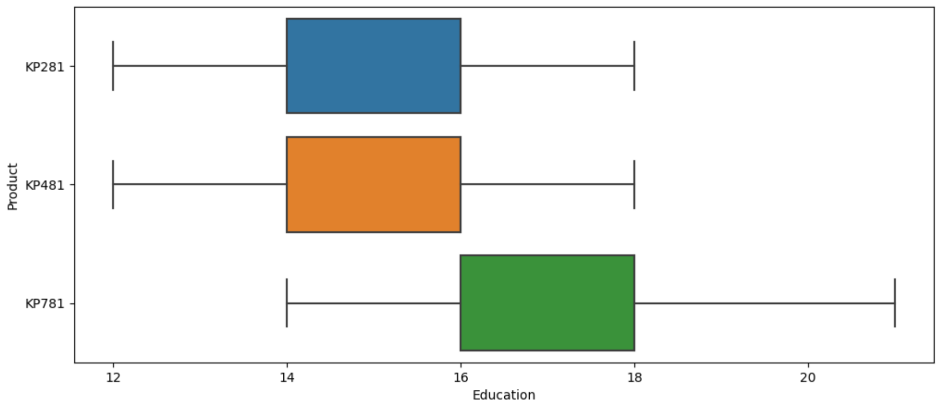


Customers with product KP781, has been able to cover more miles than other two product types.

KP481 product is the second most highest miles covering product among the customers.

KP281 product customer had covered less distance compared with other two product types

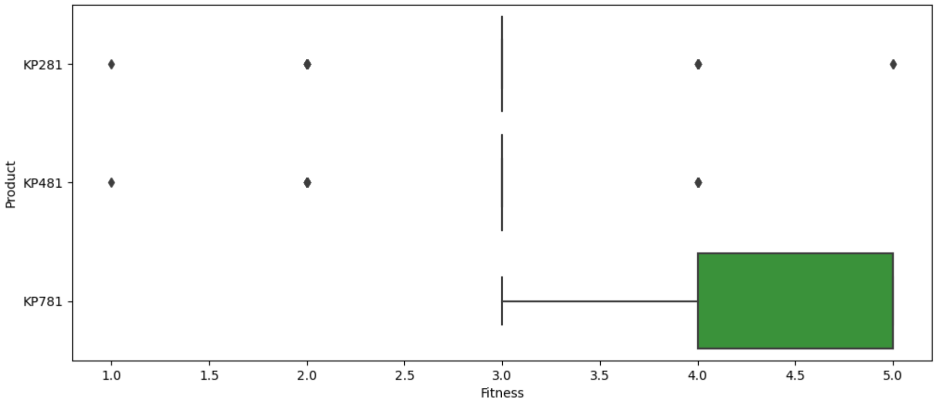
```
# Education of customers with
plt.figure(figsize=(12,5))
sns.boxplot(x='Education',y='Product',data=aerofit)
plt.show()
```



Customers with Higher education of 16 to 18 have preferred mostly product type KP781

Customers with education between 14 to 16 prefer KP281 and KP481 equally

```
# Fitness of customer with eac
plt.figure(figsize=(12,5))
sns.boxplot(x='Fitness',y='Product', data=aerofit)
plt.show()
```



Customers with excellent shape are significantly using KP781 product type

KP481 and KP281 product type are scattered across the fitness rating

Missing Value & Outlier Detection

Missing Value Detection

```
aerofit.isnull().values.any()
```

```
False
```

there are no any missing values

```
aerofit.duplicated().sum()
```

```
0
```

No duplicates as well

Outlier Detection

```
 # Outlier calculation for Mile
q_75, q_25 = np.percentile(aerofit['Miles'], [75, 25])
miles_iqr = q_75 - q_25
print("Inter Quartile Range for Miles is ",miles_iqr)
```

```
Inter Quartile Range for Miles is  48.75
```

Business Insights based on Non- Graphical and Visual Analysis

```
aerofit.Product.value_counts(normalize= True)
```

```
KP281    0.444444
KP481    0.333333
```

```
KP781    0.222222
Name: Product, dtype: float64
```

Probability of buying KP281, KP481 & KP781 are 0.44, 0.33 & 0.22 respectively

Double-click (or enter) to edit

```
aerofit.Gender.value_counts(normalize= True)
```

```
Male      0.577778
Female    0.422222
Name: Gender, dtype: float64
```

- Probability of Male customer is 0.57
- Probability of Female customer is 0.42

```
aerofit.MaritalStatus.value_counts(normalize= True)
```

```
Partnered    0.594444
Single       0.405556
Name: MaritalStatus, dtype: float64
```

Probability of Married/Partnered is 0.59 Probability of Single is 0.40

Customer Age Group Analysis

```
df_cat['age_group'] = df_cat.Age
df_cat.head()
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles	Fitness_cat
0	KP281	18	Male	14	Single	3	4	29562	112	Good
1	KP281	19	Male	15	Single	2	3	31836	75	Average
2	KP281	19	Female	14	Partnered	4	3	30699	66	Average
3	KP281	19	Male	12	Single	3	3	32973	85	Average
4	KP281	20	Male	13	Partnered	4	2	35247	47	Bad

```
age_bins = [0, 21, 35, 45, 60]
age_labels = ['Teen', 'Adult', 'Middle Age', 'Elder Age']

# Use pd.cut() to categorize the 'Age' column based on the defined bins and labels
df_cat['age_group'] = pd.cut(df_cat['Age'], bins=age_bins, labels=age_labels, right=False)
```

```
df_cat.head()
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles	Fitness_cat
0	KP281	18	Male	14	Single	3	4	29562	112	Good
1	KP281	19	Male	15	Single	2	3	31836	75	Average
2	KP281	19	Female	14	Partnered	4	3	30699	66	Average
3	KP281	19	Male	12	Single	3	3	32973	85	Average
4	KP281	20	Male	13	Partnered	4	2	35247	47	Bad

```
df_cat.age_group.value_counts()
```

```
Adult         134
Middle Age     28
Teen           10
Elder Age       8
Name: age_group, dtype: int64
```

```
df_cat.loc[df_cat.Product=='KP281'].age_group.value_counts()
```

```
Adult         57
Middle Age    14
```

```
Teen           6
Elder Age      3
Name: age_group, dtype: int64
```

```
df_cat.loc[df_cat.Product=='KP481'].age_group.value_counts()
```

```
Adult         44
Middle Age    10
Teen           4
Elder Age      2
Name: age_group, dtype: int64
```

```
df_cat.loc[df_cat.Product=='KP781'].age_group.value_counts()
```

```
Adult         33
Middle Age     4
Elder Age      3
Teen           0
Name: age_group, dtype: int64
```

```
pd.crosstab(index=[df_cat["Product"], df_cat["Fitness_category"]], columns=df_cat["Gender"])
```

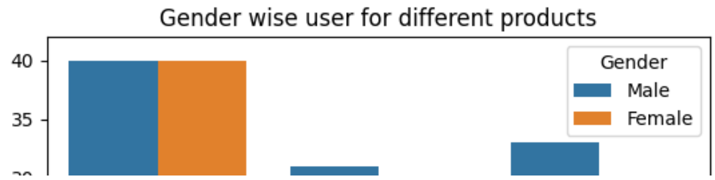
	Gender	Female	Male
Product	**Fitness_category**		
KP281	**Poor Shape**	0	1
	Bad Shape	10	4
	Average Shape	26	28
	Good Shape	3	6
	Excellent Shape	1	1
KP481	**Poor Shape**	1	0
	Bad Shape	6	6
	Average Shape	18	21
	Good Shape	4	4
KP781	**Average Shape**	1	3
	Good Shape	1	6
	Excellent Shape	5	24

Conditional and Marginal Probabilities

Two-Way Contingency Table

Marginal Probabilities

```
sns.countplot(x = "Product", data=aerofit, hue='Gender')
plt.xlabel("Products")
plt.title("Gender wise user for different products")
plt.show()
```




```
pd.crosstab([aerofit.Product],aerofit.Gender, margins=True, margins_name="All")
```

Gender / Product	Female	Male	All
KP281	40	40	80
KP481	29	31	60
KP781	7	33	40
All	76	104	180

```
np.round(((pd.crosstab([aerofit.Product],aerofit.Gender, margins=True, margins_name="All"))))
```

Gender / Product	Female	Male	All
KP281	40	40	80
KP481	29	31	60
KP781	7	33	40
All	76	104	180

```
cross_tab = pd.crosstab(index=aerofit['Product'], columns=aerofit['Gender'], margins=True, margins_name="All")

# Calculate the marginal probabilities
marginal_probabilities = cross_tab / cross_tab.loc["All", "All"]

np.round(marginal_probabilities*100, 2)
```

Gender / Product	Female	Male	All
KP281	22.22	22.22	44.44
KP481	16.11	17.22	33.33
KP781	3.89	18.33	22.22
All	42.22	57.78	100.00

Marginal Probability

Probability of Male Customer Purchasing any product is : 57.77 %

Probability of Female Customer Purchasing any product is : 42.22 %

Marginal Probability of any customer buying

- product KP281 is : 44.44 % (cheapest / entry level product)
- product KP481 is : 33.33 % (intermediate user level product)
- product KP781 is : 22.22 % (Advanced product with ease of use that help in covering longer distance)

Conditional Probabilities

```
cross_tab1 = pd.crosstab(index=aerofit['Product'], columns=aerofit['Gender'], margins=True, margins_name="All")

# Calculate the conditional probabilities for each cell with respect to the 'Gender' column
conditional_probabilities_gender = cross_tab / cross_tab.loc[:, "All"]

# Calculate the conditional probabilities for each cell with respect to the 'Product' row
conditional_probabilities_product = cross_tab.div(cross_tab["All"], axis=0)

print("Conditional Probabilities with respect to Gender:")
print(np.round(conditional_probabilities_gender*100, 2))
```

```
print("\nConditional Probabilities with respect to Product:")
print(np.round(conditional_probabilities_product*100, 2))
```

```
Conditional Probabilities with respect to Gender:
            All  Female  KP281  KP481  KP781  Male
Product
KP281     44.44     NaN    NaN    NaN    NaN   NaN
KP481     33.33     NaN    NaN    NaN    NaN   NaN
KP781     22.22     NaN    NaN    NaN    NaN   NaN
All      100.00     NaN    NaN    NaN    NaN   NaN

Conditional Probabilities with respect to Product:
Gender   Female   Male    All
Product
KP281     50.00  50.00  100.0
KP481     48.33  51.67  100.0
KP781     17.50  82.50  100.0
All       42.22  57.78  100.0
```

Probability of Selling Product

KP281 | Female = 50 %

KP481 | Female = 48 %

KP781 | Female = 18 %

KP281 | male = 50 %

KP481 | male = 52 %

KP781 | male = 83 %

Probability of Female customer buying KP281(50%) same as male(50%). KP281 is more recommended for both customers.

Probability of Male customer buying Product KP781(83) is way more than female(18%).

Probability of Female customer buying Product KP481(48%) is significantly higher than male(52%).

KP481 product is specifically recommended for Female customers who are intermediate user.

Objective: Customer Profiling for Each Product

Customer profiling based on the 3 product categories provided

KP281

• Easily affordable entry level product, which is also the maximum selling product.

• KP281 is the most popular product among the entry level customers.

• This product is easily afforded by both Male and Female customers.

• Average distance covered in this model is around 70 to 90 miles.

• Product is used 3 to 4 times a week.

• Most of the customer who have purchased the product have rated Average shape as the fitness rating.

• Younger to Elder beginner level customers prefer this product.

• Single female & Partnered male customers bought this product more than single male customers.

• Income range between 39K to 53K have preferred this product.

KP481

• This is an Intermediate level Product.

• KP481 is the second most popular product among the users.

• Fitness Level of this product users varies from Bad to Average Shape depending on their usage. Users Prefer this product mostly to cover more miles than fitness.

• Average distance covered in this product is from 70 to 130 miles per week.

• More Female users prefer this product than males.

• Probability of Female user buying KP481 is significantly higher than male.

• KP481 product is specifically recommended for Female users who are intermediate user.

• Three different age groups prefer this product - Teen, Adult and middle aged.

• Average Income of the user who buys KP481 is 49K.

• Average Usage of this product is 3 days per week.

• More Partnered users prefer this product.

• There are slightly more male buyers of the KP481.

• The distance travelled on the KP481 treadmill is roughly between 75 - 100 Miles. It is also the 2nd most distance travelled model.

• The buyers of KP481 in Single & Partnered, Male & Female are same.

• The age range of KP481 treadmill users is roughly between 24-34 years.

KP781

• Due to the High Price & being the advanced type, user prefers less of this product.

• Users use this product mainly to cover more distance.

• Users who use this product have rated excelled shape as fitness rating.

• User walk/run average 120 to 200 or more miles per week on his product.

• Users use 4 to 5 times a week at least.

• Female Users who are running average 180 miles (extensive exercise) , are using product KP781, which is higher than Male average using same product.

• Probability of Male user buying Product KP781(31.73%) is way more than female(9.21%).

• Probability of a single person buying KP781 is higher than Married users. So , KP781 is also recommended for people who are single and exercises more.

• Middle aged to higher age users tend to use this model to cover more distance.

• Average Income of KP781 buyers are over 75K per annum.

• Partnered Female bought KP781 treadmill compared to Partnered Male.

• Users who have more experience with previous aerofit products tend to buy this product

• This product is preferred by the user where the correlation between Education and Income is High.

Recommendation

1. There is a notable opportunity to encourage more female customers to engage in regular exercise, as their preference for these products appears to be relatively low. Implementing a targeted marketing campaign could effectively inspire and motivate women to adopt a healthier lifestyle.
2. The KP281 and KP481 treadmills seem to be favored by customers with an annual income between 39K - 53K Dollars. Promoting these models as budget-friendly options could attract more price-conscious consumers to consider these treadmills.
3. Given its extensive features and functionalities, the KP781 treadmill appears to be well-suited for professionals and athletes. Tailoring marketing efforts to target this specific audience would likely lead to increased interest and demand.
4. To promote the KP781 product effectively, collaborating with influencers and international athletes could significantly enhance its brand image and appeal to a wider audience.
5. Exploring opportunities to expand the market to customers beyond 50 years of age requires in-depth research to carefully evaluate the health benefits and potential drawbacks associated with treadmill usage in this age group.
6. Providing excellent customer support and encouraging users to upgrade from lower treadmill versions to more advanced models after consistent usage can enhance customer satisfaction and drive brand loyalty.
7. The KP781 treadmill could be recommended to female customers who engage in extensive exercise routines, accompanied by user-friendly guidance to facilitate ease of use for this advanced product.
8. Targeting customers above 40 years of age for KP781 promotion would be strategic, as this age group appears to be a potential key market segment for this product.