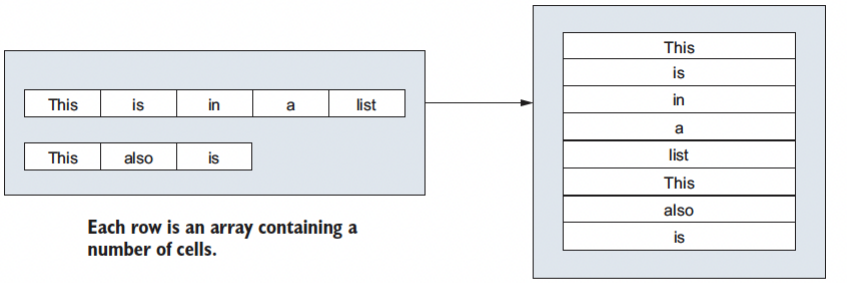
Business Case

Business Case: Aerofit - Descriptive Statistics & Probability

Suman Debnath



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.

Business Problem

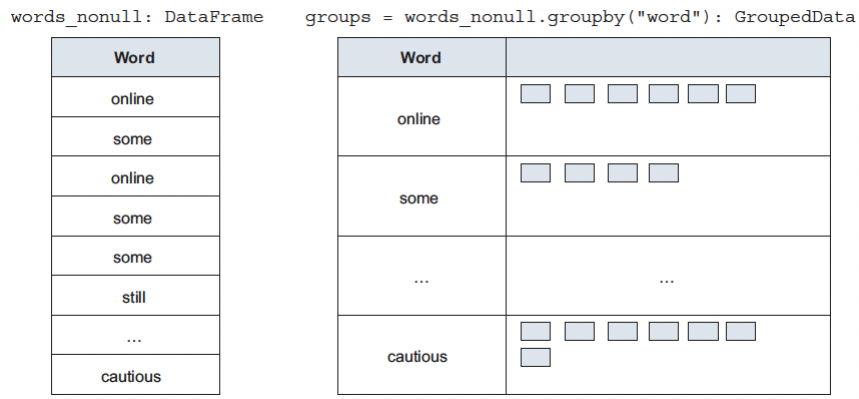
The market research team at AeroFit wants to identify the characteristics of the target audience for each type of treadmill offered by the company, to provide a better recommendation of the treadmills to the new customers. The team decides to investigate whether there are differences across the product with respect to customer characteristics.

1. 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

Link: Dataset_link

The dataset have the following fields:



Customer based profiling

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 customers.
- Fitness Level of this product users varies from Bad to Average Shape depending on their usage.

- Customers 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 customers prefer this product than males.
- Probability of Female customer buying KP481 is significantly higher than male.
- KP481 product is specifically recommended for Female customers who are intermediate user.
- Three different age groups prefer this product - Teen, Adult and middle aged.
- Average Income of the customer who buys KP481 is 49K.
- Average Usage of this product is 3 days per week.
- More Partnered customers 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 customers is roughly between 24-34 years.

KP781

- Due to the High Price & being the advanced type, customer prefers less of this product.
- Customers use this product mainly to cover more distance.
- Customers who use this product have rated excelled shape as fitness rating.
- Customer walk/run average 120 to 200 or more miles per week on his product.
- Customers use 4 to 5 times a week at least.
- Female Customers who are running average 180 miles (extensive exercise) , are using product KP781, which is higher than Male average using same product.
- Probability of Male customer buying Product KP781(31.73%) is way more than female(9.21%).
- Probability of a single person buying KP781 is higher than Married customers. So , KP781 is also recommended for people who are single and exercises more.
- Middle aged to higher age customers 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.
- Customers who have more experience with previous aerofit products tend to buy this product
- This product is preferred by the customer where the correlation between Education and Income is High.

Recommendation

- Female who prefer exercising equipments are very low here. Hence, we should run a marketing campaign on to encourage women to exercise more
- KP281 & KP481 treadmills are preferred by the customers whose annual income lies in the range of 39K - 53K Dollars. These models should promoted as budget treadmills.
- As KP781 provides more features and functionalities, the treadmill should be marketed for professionals and athletes.
- KP781 product should be promotted using influencers and other international atheletes.
- Research required for expanding market beyond 50 years of age considering health pros and cons.
- Provide customer support and recommend users to upgrade from lower versions to next level versions after consistent usages.
- KP781 can be recommended for Female customers who exercises extensively along with easy usage guidance since this type is advanced.
- Target the Age group above 40 years to recommend Product KP781.
- Education with 14 to 16 years have tendency to buy more of KP281 models

Conclusion

- KP 281 model is the most purchased model (44.4%) then KP 481 (33.3%). KP 781 is the least sold model (22.2%).
- There are more Male customers (57.8%) than Female customers (42.2%).
- Average Usage of Males is more than Average usage of Females.
- Customers buying treadmill are younger and average age of customer is 28.
- Most of the customers earns less than 70K and prefer KP 281 & KP 481 models.
- 59.4% of the customers who purchased treadmill are partnered.
- Customers average education is 16.

Future

- For KP281: Concentrate advertising broadly across gender and marital status towards individuals with annual income less than $75,000, with some college education or a bachelor's degree, who are unfit or average fitness and in their 20s or 30s.
- For KP481 : Concentrate advertising broadly across gender and marital status towards individuals with annual income less than $75,000, with some college education or a bachelor's degree, who are unfit or average fitness and in their 20s or 30s.
- For KP781: Concentrate advertising towards males who are average fitness to very fit, have a bachelors degree or advanced education, and are in their 20s or 30s.
- There may be untapped potential for targeting customers in the 40s and beyond age group, which appear to be an underserved population. Analysis indicates more than just outlying purchases of KP 781 .

- Individuals with only a high school education also appear to be an underserved population. Likely best candidates for KP 281 or KP 481 due to annual income constraints.

Detailed Analysis

Importing all the `libs`

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

Loading the `data`

In [2]:
```
# data_set = 'https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/125/original/aerofit_treadmill.csv'
data_set = 'aerofit_treadmill.csv'
```

Data Exploration

In [3]:
```
df = pd.read_csv(data_set)
```

In [4]:
```
df.shape
```

Out[4]:
```
(180, 9)
```

In [5]:
```
df.head()
```

Out[5]:

	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 [6]:
```
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
```

In [7]:
```
df.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

In [8]:
```
print(df.isnull().sum())
```

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

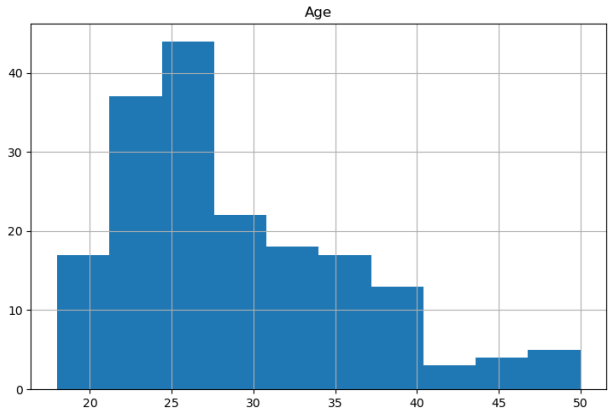
In [9]:
```
df['Product'].value_counts().reset_index()
```

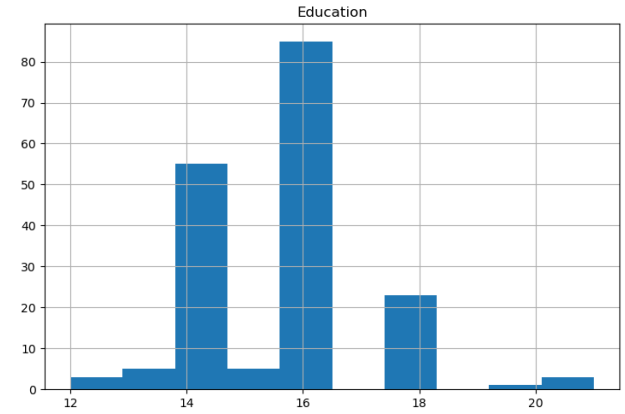
Out[9]:

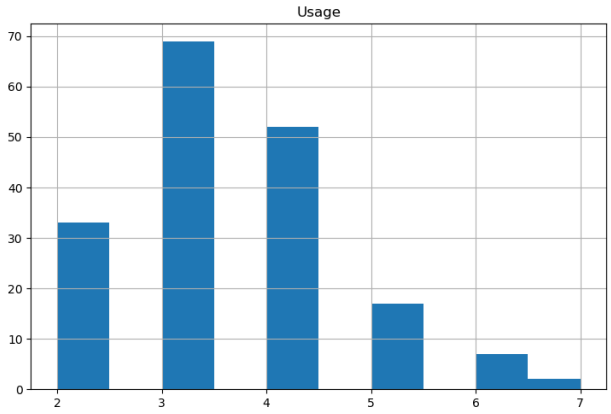
	index	Product
0	KP281	80
1	KP481	60
2	KP781	40

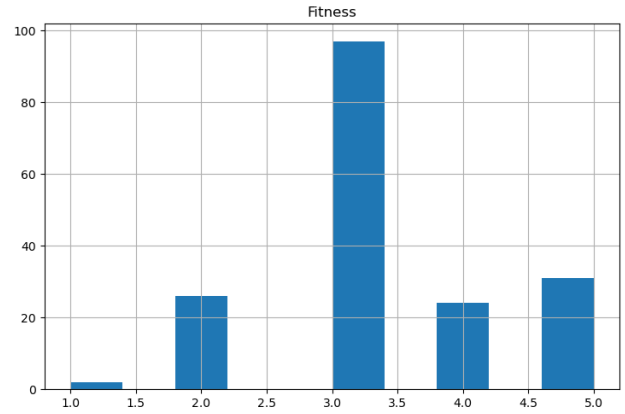
Histogram of all fields

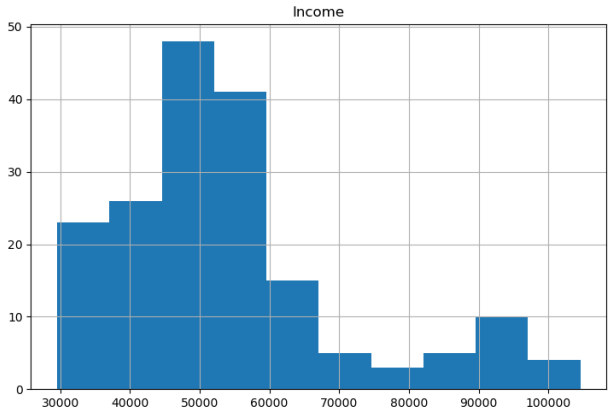
In [10]:
```
df.hist(figsize=(20,20));
```

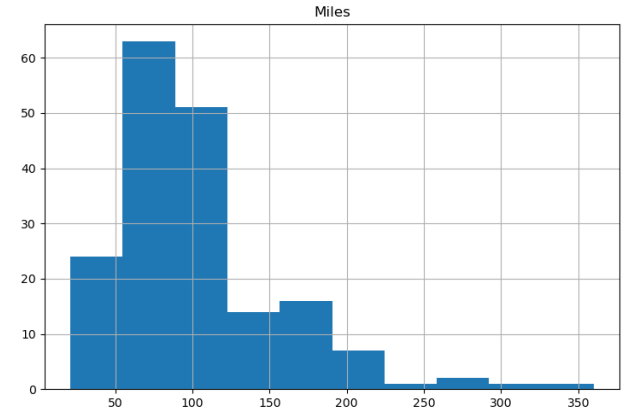




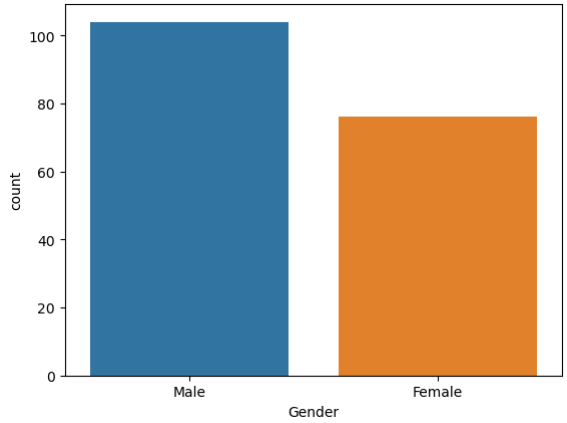




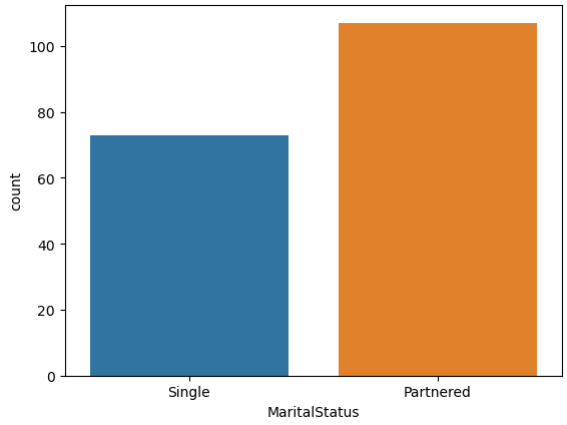




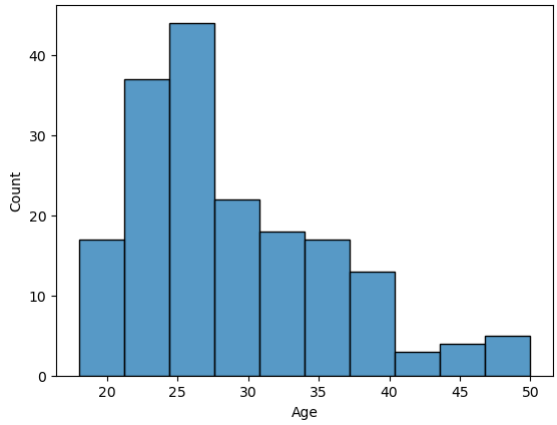
In [11]:
```
sns.countplot(x='Gender', data=df)
plt.show()
```



In [12]:
```
sns.countplot(x='MaritalStatus', data=df)
plt.show()
```

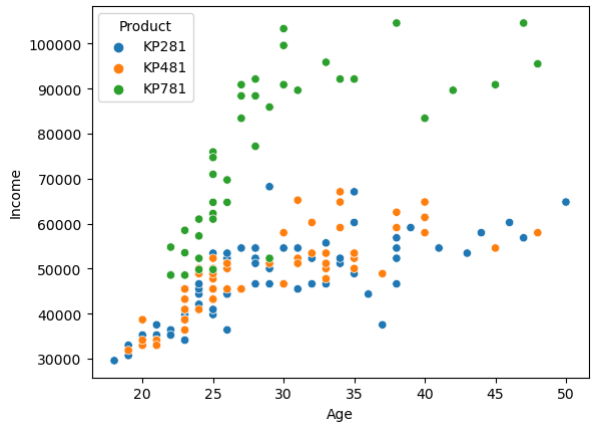


In [13]:
```
sns.histplot(df['Age'], bins=10)
plt.xlabel('Age')
plt.ylabel('Count')
plt.show()
```



Scatter Plot between `Age` and `Income`

In [14]:
```
sns.scatterplot(x='Age', y='Income',data=df, hue = 'Product')
plt.show()
```



In [ ]:

Observation

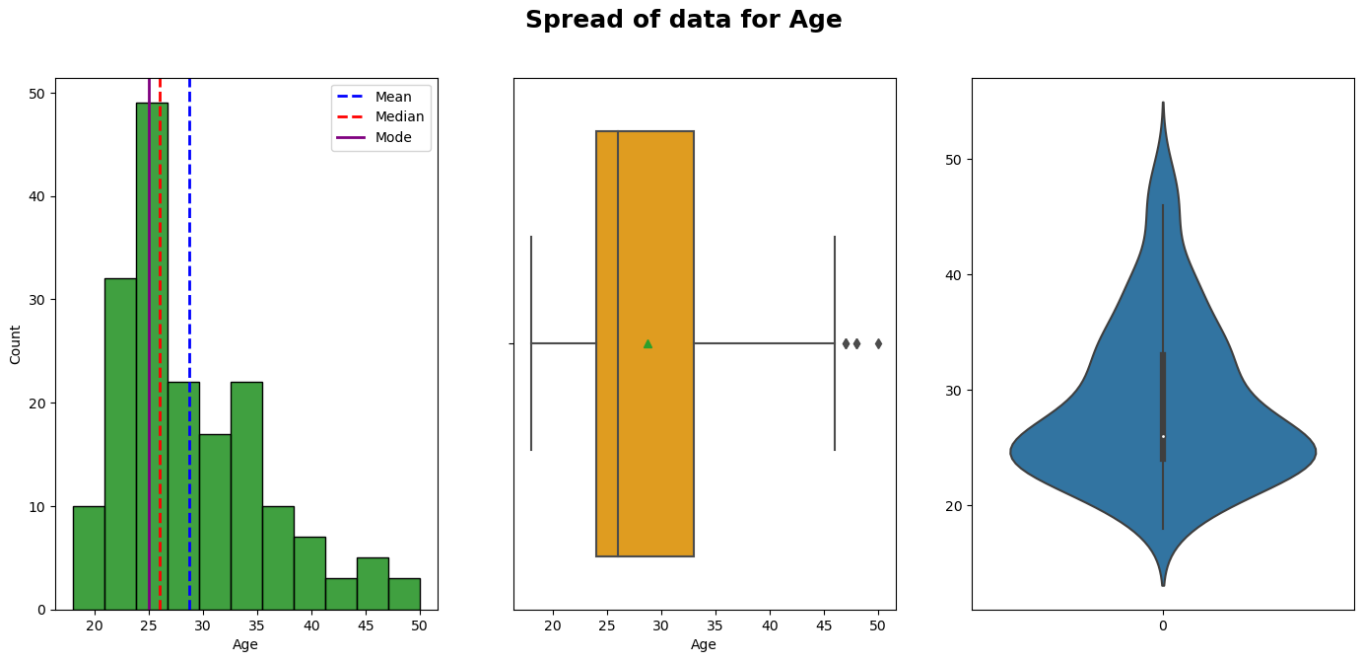
- No Null Value
- Small dataset of 180 data points
- Income has a wide spread
- The most popular product is `KP281`
- Age of customer using treadmill is between range 18-50
- Avg age is around 28 and median is 26
- Male/Female ratio is almost equally balanced (100/80)
- Age group between 22 to 35 buys the most
- Higher income people buys `KP781` most

Detect Outliers

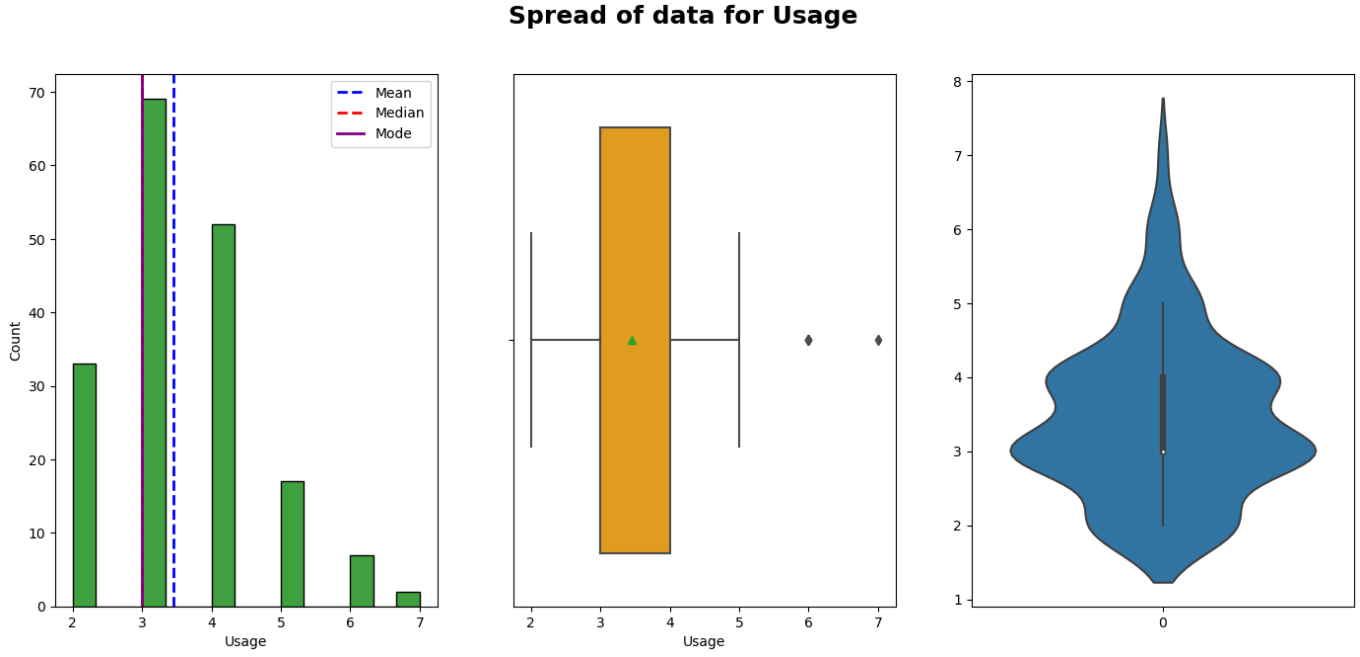
In [15]:
```
def dist_box_violin(data):
    Name = data.name.capitalize()
```

```
    fig, axes = plt.subplots(1, 3, figsize=(17, 7))
    fig.suptitle("Spread of data for " + Name, fontsize=18, fontweight='bold')

    # Histogram with mean, median, and mode
    sns.histplot(data, kde=False, color='green', ax=axes[0])
    axes[0].axvline(data.mean(), color='blue', linestyle='--', linewidth=2)
    axes[0].axvline(data.median(), color='red', linestyle='dashed', linewidth=2)
    axes[0].axvline(data.mode()[0], color='purple', linestyle='solid', linewidth=2)
    axes[0].legend({'Mean': data.mean(), 'Median': data.median(), 'Mode': data.mode()})

    # Box plot
    sns.boxplot(x=data, showmeans=True, orient='h', color="orange", ax=axes[1])

    # Violin plot
    sns.violinplot(data, ax=axes[2], showmeans=True)
```

In [16]:
```
dist_box_violin(df['Age'])
```

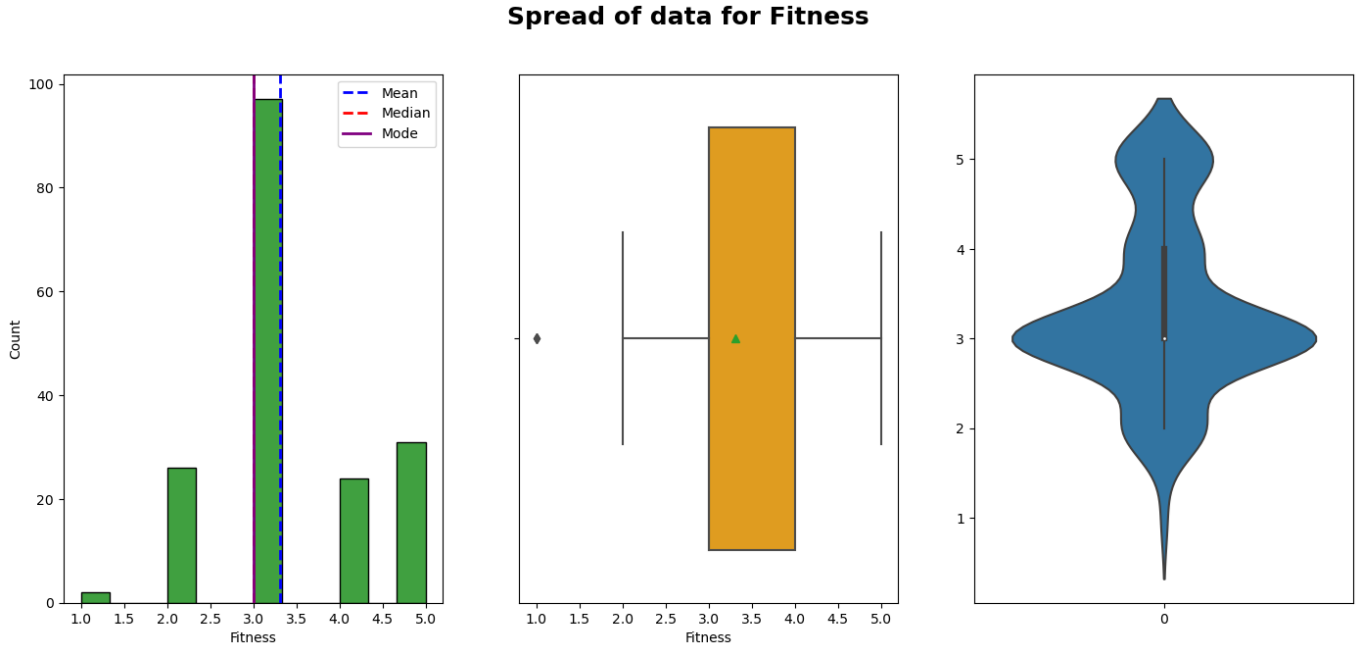


In [17]:
```
dist_box_violin(df['Usage'])
```

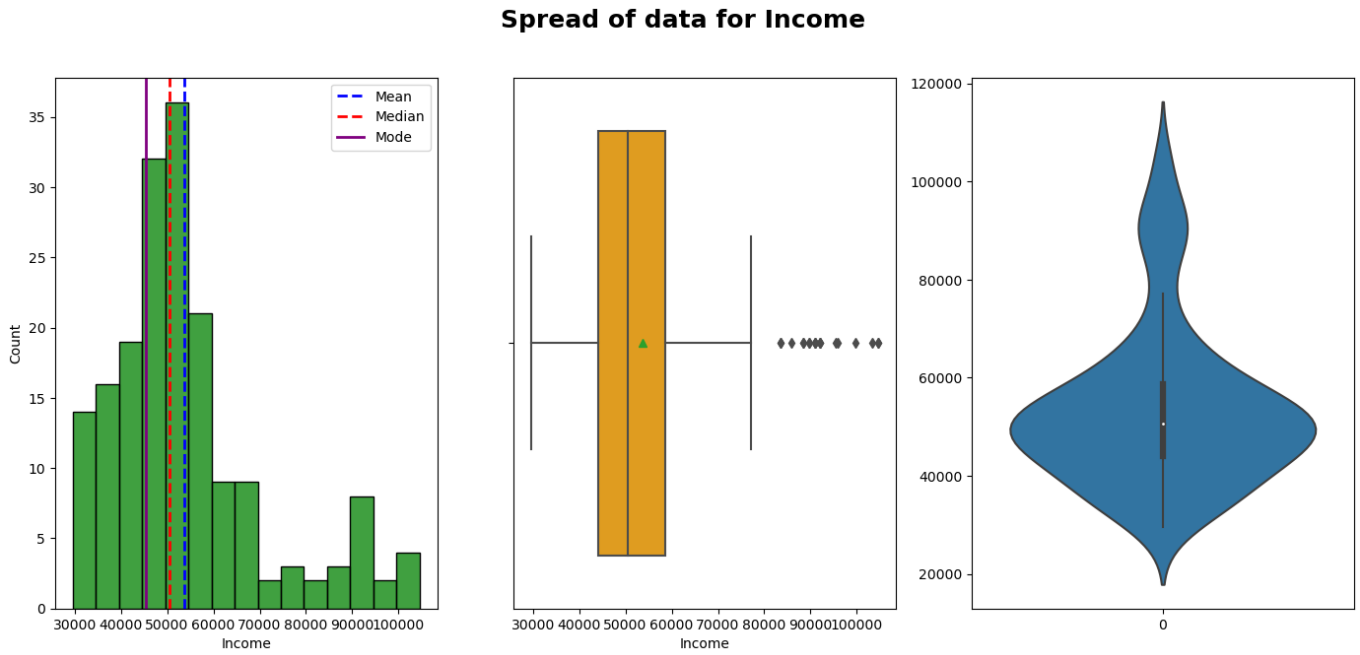


Fitness

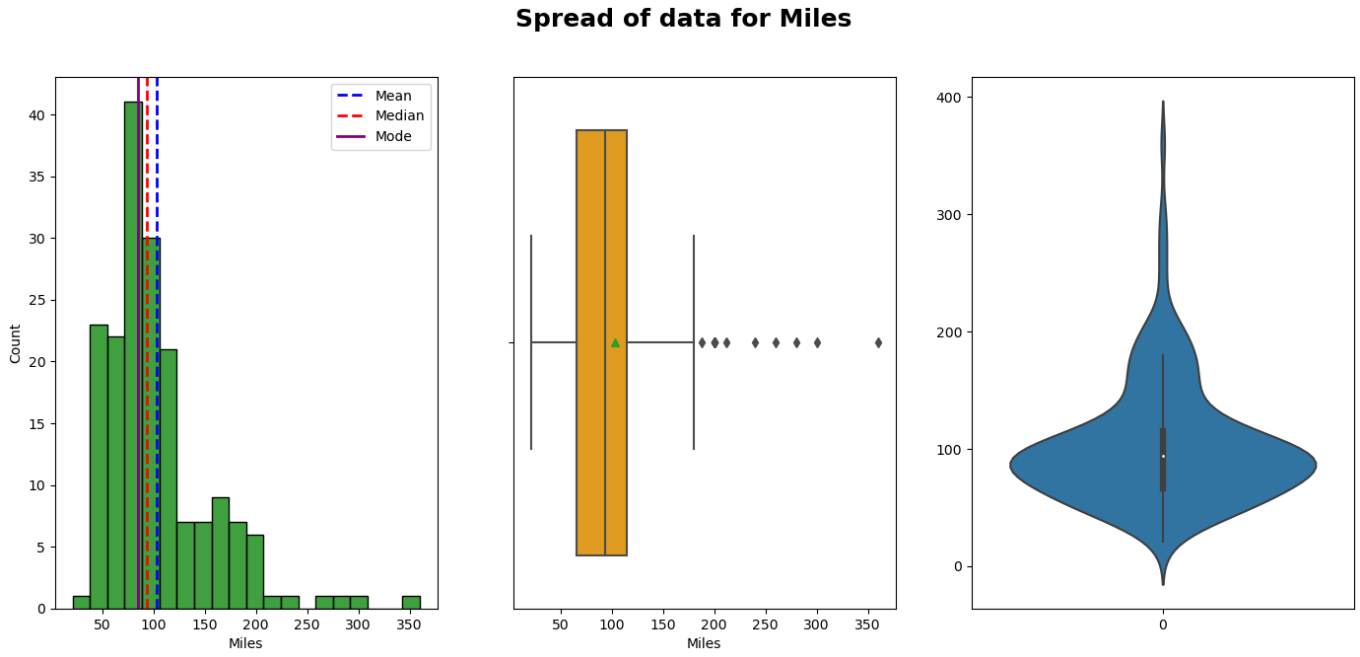
In [18]:
```
dist_box_violin(df['Fitness'])
```



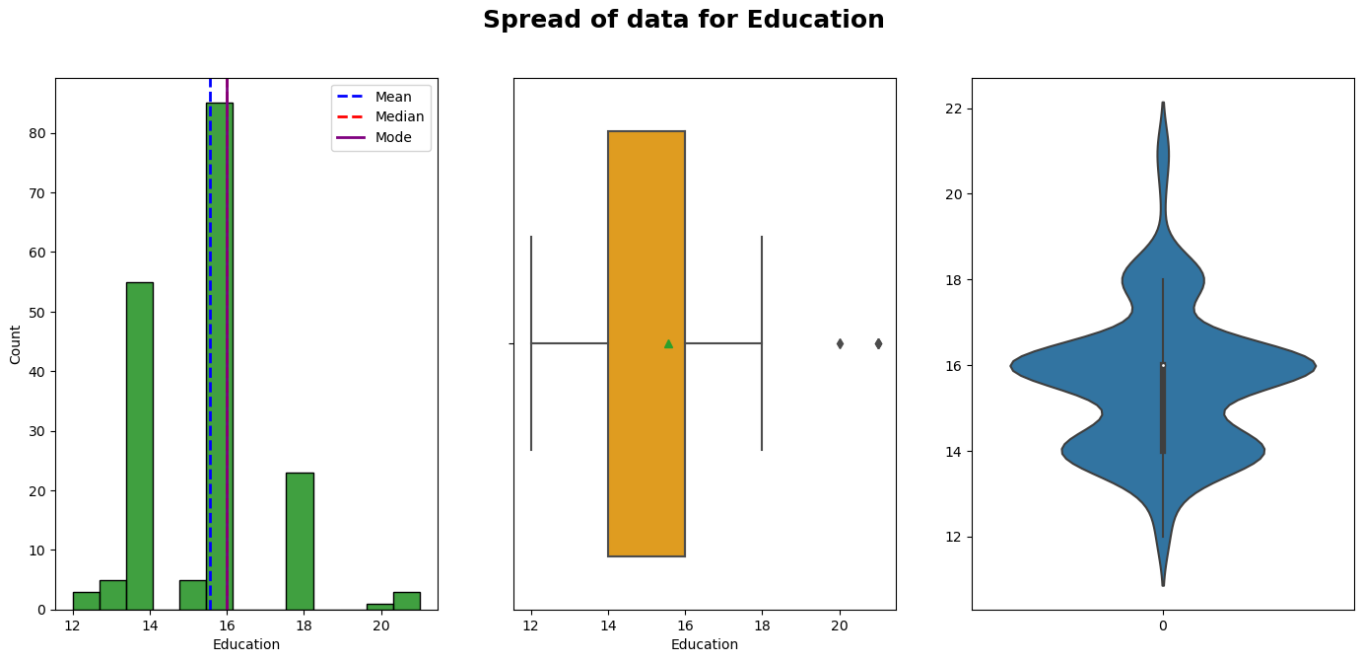
In [19]:
```
dist_box_violin(df['Income'])
```



In [20]:
```
dist_box_violin(df['Miles'])
```



In [21]:
```
dist_box_violin(df['Education'])
```



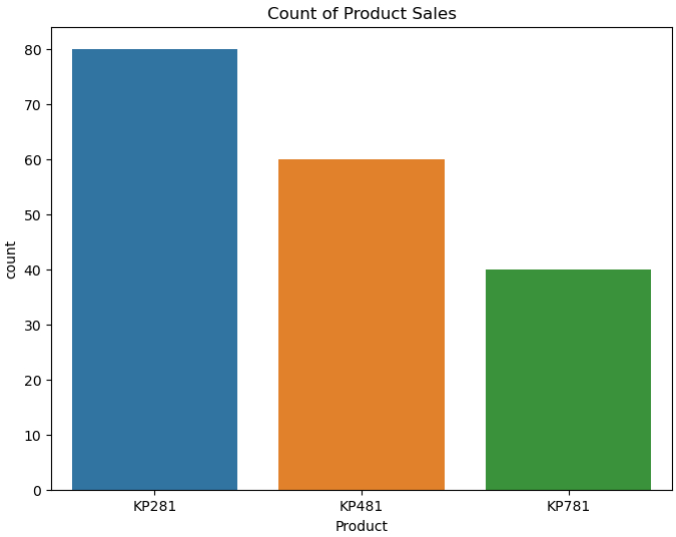
Observation

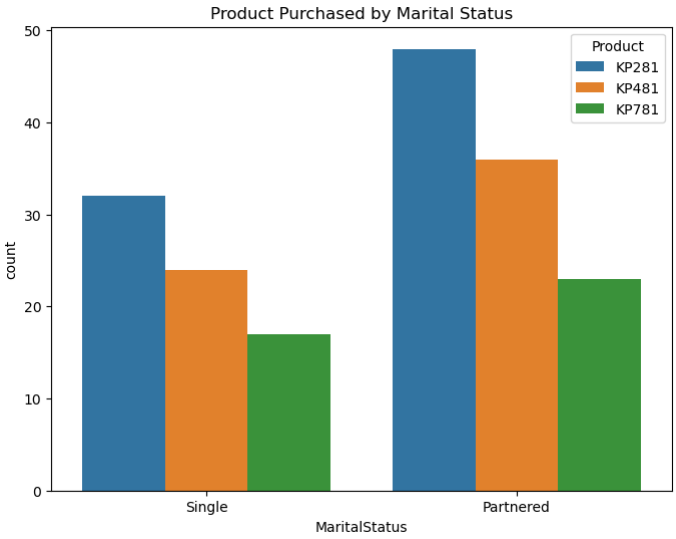
- More than 90 customers have rated their physical fitness rating as Average
- Most of customers who have purchased the product have a average income between 40K to 60K
- Highest number of customers have 16 as their Education
- 14 is the second highest education among the customers
- 20 is the least education among the customers
- 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
- 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)
- Few customers have income above 80K per annum(Outliers)
- Most customers earn from 45K to around 60K per annum

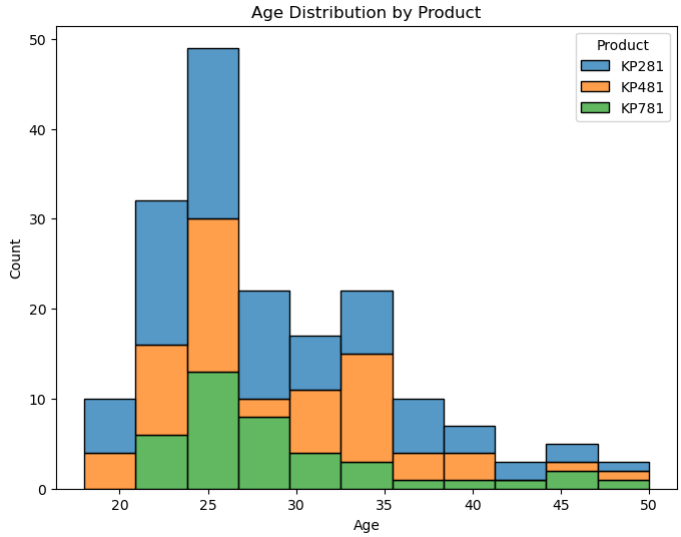
Univariate Catagorical Analysis

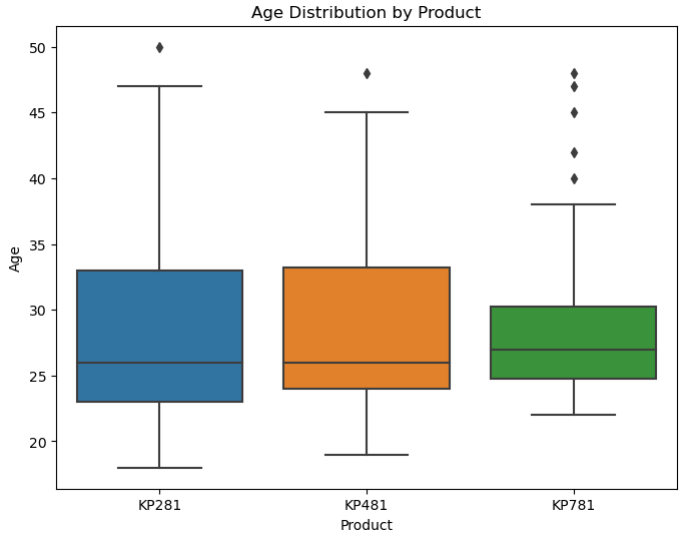
In [22]:
```
# Countplot of sales of differnt products
plt.figure(figsize=(8, 6))
sns.countplot(data=df, x='Product')
plt.title("Count of Product Sales")
plt.show()

# Countplot of MaritalStatus and Product
plt.figure(figsize=(8, 6))
sns.countplot(x='MaritalStatus', hue='Product', data=df)
plt.title('Product Purchased by Marital Status')
```

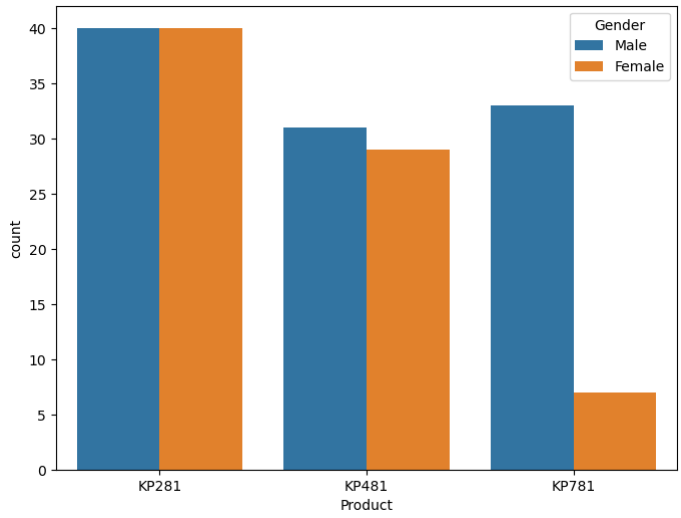
```
plt.show()

# Histogram of Age by Product
plt.figure(figsize=(8, 6))
sns.histplot(data=df, x='Age', hue='Product', multiple='stack')
plt.title('Age Distribution by Product')
plt.show()

# Boxplot of Age by Product
plt.figure(figsize=(8, 6))
sns.boxplot(x='Product', y='Age', data=df)
plt.title('Age Distribution by Product')
plt.show()


# Countplot of purchased among Male and Female
plt.figure(figsize=(8, 6))
sns.countplot(x='Product',hue='Gender', data=df)
plt.show()


# Countplot of product usage among Gender
plt.figure(figsize=(8,6))
sns.countplot(x='Usage',hue='Gender', data=df, palette='cool')
plt.show()
```

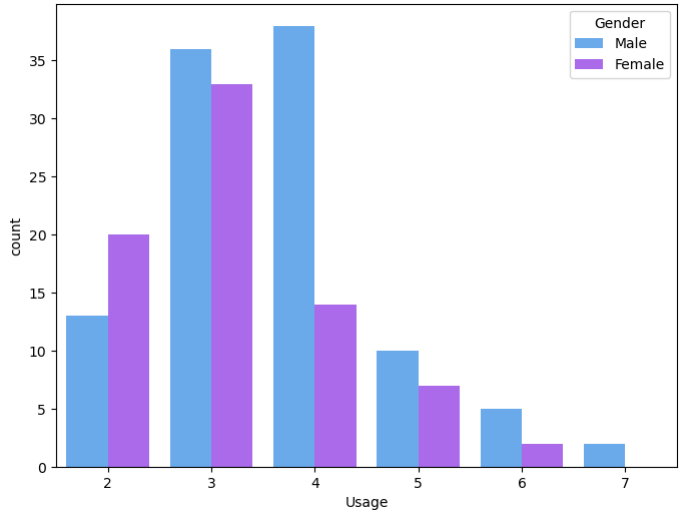





Age Distribution by Product
Product
KP281
KP481
KP781
Count
Age
0
10
20
30
40
50
20
25
30
35
40
45
50


Age Distribution by Product
Age
Product
KP281
KP481
KP781
20
25
30
35
40
45
50


Gender
Male
Female
count
Product
KP281
KP481
KP781
0
5
10
15
20
25
30
35
40



Observation

- KP281 is the most commonly purchase product type
- KP481 is the second most top product type purchased
- KP781 is the least purchased product type
- Most products purchased by Married/Partnered customer category
- Overall Male customers are the highest product purchasers
- 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

Marginal probability using `crosstab`

In [23]:
```
def get_crosstab_marg_prob(df, col1, col2, bins=None):

    # For 'Income' and 'Age' column
    if bins:
        _col2 = col2 + 'Group'
        df[_col2] = pd.cut(df[col2], bins=bins)
        col2 = _col2

    # Calculate the Crosstab
    crosstab = pd.crosstab(df[col1], df[col2])

    # Marginal probabilities
    marginal_prob = crosstab / crosstab.sum()

    # Printing the table
    crosstab_styled = crosstab.style.background_gradient(cmap='Greens')
    marginal_prob_styled = marginal_prob.style.background_gradient(cmap='Reds', axis=None)

    display(crosstab_styled)
    display(marginal_prob_styled)
```

1) `Product` and `MaritalStatus`

In [24]:
```
get_crosstab_marg_prob(df, 'Product', 'MaritalStatus')
```

MaritalStatus	Partnered	Single
Product		
KP281	48	32
KP481	36	24
KP781	23	17

MaritalStatus	Partnered	Single
Product		
KP281	0.448598	0.438356
KP481	0.336449	0.328767
KP781	0.214953	0.232877

2) `Product` and `Gender`

In [25]:
```
get_crosstab_marg_prob(df, 'Product', 'Gender')
```

Gender	Female	Male
Product		
KP281	40	40
KP481	29	31
KP781	7	33

Gender	Female	Male
Product		
KP281	0.526316	0.384615
KP481	0.381579	0.298077
KP781	0.092105	0.317308

3) `Product` and `Education`

In [26]:
```
get_crosstab_marg_prob(df, 'Product', 'Education')
```

Education	12	13	14	15	16	18	20	21
Product								
KP281	2	3	30	4	39	2	0	0
KP481	1	2	23	1	31	2	0	0
KP781	0	0	2	0	15	19	1	3

Education	12	13	14	15	16	18	20	21
Product								
KP281	0.666667	0.600000	0.545455	0.800000	0.458824	0.086957	0.000000	0.000000
KP481	0.333333	0.400000	0.418182	0.200000	0.364706	0.086957	0.000000	0.000000
KP781	0.000000	0.000000	0.036364	0.000000	0.176471	0.826087	1.000000	1.000000

4) `Product` and `Age`

In [27]:
```
bins = range(10, 150, 10)
get_crosstab_marg_prob(df, 'Product', 'Age', bins=bins)
```

AgeGroup	(10, 20]	(20, 30]	(30, 40]	(40, 50]
Product				
KP281	6	49	19	6
KP481	4	31	23	2
KP781	0	30	6	4

AgeGroup	(10, 20]	(20, 30]	(30, 40]	(40, 50]
Product				
KP281	0.600000	0.445455	0.395833	0.500000
KP481	0.400000	0.281818	0.479167	0.166667
KP781	0.000000	0.272727	0.125000	0.333333

5) `Product` and `Fitness`

In [28]:
```
get_crosstab_marg_prob(df, 'Product', 'Fitness')
```

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

Fitness	1	2	3	4	5
Product					
KP281	0.500000	0.538462	0.556701	0.375000	0.064516
KP481	0.500000	0.461538	0.402062	0.333333	0.000000
KP781	0.000000	0.000000	0.041237	0.291667	0.935484

6) `Product` and `Usage`

In [29]:
```
get_crosstab_marg_prob(df, 'Product', 'Usage')
```

Usage	2	3	4	5	6	7
Product						
KP281	19	37	22	2	0	0
KP481	14	31	12	3	0	0
KP781	0	1	18	12	7	2

Usage	2	3	4	5	6	7
Product						
KP281	0.575758	0.536232	0.423077	0.117647	0.000000	0.000000
KP481	0.424242	0.449275	0.230769	0.176471	0.000000	0.000000
KP781	0.000000	0.014493	0.346154	0.705882	1.000000	1.000000

7) `Product` and `Income`

In [30]:
```
bins = [0, 30000, 50000, 80000, float('inf')]
get_crosstab_marg_prob(df, 'Product', 'Income', bins=bins)
```

IncomeGroup	(0.0, 30000.0]	(30000.0, 50000.0]	(50000.0, 80000.0]	(80000.0, inf]
Product				
KP281	1	47	32	0
KP481	0	30	30	0
KP781	0	5	16	19

IncomeGroup	(0.0, 30000.0]	(30000.0, 50000.0]	(50000.0, 80000.0]	(80000.0, inf]
Product				
KP281	1.000000	0.573171	0.410256	0.000000
KP481	0.000000	0.365854	0.384615	0.000000
KP781	0.000000	0.060976	0.205128	1.000000

Observation

- The lower end model has the highest number of female customers the higher end model has the highest number of male customers.
- Most products purchased by Males, females are less interested in the product compared to Males
- KP281 Product is the equally preferred by both male and female genders
- KP781 Product is mostly preferred among the Male customers

Check correlation using `heat maps` or `pair plots`

1) Heat Map

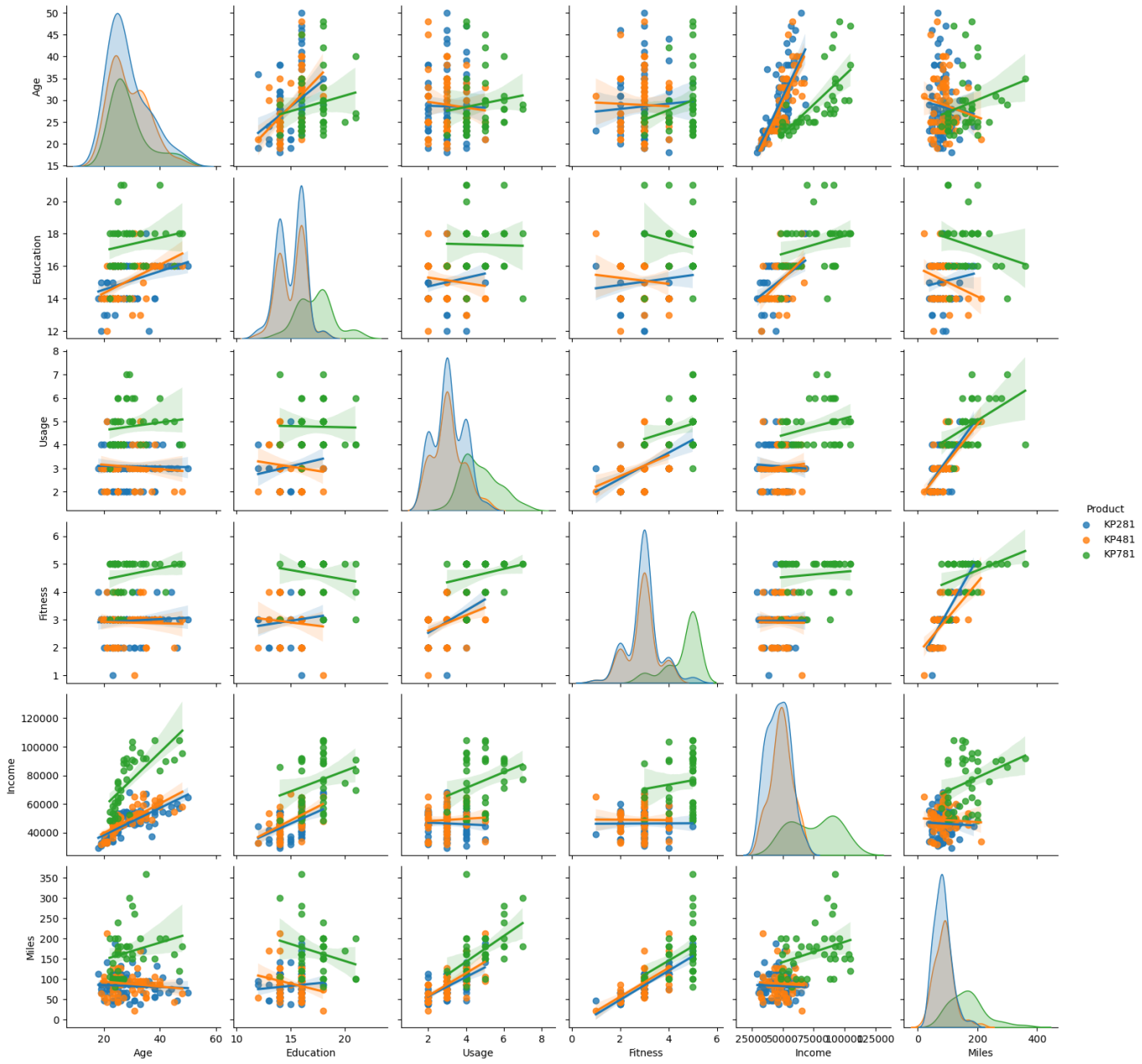
In [31]:
```
numeric_columns = df.select_dtypes(include='number')

correlation_matrix = numeric_columns.corr()

plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```

Observation

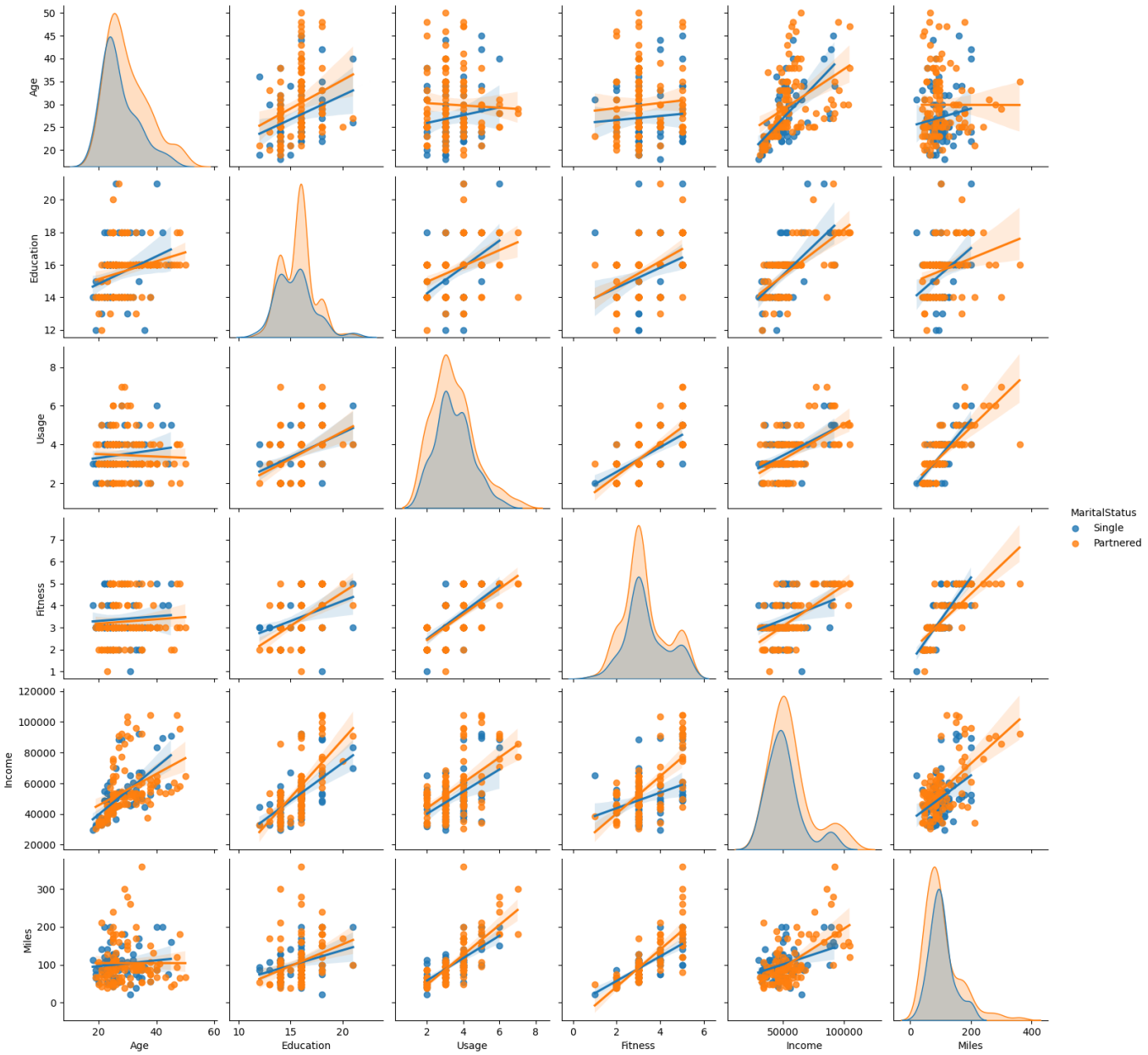
- Income and education show a great reason of buying
- Usage and fitness has the great correlation among the reason for buying
- Miles runned and fitness levels have 79 percent correlation.

2) Pair Plot

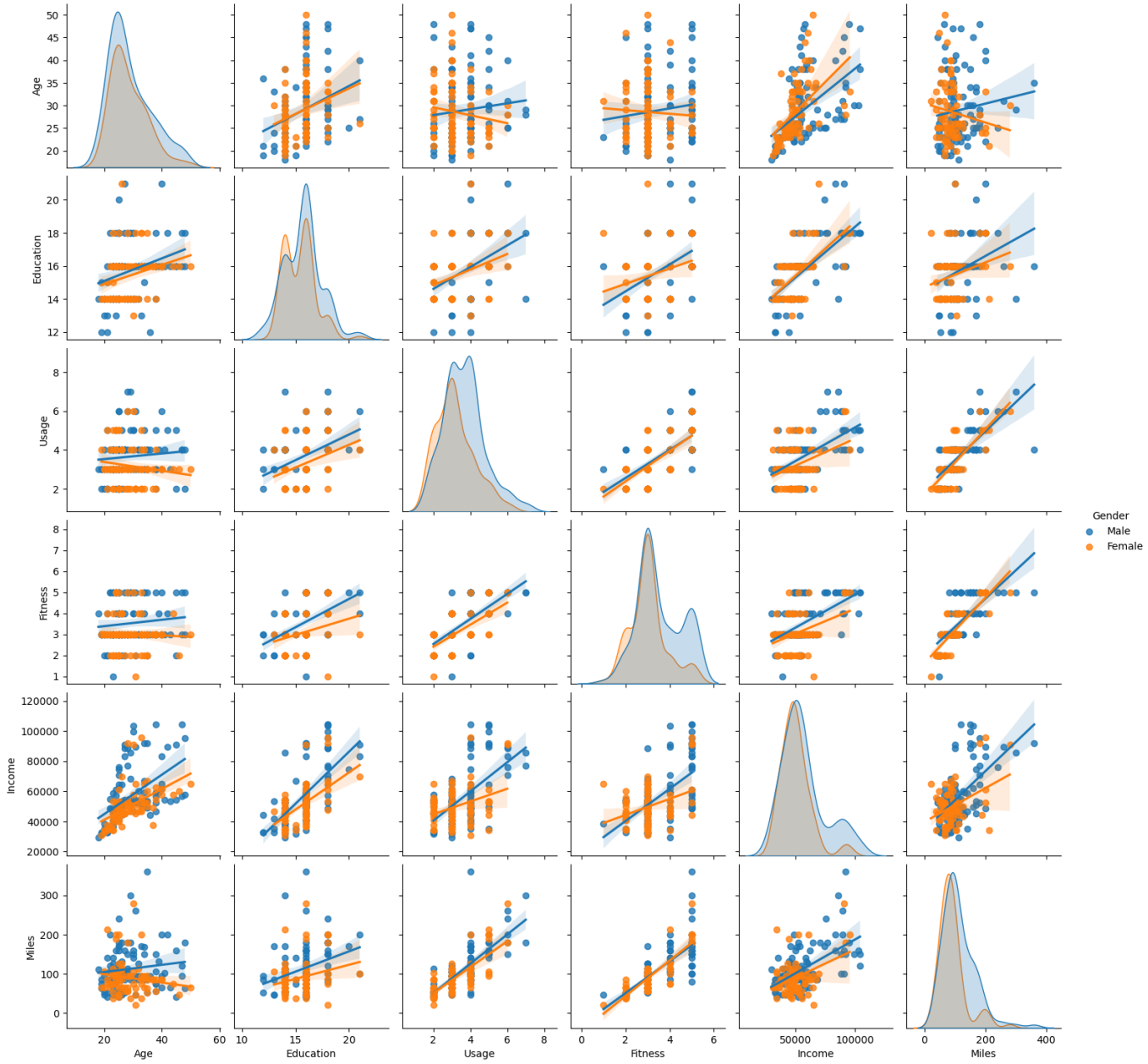
In [32]:
```
sns.pairplot(df,hue='Product',kind='reg')
plt.show()
```



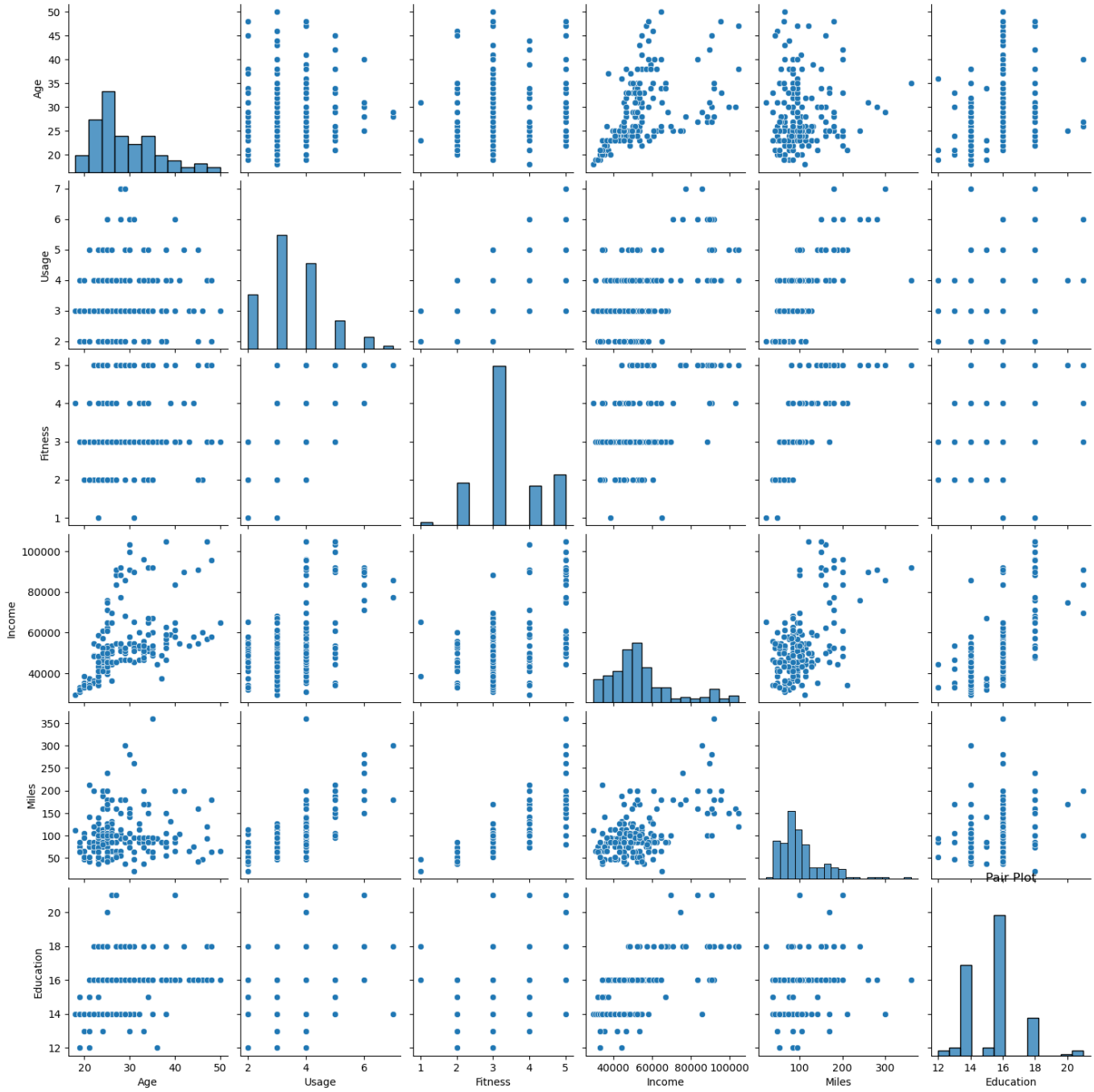
In [33]:
```
sns.pairplot(df,hue='MaritalStatus',kind='reg')
plt.show()
```



In [34]:
```
sns.pairplot(df,hue='Gender',kind='reg')
plt.show()
```



In [35]:
```
columns = ['Age', 'Usage', 'Fitness', 'Income', 'Miles', 'Education']

sns.pairplot(df[columns])
plt.title('Pair Plot')
plt.show()
```



Probability based inference

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

In [36]:
```
get_crosstab_marg_prob(df, 'Product', 'Gender')
```

Gender	Female	Male
Product		
KP281	40	40
KP481	29	31
KP781	7	33

Gender	Female	Male
Product		
KP281	0.526316	0.384615
KP481	0.381579	0.298077
KP781	0.092105	0.317308

In [37]:
```
crosstab = pd.crosstab(df['Product'], df['Gender'])
marginal_prob = crosstab / crosstab.sum()

prob_male_kp781 = marginal_prob.loc['KP781', 'Male']
print("Probability of a male customer buying KP781:", prob_male_kp781)
```

```
Probability of a male customer buying KP781: 0.3173076923076923
```

Probability for each product for the both genders

In [38]:
```
def gender_Probability(gender,df):
    print(f"Prob P(KP781) for {gender}: {round(df['KP781'][gender]/df.loc[gender].sum(),3)}")
    print(f"Prob P(KP481) for {gender}: {round(df['KP481'][gender]/df.loc[gender].sum(),3)}")
    print(f"Prob P(KP281) for {gender}: {round(df['KP281'][gender]/df.loc[gender].sum(),3)}")

df_temp = pd.crosstab(index=df['Gender'],columns=[df['Product']])
print("Prob of Male: ",round(df_temp.loc['Male'].sum()/len(df),3))
print("Prob of Female: ",round(df_temp.loc['Female'].sum()/len(df),3))
print()
gender_Probability('Male',df_temp)
print()
gender_Probability('Female',df_temp)
```

```
Prob of Male:  0.578
Prob of Female:  0.422

Prob P(KP781) for Male: 0.317
Prob P(KP481) for Male: 0.298
Prob P(KP281) for Male: 0.385

Prob P(KP781) for Female: 0.092
Prob P(KP481) for Female: 0.382
Prob P(KP281) for Female: 0.526
```

Probability of each product for given Marital Status

In [39]:
```
def ms_probability(ms_status,df):
    print(f"Prob P(KP781) for {ms_status}: {round(df['KP781'][ms_status]/df.loc[ms_status].sum(),3)}")
    print(f"Prob P(KP481) for {ms_status}: {round(df['KP481'][ms_status]/df.loc[ms_status].sum(),3)}")
    print(f"Prob P(KP281) for {ms_status}: {round(df['KP281'][ms_status]/df.loc[ms_status].sum(),3)}")

df_temp = pd.crosstab(index=df['MaritalStatus'],columns=[df['Product']])
print("Prob of P(Single): ",round(df_temp.loc['Single'].sum()/len(df),3))
print("Prob of P(Married/Partnered): ",round(df_temp.loc['Partnered'].sum()/len(df),3))
print()
ms_probability('Single',df_temp)
print()
ms_probability('Partnered',df_temp)
```

```
Prob of P(Single):  0.406
Prob of P(Married/Partnered):  0.594

Prob P(KP781) for Single: 0.233
Prob P(KP481) for Single: 0.329
Prob P(KP281) for Single: 0.438

Prob P(KP781) for Partnered: 0.215
Prob P(KP481) for Partnered: 0.336
Prob P(KP281) for Partnered: 0.449
```

In [ ]: