Task

Analyze the "aerofit_treadmill.csv" dataset to create customer profiles for each treadmill product, identify characteristics influencing purchases, and compute conditional and marginal probabilities from contingency tables to provide insights for the AeroFit market research team.

Data loading and initial exploration

Subtask:

Load the `aerofit_treadmill.csv` dataset into a pandas DataFrame. Display the first few rows, check the data types and summary statistics to understand the structure and characteristics of the dataset.

Reasoning: The first step is to load the data into a pandas DataFrame and perform initial data exploration.

In [34]:

```python
import pandas as pd

# Read the CSV file into a pandas DataFrame
df = pd.read_csv('/content/aerofit_treadmill.csv')

# Display the first 5 rows
print("First 5 rows of the DataFrame:")
display(df.head())

# Display data types of each column
print("\nData types of each column:")
display(df.info())

# Generate descriptive statistics for numerical columns
print("\nDescriptive statistics for numerical columns:")
display(df.describe())
```

```
First 5 rows of the DataFrame:
```

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Mile
0	KP281	18	Male	14	Single	3	4	29562	11
1	KP281	19	Male	15	Single	2	3	31836	7
2	KP281	19	Female	14	Partnered	4	3	30699	6
3	KP281	19	Male	12	Single	3	3	32973	8
4	KP281	20	Male	13	Partnered	4	2	35247	4

```
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
None
Descriptive statistics for numerical columns:
```

	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

Descriptive analytics and outlier detection

Subtask:

Perform descriptive statistics on the numerical features. Use visualizations like boxplots to detect outliers. Compare the mean and median of numerical features to identify potential skewness.

Reasoning: Generate descriptive statistics for numerical columns and create boxplots for each numerical column to identify potential outliers and examine the distribution.

In [35]:

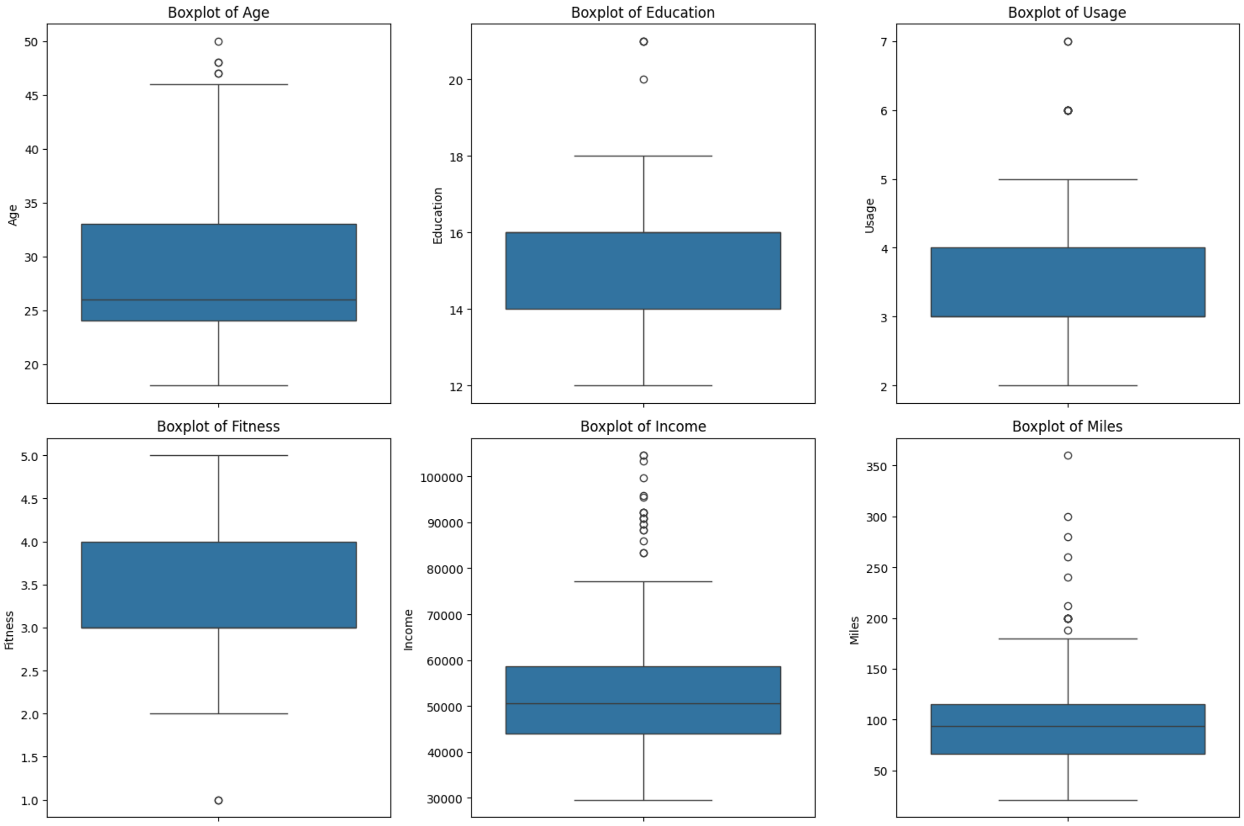
```python
import matplotlib.pyplot as plt
import seaborn as sns
```

```python
# Generate descriptive statistics for numerical columns
print("\nDescriptive statistics for numerical columns:")
display(df.describe())

# Create boxplots for each numerical column
numerical_cols = ['Age', 'Education', 'Usage', 'Fitness', 'Income', 'Miles']
plt.figure(figsize=(15, 10))
for i, col in enumerate(numerical_cols):
    plt.subplot(2, 3, i + 1)
    sns.boxplot(y=df[col])
    plt.title(f'Boxplot of {col}')
plt.tight_layout()
plt.show()
```

```
Descriptive statistics for numerical columns:
```

	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



Customer profile by product

Subtask:

Group the data by 'Product' and calculate descriptive statistics for each product to understand the typical customer profile for each treadmill model.

Reasoning: Group the data by 'Product' and calculate descriptive statistics for each product to understand the typical customer profile for each treadmill model.

In [36]:

```
# Group data by Product and calculate descriptive statistics
product_profiles = df.groupby('Product').describe()

# Print the descriptive statistics for each product
print("Descriptive statistics for each product:")
display(product_profiles)
```

```
Descriptive statistics for each product:
```

	Age								Education	
	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**	**count**	**mean**
Product										
KP281	80.0	28.55	7.221452	18.0	23.00	26.0	33.00	50.0	80.0	15.037500
KP481	60.0	28.90	6.645248	19.0	24.00	26.0	33.25	48.0	60.0	15.116667
KP781	40.0	29.10	6.971738	22.0	24.75	27.0	30.25	48.0	40.0	17.325000

3 rows × 48 columns

Analyze feature impact on product purchase

Subtask:

Use visualizations such as countplots, histplots, and boxplots to explore the relationship between customer characteristics (Age, Gender, Education, MaritalStatus, Usage, Income, Fitness, Miles) and the purchased product.

Reasoning: Create countplots to visualize the distribution of 'Gender' and 'MaritalStatus' across different 'Product' categories as requested in step 1.

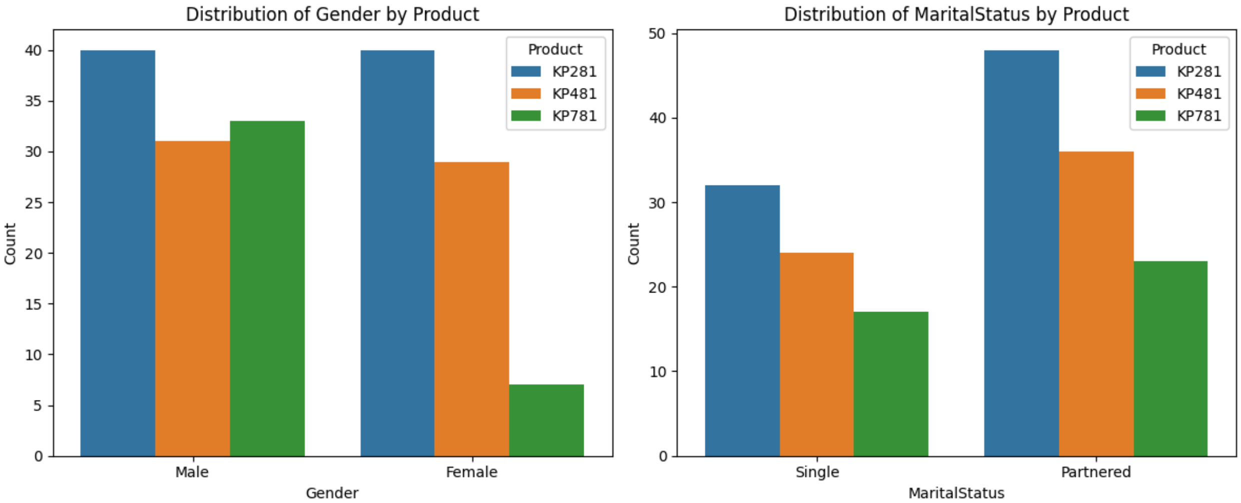
In [37]:

```
import matplotlib.pyplot as plt
import seaborn as sns

# Create countplots for Gender and MaritalStatus by Product
categorical_cols = ['Gender', 'MaritalStatus']
plt.figure(figsize=(12, 5))

for i, col in enumerate(categorical_cols):
    plt.subplot(1, 2, i + 1)
    sns.countplot(data=df, x=col, hue='Product')
    plt.title(f'Distribution of {col} by Product')
    plt.xlabel(col)
    plt.ylabel('Count')

plt.tight_layout()
plt.show()
```



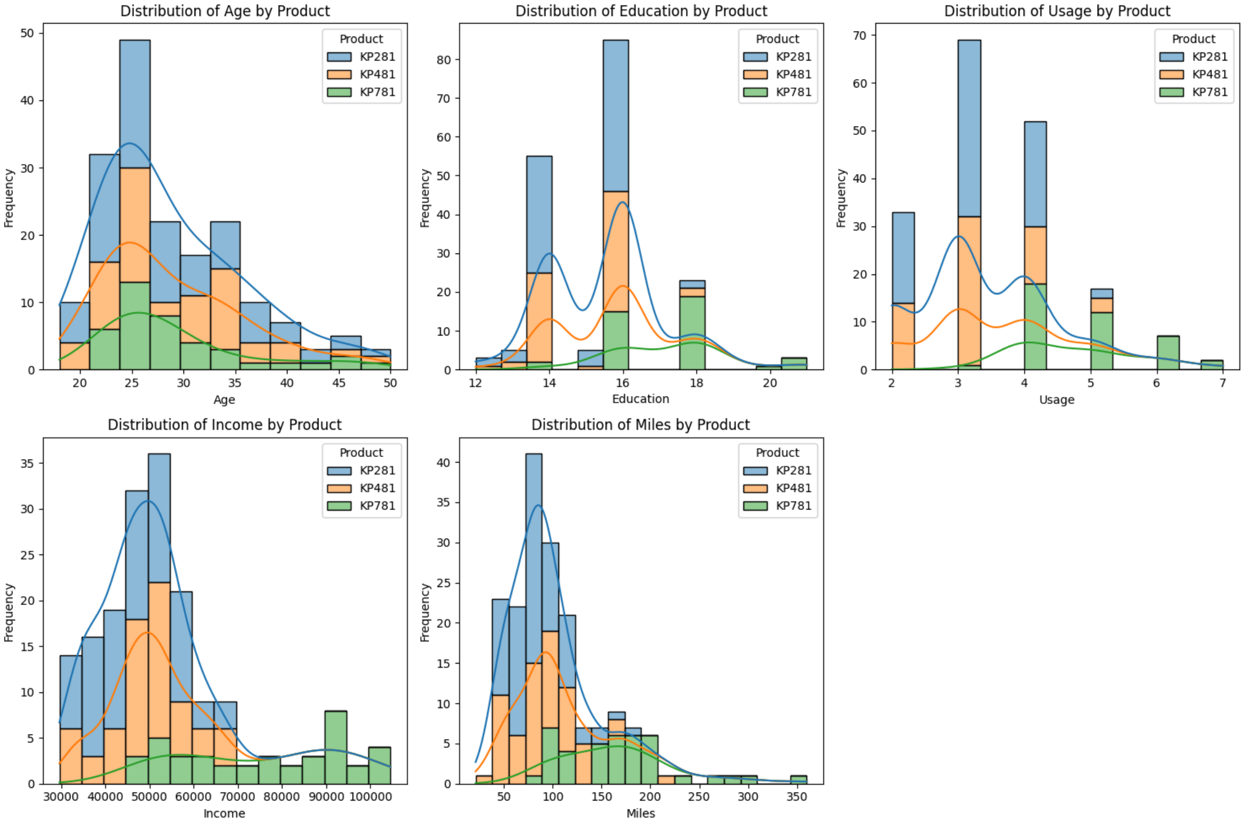
Reasoning: Create histplots to visualize the distribution of numerical features ('Age', 'Education', 'Usage', 'Income', 'Miles') for each 'Product' as requested in step 2.

In [38]:

```python
# Create histplots for numerical features by Product
numerical_cols = ['Age', 'Education', 'Usage', 'Income', 'Miles']
plt.figure(figsize=(15, 10))

for i, col in enumerate(numerical_cols):
    plt.subplot(2, 3, i + 1)
    sns.histplot(data=df, x=col, hue='Product', multiple='stack', kde=True)
    plt.title(f'Distribution of {col} by Product')
    plt.xlabel(col)
    plt.ylabel('Frequency')

plt.tight_layout()
plt.show()
```



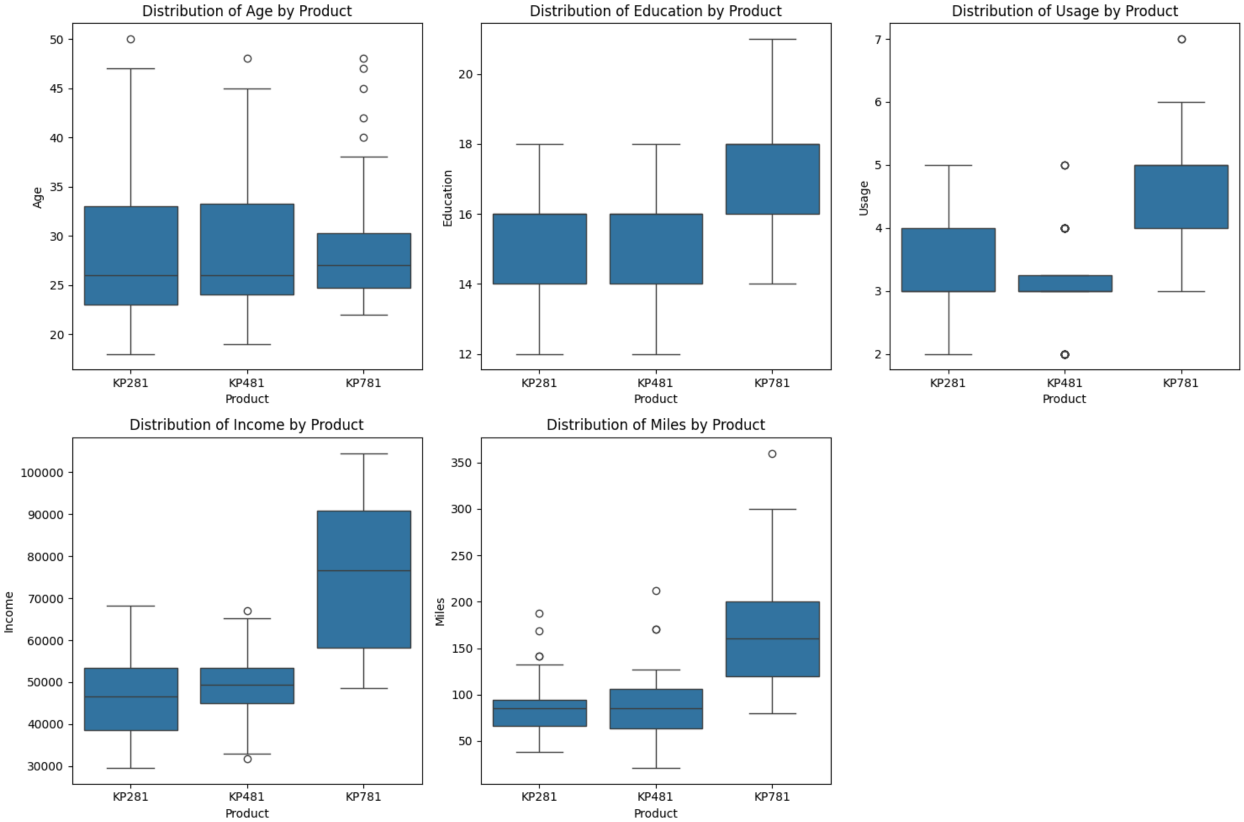
Reasoning: Create boxplots to visualize the distribution of numerical features ('Age', 'Education', 'Usage', 'Income', 'Miles') across different 'Product' categories to identify differences in central tendency and spread, as requested in step 3.

In [39]:

```python
# Create boxplots for numerical features by Product
numerical_cols = ['Age', 'Education', 'Usage', 'Income', 'Miles']
plt.figure(figsize=(15, 10))

for i, col in enumerate(numerical_cols):
    plt.subplot(2, 3, i + 1)
    sns.boxplot(data=df, x='Product', y=col)
    plt.title(f'Distribution of {col} by Product')
    plt.xlabel('Product')
    plt.ylabel(col)

plt.tight_layout()
plt.show()
```


Distribution of Age by Product

Distribution of Education by Product

Distribution of Usage by Product

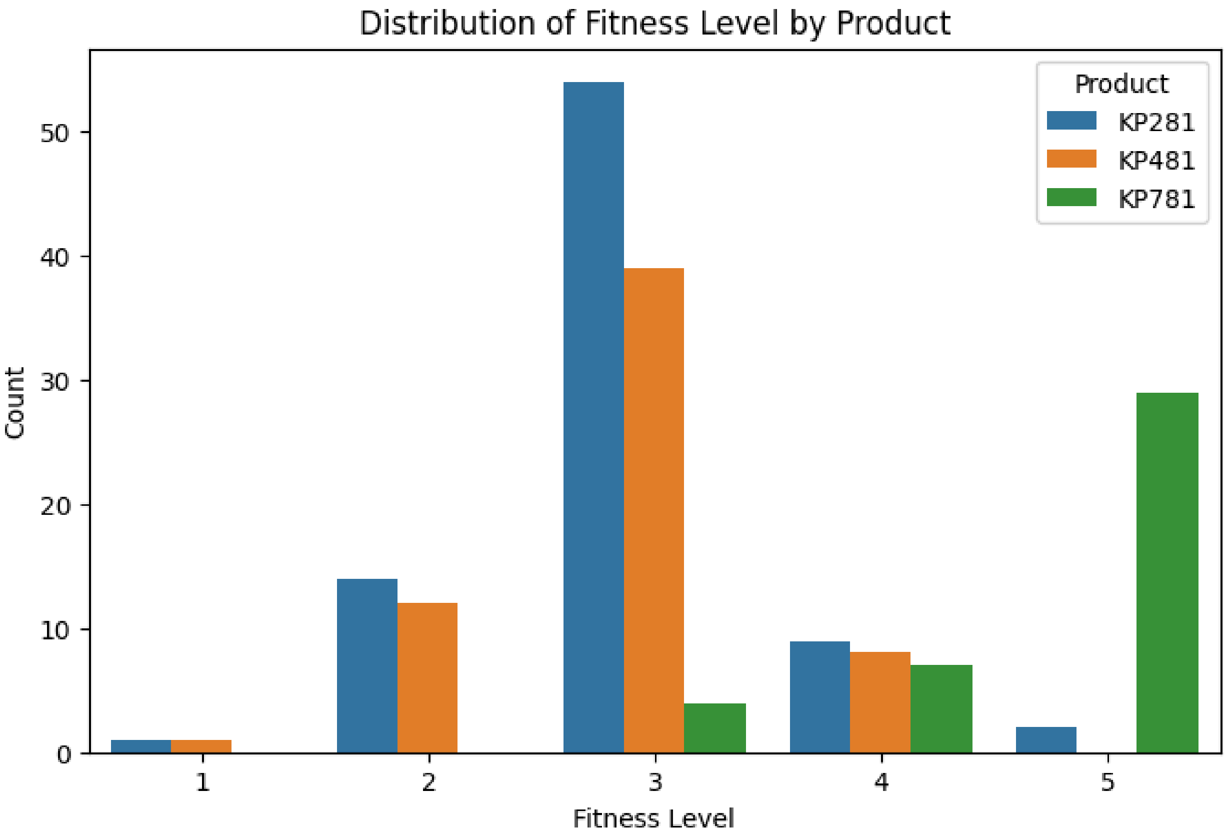
Distribution of Income by Product

Distribution of Miles by Product

Reasoning: Create countplots or boxplots to visualize the relationship between 'Fitness' and 'Product' as requested in step 4. A countplot is suitable for visualizing the distribution of a categorical variable like 'Fitness' across different products.

In [40]:

```python
# Create a countplot for Fitness by Product
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x='Fitness', hue='Product')
plt.title('Distribution of Fitness Level by Product')
plt.xlabel('Fitness Level')
plt.ylabel('Count')
plt.show()
```



Contingency tables and probabilities

Subtask:

Construct two-way contingency tables for 'Product' and categorical features like 'Gender' and 'MaritalStatus'. Compute conditional and marginal probabilities from these tables to understand the distribution of product purchases based on these characteristics.

Reasoning: Create contingency tables and calculate conditional and marginal probabilities for 'Product' with 'Gender' and 'MaritalStatus' as instructed.

Summary:

Data Analysis Key Findings

- **Customer Profiles by Product:**
    - **KP281:** Entry-level, customers are typically younger adults (mean age ~28.56), have lower average income (~$46,418.03), lower usage (~2.99 days/week) and fitness

    - (~2.91 rating), and expect fewer miles (~75.9 miles/week). Gender and marital status are relatively balanced.
    - **KP481:** Mid-range, customers have similar age (~28.90) to KP281, slightly higher average income (~$48,973.65), moderate usage (~3.07 days/week) and fitness (~3.0), and moderate expected miles (~87.9 miles/week). Gender and marital status are relatively balanced.
    - **KP781:** High-end, customers have a slightly higher mean age (~30.44), significantly higher average income (~$75,441.57), high usage (~4.77 days/week) and fitness (~4.62 rating, often 4 or 5), and high expected miles (~166.9 miles/week). This model is predominantly purchased by males (82.5% are male).
- **Impact of Characteristics:**
    - Income, fitness level, usage frequency, and expected miles are strong predictors of product purchase, particularly for the high-end KP781.
    - Gender is a significant factor for the KP781, which is heavily favored by males.
    - Age and education show some variation but are less influential than income, fitness, usage, and miles.
    - Marital status does not appear to significantly influence product choice.
- **Probabilities:**
    - Conditional probabilities highlight that for the KP781 model, 82.5% of buyers are male, compared to 51.25% for KP281 and 50.0% for KP481.
    - Conditional probabilities based on marital status show a relatively even split between partnered and single customers for all product models, indicating marital status is not a strong differentiator.

Insights or Next Steps

- Tailor marketing campaigns to the distinct customer profiles, focusing on income levels, fitness goals, and usage expectations for each product segment.
- Explore opportunities to attract more female customers to the higher-end KP781 model, potentially by highlighting different benefits or features.