sales analysis project

December 1, 2023

0.1 Project Title: Sales Analysis

0.1.1 Overview:

This project aims to analyze and derive insights from consumer behavior data, particularly focusing on purchasing patterns and preferences. Through visual representations, the study delves into various aspects, including demographic trends, regional influences, and sector-specific spending habits.

0.1.2 Key Observations:

- **Gender-based Purchasing Power:**
  - Evaluate and compare the purchasing power between genders based on comprehensive data visualizations.
- **Age Group Dynamics:**
  - Explore the dominant age group of buyers to better understand and tailor marketing strategies.
- **Geographical Impact:**
  - Analyze regional contributions to identify key markets and potential growth areas.
- **Marital Status Influence:**
  - Investigate the correlation between marital status, particularly among women, and purchasing behavior.
- **Sector-Specific Expenditure:**
  - Examine the spending patterns of professionals in critical sectors like IT, Healthcare, and Aviation.
- **Category-wise Spending Analysis:**
  - Break down spending across categories such as Food, Clothing, and Electronics to inform product development and promotional strategies.

0.1.3 Methodology:

Utilizing visualizations and statistical analyses, this project aims to provide actionable insights for stakeholders to make informed decisions regarding marketing, inventory management, and customer engagement.

0.1.4 Expected Outcomes:

The project anticipates uncovering nuanced consumer behavior trends that can be leveraged for strategic business decisions. Insights gained will contribute to optimizing marketing campaigns, refining product offerings, and enhancing overall customer satisfaction.

[1]:
```
# Import NumPy for numerical operations
import numpy as np

# Import Pandas for data manipulation and analysis
import pandas as pd

# Import Matplotlib for basic plotting
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

# Import Seaborn for statistical data visualization
import seaborn as sns
```

[2]:
```
sns.set_style("darkgrid")
matplotlib.rcParams['font.size'] = 14
matplotlib.rcParams['figure.figsize'] = (9, 5)
matplotlib.rcParams['figure.facecolor'] = '#00000000'
```

[3]:
```
file_path = r"C:
 ↪\Users\dell1\Downloads\Python_Diwali_Sales_Analysis-main\Python_Diwali_Sales_Analysis-main\
 ↪Sales Data.csv"
df = pd.read_csv(file_path, encoding='latin-1')
print(df)
```

```
       User_ID    Cust_name Product_ID Gender Age Group  Age  Marital_Status  \
0      1002903    Sanskriti  P00125942      F     26-35   28               0
1      1000732       Kartik  P00110942      F     26-35   35               1
2      1001990        Bindu  P00118542      F     26-35   35               1
3      1001425       Sudevi  P00237842      M      0-17   16               0
4      1000588         Joni  P00057942      M     26-35   28               1
…          …          …         …    …       …  …             …
11246  1000695      Manning  P00296942      M     18-25   19               1
11247  1004089  Reichenbach  P00171342      M     26-35   33               0
11248  1001209        Oshin  P00201342      F     36-45   40               0
11249  1004023       Noonan  P00059442      M     36-45   37               0
11250  1002744      Brumley  P00281742      F     18-25   19               0

                State      Zone       Occupation Product_Category  Orders  \
0         Maharashtra   Western       Healthcare             Auto       1
1      Andhra Pradesh  Southern             Govt             Auto       3
2       Uttar Pradesh   Central       Automobile             Auto       3
3           Karnataka  Southern     Construction             Auto       2
4             Gujarat   Western  Food Processing             Auto       2
…                 …         …                …              …     …
11246     Maharashtra   Western         Chemical           Office       4
11247         Haryana  Northern       Healthcare       Veterinary       3
11248  Madhya Pradesh   Central          Textile           Office       4
```

```
11249       Karnataka  Southern      Agriculture           Office       3
11250     Maharashtra   Western       Healthcare           Office       3

         Amount  Status  unnamed1
0       23952.0     NaN       NaN
1       23934.0     NaN       NaN
2       23924.0     NaN       NaN
3       23912.0     NaN       NaN
4       23877.0     NaN       NaN
…           …     …         …
11246     370.0     NaN       NaN
11247     367.0     NaN       NaN
11248     213.0     NaN       NaN
11249     206.0     NaN       NaN
11250     188.0     NaN       NaN

[11251 rows x 15 columns]
```

0.2 DATA CLEANING AND PREPARATION

[4]:
```
df.isnull().sum()
```

[4]:
```
User_ID                0
Cust_name              0
Product_ID             0
Gender                 0
Age Group              0
Age                    0
Marital_Status         0
State                  0
Zone                   0
Occupation             0
Product_Category       0
Orders                 0
Amount                12
Status             11251
unnamed1           11251
dtype: int64
```

[5]:
```
# Specify the column(s) to remove
columns_to_remove = ['Status', 'unnamed1']

# Use the del keyword to remove columns
for column in columns_to_remove:
    del df[column]
```

[6]:
```
df
```

[6]:
```
        User_ID    Cust_name Product_ID Gender Age Group  Age  Marital_Status  \
0       1002903    Sanskriti  P00125942      F     26-35   28               0
1       1000732       Kartik  P00110942      F     26-35   35               1
2       1001990        Bindu  P00118542      F     26-35   35               1
3       1001425       Sudevi  P00237842      M      0-17   16               0
4       1000588         Joni  P00057942      M     26-35   28               1
…         …          …         …    …       …  …            …
11246   1000695      Manning  P00296942      M     18-25   19               1
11247   1004089  Reichenbach  P00171342      M     26-35   33               0
11248   1001209        Oshin  P00201342      F     36-45   40               0
11249   1004023       Noonan  P00059442      M     36-45   37               0
11250   1002744      Brumley  P00281742      F     18-25   19               0

                State      Zone       Occupation Product_Category  Orders  \
0         Maharashtra   Western       Healthcare             Auto       1
1      Andhra Pradesh  Southern             Govt             Auto       3
2       Uttar Pradesh   Central       Automobile             Auto       3
3           Karnataka  Southern     Construction             Auto       2
4             Gujarat   Western  Food Processing             Auto       2
…               …       …               …              …     …
11246     Maharashtra   Western         Chemical           Office       4
11247         Haryana  Northern       Healthcare       Veterinary       3
11248  Madhya Pradesh   Central          Textile           Office       4
11249       Karnataka  Southern      Agriculture           Office       3
11250     Maharashtra   Western       Healthcare           Office       3

        Amount
0      23952.0
1      23934.0
2      23924.0
3      23912.0
4      23877.0
…        …
11246    370.0
11247    367.0
11248    213.0
11249    206.0
11250    188.0

[11251 rows x 13 columns]
```

[7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11251 entries, 0 to 11250
Data columns (total 13 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
```

```
 0   User_ID           11251 non-null  int64
 1   Cust_name         11251 non-null  object
 2   Product_ID        11251 non-null  object
 3   Gender            11251 non-null  object
 4   Age Group         11251 non-null  object
 5   Age               11251 non-null  int64
 6   Marital_Status    11251 non-null  int64
 7   State             11251 non-null  object
 8   Zone              11251 non-null  object
 9   Occupation        11251 non-null  object
 10  Product_Category  11251 non-null  object
 11  Orders            11251 non-null  int64
 12  Amount            11239 non-null  float64
dtypes: float64(1), int64(4), object(8)
memory usage: 1.1+ MB
```

[8]:
```
df.dropna(inplace = True)
```

[9]:
```
df.isnull().sum()
```

[9]:
```
User_ID             0
Cust_name           0
Product_ID          0
Gender              0
Age Group           0
Age                 0
Marital_Status      0
State               0
Zone                0
Occupation          0
Product_Category    0
Orders              0
Amount              0
dtype: int64
```

[10]:
```
df.sample(10)
```

[10]:

	User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	\
4726	1000166	Angele	P00022542	F	18-25	24	0	
10288	1001758	Edelman	P00321942	M	26-35	30	0	
6277	1003641	Cacioppo	P00041542	F	26-35	34	0	
2434	1001278	Phalguni	P00199442	F	18-25	18	0	
3532	1000695	Manning	P00217742	M	18-25	20	1	
1829	1002801	Moffitt	P00310242	M	26-35	27	0	
8266	1001364	D'Ascenzo	P00198042	M	46-50	47	0	
980	1001491	James	P00240142	M	18-25	21	0	
9627	1005795	Conant	P00250942	F	26-35	34	0	
10977	1005136	Aniket	P00031042	F	26-35	30	0	

```
                State      Zone    Occupation       Product_Category  Orders  \
4726      Uttarakhand   Central         Media     Clothing & Apparel       1
10288           Delhi   Central       Banking        Household items       4
6277        Rajasthan  Northern      Chemical  Electronics & Gadgets       2
2434            Delhi   Central      Aviation                   Food       1
3532    Uttar Pradesh   Central        Retail                   Food       3
1829            Delhi   Central  Construction                   Food       4
8266   Madhya Pradesh   Central      Aviation  Electronics & Gadgets       1
980    Andhra Pradesh  Southern         Media     Hand & Power Tools       4
9627       Maharashtra   Western    Healthcare        Household items       1
10977           Delhi   Central    Healthcare                 Beauty       1

        Amount
4726    8891.0
10288   3060.0
6277    7884.0
2434   15165.0
3532   11678.0
1829   15728.0
8266    5875.0
980    18789.0
9627    3866.0
10977   1606.0
```

[11]:
```
df['Amount'] = df['Amount'].astype('int')
```

[12]:
```
df['total_amount'] = df['Orders'] * df['Amount']
```

[13]:
```
df
```

[13]:
```
        User_ID    Cust_name Product_ID Gender Age Group  Age  Marital_Status  \
0       1002903    Sanskriti  P00125942      F     26-35   28               0
1       1000732       Kartik  P00110942      F     26-35   35               1
2       1001990        Bindu  P00118542      F     26-35   35               1
3       1001425       Sudevi  P00237842      M      0-17   16               0
4       1000588         Joni  P00057942      M     26-35   28               1
…         …           …         …    …       …  …             …
11246   1000695      Manning  P00296942      M     18-25   19               1
11247   1004089  Reichenbach  P00171342      M     26-35   33               0
11248   1001209        Oshin  P00201342      F     36-45   40               0
11249   1004023       Noonan  P00059442      M     36-45   37               0
11250   1002744      Brumley  P00281742      F     18-25   19               0

                State      Zone       Occupation Product_Category  Orders  \
0         Maharashtra   Western       Healthcare             Auto       1
1       Andhra Pradesh  Southern             Govt             Auto       3
```

```
2       Uttar Pradesh   Central       Automobile             Auto       3
3           Karnataka  Southern     Construction             Auto       2
4             Gujarat   Western  Food Processing             Auto       2
…                …       …               …               …     …
11246     Maharashtra   Western         Chemical           Office       4
11247         Haryana  Northern       Healthcare       Veterinary       3
11248  Madhya Pradesh   Central          Textile           Office       4
11249       Karnataka  Southern      Agriculture           Office       3
11250     Maharashtra   Western       Healthcare           Office       3

        Amount  total_amount
0        23952         23952
1        23934         71802
2        23924         71772
3        23912         47824
4        23877         47754
…        …             …
11246      370          1480
11247      367          1101
11248      213           852
11249      206           618
11250      188           564

[11239 rows x 14 columns]
```

1 Visualization Explanation

1.1 Total Amount by Gender

- The first subplot shows the total sales amount categorized by gender.
- Blue bars represent sales for one gender, and the orange bars represent the other.
- Provides a clear comparison of sales contribution between different genders.

1.2 Total Amount by Marital Status

- The second subplot visualizes the total sales amount based on both gender and marital status.
- Stacked bars show the contribution of each gender to the total sales for various marital statuses.
- Allows an analysis of how marital status impacts sales, considering both genders.

1.3 Amount Spent by Age Group

- The third subplot displays the total sales amount for different age groups.
- Helps understand the distribution of sales across various age categories.
- Colors represent different amounts spent within each age group.

1.4 Total Amount by State

- The fourth subplot focuses on the total sales amount in different states.

- The top 10 states with the highest sales are highlighted.
- Provides insights into regional variations in sales.

1.5 Total Amount by Occupation

- The fifth subplot illustrates the total sales amount for different occupations.
- The top 10 occupations with the highest sales are showcased.
- Allows for an analysis of sales patterns across various professions.

1.6 Total Amount by Product Category

- The sixth subplot visualizes the total sales amount for different product categories.
- The top 10 product categories with the highest sales are presented.
- Helps identify the most lucrative product categories.

Overall Presentation: - The entire visualization is organized in a 3x2 grid, providing a comprehensive overview of sales data. - Each subplot is labeled with a title to guide interpretation. - The color palettes and legends assist in distinguishing different categories within each subplot. - The layout is designed for clarity and comparison across various demographic and product-related factors.

[14]:
```
sales_gen = df.groupby(['Gender'], as_index=False)['total_amount'].sum().
↪sort_values(by="total_amount", ascending=False)
```

[15]:
```
marital_spending = df.groupby(['Marital_Status','Gender'] ,␣
↪as_index=False)['total_amount'].sum().sort_values(by='total_amount',␣
↪ascending=False)
```

[16]:
```
spend_by_age = df.groupby(['Age Group'] , as_index=False)['total_amount'].sum().
↪sort_values(by='total_amount', ascending=False)
```

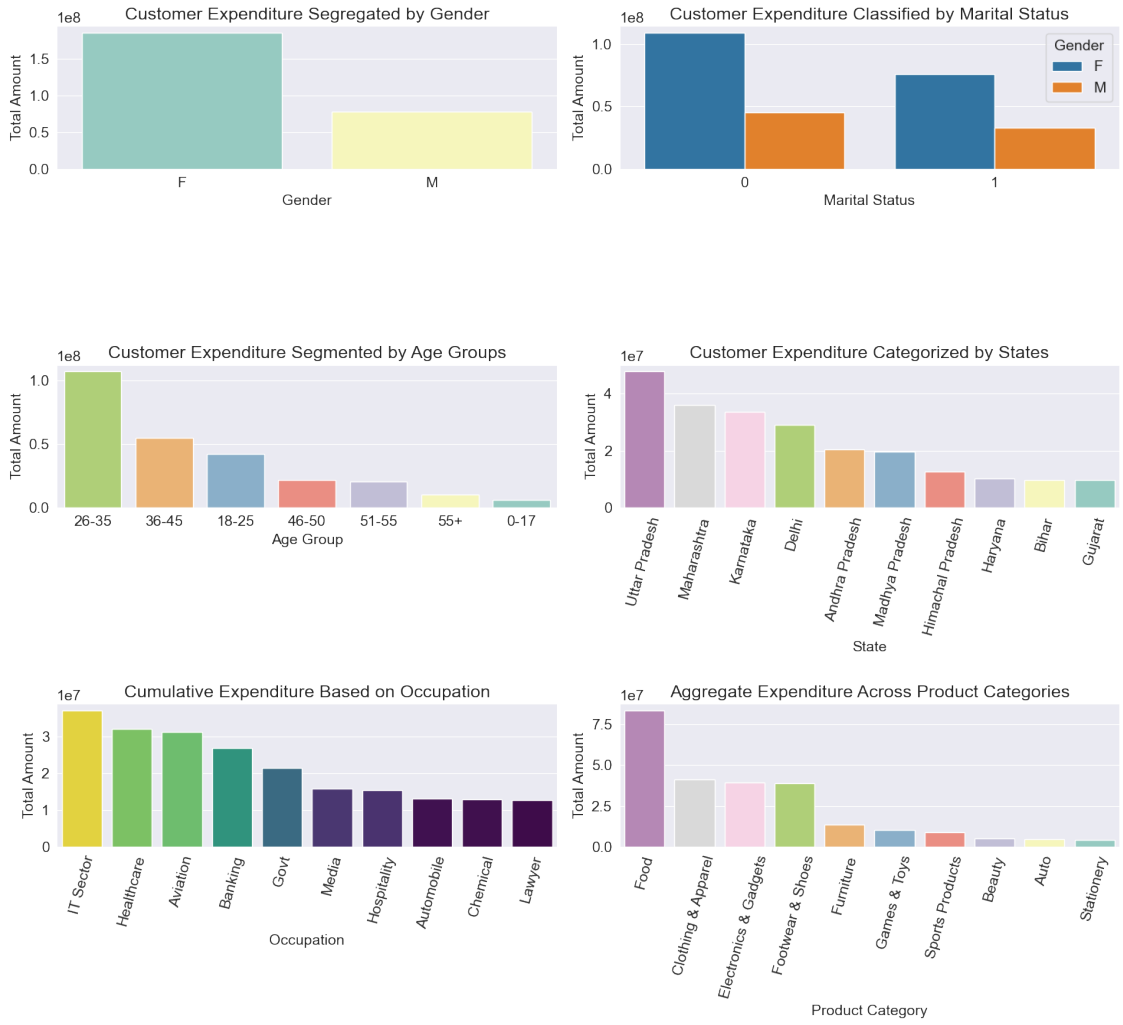
[17]:
```
top_states = df.groupby(['State'] , as_index=False)['total_amount'].sum().
↪sort_values(by='total_amount', ascending=False).head(10)
```

[18]:
```
top_occupation = df.groupby(['Occupation'] , as_index=False)['total_amount'].
↪sum().sort_values(by='total_amount', ascending=False).head(10)
```

[19]:
```
top_products = df.groupby(['Product_Category'] ,␣
↪as_index=False)['total_amount'].sum().sort_values(by='total_amount',␣
↪ascending=False).head(10)
```

[20]:
```
# Set up subplots
fig, axes = plt.subplots(3, 2, figsize=(15, 15))
fig.suptitle('Analysis of Sales Data by Total Amount Spent', fontsize=30)

# Plot 1: Gender
sns.barplot(x='Gender', y='total_amount', data=sales_gen, palette='Set3' ,␣
↪legend=False, hue='Gender', ax=axes[0, 0])
axes[0, 0].set_title('Customer Expenditure Segregated by Gender')
```

```
axes[0, 0].set_xlabel('Gender')
axes[0, 0].set_ylabel('Total Amount')

# Plot 2: Marital Status
sns.barplot(x="Marital_Status", y="total_amount", hue="Gender",␣
 ↪data=marital_spending, ax=axes[0, 1])
axes[0, 1].set_title('Customer Expenditure Classified by Marital Status')
axes[0, 1].set_xlabel('Marital Status')
axes[0, 1].set_ylabel('Total Amount')

# Plot 3: Age
sns.barplot(y='total_amount', x='Age Group', data=spend_by_age, palette='Set3',␣
 ↪hue='total_amount', legend=False, ax=axes[1, 0])
axes[1, 0].set_title('Customer Expenditure Segmented by Age Groups')
axes[1, 0].set_xlabel('Age Group')
axes[1, 0].set_ylabel('Total Amount')

# Plot 4: States
sns.barplot(y='total_amount', x='State', data=top_states, palette='Set3',␣
 ↪hue='total_amount', legend=False, ax=axes[1, 1])
axes[1, 1].set_title('Customer Expenditure Categorized by States')
axes[1, 1].set_xlabel('State')
axes[1, 1].set_ylabel('Total Amount')
axes[1, 1].tick_params(axis='x', rotation=75)

# Plot 5: Occupation
sns.barplot(y='total_amount', x='Occupation', data=top_occupation,␣
 ↪palette='viridis', hue='total_amount', legend=False, ax=axes[2, 0])
axes[2, 0].set_title('Cumulative Expenditure Based on Occupation')
axes[2, 0].set_xlabel('Occupation')
axes[2, 0].set_ylabel('Total Amount')
axes[2, 0].tick_params(axis='x', rotation=75)

# Plot 6: Product Category
sns.barplot(y='total_amount', x='Product_Category', data=top_products,␣
 ↪palette='Set3', hue='total_amount', legend=False, ax=axes[2, 1])
axes[2, 1].set_title('Aggregate Expenditure Across Product Categories')
axes[2, 1].set_xlabel('Product Category')
axes[2, 1].set_ylabel('Total Amount')
axes[2, 1].tick_params(axis='x', rotation=75)

# Adjust layout
plt.tight_layout(rect=[0, 0, 1, 0.96])

# Show the plots
plt.show()
```



1.6.1 Key Observations from the Graphs

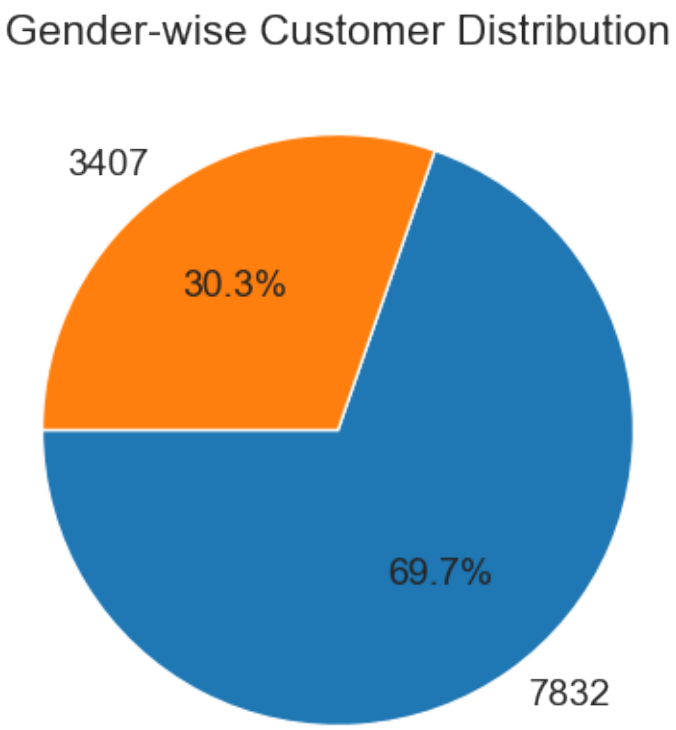
- **Female Purchasing Power:**
  - The graphs highlight that the purchasing power of females exceeds that of males.
- **Dominant Age Group:**
  - Most buyers fall within the age group of 26-35 years, as evidenced by the graphical representation.
- **Geographical Insights:**
  - The states of Uttar Pradesh, Maharashtra, and Karnataka stand out as the primary contributors to the highest number of orders and total sales/amount.
- **Marital Status Influence:**
  - The data reveals that a significant portion of buyers consists of married women, indicating both a strong presence and substantial purchasing power.

- **Sector-Specific Spending:**
  - Professionals working in the IT, Healthcare, and Aviation sectors are identified as the primary contributors to the highest expenditure, as depicted in the graphs.
- **Top Spending Categories:**
  - The major expenditure is concentrated in the Food, Clothing, and Electronics categories, reflecting the dominant areas of consumer spending.

1.7 Asking and Answering Questions

1.7.1 Q1. What is the distribution of male and female buyers?

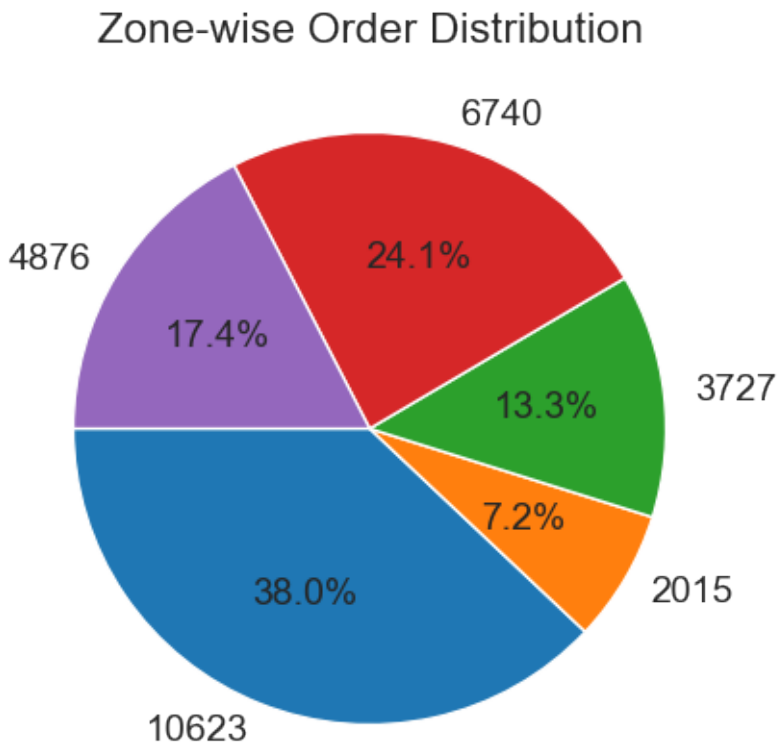
[21]:
```
gender_counts = df.Gender.value_counts()
plt.figure(figsize=(9,5))
plt.title('Gender-wise Customer Distribution', fontsize=16)
plt.pie(gender_counts, labels=gender_counts, autopct='%1.1f%%', startangle=180);
```



1.7.2 Q2. Determine the zone with the highest order count.

[22]:
```
zone_order_counts = df.groupby('Zone')['Orders'].sum()
plt.figure(figsize=(9, 5))
plt.title('Zone-wise Order Distribution', fontsize=16)
```

```
plt.pie(zone_order_counts, labels=zone_order_counts, autopct='%1.1f%%',␣
 ↪startangle=180);
plt.show()
```



[23]:
```
df.columns
```

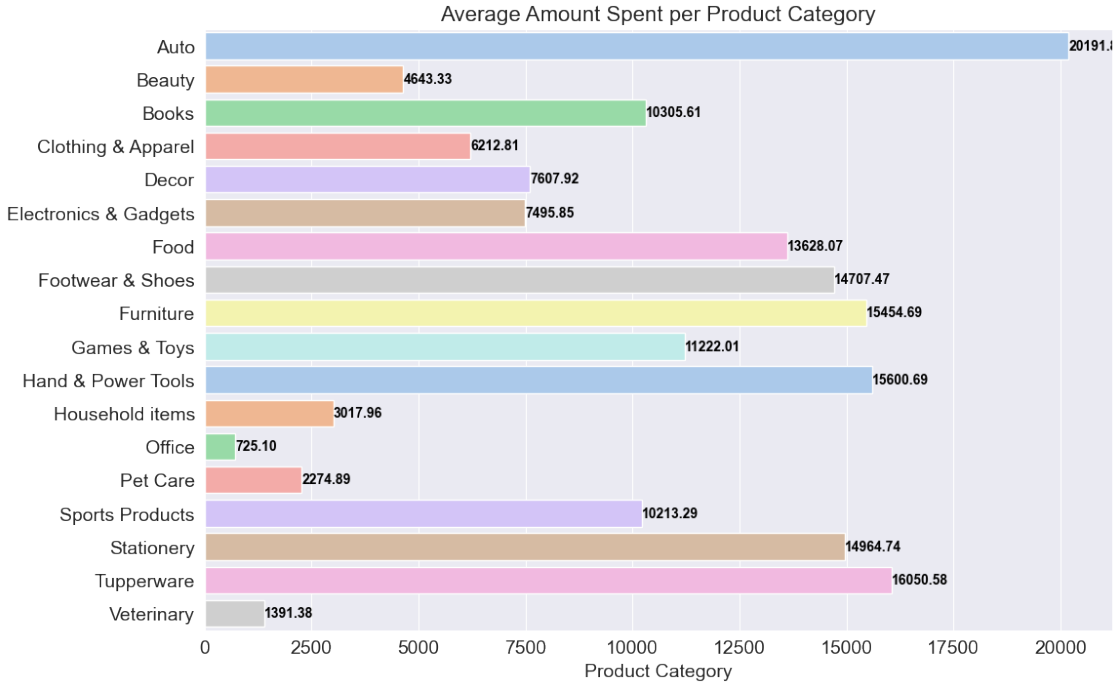
[23]:
```
Index(['User_ID', 'Cust_name', 'Product_ID', 'Gender', 'Age Group', 'Age',
       'Marital_Status', 'State', 'Zone', 'Occupation', 'Product_Category',
       'Orders', 'Amount', 'total_amount'],
      dtype='object')
```

1.7.3 Q3. What is the average amount spent on each product?

[24]:
```
average_amount_per_product = df.
 ↪groupby(['Product_Category'],as_index=False)['Amount'].mean()
```

[25]:
```
plt.figure(figsize=(12, 8))
ax=sns.barplot(x='Amount', y='Product_Category',␣
 ↪data=average_amount_per_product,␣
 ↪palette='pastel',legend=False,hue='Product_Category')
for bars in ax.containers:
```

```
    ax.bar_label(bars, fmt='%.2f', label_type='edge', fontsize=10,␣
 ↪color='black', weight='bold', clip_on=True)
plt.title('Average Amount Spent per Product Category', fontsize=16)
plt.xlabel('Product Category')
plt.ylabel(None)
plt.show();
```



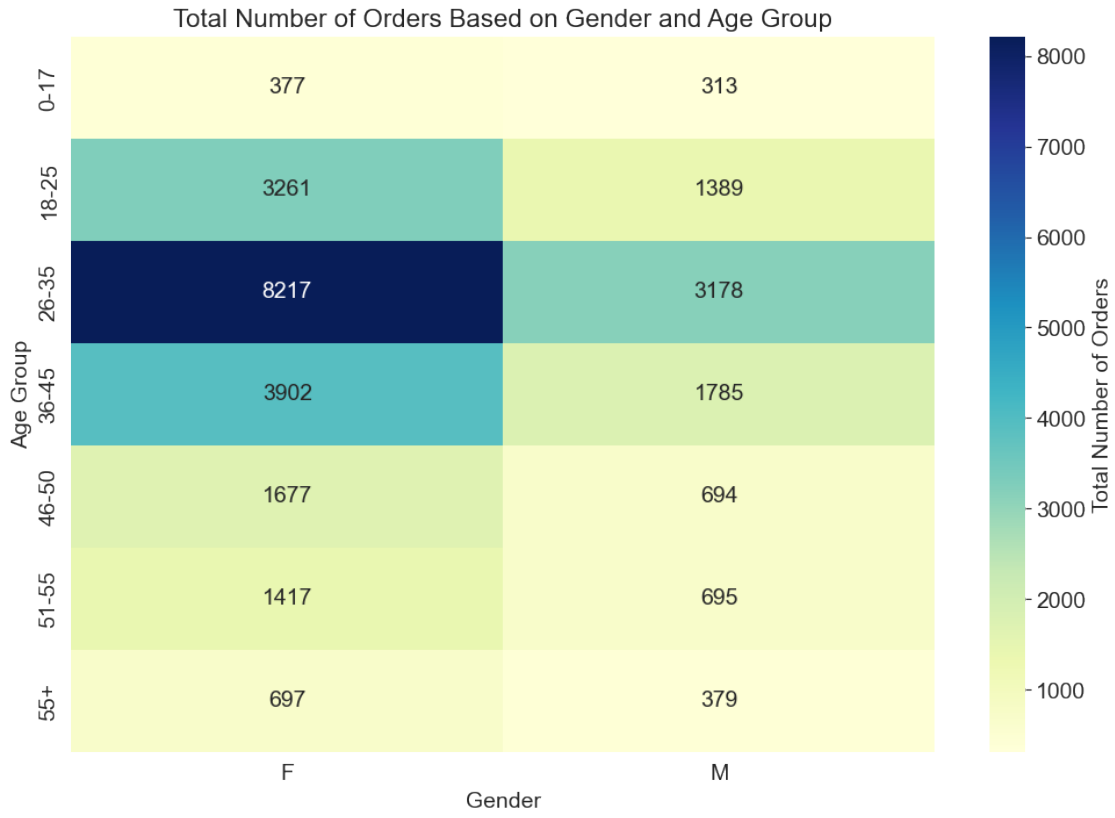
1.7.4 Q4. Are there any patterns in the total number of orders based on gender and age group?

[26]:
```
pivot_table = df.pivot_table(index='Age Group', columns='Gender',␣
 ↪values='Orders', aggfunc='sum', fill_value=0)
pivot_table
```

[26]:

Gender	F	M
Age Group		
0-17	377	313
18-25	3261	1389
26-35	8217	3178
36-45	3902	1785
46-50	1677	694
51-55	1417	695
55+	697	379

[27]:
```
plt.figure(figsize=(12, 8))
sns.heatmap(pivot_table, annot=True, cmap='YlGnBu', fmt='d', cbar_kws={'label':␣
↪'Total Number of Orders'})
plt.title('Total Number of Orders Based on Gender and Age Group', fontsize=16)
plt.xlabel('Gender', fontsize=14)
plt.ylabel('Age Group', fontsize=14)
plt.show();
```



The visual representation strongly suggests that the peak volume of orders was driven by a dynamic demographic—specifically, women in the vibrant age range of 26 to 35. This cohort emerges as the frontrunner in terms of placing the highest number of orders, adding a compelling layer to our understanding of consumer behavior.