LONDON METROPOLITAN UNIVERSITY

Islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CC5061NI Applied Data Science

60% Individual Coursework

Academic Semester: Spring Semester 2026

Credit: 15 credit Semester long module

Student Name: Abiram Bhatta

London Met ID: NP01AI4A240144

College ID: 24046558

Assignment Due Date: Monday, May 18, 2026

Assignment Submission Date: Friday, May 15, 2026

Submitted To: Subarna Sapkota

I confirm that I understand my coursework needs to be submitted online via MST Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded

Table of Contents

Table of Figures

1 Introduction

This technical report analyses the Diwali Sales Transaction dataset, a retail/e-commerce transaction dataset from a festive sales period. It combines customer, location, product, order and revenue fields to identify the customer segments and regions that contribute most strongly to sales performance.

1.1 Overview of Dataset

The raw dataset contains 11,251 rows and 15 columns. Each row represents one customer transaction, and the main outcome variable is Amount because it directly reflects sales revenue.

1.2 Objectives of the Analysis

The analysis aims to inspect data quality, prepare and clean the dataset, create useful sales categories, evaluate descriptive and correlation patterns, visualise customer/product sales trends, and apply ANOVA and Chi-square tests to support conclusions.

2 Data Understanding

Data understanding examines the data source, fields, structure and quality issues. It is required because later cleaning, visualisation and testing decisions depend on correctly interpreting the raw dataset.

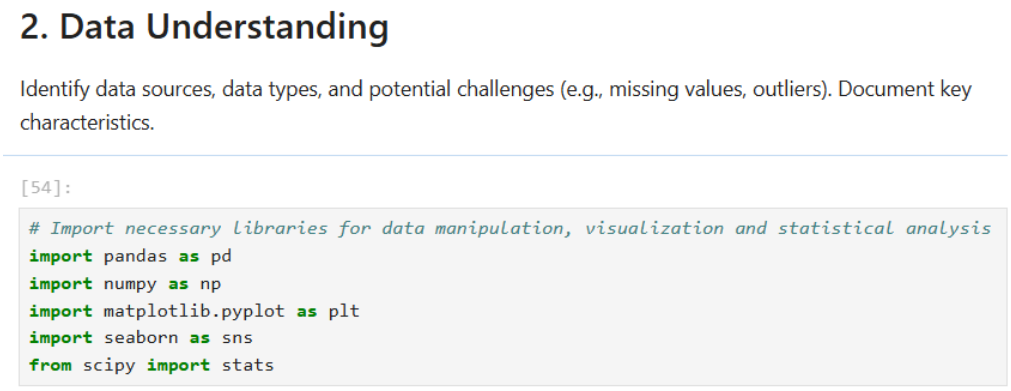


Figure 1: Importing required libraries

```
# Display the first few rows to inspect the data structure
df.head()
```

User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount	Status	unnamed1
1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0	NaN	NaN
1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0	NaN	NaN
1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0	NaN	NaN
1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912.0	NaN	NaN
1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877.0	NaN	NaN

Figure 2: Table Data

2.1 Dataset Overview

- **Dataset Name:** Diwali Sales Transaction Data
- **Domain:** Retail / E-commerce
- **Record Type:** Transactional
- **Initial Dimensions:** The raw dataset consists of **15 columns** and approximately **11,251 rows**.

```
# Check the data types and non-null counts to understand the variables
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11251 entries, 0 to 11250
Data columns (total 15 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
 13  Status            0 non-null      float64
 14  unnamed1          0 non-null      float64
dtypes: float64(3), int64(4), object(8)
memory usage: 1.3+ MB
```

Figure 3: using df.info()

```
# Confirming the exact dimensions (Rows, Columns)
print("\nTotal Rows: {}".format(df.shape[0]))
print("Total Columns: {}".format(df.shape[1]))
```

```
Total Rows: 11251
Total Columns: 15
```

Figure 4: using df.shape()

The dataset belongs to the retail/e-commerce domain and is transactional. It combines demographic, geographic and transaction fields to identify customer purchasing patterns and festival-period sales performance.

The raw dataset occupies approximately 1.3+ MB of memory in pandas, contains 11,251 observations and has 15 fields. Initial inspection shows that Amount has 12 missing values, while Status and unnamed1 are empty artifact columns.

2.2 Data Description Table

Appendix G: Full Data Description Table provides the complete data description table for the dataset.

2.3 Identified Challenges

The key issues were 12 missing Amount values, empty artifact columns, privacy-sensitive identifiers, and Age Group needing ordinal treatment. Product_Category has 18 categories and numerical correlations with Amount are weak, so the analysis needs careful grouping and interpretation. Detailed challenge points are listed in **Appendix A: Detailed Objectives and Data Challenges.**

3 Data Preparation

3.1 Import the Dataset

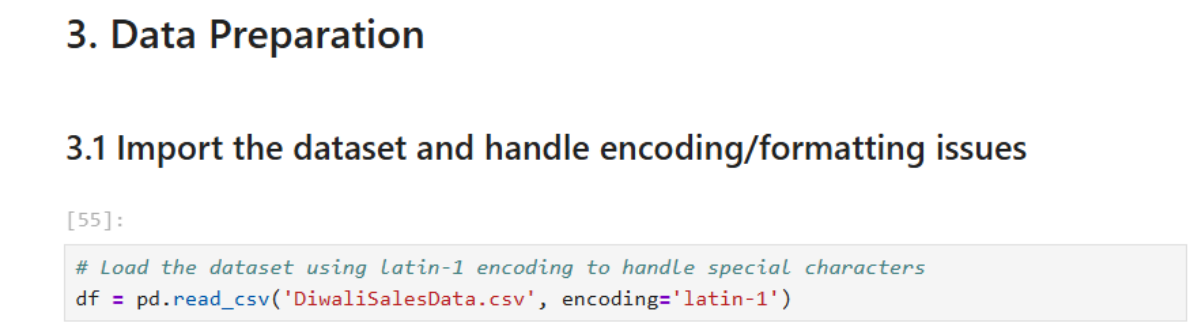


Figure 5: importing the dataset using pandas with latin-1 encoding

```
# Display the first few rows to inspect the data structure
df.head()
```

User_ID	Cust_name	Product_ID	Gender	Age Group	Age	Marital_Status	State	Zone	Occupation	Product_Category	Orders	Amount	Status	unnamed1
1002903	Sanskriti	P00125942	F	26-35	28	0	Maharashtra	Western	Healthcare	Auto	1	23952.0	NaN	NaN
1000732	Kartik	P00110942	F	26-35	35	1	Andhra Pradesh	Southern	Govt	Auto	3	23934.0	NaN	NaN
1001990	Bindu	P00118542	F	26-35	35	1	Uttar Pradesh	Central	Automobile	Auto	3	23924.0	NaN	NaN
1001425	Sudevi	P00237842	M	0-17	16	0	Karnataka	Southern	Construction	Auto	2	23912.0	NaN	NaN
1000588	Joni	P00057942	M	26-35	28	1	Gujarat	Western	Food Processing	Auto	2	23877.0	NaN	NaN

Figure 6: head() output showing the first five transaction records

The head() output confirms that the dataset loaded correctly and that the columns are aligned. The first rows include customer, product, demographic, geographic and transaction fields, which confirms that the file contains retail sales transaction records.

3.2 Dataset Insight

The dataset structure was inspected using df.info(), df.dtypes and df.shape. These outputs confirm the number of rows and columns, identify object, integer and float columns, and highlight missing values in Amount, Status and unnamed1.

```
[14]: # Utilizing df.info() to see a high-level summary of the dataset structure
      df.info()

      <class 'pandas.core.frame.DataFrame'>
      RangeIndex: 11251 entries, 0 to 11250
      Data columns (total 15 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
       13  Status            0 non-null      float64
       14  unnamed1          0 non-null      float64
      dtypes: float64(3), int64(4), object(8)
      memory usage: 1.3+ MB
```

Figure 7: df.info() output showing non-null counts, data types and memory usage.

```
:  # Specifically checking data types for each feature
   print("Detailed Column Data Types:\n{}".format(df.dtypes))

   Detailed Column Data Types:
   User_ID              int64
   Cust_name           object
   Product_ID          object
   Gender              object
   Age Group           object
   Age                  int64
   Marital_Status       int64
   State               object
   Zone                object
   Occupation          object
   Product_Category    object
   Orders               int64
   Amount             float64
   Status             float64
   unnamed1           float64
   dtype: object
```

Figure 8: Detailed column data types from df.dtypes.

```
# Confirming the exact dimensions (Rows, Columns)
print("\nTotal Rows: {}".format(df.shape[0]))
print("Total Columns: {}".format(df.shape[1]))
```

```
Total Rows: 11251
Total Columns: 15
```

Figure 9: Dataset dimensions showing 11,251 rows and 15 columns.

The raw dataset contains 11,251 rows and 15 columns. It contains integer variables such as User_ID, Age, Marital_Status and Orders; float variables such as Amount, Status and unnamed1; and categorical/object variables such as Gender, State, Zone, Occupation and Product_Category.

3.2.1 Business Relivence :

Appendix H: Business Relevance of Key Columns summarises the business relevance of the key variables: Amount, Orders, demographic fields, location fields and Occupation.

4 Data Cleaning

Data cleaning improves reliability by converting variables into suitable formats, creating useful categories, removing irrelevant columns and handling missing values while keeping variables that support customer, product and sales analysis.

4.1 Convert Age Group to Ordinal Categorical

Task: Map the seven specific age ranges from the dataset into broad categories: **Teen**, **Adult**, and **Senior**, and establish a mathematical order.

4. Data Cleaning

4.1 Convert Age Group to Ordinal Categorical

```
# Define mapping: Teen (0-17), Adult (18-55), Senior (55+)
age_mapping = {
    '0-17': 'Teen',
    '18-25': 'Adult',
    '26-35': 'Adult',
    '36-45': 'Adult',
    '46-50': 'Adult',
    '51-55': 'Adult',
    '55+': 'Senior'
}
```

```
# Mapping the existing column to new broad categories
df['Age Group'] = df['Age Group'].map(age_mapping)

# Converting to an ordinal categorical type with a defined sequence
df['Age Group'] = pd.Categorical(df['Age Group'], categories=['Teen', 'Adult', 'Senior'], ordered=True)
```

```
# Verifying the conversion
print("Age Group data type: {}".format(df['Age Group'].dtype))
print("Unique Categories: {}".format(df['Age Group'].unique()))
```

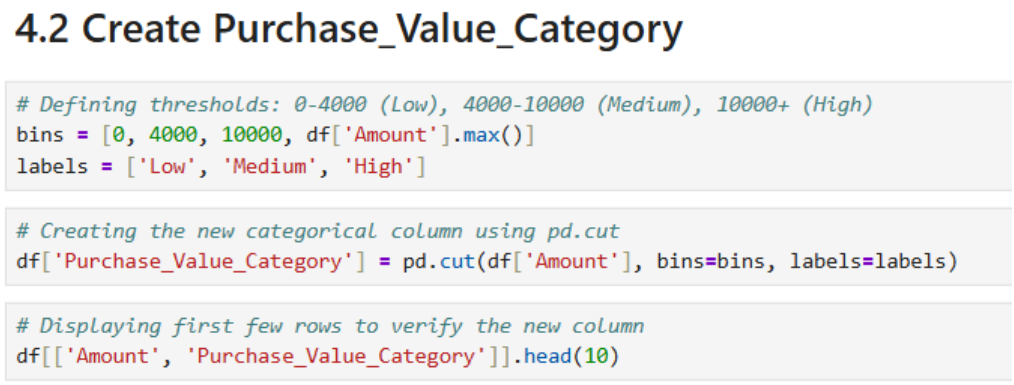
```
Age Group data type: category
Unique Categories: ['Adult', 'Teen', 'Senior']
Categories (3, object): ['Teen' < 'Adult' < 'Senior']
```

Figure 10: Code and output converting Age Group into an ordered categorical variable.

Ordinal encoding matters because age categories are not just labels; they have a meaningful chronological order. This helps later analysis present age groups in a logical order rather than alphabetically.

4.2 Create Purchase_Value_Category Column

The continuous Amount variable was converted into Low, Medium and High transaction bands using pd.cut(). The thresholds used were 0-4000 for Low, 4000-10000 for Medium and above 10000 for High. These thresholds separate small purchases, mid-range purchases and high-value festival purchases.



4.2 Create Purchase_Value_Category

```
# Defining thresholds: 0-4000 (Low), 4000-10000 (Medium), 10000+ (High)
bins = [0, 4000, 10000, df['Amount'].max()]
labels = ['Low', 'Medium', 'High']
```

```
# Creating the new categorical column using pd.cut
df['Purchase_Value_Category'] = pd.cut(df['Amount'], bins=bins, labels=labels)
```

```
# Displaying first few rows to verify the new column
df[['Amount', 'Purchase_Value_Category']].head(10)
```

Figure 11: Code creating Purchase_Value_Category from Amount.

	Amount	Purchase_Value_Category
0	23952.00	High
1	23934.00	High
2	23924.00	High
3	23912.00	High
4	23877.00	High
5	23877.00	High
6	23841.00	High
7	NaN	NaN
8	23809.00	High
9	23799.99	High

Figure 12: Output of Purchase_Value_Category from Amount

This category simplifies interpretation because managers can compare customer groups by spending level rather than only by raw transaction value.

4.3 Drop Irrelevant Columns

Cust_name, User_ID, unnamed1 and Status were removed because identifiers are not needed for aggregate analysis and the last two columns contain no meaningful data.

3.3 Drop Irrelevant Columns

```
# Identifying irrelevant artifact columns and personal identifiers
columns_to_drop = ['unnamed1', 'User_ID', 'Cust_name']

# Removing columns from the dataframe
df.drop(columns_to_drop, axis=1, inplace=True)

print("Remaining Columns:\n- " + "\n- ".join(df.columns.tolist()))
```

```
Remaining Columns:
- Product_ID
- Gender
- Age Group
- Age
- Marital_Status
- State
- Zone
- Occupation
- Product_Category
- Orders
- Amount
- Status
- Purchase_Value_Category
```

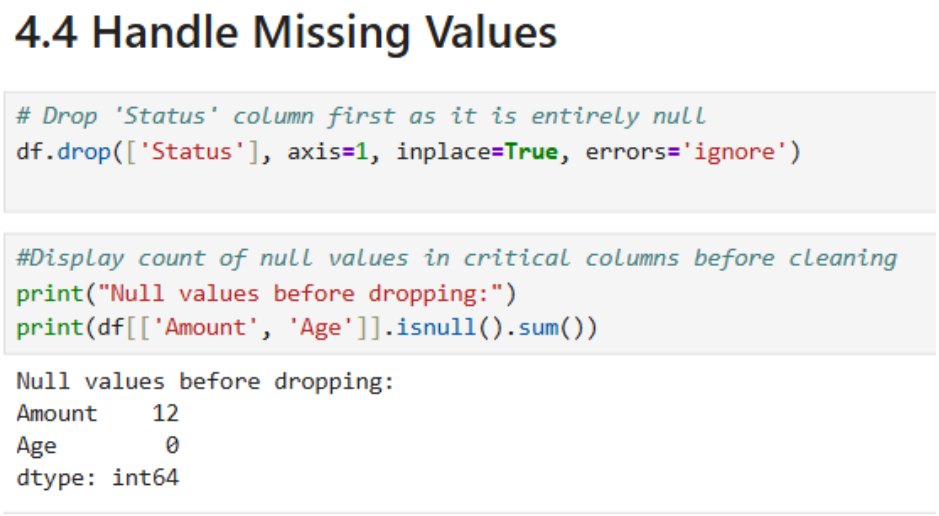
Figure 13: code and output for dropping irrelevant columns.

This kept the analysis focused on useful demographic and transaction variables.

4.4 Handle Missing Values

4.4.1 Techniques for Managing Missing Data

Appendix B: Techniques for Managing Missing Data summarises common missing-data techniques. Dropping rows was selected because Amount is the main revenue variable.



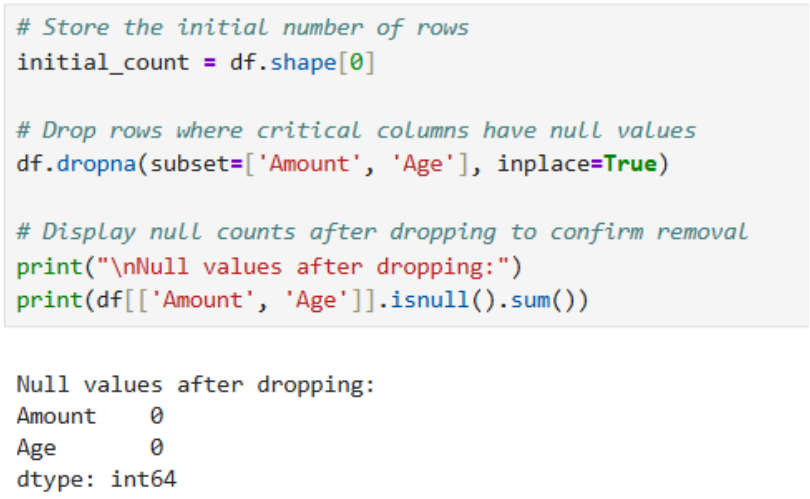
4.4 Handle Missing Values

```
# Drop 'Status' column first as it is entirely null
df.drop(['Status'], axis=1, inplace=True, errors='ignore')
```

```
#Display count of null values in critical columns before cleaning
print("Null values before dropping:")
print(df[['Amount', 'Age']].isnull().sum())
```

```
Null values before dropping:
Amount    12
Age        0
dtype: int64
```

Figure 14: Count of null values before dropping in critical coloumn



```
# Store the initial number of rows
initial_count = df.shape[0]

# Drop rows where critical columns have null values
df.dropna(subset=['Amount', 'Age'], inplace=True)

# Display null counts after dropping to confirm removal
print("\nNull values after dropping:")
print(df[['Amount', 'Age']].isnull().sum())
```

```
Null values after dropping:
Amount    0
Age       0
dtype: int64
```

Figure 15: Count of null values after dropping in critical column

```
# Calculate and display the total number of rows removed
final_count = df.shape[0]
rows_removed = initial_count - final_count
print("\nTotal rows removed from the dataset:", rows_removed)
```

```
Total rows removed from the dataset: 12
```

Figure 16: Calculating the total number of rows removed

```
#Display the final shape of the dataset for verification
print("\nFinal dataset shape (Rows, Columns):")
print(df.shape)
```

```
Final dataset shape (Rows, Columns):
(11239, 12)
```

Figure 17: Final Shape of the dataset

In this dataset, Status was removed as an empty artifact column. Rows with missing Amount were dropped because Amount is the primary sales outcome. The output shows that 12 rows were removed, reducing the dataset from 11,251 rows to 11,239 rows. This avoids introducing artificial sales values through imputation.

4.5 List Unique Values for Categorical Columns

The distinct values in categorical columns were inspected to understand category diversity and detect unexpected labels. This step is important before creating plots or grouped summaries.

4.5 Unique Values for Categorical Columns

```
# List distinct values for categorical columns like Product_Category, Zone, occupation
print('Unique Product Categories:', df['Product_Category'].unique())
print('\nUnique Zones:', df['Zone'].unique())
print("\nUnique Occupations:",df['Occupation'].unique())
```

```
Unique Product Categories: ['Auto' 'Hand & Power Tools' 'Stationery' 'Tupperware' 'Footwear & Shoes'
 'Furniture' 'Food' 'Games & Toys' 'Sports Products' 'Books'
 'Electronics & Gadgets' 'Decor' 'Clothing & Apparel' 'Beauty'
 'Household items' 'Pet Care' 'Veterinary' 'Office']

Unique Zones: ['Western' 'Southern' 'Central' 'Northern' 'Eastern']

Unique Occupations: ['Healthcare' 'Govt' 'Automobile' 'Construction' 'Food Processing'
 'Lawyer' 'Media' 'Banking' 'Retail' 'IT Sector' 'Aviation' 'Hospitality'
 'Agriculture' 'Textile' 'Chemical']
```

Figure 18: Unique values for Product_Category, Zone and Occupation.

```
# Counting the number of unique entries for the report interpretation
print("\nTotal Unique Product Categories: {}".format(df['Product_Category'].nunique()))
print("Total Unique Zones: {}".format(df['Zone'].nunique()))
```

```
Total Unique Product Categories: 18
Total Unique Zones: 5
```

Figure 19: Total number of unique product categories and unique zones

Product_Category contains 18 unique categories and Zone contains 5 regions. No unexpected duplicate spellings were identified in the displayed key categories.

5 Data Analysis

Data analysis summarises numerical variables and measures relationships between them. This section includes descriptive statistics, skewness, kurtosis and correlation analysis.

5.1 Summary Statistics

Mean, median, mode, skewness and kurtosis were used to summarise the numerical variables. **Appendix C: Summary Statistics Definitions** provides the detailed definitions of these statistical measures.

5.1 Summary Statistics

```
# Calculate mean, std, skewness and kurtosis for numerical columns
num_cols = ['Age', 'Orders', 'Amount']
stats_df = pd.DataFrame({
    'Mean': df[num_cols].mean(),
    'Std': df[num_cols].std(),
    'Skewness': df[num_cols].skew(),
    'Kurtosis': df[num_cols].kurt()
})
print(stats_df)
```

```
                Mean          Std  Skewness  Kurtosis
Age        35.410357    12.753866  1.185478  2.472002
Orders      2.489634     1.114967  0.019284 -1.352541
Amount   9453.610858  5222.355869  0.558026 -0.540209
```

Figure 20: Code and output to calculate the mean, standard deviation, skewness and kurtosis

```python
# Providing a comprehensive overview including count, min, max, and quartiles
summary_stats = df[num_cols].describe()
skew_val = df[num_cols].skew()
kurt_val = df[num_cols].kurt()

print("Comprehensive Summary Statistics:")
print(summary_stats)
print("\nSkewness Values:\n", skew_val)
print("\nKurtosis Values:\n", kurt_val)
```

```
Comprehensive Summary Statistics:
                Age        Orders        Amount
count  11239.000000  11239.000000  11239.000000
mean      35.410357      2.489634   9453.610858
std       12.753866      1.114967   5222.355869
min       12.000000      1.000000    188.000000
25%       27.000000      2.000000   5443.000000
50%       33.000000      2.000000   8109.000000
75%       43.000000      3.000000  12675.000000
max       92.000000      4.000000  23952.000000

Skewness Values:
 Age       1.185478
Orders     0.019284
Amount     0.558026
dtype: float64

Kurtosis Values:
 Age       2.472002
Orders    -1.352541
Amount    -0.540209
dtype: float64
```

Figure 21: Code used to compute summary statistics, skewness, kurtosis

```python
# Plotting Age, Orders, and Amount to visually assess skewness and kurtosis
plt.figure(figsize=(15, 5))

# Plotting Age
plt.subplot(1, 3, 1)
sns.kdeplot(df['Age'], fill=True, color='teal')
plt.title('Distribution Curve: Age')

# Plotting Orders
plt.subplot(1, 3, 2)
sns.kdeplot(df['Orders'], fill=True, color='orange')
plt.title('Distribution Curve: Orders')

# Plotting Amount
plt.subplot(1, 3, 3)
sns.kdeplot(df['Amount'], fill=True, color='purple')
plt.title('Distribution Curve: Amount')

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

Figure 22: Code used to compute summary statistics, skewness, kurtosis curves

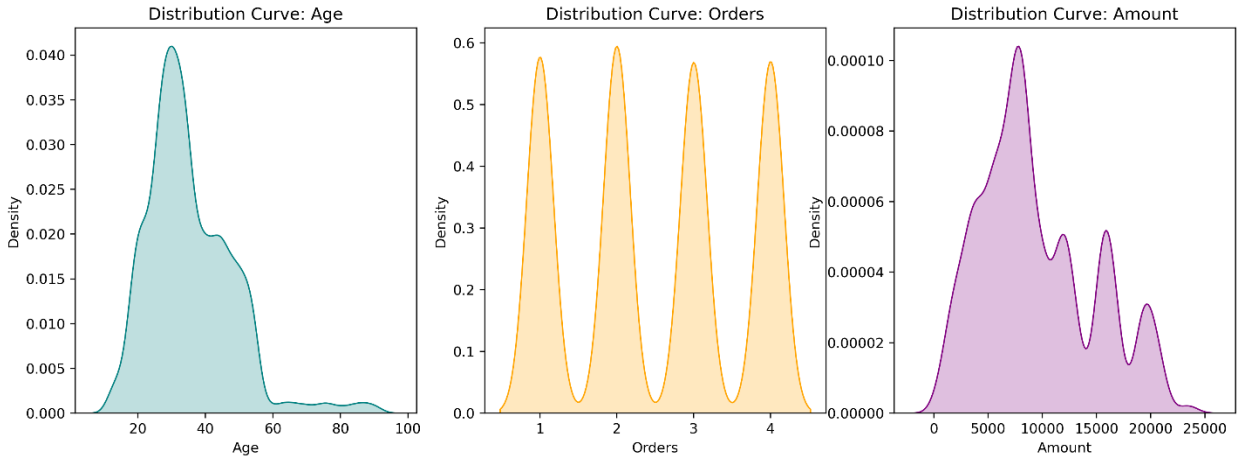


Figure 23: Distribution curves

The statistical profiling shows that Amount is positively skewed with some high-value transactions, Age is relatively balanced, and Orders is a discrete purchase-count variable. These results support using descriptive statistics alongside visuals because averages alone do not show the full spread of festival spending.

5.2 Correlation Analysis

5.2.1 Definitions and Importance

Correlation matrices and heatmaps were used to identify the strength and direction of relationships between numerical variables. Definitions are summarised in **Appendix D: Correlation Concepts and Calculation Methods**.

5.2.2 Calculation Methods

Pearson correlation was used to measure linear relationships between Age, Orders and Amount. Spearman correlation is another method for ranked or monotonic relationships and is explained in **Appendix D: Correlation Concepts and Calculation Methods**.

5.2 Correlation Analysis

```
# Calculate pairwise correlations for Age, Orders and Amount
plt.figure(figsize=(10,6))
sns.heatmap(df[num_cols].corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```

Figure 24: Correlation Calculation

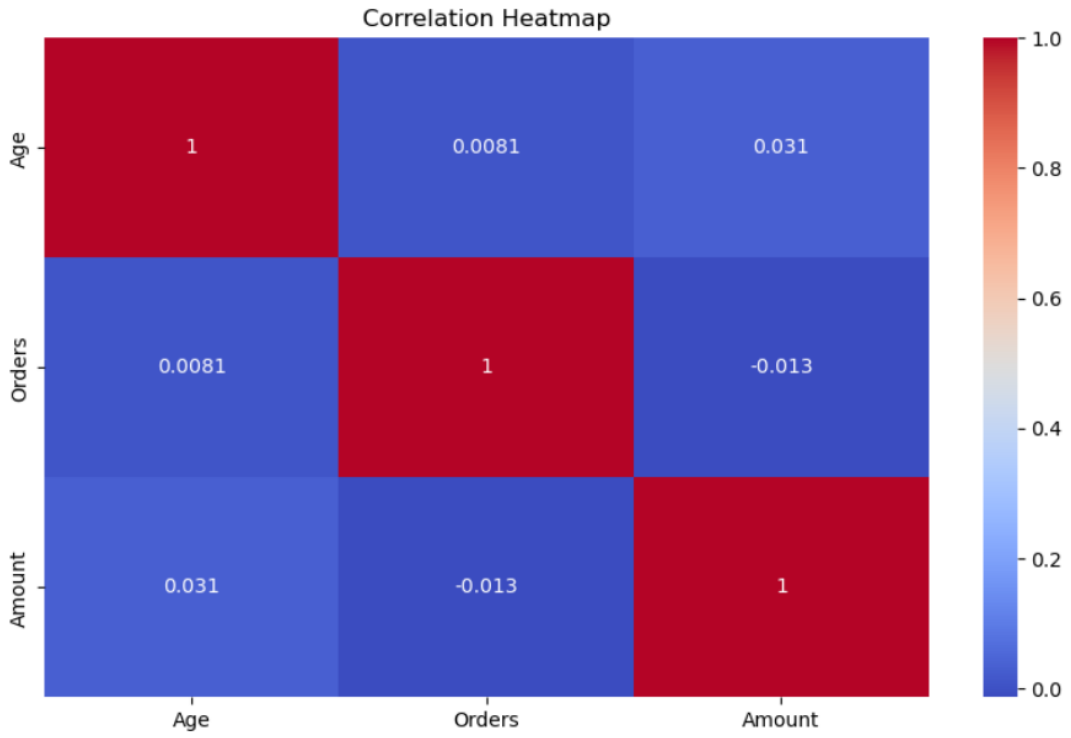


Figure 25: Heat Map

The correlation analysis shows weak pairwise relationships between the numerical variables. Orders and Amount have a negligible negative correlation ($r \approx -0.013$), while Age and Amount have a negligible positive correlation ($r \approx 0.031$), so neither variable is a strong linear predictor of spending.

6. Data Exploration

Data exploration uses visualisations to identify interpretable business patterns. Each chart includes a title, labelled axes and a short interpretation.

6.1 Top 5 States by Total Amount

6.1 Top 5 States by Total Amount

```
# Sum Amount by State and visualize top 5
top_states = df.groupby('State')['Amount'].sum().sort_values(ascending=False).head(5)
top_states.plot(kind='bar', color='skyblue')
plt.title('Top 5 States by Total Amount')
plt.ylabel('Total Amount')
plt.show()
```

Figure 26: Code for ranking the top five states by total Amount.

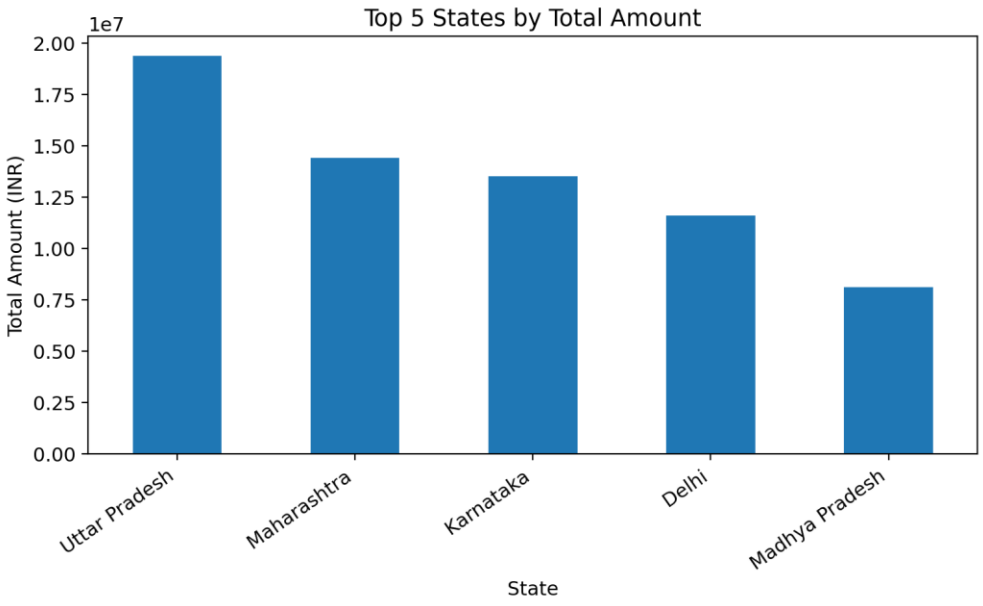


Figure 27: Output for ranking the top five states by total Amount.

Uttar Pradesh generated the highest total Amount at 19,374,968.00, followed by Maharashtra and Karnataka. These states represent the strongest revenue-contributing regions in the dataset and could be prioritised for stock planning, promotional targeting and seasonal marketing.

6.2 Distribution of Gender across Product_Category

6.2 Distribution of Gender across Product_Category

```
# Visualize how gender is distributed across different product categories
plt.figure(figsize=(15,6))
sns.countplot(data=df, x='Product_Category', hue='Gender')
plt.xticks(rotation=45)
plt.title('Gender Distribution across Product Categories')
plt.show()
```

Figure 28: Code for generating visual of distribution across different products categories based on gender

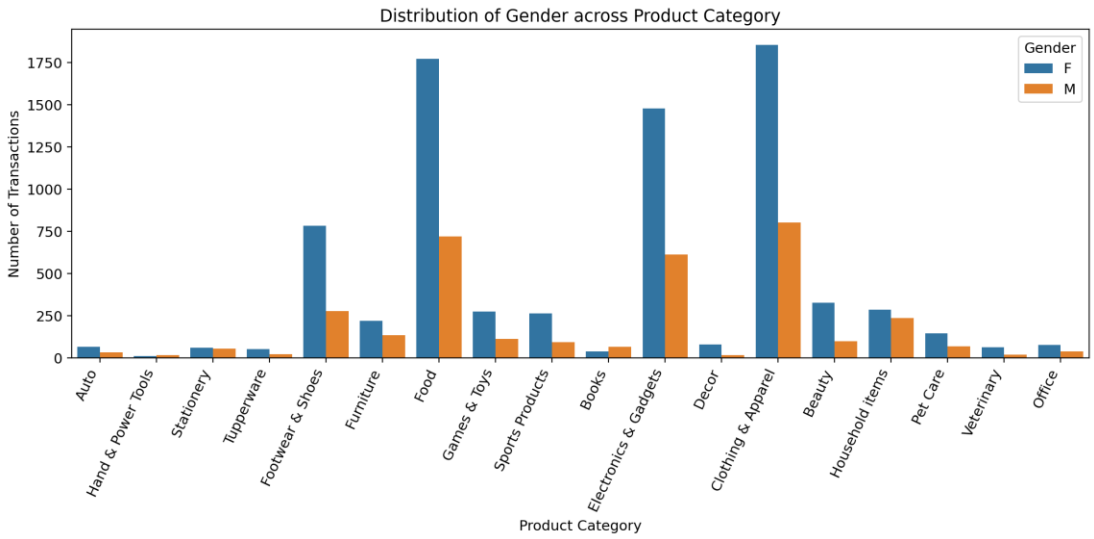


Figure 29: Output of distribution across different products categories based on gender

The dataset contains 7,832 female and 3,407 male transactions after cleaning. Female customers appear more frequently in most high-volume product categories, especially Clothing & Apparel, Food and Electronics & Gadgets. This indicates that gender-based segmentation may be useful for promotional campaigns, although interpretation should consider that the dataset itself contains more female transactions overall.

6.3 Age Group vs Average Amount

6.3 Age Group vs Average Amount

```
# Compare average spend amount for each age cohort
df.groupby('Age Group')['Amount'].mean().plot(kind='bar', color='salmon')
plt.title('Average Amount Spent by Age Group')
plt.ylabel('Average Amount')
plt.show()
```

Figure 30: Code for average Amount by ordinal age group

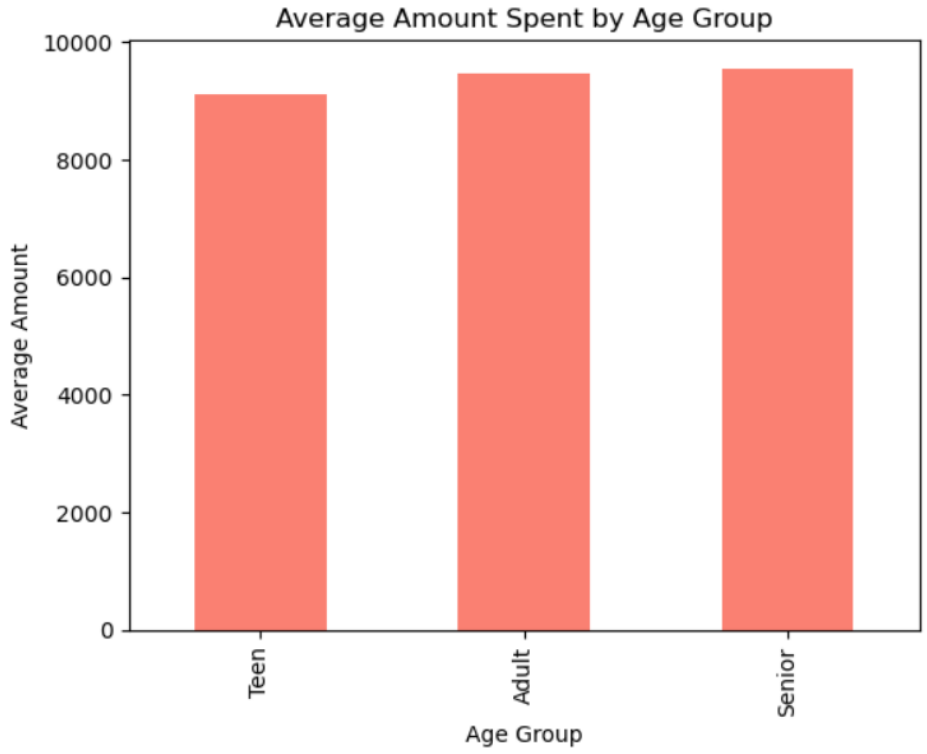


Figure 31: Average Amount by ordinal age group

Average spending increases slightly from Teen (9,120.45) to Adult (9,458.78) and Senior (9,557.35). The differences are not extremely large, but the pattern suggests that older groups may have slightly higher purchasing power or preference for higher-value items.

6.4 Marital_Status Impact on Orders

6.4 Marital_Status Impact on Orders

```
# 1. Permanent mapping: Replace 0 and 1 with text labels
df['Marital_Status'] = df['Marital_Status'].replace({0: 'Unmarried', 1: 'Married'})
```

```
# 2. Plotting: Clean bar chart without error bars or 0/1 labels
plt.figure(figsize=(8,6))
sns.barplot(
    data=df,
    x='Marital_Status',
    y='Orders',
    hue='Marital_Status',
    estimator=np.mean,
    errorbar=None,
    palette='viridis',
    legend=False
)
plt.title('Average Order Frequency by Marital Status')
plt.xlabel('Marital Status')
plt.ylabel('Average Number of Orders')
plt.show()
```

Figure 31: Code for average Orders by Marital_Status.

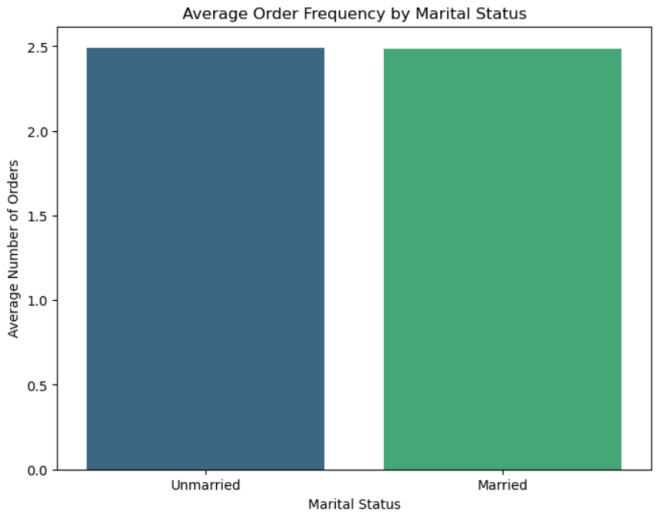


Figure 32: Average orders by marital status.

The average order count is very similar for both groups: unmarried customers average 2.493 orders, while married customers average 2.485 orders. This indicates that marital status has little practical impact on average order quantity in this dataset. However, unmarried customers contribute more total orders (16,249) because there are more unmarried transactions.

6.5 Categorised Analysis with Grouped Bar Chart

The grouped bar chart ranks product categories by average Amount and then compares those high-value categories across zones. To keep the chart readable, the top eight categories by overall average Amount are shown.

5.5 Categorized Analysis: Zone and Product Category

```
# Rank product category by average Amount, grouped by Zone
zone_cat_avg = df.groupby(['Zone', 'Product_Category'])['Amount'].mean().reset_index()
zone_cat_avg = zone_cat_avg.sort_values(['Zone', 'Amount'], ascending=[True, False])
plt.figure(figsize=(16,10))
sns.barplot(data=zone_cat_avg, x='Product_Category', y='Amount', hue='Zone')
plt.xticks(rotation=45)
plt.title('Average Amount by Category Grouped by Zone (Ranked)')
plt.legend(title='Zone', bbox_to_anchor=(1.05, 1), loc='upper left')
plt.tight_layout()
plt.show()
```

Figure 33: Code for product category ranking grouped by Zone.

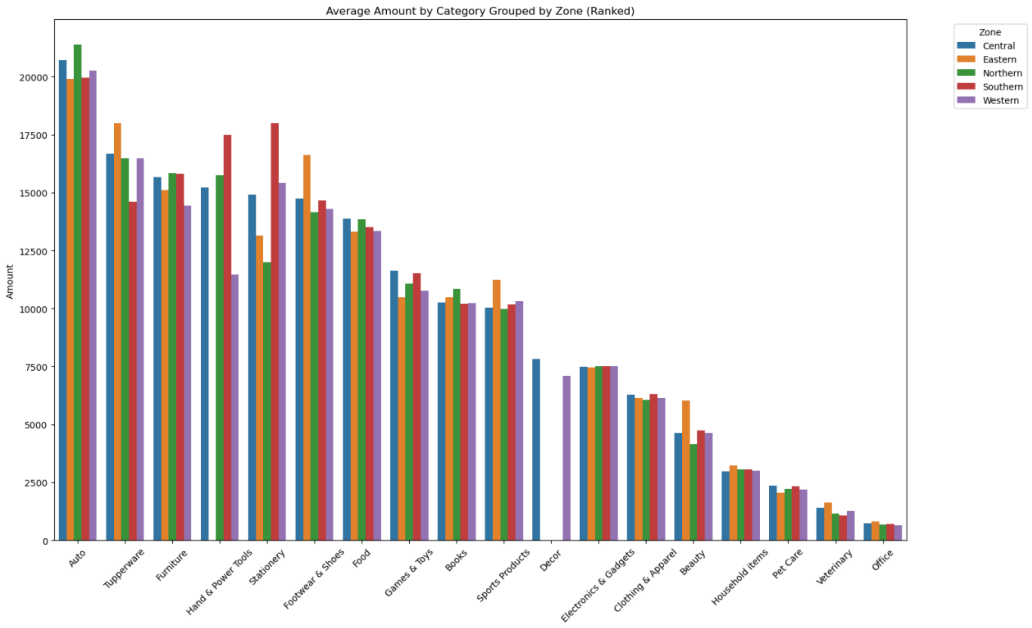


Figure 34: Average Amount by top product categories across zones

Auto has the highest overall average Amount at 20,191.86, followed by Tupperware, Hand & Power Tools, Furniture and Stationery. The grouped chart shows that the average spend varies by zone for several categories, which suggests that regional preferences and product mix influence sales value. These differences could guide targeted regional promotions and zone-specific inventory allocation.

7. Statistical Testing

Hypothesis testing was used to check whether observed patterns were statistically meaningful. **Appendix E: Hypothesis Testing Background** provides additional background on hypothesis testing; this section presents the hypotheses, p-values and conclusions. Detailed statistical and test interpretations are provided in **Appendix F: Detailed Statistical and Test Interpretations.**

7.1 ANOVA Test - Amount by Occupation

The one-way ANOVA test compares the mean Amount across multiple occupation groups.

- H0: The mean Amount is equal across all occupation groups.
- H1: The mean Amount differs significantly across at least two occupation groups.

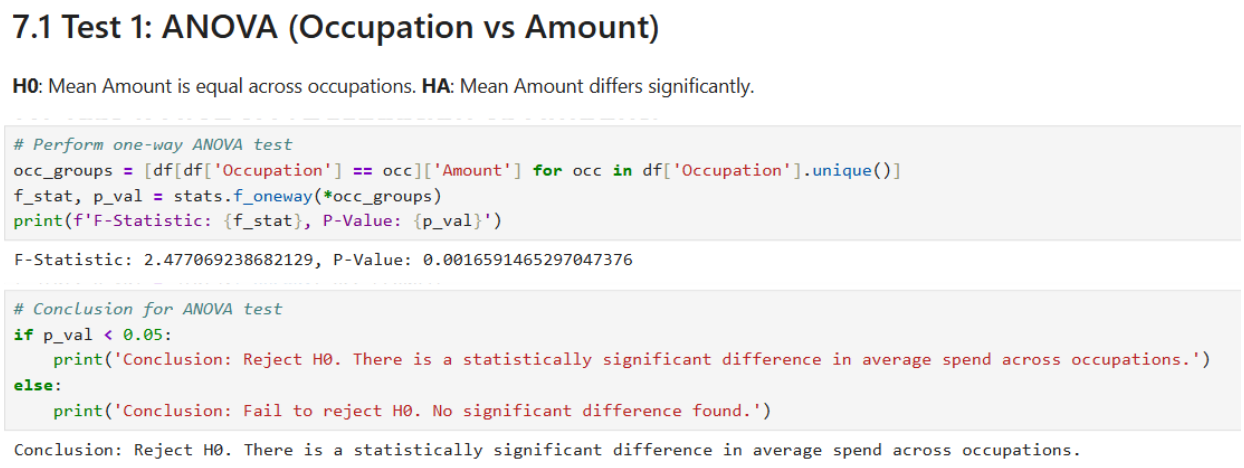


Figure 35: Code and output for the ANOVA test comparing Amount across Occupation groups

The ANOVA F-statistic is 2.477069 and the p-value is 0.001659146530. Since the p-value is below 0.05, the null hypothesis is rejected, showing that average purchase Amount differs across at least two occupation groups.

7.2 Chi-Square Test - Product_Category vs Zone

The Chi-square test of independence checks whether Product_Category and Zone are independent categorical variables.

- H0: There is no association between Product_Category and Zone.
- H1: There is a significant association between Product_Category and Zone.

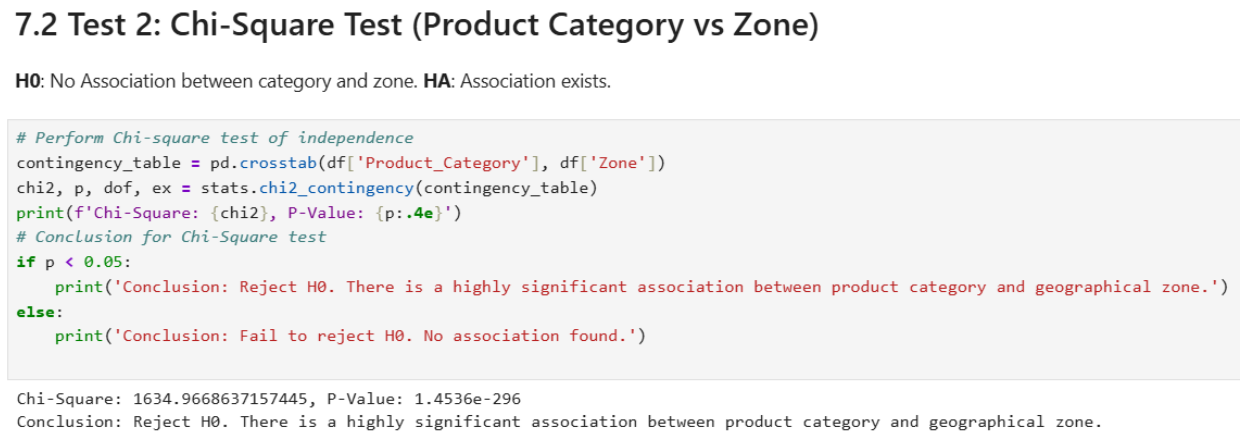


Figure 36: Code and output for the Chi-square test of Product_Category and Zone.

The Chi-square statistic is 1634.966864, the p-value is 1.454e-296, and the degrees of freedom are 68. Since the p-value is far below 0.05, the null hypothesis is rejected, showing a significant association between Product_Category and Zone.

Conclusion

The Diwali Sales Transaction dataset was analysed to understand customer purchasing behaviour, sales performance, and demographic, geographic and product-related patterns. Preparation improved reliability by removing irrelevant columns, handling missing values and transforming variables into meaningful formats.

The analysis showed that customer spending is unevenly distributed, with most customers making moderate purchases and a few making high-value transactions. Correlation results showed that Age and Orders have negligible linear relationships with Amount, so grouped patterns and statistical tests provide stronger insight than simple pairwise correlations.

Visual analysis showed that sales contributions vary by state, gender, age group, product category and region. These insights support marketing, inventory planning and customer targeting.

Statistical tests supported the findings: ANOVA showed significant differences in Amount across occupations, and Chi-square showed a strong association between Product_Category and Zone. Overall, occupation, region and product preferences are important for understanding Diwali sales performance.

References

Bruce, P., Bruce, A. and Gedeck, P. (2020) *Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python*. 2nd edn. Sebastopol: O'Reilly Media.

McKinney, W. (2022) *Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter*. 3rd edn. Sebastopol: O'Reilly Media.

Osborne, J.W. (2013) *Best Practices in Data Cleaning: A Complete Guide to Everything You Need to Do Before and After Collecting Your Data*. Los Angeles: SAGE Publications.

Waskom, M.L. (2021) 'Seaborn: Statistical data visualization', *Journal of Open Source Software*, 6(60), p. 3021. doi: 10.21105/joss.03021.

Appendix

The appendix contains supporting methodological detail, extended definitions and full tables used to support the main analysis.

Appendix A: Detailed Objectives and Data Challenges

Detailed objectives of the analysis:

- Identify the data source, data types, memory usage and potential quality issues.
- Prepare the dataset using correct encoding and inspect its structure using Python.
- Create useful categories, remove irrelevant columns and handle missing values.
- Compute descriptive statistics, skewness, kurtosis and correlations.
- Visualise sales patterns by state, gender, age group, marital status, product category and zone.
- Apply ANOVA and Chi-square tests to support analytical conclusions.

Detailed data-quality challenges:

- Amount contains 12 missing values and is required for revenue-based analysis.
- Status and unnamed1 contain no meaningful values and should be removed.
- Cust_name and User_ID identify individuals and are not required for aggregate analysis.
- Age Group has a meaningful order and should be converted into an ordinal categorical variable.
- Product_Category has 18 categories, so visualisations need readable labels and careful grouping.
- The numerical variables show only weak pairwise relationships overall. Age and Amount have r = 0.031, while Orders and Amount have r = -0.013.

Appendix B: Techniques for Managing Missing Data

Technique	Description	Example
Mean imputation	Replaces missing numerical values with the column average.	A missing Age value could be filled with the average age of all other customers.
Median imputation	Replaces missing values with the middle value of the column and is less affected by outliers.	For Amount, the median can be safer than the mean when a few festival purchases are very high.
Mode imputation	Replaces missing values with the most frequent category.	A missing Gender or Product_Category value could be filled with the most common category.
Dropping rows	Removes records where critical values are missing.	A row with missing Amount was removed because it cannot support revenue analysis.

Appendix C: Summary Statistics Definitions

Measure	Meaning	Interpretation in analysis
Mean	The mathematical average of a variable.	Useful for estimating the typical value, such as average purchase Amount.
Median	The middle value after sorting the data.	Useful when outliers may distort the mean.
Mode	The most frequently occurring value.	Useful for common order quantities or common categories.

Skewness	Measures whether a distribution is symmetrical or pulled to one side.	Positive skew means a few high values pull the mean upward.
Positive skew	A distribution with a long right-hand tail.	Most values are lower, but some high values exist.
Negative skew	A distribution with a long left-hand tail.	Most values are higher, with some low values.
Zero skew	A distribution that is close to symmetrical.	Values are balanced around the mean.
Kurtosis	Measures the peak and tail weight of a distribution.	Helps identify whether extreme values are more likely.
Leptokurtic	A sharper peak with heavier tails.	Indicates a higher chance of extreme values.
Platykurtic	A flatter distribution with thinner tails.	Indicates fewer extreme values.
Mesokurtic	A moderate distribution similar to normal distribution.	Indicates average tail weight.

Appendix D: Correlation Concepts and Calculation Methods

Concept / Method	**Explanation**	**Use in this analysis**
Correlation matrix	A table showing correlation coefficients between numerical variables.	Used to compare Age, Orders and Amount.
Heatmap	A visual display where colour intensity represents relationship strength.	Used to identify relationship patterns quickly.
Pearson correlation	Measures linear relationships between continuous variables.	Used to assess Age vs Amount and Orders vs Amount.

Spearman correlation	Measures monotonic relationships using ranked values.	Useful when variables are ordinal or not linearly related.

Appendix E: Hypothesis Testing Background

Hypothesis testing is a statistical process used to decide whether patterns found in sample data are likely to represent real differences or associations. A null hypothesis (H0) states that no difference or association exists, while an alternative hypothesis (H1) states that a difference or association exists. In business analysis, hypothesis testing supports evidence-based decisions, such as whether customer groups spend differently or whether product preferences vary by region.

Term	**Meaning**
p-value	The probability of observing the result, or a more extreme result, if the null hypothesis is true.
F-statistic	The ANOVA test statistic comparing variation between groups with variation within groups.
Chi-square statistic	A test statistic measuring the difference between observed and expected frequencies.
Degrees of freedom	The number of independent comparisons available in the statistical test.
Expected values	Frequencies expected if the two categorical variables were independent.

Appendix F: Detailed Statistical and Test Interpretations

Area	Detailed interpretation
Amount	Amount is positively skewed, suggesting that most customers spend a moderate amount while a smaller group of high-value shoppers pulls the mean upward. The leptokurtic pattern indicates concentration around certain price points with some extreme high-value purchases.
Age	The Age distribution is relatively balanced, meaning the dataset captures a broad range of shoppers and supports demographic comparison.
Orders	Orders is a discrete transaction-count variable. The mode is useful because it identifies the most common purchase volume per customer.
Orders and Amount	The correlation is negligible ($r \approx -0.013$), indicating that order quantity does not have a meaningful linear relationship with total spending in this dataset.
Age and Amount	The correlation is negligible and positive ($r \approx 0.031$), meaning age is not a reliable predictor of total purchase Amount.
Overall correlations	The numerical variables do not show strong relationships with Amount, so other factors such as product type, occupation and region are more useful for interpretation.
ANOVA result	The F-statistic is 2.477069 and the p-value is 0.001659146530. Since the p-value is below 0.05, the null hypothesis is rejected and average Amount differs across at least two occupation groups.
Chi-square result	The Chi-square statistic is 1634.966864, the p-value is 1.454e-296 and the degrees of freedom are 68. Since the p-value is far below 0.05, Product_Category and Zone are significantly associated.

Appendix G: Full Data Description Table

Column Name	Data Type	Description	Potential Issues
User_ID	Integer	Unique identifier for each customer.	May be dropped.
Cust_name	String	Full name of the customer.	To be dropped for privacy.
Product_ID	String	Unique identifier for the product purchased.	None.
Gender	String	Gender of the customer.	None.
Age Group	Categorical	Age range of the customer.	Needs ordinal mapping.
Age	Integer	Actual age of the customer.	Potential outliers.
Marital_Status	Binary	0 for Single, 1 for Married.	None.
State	String	Location of the customer.	None.
Zone	String	Regional zone.	None.
Occupation	String	Profession of the customer.	None.
Product_Category	String	Broad category of the product purchased.	18 unique categories.
Orders	Integer	Number of items purchased in the transaction.	Indicates buying behaviour.
Amount	Float	Total sales revenue for the transaction.	Missing values identified.
Status	Null	Empty column with no data.	To be dropped.
unnamed1	Null	Artifact column with no meaningful data.	To be dropped.

Appendix H: Business Relevance of Key Columns

Column	Business Relevance
Amount	Represents total revenue generated from each transaction and acts as the primary metric for evaluating sales performance across customer segments and regions.
Orders	Indicates the number of items purchased in one transaction and helps identify bulk-buying behaviour during the Diwali sales period.
Gender and Age Group	Provide demographic insights that help identify customer segments with different purchasing patterns and support targeted marketing.
State and Zone	Provide geographic information for analysing regional sales trends and supporting inventory distribution in high-demand areas.
Occupation	Reflects professional background and helps identify spending patterns that may relate to income or lifestyle differences.