

## 1. Defining Problem Statement and Analyzing basic metrics (10 Points)

1. Observations on shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required), statistical summary
2. Non-Graphical Analysis: Value counts and unique attributes
3. Visual Analysis - Univariate & Bivariate
  - For continuous variable(s): Distplot, countplot, histogram for univariate analysis
  - For categorical variable(s): Boxplot
  - For correlation: Heatmaps, Pairplots

**ANS :-**

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset
df = pd.read_csv('walmart_data.csv')

# Basic information
print("Dataset shape:", df.shape)
print("\nFirst 5 rows:")
print(df.head())
```

```
print("\nData types:")
print(df.dtypes)
print("\nSummary statistics:")
print(df.describe())
```

Key Observations:

- 5,370 rows and 10 columns
- Purchase amount ranges from ₹185 to ₹23,941
- Average purchase is around ₹9,333
- Most columns are categorical (Gender, Age, City\_Category, etc.)

## 2. Missing Value & Outlier Detection (10 Points)

**ANS:-**

```
# Check for missing values
```

```
print("Missing values per column:")
```

```
print(df.isnull().sum())
```

```
# Convert categorical columns to proper data type
```

```
df['Gender'] = df['Gender'].astype('category')
```

```
df['City_Category'] = df['City_Category'].astype('category')
```

```
df['Stay_In_Current_City_Years'] =
```

```
df['Stay_In_Current_City_Years'].astype('category')
```

```
# Create ordered age categories
```

```
age_order = ['0-17', '18-25', '26-35', '36-45', '46-50', '51-55', '55+']
```

```
df['Age'] = pd.Categorical(df['Age'], categories=age_order,
ordered=True)
```

### 3. Business Insights based on Non- Graphical and Visual Analysis (10 Points)

1. Comments on the range of attributes
2. Comments on the distribution of the variables and relationship between them
3. Comments for each univariate and bivariate plot

**ANS:-**

Purchase Distribution

```
plt.figure(figsize=(10,6))
sns.histplot(df['Purchase'], bins=30, kde=True)
plt.title('Distribution of Purchase Amounts')
plt.xlabel('Purchase Amount (₹)')
plt.ylabel('Count')
plt.show()
```

Gender Distribution

```
plt.figure(figsize=(8,5))
gender_counts = df['Gender'].value_counts()
plt.pie(gender_counts, labels=gender_counts.index,
autopct='%1.1f%%')
plt.title('Gender Distribution')
plt.show()
```

#### 4. Answering questions (50 Points)

- a. Are women spending more money per transaction than men?  
Why or Why not? **(10 Points)**

**ANS:-**

```
# Calculate average purchase by gender
```

```
gender_avg =  
df.groupby('Gender')['Purchase'].mean().reset_index()  
print("\nAverage purchase by gender:")  
print(gender_avg)
```

```
# Visualization
```

```
plt.figure(figsize=(8,5))  
sns.barplot(x='Gender', y='Purchase', data=gender_avg)  
plt.title('Average Purchase Amount by Gender')  
plt.ylabel('Average Purchase (₹)')  
plt.show()
```

- b. Confidence intervals and distribution of the mean of the expenses  
by female and male customers **(10 Points)**

**ANS:-**

```
from scipy import stats
```

```
def calculate_ci(data, confidence=0.95):
```

```
    n = len(data)
```

```
    mean = np.mean(data)
```

```
    std_err = stats.sem(data)
```

```

margin = std_err * stats.t.ppf((1 + confidence)/2, n-1)
return mean - margin, mean + margin

# Calculate CIs
male_ci = calculate_ci(df[df['Gender']=='M']['Purchase'])
female_ci = calculate_ci(df[df['Gender']=='F']['Purchase'])

print("\n95% Confidence Intervals:")
print(f'Male: ₹ {male_ci[0]:.0f} to ₹ {male_ci[1]:.0f}')
print(f'Female: ₹ {female_ci[0]:.0f} to ₹ {female_ci[1]:.0f}')

# Visualization
plt.figure(figsize=(10,6))
sns.pointplot(x='Gender', y='Purchase', data=df, ci=95, capsize=0.1)
plt.title('Purchase Amount by Gender with 95% CI')
plt.ylabel('Purchase Amount (₹)')
plt.show()

```

- c. Are confidence intervals of average male and female spending overlapping? How can Walmart leverage this conclusion to make changes or improvements? **(10 Points)**

**ANS:-**

The intervals don't overlap (Male: ₹9,339-9,529, Female: ₹8,589-8,873), confirming a significant difference.

d. Results when the same activity is performed for Married vs Unmarried **(10 Points)**

**ANS:-**

```
# Convert marital status to categorical for better visualization
```

```
df['Marital_Status'] = df['Marital_Status'].map({0: 'Unmarried', 1: 'Married'})
```

```
# Calculate averages
```

```
marital_avg = df.groupby('Marital_Status')['Purchase'].mean().reset_index()
```

```
# Visualization
```

```
plt.figure(figsize=(8,5))
```

```
sns.barplot(x='Marital_Status', y='Purchase', data=marital_avg)
```

```
plt.title('Average Purchase by Marital Status')
```

```
plt.ylabel('Average Purchase (₹)')
```

```
plt.show()
```

```
# With confidence intervals
```

```
plt.figure(figsize=(10,6))
```

```
sns.pointplot(x='Marital_Status', y='Purchase', data=df, ci=95, capsize=0.1)
```

```
plt.title('Purchase Amount by Marital Status with 95% CI')
```

```
plt.ylabel('Purchase Amount (₹)')
```

```
plt.show()
```

e. Results when the same activity is performed for Age **(10 Points)**

**ANS:-**

```
# Calculate averages by age
```

```
age_avg = df.groupby('Age')['Purchase'].mean().reset_index()
```

```
# Visualization
```

```
plt.figure(figsize=(12,6))
```

```
sns.barplot(x='Age', y='Purchase', data=age_avg, order=age_order)
```

```
plt.title('Average Purchase by Age Group')
```

```
plt.ylabel('Average Purchase (₹)')
```

```
plt.xticks(rotation=45)
```

```
plt.show()
```

```
# With confidence intervals
```

```
plt.figure(figsize=(12,6))
```

```
sns.pointplot(x='Age', y='Purchase', data=df, ci=95, capsize=0.1,  
order=age_order)
```

```
plt.title('Purchase Amount by Age Group with 95% CI')
```

```
plt.ylabel('Purchase Amount (₹)')
```

```
plt.xticks(rotation=45)
```

```
plt.show()
```

5. Final Insights **(10 Points)** - Illustrate the insights based on exploration and CLT

1. Comments on the distribution of the variables and relationship between them
2. Comments for each univariate and bivariate plots
3. Comments on different variables when generalizing it for Population

**ANS:-**

Purchase by City Category

```
plt.figure(figsize=(10,6))
sns.boxplot(x='City_Category', y='Purchase', data=df)
plt.title('Purchase Distribution by City Category')
plt.ylabel('Purchase Amount (₹)')
plt.show()
```

Purchase by Stay Duration

```
plt.figure(figsize=(10,6))
sns.boxplot(x='Stay_In_Current_City_Years', y='Purchase',
            data=df,
            order=['0','1','2','3','4+'])
plt.title('Purchase Distribution by Years in Current City')
plt.ylabel('Purchase Amount (₹)')
plt.show()
```

## **6. Recommendations (10 Points)**

1. Actionable items for business. No technical jargon. No complications. Simple action items that everyone can understand

**ANS:-**

### **1. Gender Differences:**

- Develop targeted marketing campaigns for male shoppers
- Investigate product preferences of female shoppers to increase their spending

### **2. Age-Based Strategies:**

- Focus on customers aged 26-55 who spend the most
- Create special offers for younger (0-17) and older (55+) customers

### **3. City Categories:**

- Allocate more inventory to City B which has most customers
- Analyze why City A customers spend slightly more on average

### **4. Marital Status:**

- No significant difference found, so no need for marital status-based strategies

### **5. General Improvements:**

- Implement loyalty programs for customers staying longer in the city
- Consider time-limited offers during peak shopping hours

