



```
# **FUTURESKILLS AI BOOTCAMP ASSIGNMENT 2**
```

```
---
```

```
**Problem Statement:**
```

```
Analyze a retail sales dataset to explore relationships between sales, profit, quantity, and discounts using data preprocessing, EDA, and correlation analysis for better business insights.
```

```
---
```

## FUTURESKILLS AI BOOTCAMP ASSIGNMENT 2

### Problem Statement:

Analyze a retail sales dataset to explore relationships between sales, profit, quantity, and discounts using data preprocessing, EDA, and correlation analysis for better business insights.

We will import pandas for data handling and Matplotlib/Seaborn for visualization.

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

To Read the Excel file into a DataFrame and display basic information.

```
df = pd.read_excel('/content/sales_data.csv.xlsx') # Update path if needed
print(df.head()) # Show the first few rows
print(df.info()) # Display column details
```

```
➡
```

	Order ID	Product	Category	Sales	Quantity	Profit	Discount	\
0	101	Laptop	Electronics	1500	2	300	0.10	
1	102	Smartphone	Electronics	700	3	150	0.20	
2	103	Desk	Furniture	200	1	50	0.15	
3	104	Chair	Furniture	100	4	20	0.10	
4	105	Television	Electronics	800	1	120	0.05	

	Region	Date
0	East	2023-01-15
1	West	2023-02-10
2	South	2023-02-20
3	North	2023-03-05
4	East	2023-04-12

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Order ID    50 non-null    int64
1   Product     50 non-null    object
2   Category    50 non-null    object
3   Sales       50 non-null    int64
4   Quantity    50 non-null    int64
5   Profit      50 non-null    int64
6   Discount    50 non-null    float64
7   Region      50 non-null    object
8   Date        50 non-null    datetime64[ns]
dtypes: datetime64[ns](1), float64(1), int64(4), object(3)
memory usage: 3.6+ KB
None
```

To display all column names

```
print(df.columns) # Display all column names
```

```
Index(['Order ID', 'Product', 'Category', 'Sales', 'Quantity', 'Profit',  
      'Discount', 'Region', 'Date'],  
      dtype='object')
```

## Bar Plot

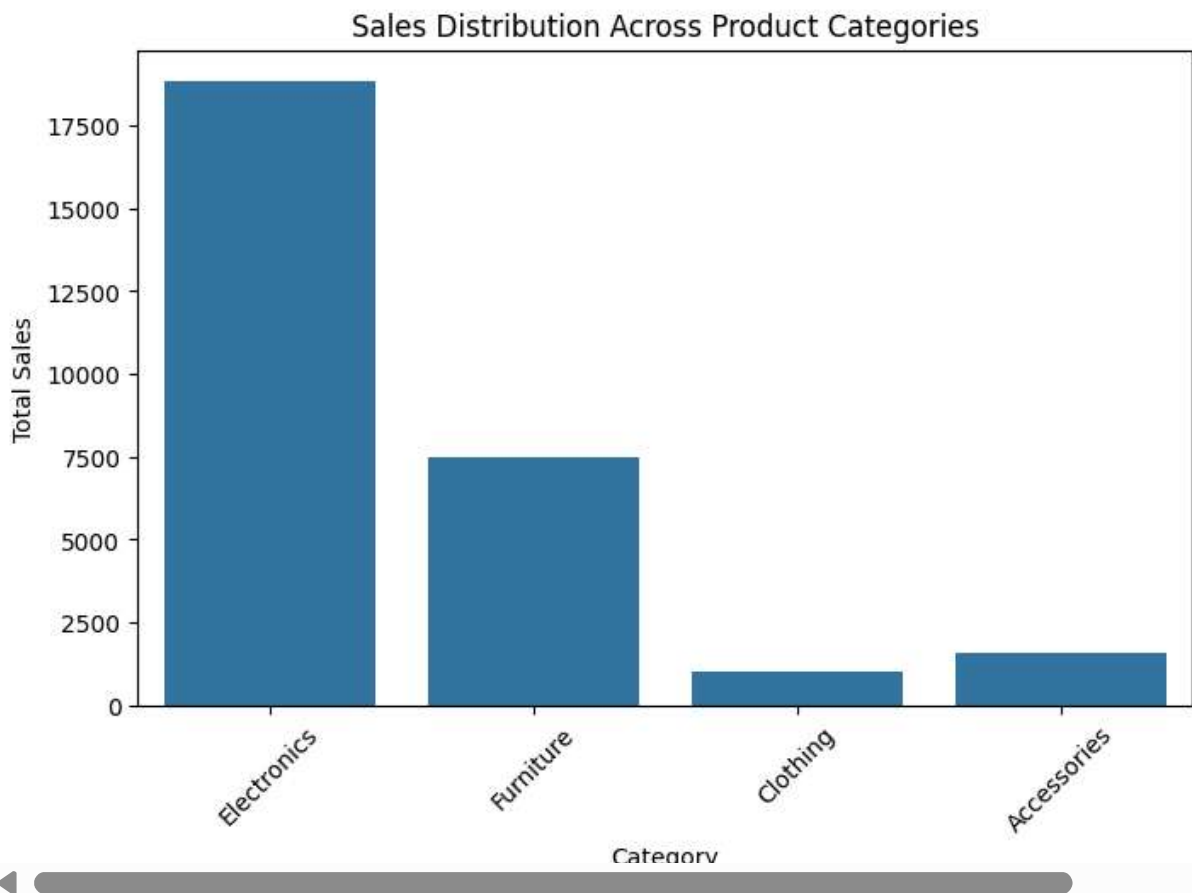
To show total sales for each Category.

```
plt.figure(figsize=(8, 5))  
sns.barplot(x=df['Category'], y=df['Sales'], estimator=sum, ci=None)  
plt.xticks(rotation=45)  
plt.title("Sales Distribution Across Product Categories")  
plt.xlabel("Category")  
plt.ylabel("Total Sales")  
plt.show()
```

<ipython-input-7-619f1feadfd8>:2: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

```
sns.barplot(x=df['Category'], y=df['Sales'], estimator=sum, ci=None)
```

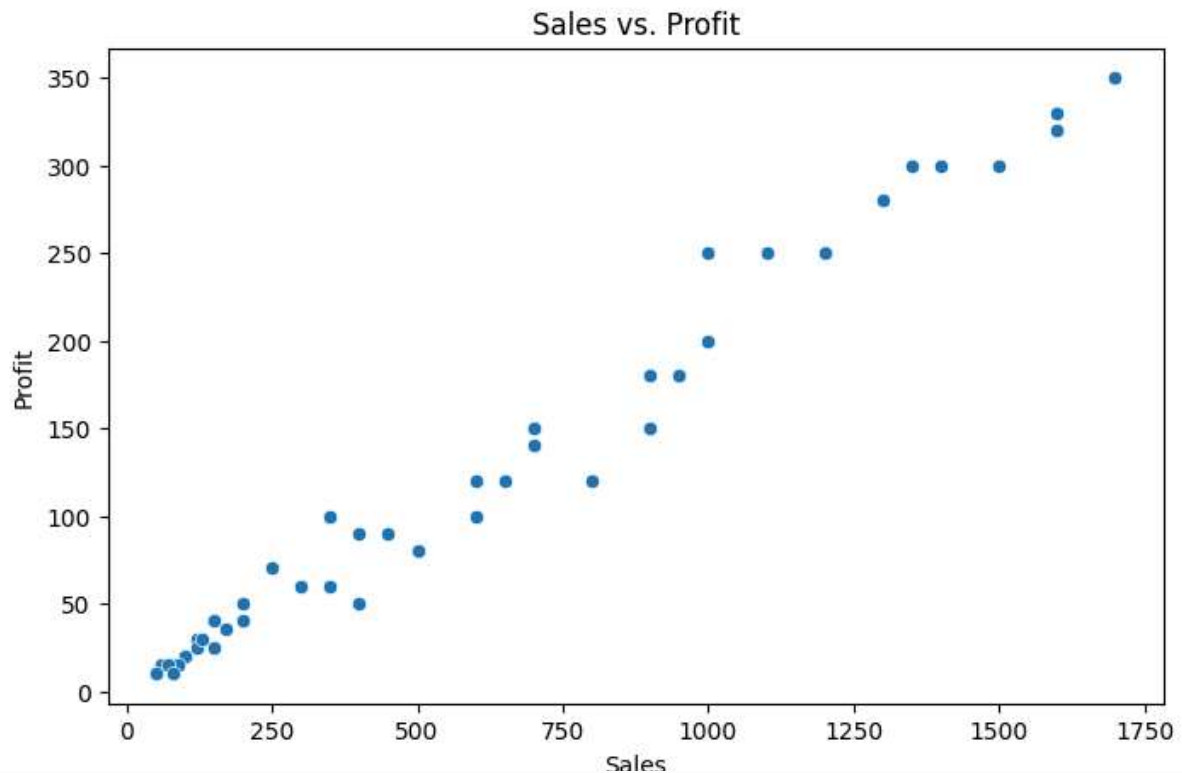


## Scatter Plot

To Visualise the relationship between Sales and Profit.

```
plt.figure(figsize=(8, 5))  
sns.scatterplot(x=df['Sales'], y=df['Profit'])
```

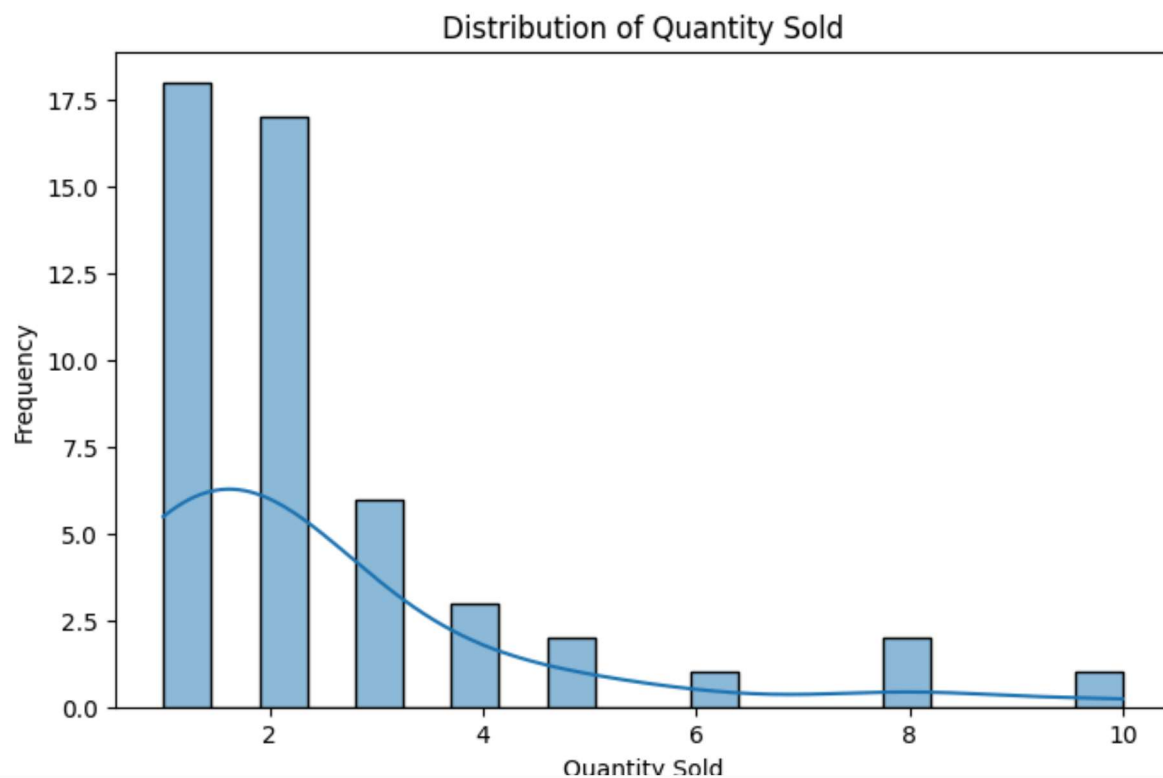
```
plt.title("Sales vs. Profit")  
plt.xlabel("Sales")  
plt.ylabel("Profit")  
plt.show()
```



## Histogram

To show how Quantity sold is distributed.

```
plt.figure(figsize=(8, 5))  
sns.histplot(df['Quantity'], bins=20, kde=True)  
plt.title("Distribution of Quantity Sold")  
plt.xlabel("Quantity Sold")  
plt.ylabel("Frequency")  
plt.show()
```



### Box Plot

To compare Discounts across different Regions.

```
plt.figure(figsize=(8, 5))
sns.boxplot(x=df['Region'], y=df['Discount'])
plt.xticks(rotation=45)
plt.title("Discount Comparison by Region")
plt.xlabel("Region")
plt.ylabel("Discount")
plt.show()
```



## Discount Comparison by Region

To show the correlation between numerical columns.

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

# Select only numerical columns
df_numeric = df.select_dtypes(include=['number'])

# Plot heatmap
plt.figure(figsize=(10, 6))
sns.heatmap(df_numeric.corr(), annot=True, cmap="coolwarm", fmt=".2f", linewidths=0.5)
plt.title("Correlation Heatmap")
plt.show()
```

