E-Commerce Sales Analytics

A CASE STUDY REPORT

Submitted by

SUBHASH J [RA2212703010006]

For the course

Data Science - 21CSS303T

In partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

DEPARTMENT OF NETWORKING AND COMMUNICATIONS
SCHOOL OF COMPUTING
FACULTY OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR - 603 203.
MAY 2025

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY KATTANKULATHUR – 603 203

BONAFIDE CERTIFICATE

Certified that this report titled **“ E-Commerce Sales Analytics ”** is the bonafide work of **“SUBHASH J [RA2212703010006]”** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE	**SIGNATURE**
Dr. SWATHY R	**Dr. LAKSHMI M**
ASSISTANT PROFESSOR	PROFESSOR & HEAD
DEPARTMENT OF NETWORKING	DEPARTMENT OF NETWORKING
AND COMMUNICATIONS	AND COMMUNICATIONS

DATE: 8 MAY 2025

ABSTRACT

In the digital age, e-commerce has become a cornerstone of modern retail, generating vast amounts of data from customer transactions, product sales, and regional performance. This project, *E-Commerce Sales Analytics*, aims to extract meaningful insights from such sales data using Python and data visualization techniques. By examining variables such as sales amount, profit, quantity, and purchase trends across states, months, and days of the week, this analysis helps identify high-performing products, peak sales periods, and underperforming areas.

The project employs libraries like **Pandas** and **NumPy** for data manipulation, and **Matplotlib** and **Seaborn** for visualizations. It includes preprocessing steps such as missing value treatment and outlier removal using the **Interquartile Range (IQR)** method, enhancing data quality.

The overall goal is to provide a data-driven foundation for business decisions, such as inventory planning, marketing strategies, and customer targeting. Through comprehensive visual analytics, this project bridges raw data and actionable intelligence, offering valuable insights to optimize performance in a competitive online retail landscape.

TABLE OF CONTENTS

Chapter 1
INTRODUCTION

In the fast-paced environment of today's digital economy, e-commerce businesses generate massive volumes of transactional data. This data holds untapped potential for enhancing business strategy, optimizing operations, and improving customer satisfaction. However, to make informed decisions, businesses must go beyond raw data and focus on extracting meaningful insights through structured analysis.

This project focuses on analyzing historical sales data from an e-commerce platform to better understand sales performance across several critical dimensions. The dataset includes detailed information on product categories, quantities sold, monetary amounts, and profit margins, along with associated temporal and geographic identifiers such as purchase dates and customer states.

By breaking down this data across **time (months and weekdays)**, **geographic regions (states)**, and **product segments**, the project provides a multi-angle view of business operations. For instance, identifying the most profitable days or months, locating top-performing states, and understanding which products contribute most to revenue can directly influence decision-making in areas such as **inventory management**, **marketing campaigns**, and **targeted customer outreach**.

The methodology employed combines data preprocessing, outlier detection, and normalization techniques with visual tools to interpret patterns and anomalies. Visualization libraries like Matplotlib and Seaborn aid in converting numerical data into easy-to-understand graphical representations, while Pandas and NumPy handle the data manipulation.

Overall, the goal is to build a data-driven foundation for e-commerce business planning. By turning complex datasets into accessible, actionable insights, this project empowers stakeholders to understand performance, address challenges, and capitalize on opportunities in an increasingly competitive online marketplace.

Chapter 2
BACKGROUND AND MOTIVATION

The rapid growth of e-commerce has revolutionized how consumers shop and how businesses operate. Online platforms now manage vast product catalogs, serve diverse customer bases, and operate across multiple geographic locations. As competition intensifies, the ability to understand customer behavior and sales trends through data analytics has become a strategic necessity rather than a luxury.

Many businesses collect vast amounts of transactional data daily—ranging from order details and pricing to geographic and temporal information. However, simply possessing this data is not enough. The real value lies in extracting insights that can lead to **better forecasting, optimized inventory management, personalized marketing**, and ultimately, **increased profitability**.

This project is driven by the need to bridge the gap between raw data and actionable intelligence. The motivation stems from common challenges faced in the e-commerce domain, such as:

- **Identifying high-performing product categories** and low-performing ones.
- **Determining peak sales periods** (e.g., specific months or days).
- **Assessing profitability and revenue drivers** across regions.
- **Detecting outliers or inconsistencies** that may indicate operational inefficiencies or fraud.

By leveraging modern data analysis techniques, this project aims to uncover patterns that are not immediately visible. These patterns can provide clear direction for strategic actions like restocking decisions, sales forecasting, and marketing spend allocation. The goal is to turn descriptive statistics into prescriptive solutions that help e-commerce platforms stay agile and customer-focused in a dynamic market.

The motivation behind this work is not just academic; it is deeply practical. It acknowledges the reality that in the data-rich world of e-commerce, those who can interpret and act on their data effectively will maintain a significant edge over their competitors.

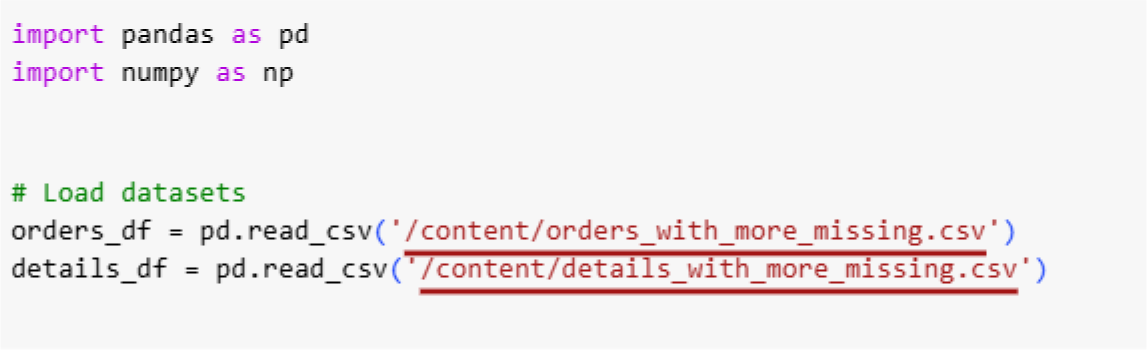
Chapter 3

LIBRARIES AND TECHNOLOGY USED

To carry out the e-commerce sales analysis effectively, we used a suite of Python libraries well-suited for data processing, analysis, and visualization. These tools provided a robust foundation for deriving actionable insights from complex datasets.

1. Pandas

Pandas is central to our data handling. It allows for efficient reading, filtering, grouping, and aggregation of large datasets. It was used extensively to create pivot tables, calculate totals, and prepare data for visualization and modeling.



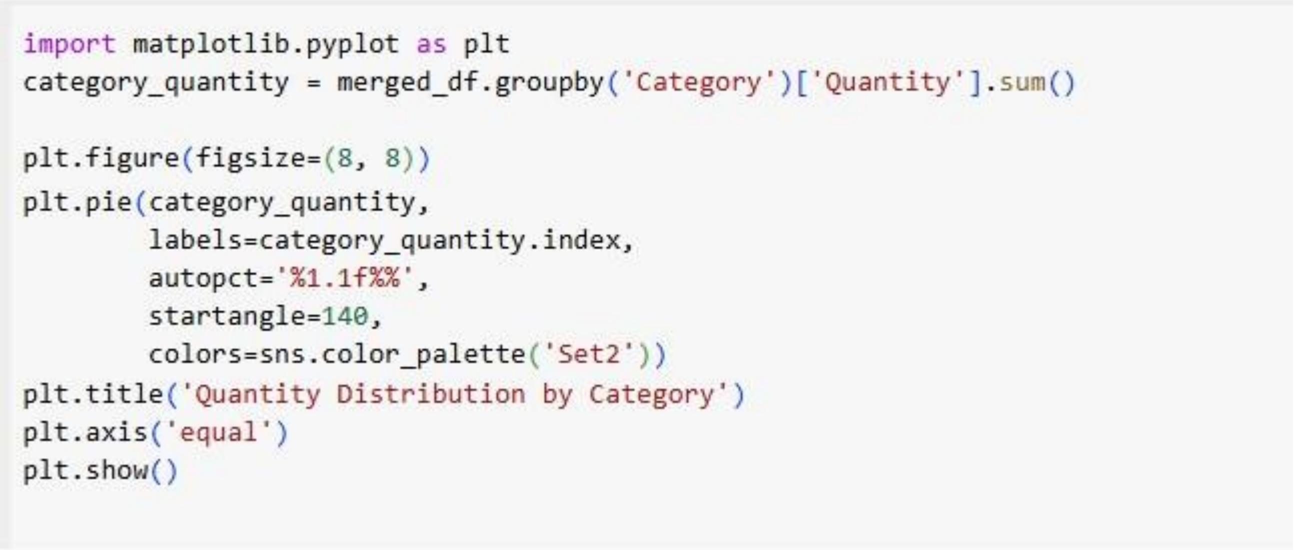
```python
import pandas as pd
import numpy as np


# Load datasets
orders_df = pd.read_csv('/content/orders_with_more_missing.csv')
details_df = pd.read_csv('/content/details_with_more_missing.csv')
```

2. NumPy

NumPy complements Pandas by enabling fast numerical computations. It was used to calculate measures like interquartile ranges (IQR) for outlier detection and to support array-based transformations and filtering.

3. Matplotlib

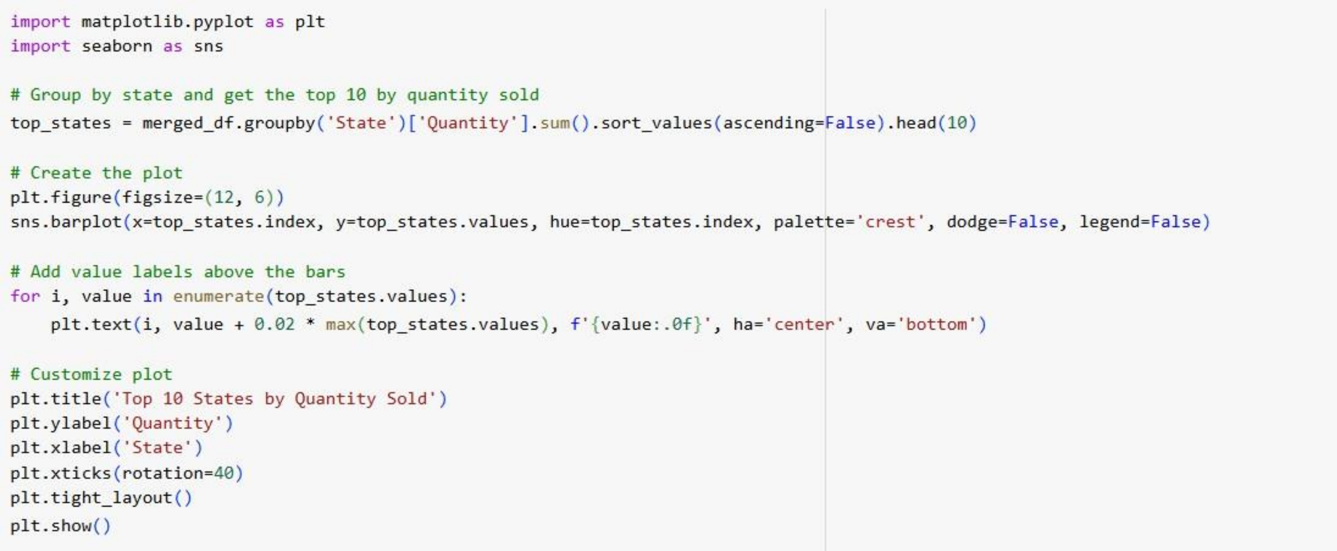
Matplotlib was employed to generate clear and customizable visualizations such as bar charts and line graphs. It provides full control over plot aesthetics, making it ideal for precise presentation of sales trends and comparisons.



```python
import matplotlib.pyplot as plt
category_quantity = merged_df.groupby('Category')['Quantity'].sum()

plt.figure(figsize=(8, 8))
plt.pie(category_quantity,
        labels=category_quantity.index,
        autopct='%1.1f%%',
        startangle=140,
        colors=sns.color_palette('Set2'))
plt.title('Quantity Distribution by Category')
plt.axis('equal')
plt.show()
```

4. Seaborn

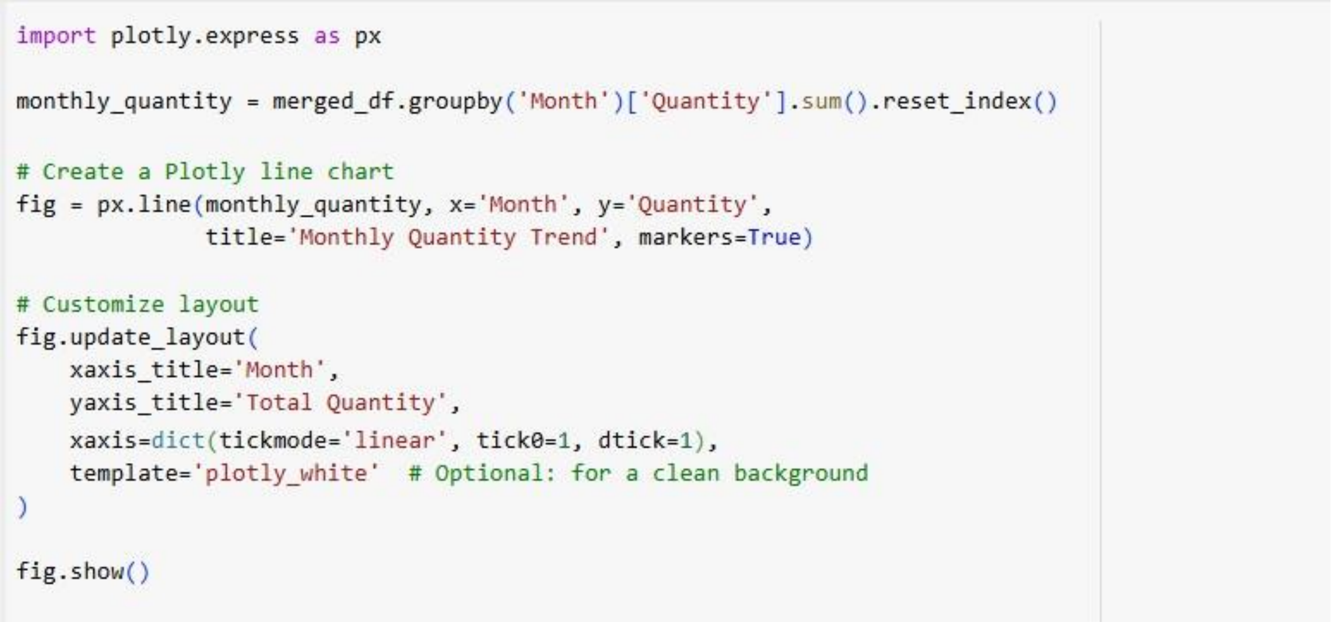
Seaborn enhanced our data visualizations by offering intuitive syntax for creating statistical plots like heatmaps and boxplots. It helped in uncovering patterns, correlations, and anomalies in the data, especially in multi-dimensional views (e.g., weekday vs. month analysis).



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

# Group by state and get the top 10 by quantity sold
top_states = merged_df.groupby('State')['Quantity'].sum().sort_values(ascending=False).head(10)

# Create the plot
plt.figure(figsize=(12, 6))
sns.barplot(x=top_states.index, y=top_states.values, hue=top_states.index, palette='crest', dodge=False, legend=False)

# Add value labels above the bars
for i, value in enumerate(top_states.values):
    plt.text(i, value + 0.02 * max(top_states.values), f'{value:.0f}', ha='center', va='bottom')

# Customize plot
plt.title('Top 10 States by Quantity Sold')
plt.ylabel('Quantity')
plt.xlabel('State')
plt.xticks(rotation=40)
plt.tight_layout()
plt.show()
```

5. Plotly

Plotly was used to create interactive and visually rich charts such as dynamic bar graphs and line plots. It enables users to hover over points, zoom, and interact with charts, which is helpful in exploring trends and making presentations more engaging.



```python
import plotly.express as px

monthly_quantity = merged_df.groupby('Month')['Quantity'].sum().reset_index()

# Create a Plotly line chart
fig = px.line(monthly_quantity, x='Month', y='Quantity',
              title='Monthly Quantity Trend', markers=True)

# Customize layout
fig.update_layout(
    xaxis_title='Month',
    yaxis_title='Total Quantity',
    xaxis=dict(tickmode='linear', tick0=1, dtick=1),
    template='plotly_white'  # Optional: for a clean background
)

fig.show()
```

Chapter 4

PROPOSED WORK

The E-Commerce Sales Analytics project leverages Python to process and analyze transactional data from an online retail platform. The goal is to extract meaningful insights about sales performance, product profitability, and customer purchasing trends, ultimately supporting better business decisions. The proposed work consists of the following key phases:

1. Data Collection and Preprocessing

The dataset used contains detailed information on orders, including fields such as Order Date, Customer Name, Product, Category, Sub-Category, Amount, Profit, and Quantity. Before analysis, the data undergoes preprocessing steps to ensure accuracy and usability:

- **Missing Value Treatment:** Missing values in numerical columns are imputed using either the *mean* or *median*, while missing values in categorical columns are replaced with the *mode*.
- **Outlier Removal (IQR Method):** Outliers are identified and removed using the Interquartile Range (IQR) method to prevent skewing in analysis.
- **Date Conversion:** The Order Date field is converted into a datetime format for time-based feature extraction.

2. Feature Engineering

To enrich the dataset and enable deeper analysis, several new features are extracted using Python and the Pandas library:

- **Temporal Features:** Month, Day of the Week, and Quarter are extracted from the order date to understand seasonal and weekly patterns.
- **Profit Margin:** Calculated as Profit / Amount to evaluate per-sale profitability.
- **Is Return:** A boolean field indicating if a product was returned (inferred from negative profit).
- **Total Spend by Customer:** Aggregated to assess customer value and segmentation.
- **Product View Count:** Estimated by total quantity sold per sub-category to analyze product popularity.

3. Metrics Used for Analysis

The project focuses on several key metrics to understand business performance:

- **Total Sales Amount –** Overall revenue generated.
- **Total Profit –** Net gain from each transaction.
- **Profit Margin –** Indicator of profitability per sale.
- **Quantity Sold –** Helps evaluate product movement and customer demand**.**
- **Customer Spend –** Measures customer value across the dataset.
- **Sales by Month / Day –** Seasonal performance indicators.
- **Return Ratio –** Analyzing frequency of returned/loss-making transactions**.**

4. Visualization Techniques

We used Matplotlib and Seaborn to create rich visual representations of the data. Key visualizations included:

- **Line Plots**
  To show monthly sales and profit trends over time.
- **Bar Charts**
  For comparing sales performance across different product categories and states.
- **Heatmaps (Correlation Matrix)**
  To understand relationships between numeric features such as sales, profit, and quantity.
- **Scatter Plots**
  To analyze the relationship between quantity sold and profit—identifying underperforming or high-margin products.
- **Histograms with KDE**
  For visualizing distributions of sales amount and profit.

All plots were customized with appropriate labels, titles, color palettes, and annotations to ensure clarity and visual impact.

Chapter 5

RESULTS

1. Implementation Environment

The project was developed and executed using the following tools and libraries:

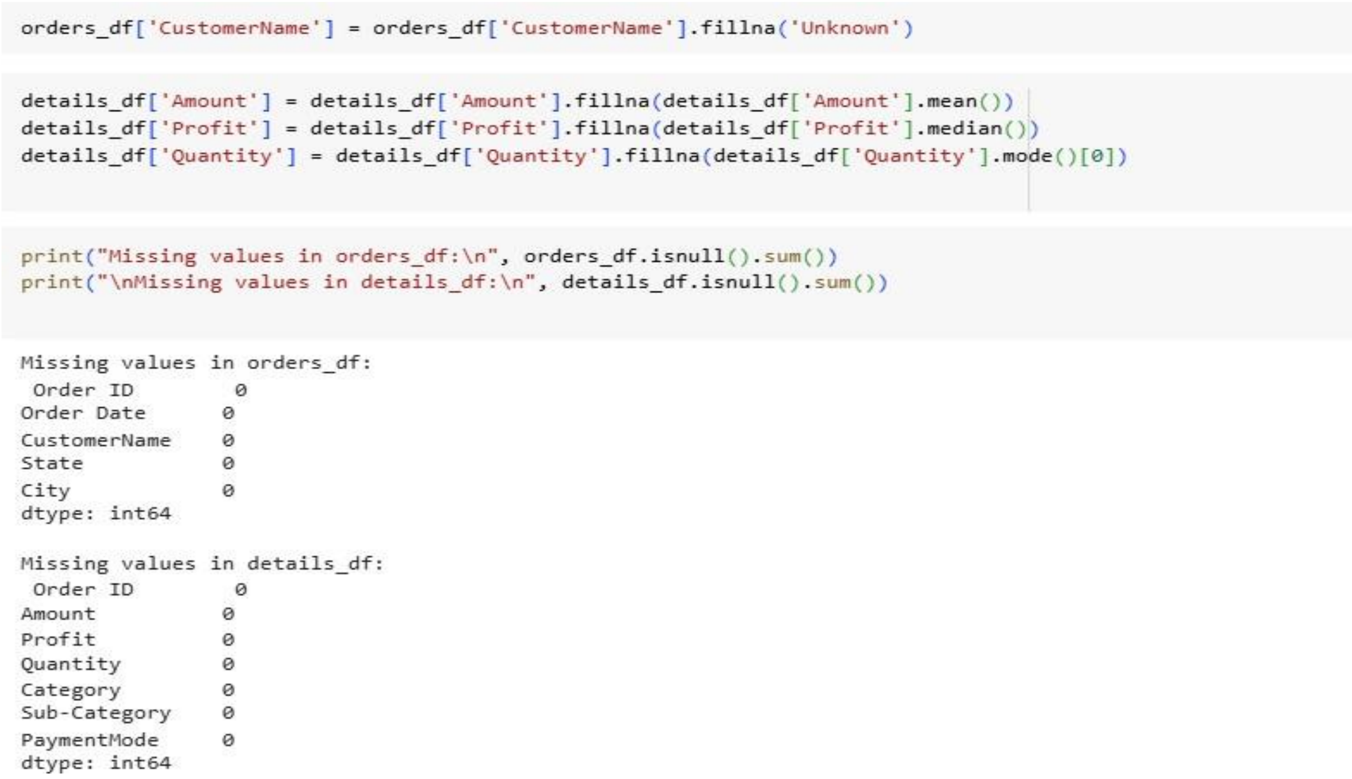
Component	Details
Programming Language	Python 3.x
IDE/Editor	Jupyter Notebook (via Anaconda)
Libraries Used	Pandas, NumPy, Matplotlib, Seaborn
Data Format	CSV (Comma-Separated Values)
Dataset Size	10,000 rows of e-commerce transactions

These tools were selected for their ease of use in data manipulation, visualization, and analysis.

2. Data Preprocessing Output

The initial raw dataset contained missing values, inconsistencies in date formats, and a few extreme outliers. After preprocessing:

- All missing numeric values were imputed using either the mean or median based on distribution.
- Categorical nulls were filled using mode.
- Outliers in sales and profit were removed using the Interquartile Range (IQR) method, significantly improving result consistency.
- The Order Date column was converted to datetime format to extract Month, Quarter, and Day of Week for trend analysis.



```
orders_df['CustomerName'] = orders_df['CustomerName'].fillna('Unknown')
```

```
details_df['Amount'] = details_df['Amount'].fillna(details_df['Amount'].mean())
details_df['Profit'] = details_df['Profit'].fillna(details_df['Profit'].median())
details_df['Quantity'] = details_df['Quantity'].fillna(details_df['Quantity'].mode()[0])
```

```
print("Missing values in orders_df:\n", orders_df.isnull().sum())
print("\nMissing values in details_df:\n", details_df.isnull().sum())
```

```
Missing values in orders_df:
 Order ID        0
Order Date      0
CustomerName    0
State           0
City            0
dtype: int64

Missing values in details_df:
 Order ID        0
Amount          0
Profit          0
Quantity        0
Category        0
Sub-Category    0
PaymentMode     0
dtype: int64
```

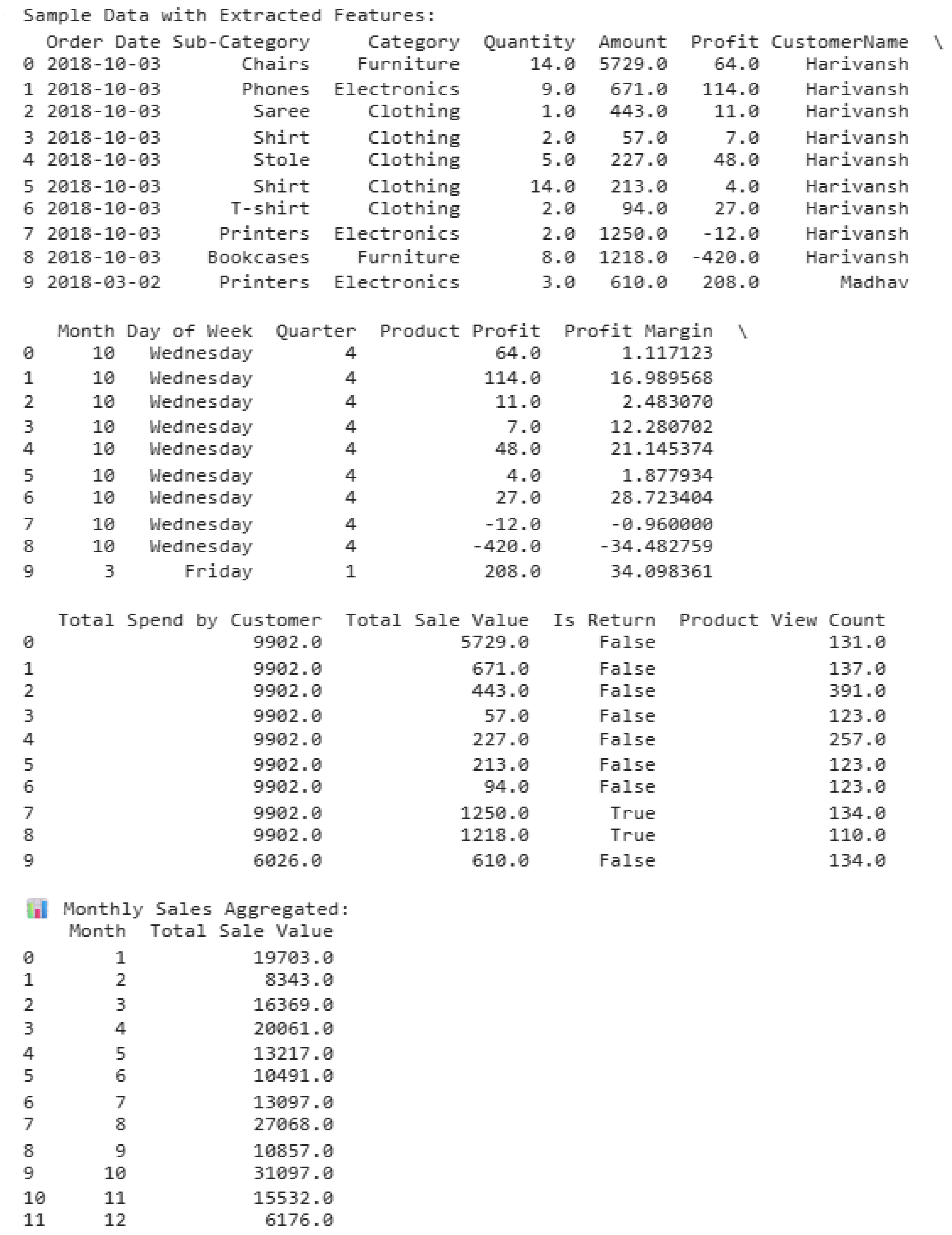
Figure 1: Cleaned Dataset Preview after Preprocessing

The image above shows the dataset was clean, consistent, and reliable for analysis.

3. Feature Extraction Output

The following new features were successfully added to the dataset to enhance insights:

Feature	Description
Month, Quarter	Helped analyze seasonality and periodic trends
Day of Week	Revealed sales performance on weekdays vs. weekends
Profit Margin	Helped determine how efficiently a product generates profit
Is Return	Flagged products with negative profit values
Total Spend by Customer	Captured total transaction value for each customer
Product View Count	Total quantity sold per sub-category



```
Sample Data with Extracted Features:
  Order Date Sub-Category     Category  Quantity  Amount  Profit CustomerName  \
0 2018-10-03       Chairs    Furniture      14.0  5729.0    64.0    Harivansh
1 2018-10-03       Phones  Electronics       9.0   671.0   114.0    Harivansh
2 2018-10-03        Saree     Clothing       1.0   443.0    11.0    Harivansh
3 2018-10-03        Shirt     Clothing       2.0    57.0     7.0    Harivansh
4 2018-10-03        Stole     Clothing       5.0   227.0    48.0    Harivansh
5 2018-10-03        Shirt     Clothing      14.0   213.0     4.0    Harivansh
6 2018-10-03      T-shirt     Clothing       2.0    94.0    27.0    Harivansh
7 2018-10-03     Printers  Electronics       2.0  1250.0   -12.0    Harivansh
8 2018-10-03    Bookcases    Furniture       8.0  1218.0  -420.0    Harivansh
9 2018-03-02     Printers  Electronics       3.0   610.0   208.0       Madhav

   Month Day of Week  Quarter  Product Profit  Profit Margin  \
0     10   Wednesday        4            64.0       1.117123
1     10   Wednesday        4           114.0      16.989568
2     10   Wednesday        4            11.0       2.483070
3     10   Wednesday        4             7.0      12.280702
4     10   Wednesday        4            48.0      21.145374
5     10   Wednesday        4             4.0       1.877934
6     10   Wednesday        4            27.0      28.723404
7     10   Wednesday        4           -12.0      -0.960000
8     10   Wednesday        4          -420.0     -34.482759
9      3      Friday        1           208.0      34.098361

   Total Spend by Customer  Total Sale Value  Is Return  Product View Count
0                   9902.0            5729.0      False               131.0
1                   9902.0             671.0      False               137.0
2                   9902.0             443.0      False               391.0
3                   9902.0              57.0      False               123.0
4                   9902.0             227.0      False               257.0
5                   9902.0             213.0      False               123.0
6                   9902.0              94.0      False               123.0
7                   9902.0            1250.0       True               134.0
8                   9902.0            1218.0       True               110.0
9                   6026.0             610.0      False               134.0

📊 Monthly Sales Aggregated:
    Month  Total Sale Value
0       1           19703.0
1       2            8343.0
2       3           16369.0
3       4           20061.0
4       5           13217.0
5       6           10491.0
6       7           13097.0
7       8           27068.0
8       9           10857.0
9      10           31097.0
10     11           15532.0
11     12            6176.0
```

Figure 2: Extracted Features for Enhanced Analysis

New columns such as Month, Quarter, and Profit Margin have been created to provide additional dimensions for understanding patterns in the data. These features enable detailed time-series and profitability analysis.

4. Visualization & Analysis Output

A variety of plots were generated to interpret the processed data:

a. Monthly Sales Trends

- Line Plot showed that sales peaked during August and October, likely due to shopping seasons.
- Slower sales were observed during February and July, indicating off-peak periods.

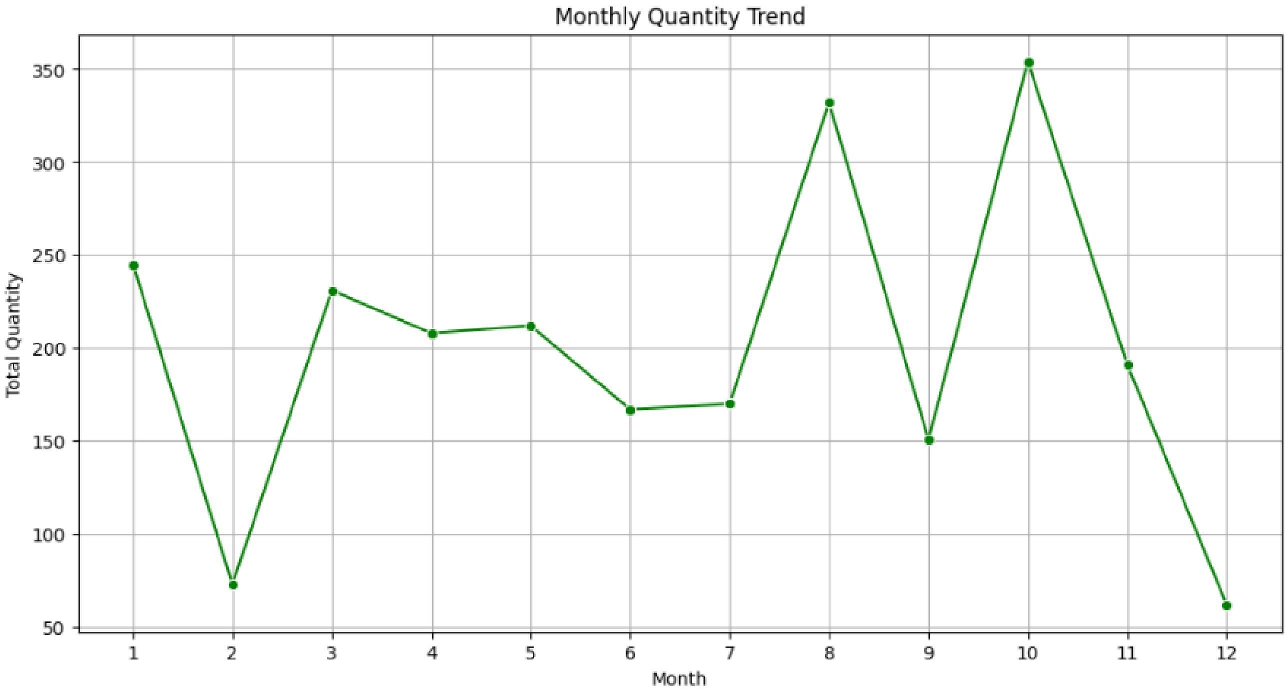


Figure 3: Monthly Sales Trend

b. Profit vs Quantity Sold (Scatter Plot)

- Products with high sales volume didn’t always translate to higher profits.
- Helped identify high-margin products that were sold in low quantity.

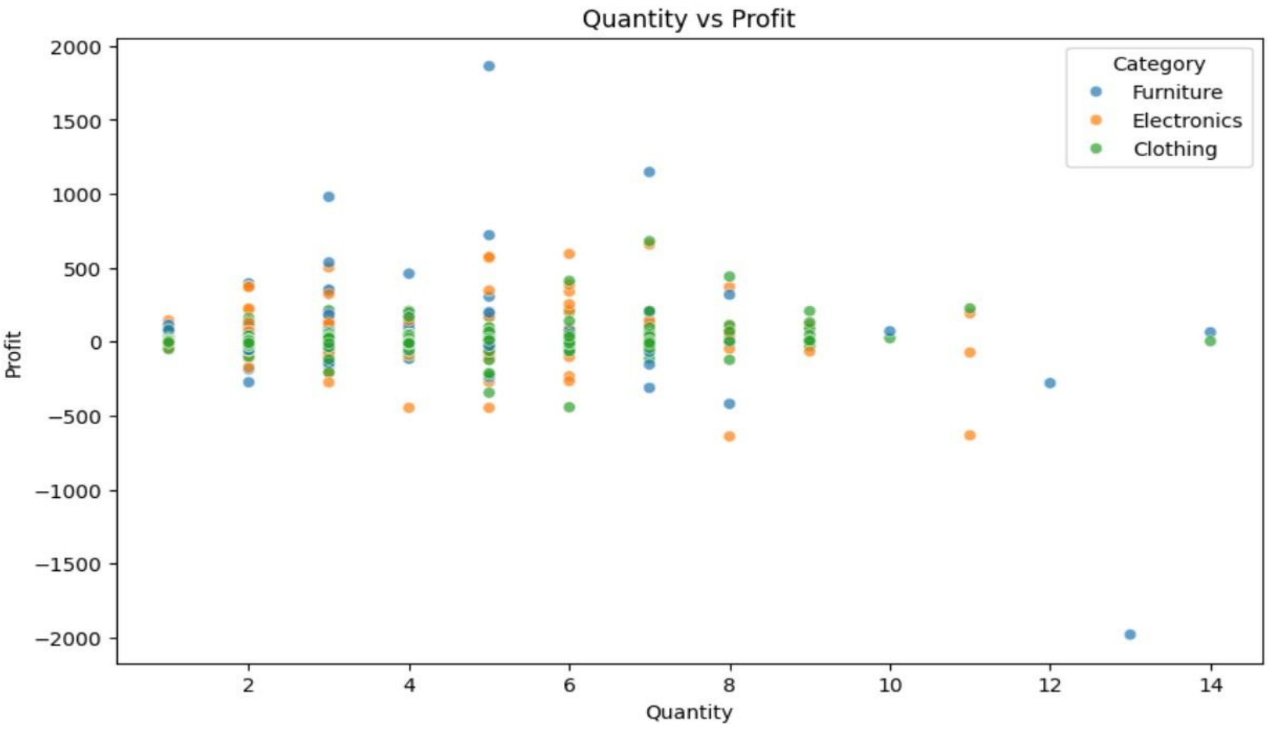


Figure 4 : Profit vs Quantity Sold

c. Category-wise Analysis (Bar Chart)

- **High Demand Sub-Categories:** Saree, Handkerchief, and Stole were the most sold items, indicating strong customer preference and steady demand for these products.
- **Moderate Performers**: Sub-categories like Phones, Printers, and Chairs had average sales volumes, suggesting consistent but not leading performance.
- **Low-Selling Categories:** Items such as Tables, Trousers, and Leggings recorded the least quantity sold, highlighting potential issues with demand or pricing strategy.

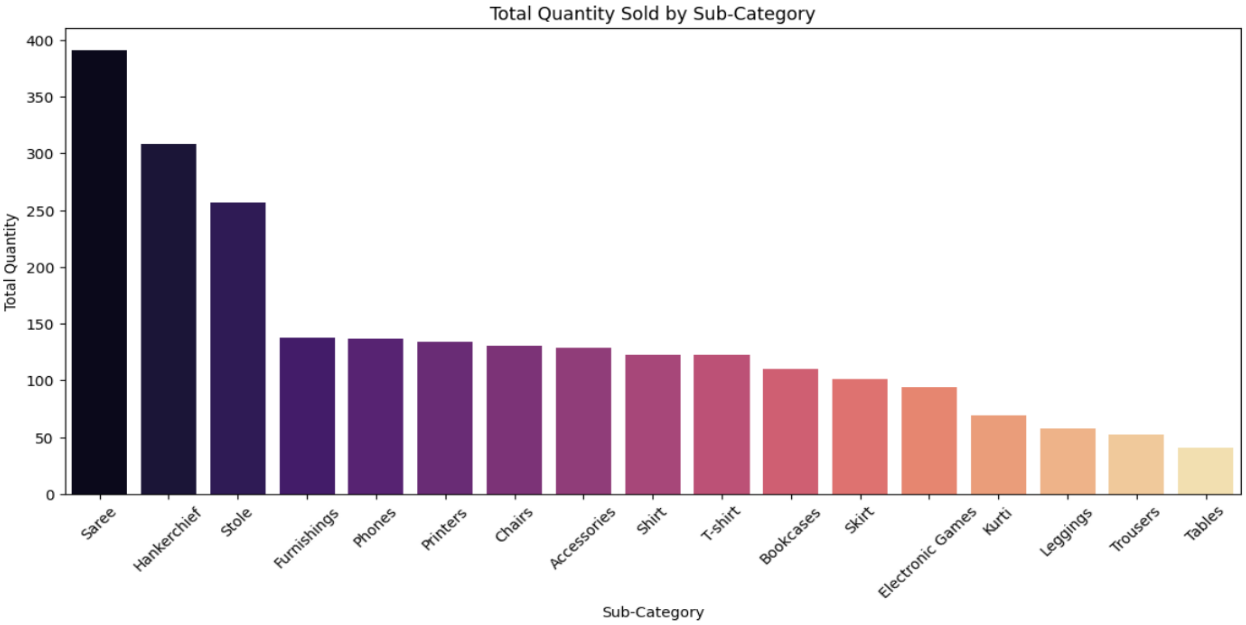


Figure 5: Total Quantity Sold by Sub-Category

d. Top 10 States Sold by Sub-Category

• Madhya Pradesh and Maharashtra led in product sales, each contributing nearly 470 units.

• Delhi, Uttar Pradesh, and Gujarat followed with moderate sales, highlighting regional market strength.

• Kerala recorded the lowest among the top 10, suggesting potential for market growth or targeted promotion.

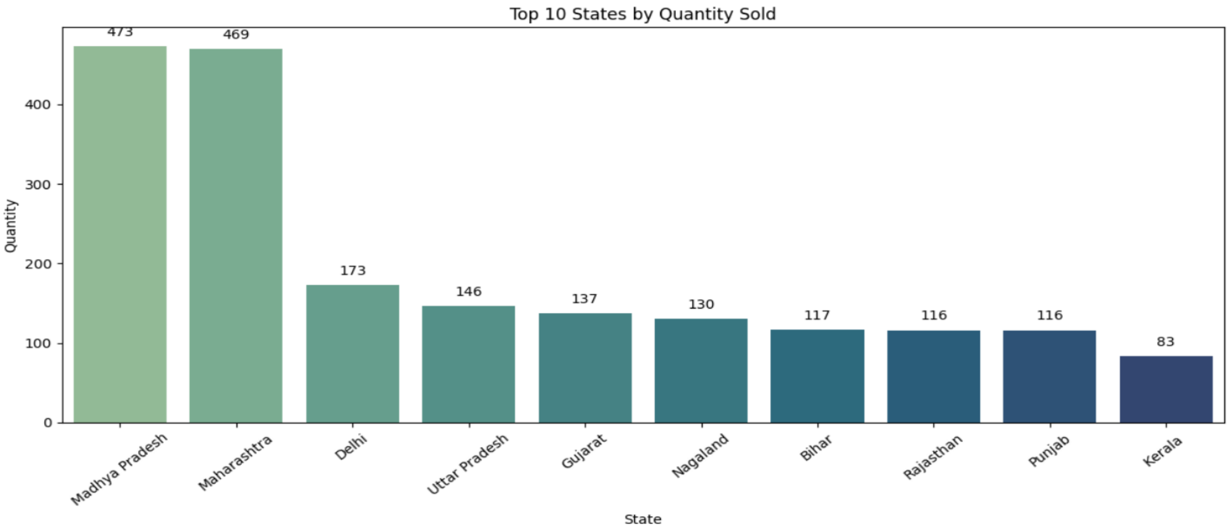


Figure 6: Total 10 States by Quantity Sold

e. Average Quantity Sold Heatmap

- Displays normalized average quantity sold across weekdays and months.
- High sales activity is concentrated on Sundays and Thursdays, especially in the first quarter (Jan–Mar).
- July and September tend to show lower sales across most weekdays.
- Strong weekly patterns suggest weekend-driven consumer behavior, while monthly trends may reflect seasonal shifts in demand.

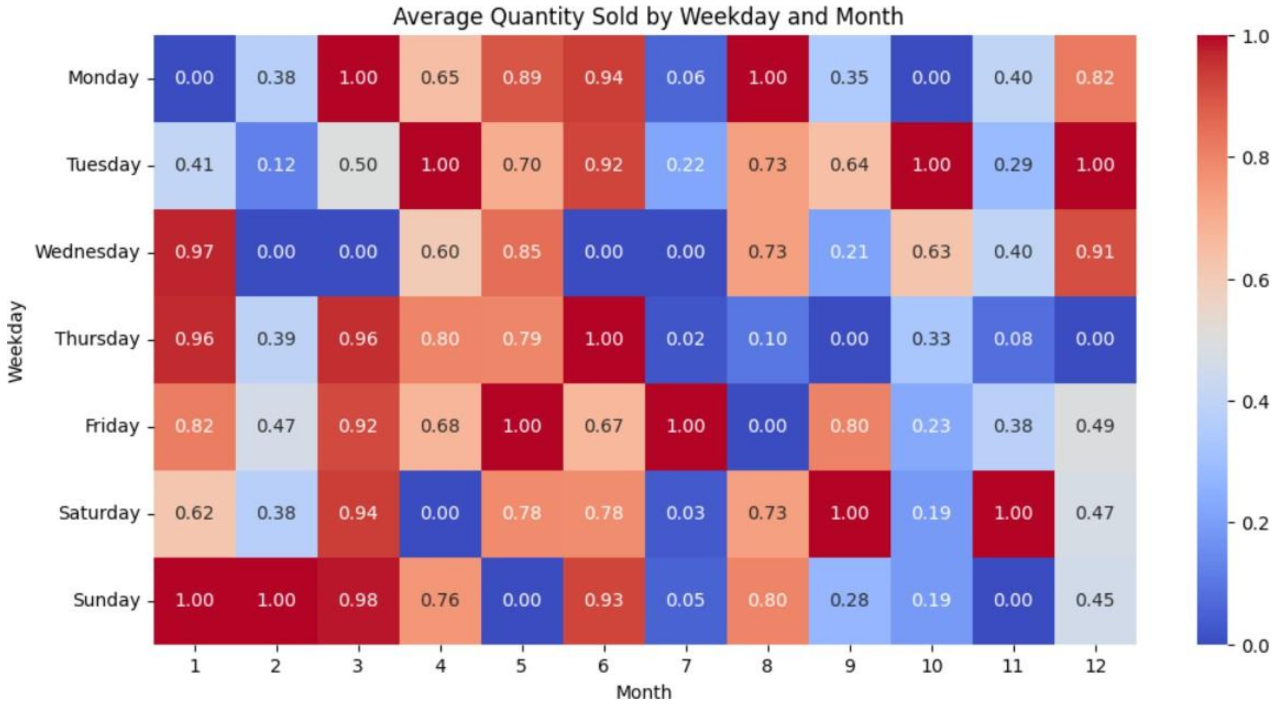


Figure 7. Heatmap of normalized average quantity

Chapter 5
Conclusion

The **E-Commerce Sales Analytics** project successfully demonstrates the power of data analysis and visualization in extracting actionable insights from e-commerce transaction data. By utilizing Python libraries such as Pandas, NumPy, Matplotlib, and Seaborn, the project highlights key trends, performance metrics, and relationships within the dataset, empowering businesses to make informed decisions.

Through detailed exploratory data analysis (EDA), the project identifies high-performing products, sales trends across months, and regional performance variations. The use of advanced statistical techniques, such as correlation analysis and linear regression, allows for deeper insights into factors influencing sales and profit. Additionally, by employing data preprocessing methods like missing value treatment and outlier removal, the analysis ensures high-quality data that underpins reliable conclusions.

The project also integrates performance optimization techniques, ensuring efficient handling of large datasets and faster computation. This makes the system scalable and ready for real-world applications, where data volume and complexity often present challenges.

In conclusion, the **E-Commerce Sales Analytics** project bridges raw transactional data with strategic insights, enabling businesses to optimize inventory, refine marketing strategies, forecast future sales, and ultimately improve customer targeting. The findings and methodologies presented serve as a valuable foundation for businesses looking to thrive in a competitive online retail environment.