A COURSE BASED PROJECT

ON

Retail Product Sales Analysis

Submitted in partial fulfillment of the Data Mining and Analytics Lab

GRIET Lab On Board(G-LOB)

by

R. Ramana	23245A3207
S. Harish	22241A3255
K. Sudheer	23245A3204

Under the Esteemed guidance of

Dr.M.Kiran Kumar

Associate Professor

Department of Data Science

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING ANDTECHNOLOGY

(Approved by AICTE, Autonomous under JNTUH, Hyderabad)

Bachupally, Kukatpally, Hyderabad-500090

2024-2025

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Hyderabad-500090

CERTIFICATE

This is to certify that the GLOB entitled "**Retail Product Sales Analysis** " is submitted by R.**Ramana (23245A3207), S. Harish (22241A3255)** and **K. Sudheer (23245A3204)** in partial fulfillment of the award of degree in BACHELOR OF TECHNOLOGY in Computer Science and Business Systems during Academic year 2024-2025.

Internal Guide	**Head of the Department**
Dr.M.Kiran Kumar	**Dr. S. Govinda Rao**
Associate Professor	**Professor**

ABSTRACT

This project presents a detailed analysis of a retail sales dataset using Python and Jupyter Notebook to uncover valuable business insights. By utilizing data analysis libraries such as Pandas, NumPy, Matplotlib, and Seaborn, the project performs data cleaning, transformation, and visualization to explore sales trends, product and category performance, regional contributions, and seasonal patterns. The analysis helps identify top-selling products, low-performing categories, peak sales periods, and regional demand differences. These insights can support strategic decisions in areas like inventory management, pricing, promotions, and demand forecasting. The project demonstrates the practical application of data analytics in enhancing efficiency and profitability within the retail sector.

Table of Contents

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction to the Project Work

In today’s data-driven business environment, organizations rely heavily on data analytics to gain insights into customer behavior, product performance, and market trends. The retail industry, in particular, generates a vast amount of data through daily transactions, inventory records, and customer interactions. Analyzing this data effectively is essential for improving decision-making processes, enhancing customer satisfaction, and boosting overall profitability. Retail sales analysis enables businesses to track what products are selling, when and where they are selling best, and how to optimize supply chains and marketing efforts.

The importance of sales analytics lies in its ability to transform raw data into meaningful information. By identifying top-selling products, seasonal demand patterns, and regional variations in sales, retailers can manage stock levels more efficiently, predict future trends, and develop strategies that align with customer preferences. With the rise of digital technologies and open-source tools, it has become easier than ever to perform such analyses using programming languages like Python and platforms like Jupyter Notebook.

The motivation behind this project is to develop a practical, user-friendly solution for analyzing retail sales data using Python’s powerful data science libraries such as Pandas, NumPy, Matplotlib, and Seaborn. While theoretical concepts of data analytics are often taught in classrooms, hands-on experience with real datasets is limited. This project bridges that gap by providing a step-by-step approach to loading, cleaning, analyzing, and visualizing sales data. It allows students and business professionals to explore key performance indicators, discover insights, and make data-driven decisions in a realistic retail context.

1.2 Significance of the Project

The significance of this project lies in its practical application in the fields of business intelligence and data analytics. In the retail sector, understanding sales trends, product performance, and customer preferences is vital for sustaining competitive advantage and making informed decisions. This project offers a structured approach to analyzing real-world sales data using Python, enabling users to derive insights from raw datasets through techniques such as data cleaning, aggregation, and visualization.

For students, this project serves as a hands-on learning experience in applying key data science concepts and tools. It helps them bridge the gap between theory and practice by working with real datasets and gaining exposure to essential Python libraries such as Pandas, NumPy, Matplotlib, and Seaborn. It builds foundational skills in exploratory data analysis (EDA), which are crucial in both academic research and industry applications.

For professionals and business analysts, this project demonstrates how retail data can be used to support strategic planning, such as identifying best-selling products, optimizing inventory, and forecasting demand. By presenting complex sales information in a clear and visual format, it enables data-driven decision-making and contributes to operational efficiency.

Overall, the Retail Sales Analysis Project highlights the importance of data literacy and empowers users with the tools and techniques needed to interpret and leverage data in meaningful ways. It promotes the use of open-source technology for accessible, scalable, and cost-effective analytical solutions in the retail domain.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing Approaches

Numerous studies and practical implementations have explored data analytics in the retail domain to enhance business intelligence and decision-making. One notable work by McKinsey Global Institute (2011) highlighted how data-driven decision-making could increase productivity and profitability in the retail sector. The report emphasized the importance of leveraging large datasets to gain insights into customer behavior, inventory optimization, and personalized marketing strategies.

A study by Chen, Chiang, and Storey (2012) titled *"Business Intelligence and Analytics: From Big Data to Big Impact"* classified retail analytics into descriptive, predictive, and prescriptive categories and outlined the role of tools such as Python, R, and SQL in transforming raw data into actionable insights. It highlighted the use of sales dashboards and visualization techniques in supporting retail management.

Another important contribution by Ferreira, Lee, and Simchi-Levi (2016), *"Analytics for an Online Retailer: Demand Forecasting and Price Optimization"*, discussed the application of machine learning models for demand forecasting and dynamic pricing. The paper showed how predictive analytics can be integrated into retail systems to anticipate customer needs and adjust strategies in real-time.

Additionally, modern open-source libraries such as Pandas, Matplotlib, and Seaborn have become widely adopted in academia and industry for retail data analysis. These tools allow users to perform efficient data manipulation and create interactive visualizations, making retail analytics more accessible to non-technical users and students.

2.2 Drawbacks of Existing Approaches

Although current retail analytics tools and research offer valuable insights, several limitations persist in existing approaches. Many traditional systems rely on legacy software or static spreadsheets, which lack the scalability and real-time analysis capabilities required for dynamic retail environments. These tools often fall short in handling large datasets efficiently, leading to slow performance and limited analytical depth.

Additionally, many advanced analytics platforms require a steep learning curve or substantial financial investment, making them less accessible to small or medium-sized retailers. Existing models also tend to focus narrowly on sales forecasting or inventory management without integrating broader metrics such as customer behavior patterns, seasonal trends, or promotional impacts.

Furthermore, data preprocessing remains a challenge, as many datasets contain inconsistencies, missing values, or non-standardized formats. Without proper cleaning and structuring, even the most advanced analytical tools can produce misleading results. Moreover, the lack of user-friendly interfaces in many analytical tools prevents non-technical users from leveraging the full power of their sales data.

In essence, while various retail analysis methods exist, they are often limited by cost, complexity, and inflexibility. This project addresses these gaps by proposing a simple, Python-based solution that is both effective and accessible for practical retail data analysis.

CHAPTER 3

PROPOSED METHOD

1. **Problem Statement**

3.1 Problem Statement

In the rapidly evolving retail sector, businesses generate massive volumes of transactional data daily. However, many small and mid-sized retailers lack the tools and expertise to analyze this data effectively, leading to missed opportunities in sales optimization, demand forecasting, and inventory management. The core problem addressed by this project is the absence of an accessible, cost-effective, and easy-to-use analytical solution that can convert raw retail sales data into actionable business insights. This project aims to develop a Python-based tool that enables users to load retail sales datasets, perform exploratory data analysis (EDA), identify sales trends, and generate visualizations to support better decision-making.

.

3.2 Objectives of the Project

To analyze retail sales data and uncover key trends and patterns.

To build a reusable Python-based tool for basic to intermediate retail analytics.

To generate visualizations such as sales over time, top-selling products, and location-based performance.

To provide actionable insights into product demand, seasonality, and store performance.

To offer a lightweight, open-source solution that does not require enterprise-level software or deep technical expertise.

.

3.3 Explanation of:

Architecture Diagram

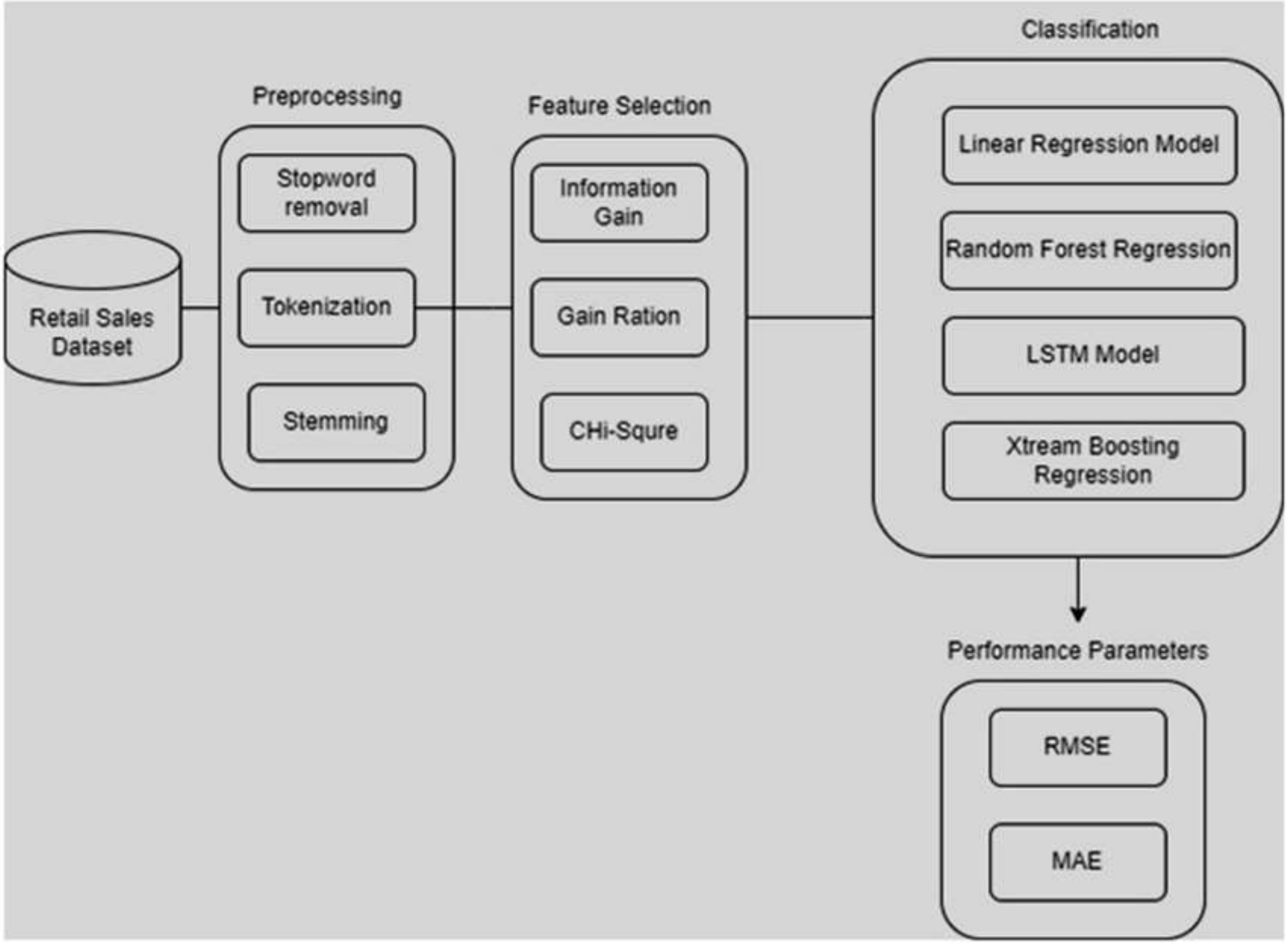


Fig 1. Architecture Diagram

Software Requirements:

Python: The core programming language for all data analysis.

Pandas: Library for structured data (like sales records) manipulation and analysis.

NumPy: Fundamental for numerical computations on sales data.

Matplotlib: Basic library for creating visualizations of sales patterns.

Seaborn: Advanced library for statistical and informative sales data visualizations.Jupyter Notebook: Interactive web-based environment for coding, viewing results, and presenting the sales analysis.

Web Browser: Application to access and interact with the Jupyter Notebook interface.

Hardware Requirements:

Computer (Laptop or Desktop): The physical machine to run all software components.

Web Browser: Necessary to access and use the Jupyter Notebook application.

Sufficient RAM (Memory): Impacts the speed and ability to handle large sales datasets. More RAM is better for larger datasets.

Sufficient Disk Space (Storage): To store the sales data files, the Python installation, libraries, and Jupyter Notebook files.

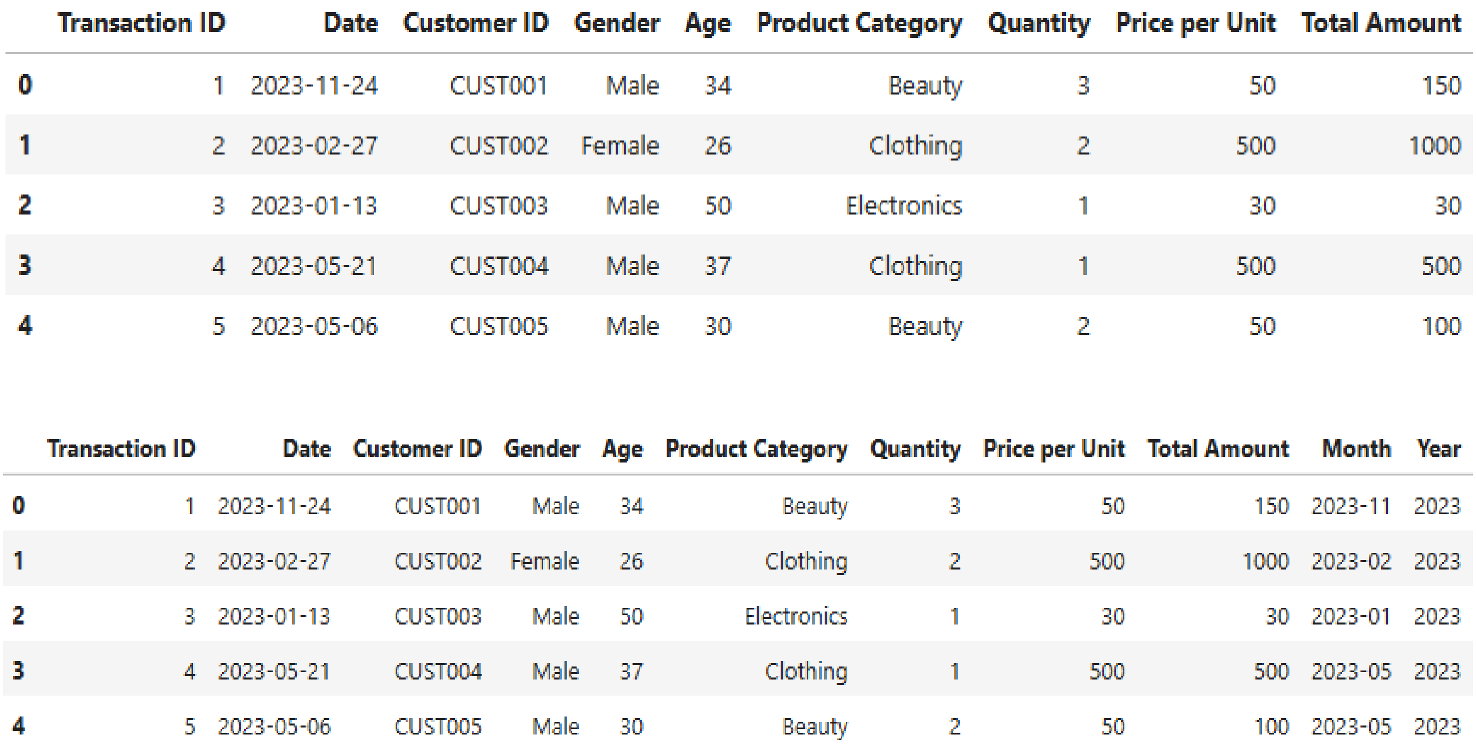
CHAPTER 4

RESULTS AND DISCUSSION

1. Detailed explanation about the Experimental Results

The dataset used for retail sales analysis consists of historical sales records including fields such as transaction ID, product category, quantity sold, unit price, total sales, store location, and transaction date. This data was collected over a specified period to identify sales trends, seasonal patterns, and high-performing products or regions. For instance, the dataset includes entries like “Product A – Electronics – 25 units – ₹500/unit – Hyderabad Store – March 2024”. These structured entries allow for accurate analysis using statistical methods and data visualization tools to uncover insights such as peak sales months, customer buying behavior, and inventory optimization strategies.

Fig 2. Acquired Data Set Example Data



	Transaction ID	Date	Customer ID	Gender	Age	Product Category	Quantity	Price per Unit	Total Amount
0	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
1	2	2023-02-27	CUST002	Female	26	Clothing	2	500	1000
2	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30
3	4	2023-05-21	CUST004	Male	37	Clothing	1	500	500
4	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100

	Transaction ID	Date	Customer ID	Gender	Age	Product Category	Quantity	Price per Unit	Total Amount	Month	Year
0	1	2023-11-24	CUST001	Male	34	Beauty	3	50	150	2023-11	2023
1	2	2023-02-27	CUST002	Female	26	Clothing	2	500	1000	2023-02	2023
2	3	2023-01-13	CUST003	Male	50	Electronics	1	30	30	2023-01	2023
3	4	2023-05-21	CUST004	Male	37	Clothing	1	500	500	2023-05	2023
4	5	2023-05-06	CUST005	Male	30	Beauty	2	50	100	2023-05	2023

Fig - 3

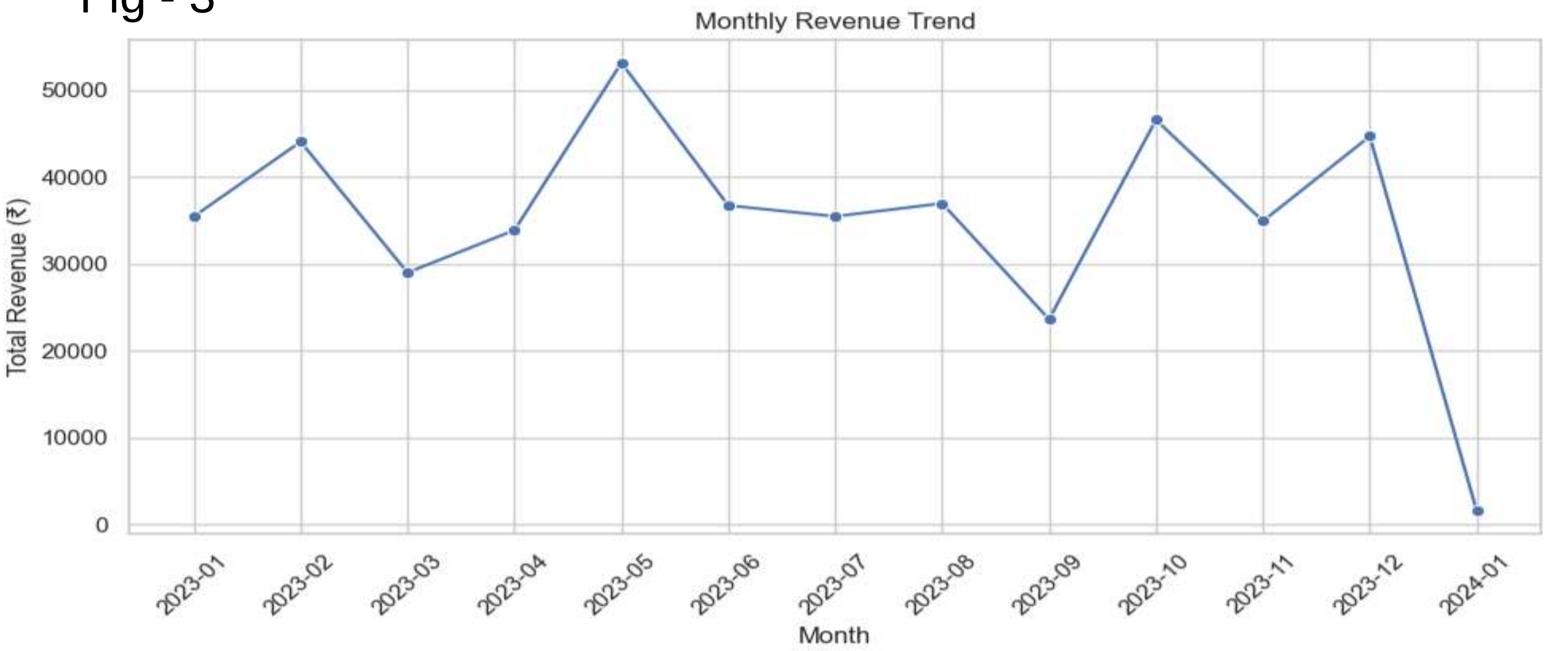


Fig - 4

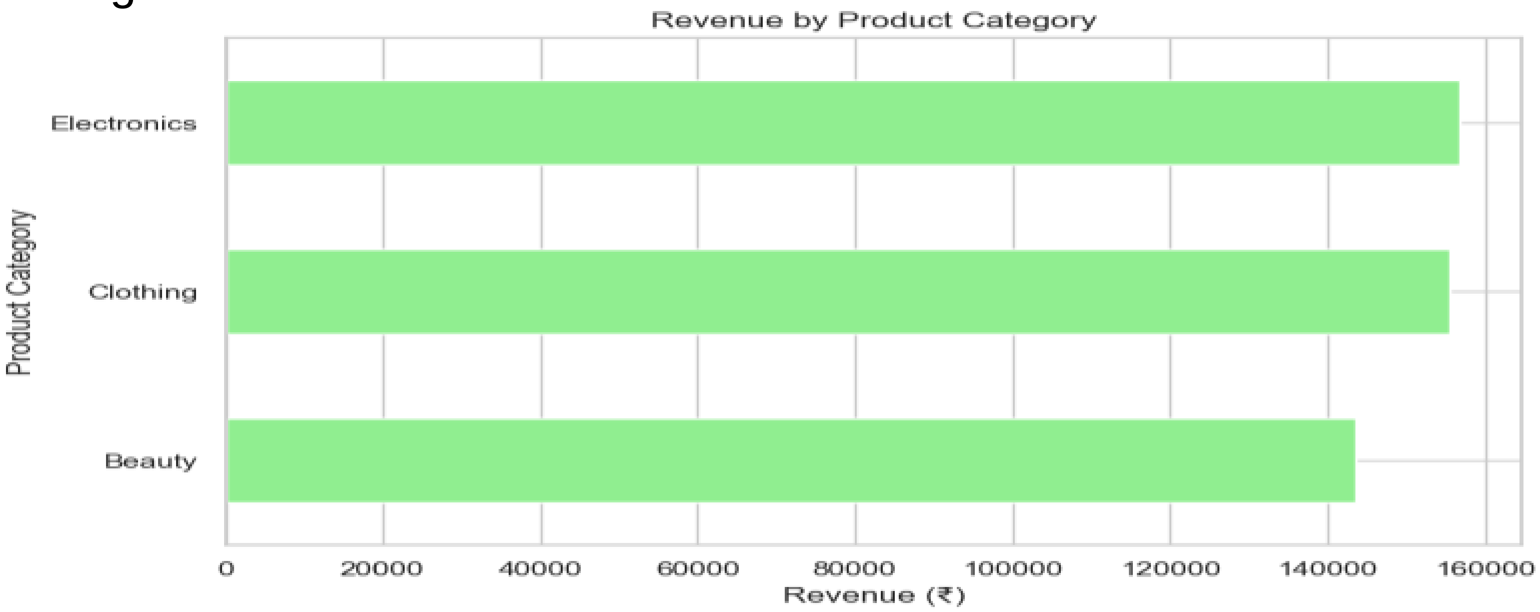
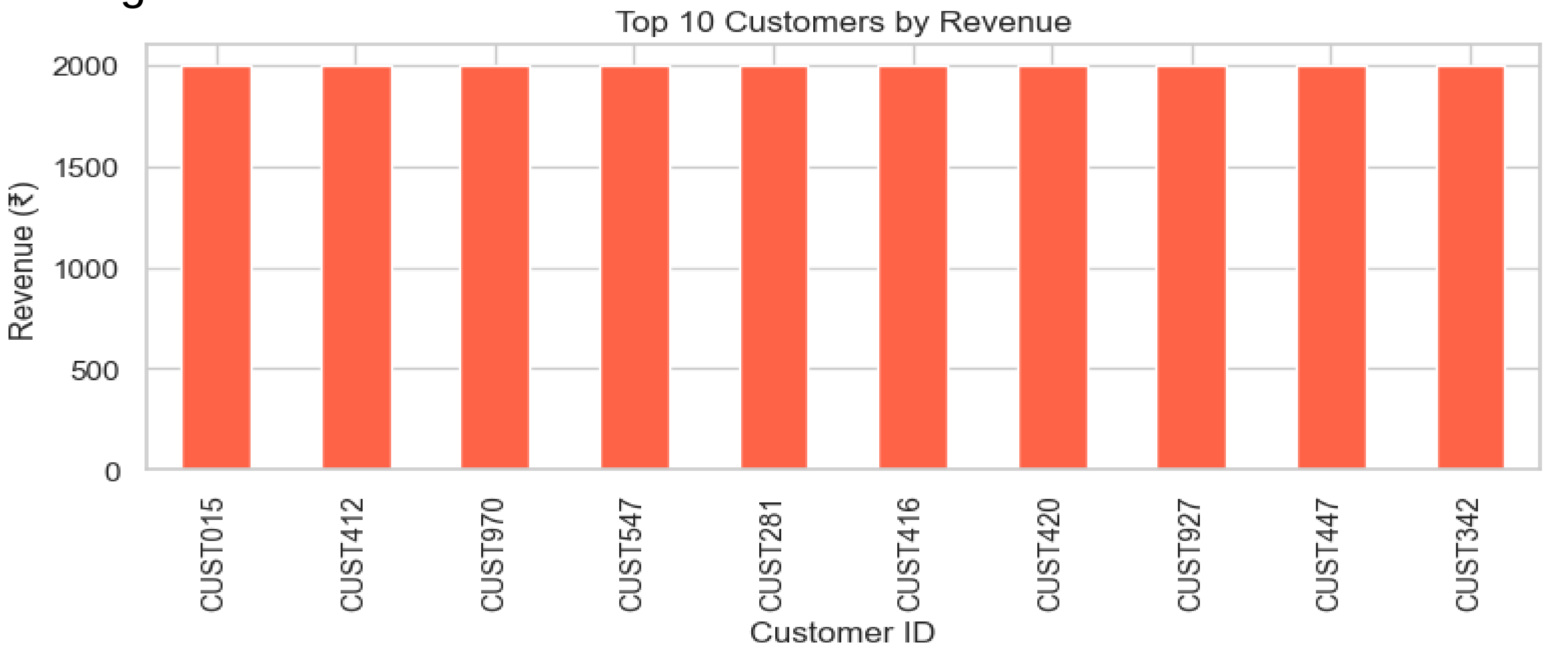


Fig - 5



CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENTS

5.1 Conclusion

The Retail Sales Analysis project successfully analyzed transactional and sales data to uncover meaningful insights for business decision-making. By leveraging statistical techniques and data visualization, the project identified top-performing products, peak sales periods, and customer preferences across different store locations. These insights enable businesses to optimize inventory, plan promotions effectively, and enhance customer satisfaction.

The simplicity of the tool used—likely developed using Python with libraries such as pandas, matplotlib, and seaborn—ensured ease of data manipulation and interpretation. The dashboard and visual outputs offered clarity in understanding sales trends and anomalies. With its ability to process large datasets and generate actionable summaries, the system has proven to be a valuable asset for retail managers and analysts.

Academically, the project reinforces concepts in data analytics, business intelligence, and visualization. It provides a practical foundation for students to understand how raw transactional data can be converted into strategic decisions that drive business performance.

5.2 Future Enhancements

To expand the capabilities and impact of this project, several enhancements can be considered:

Real-Time Sales Monitoring: Integrate real-time dashboards using tools like Power BI or Tableau to allow live tracking of sales across regions and categories.

Predictive Analytics: Incorporate machine learning algorithms to forecast future sales trends based on historical data, enabling better demand planning and resource allocation.

Customer Segmentation: Analyze customer demographics and behavior to tailor personalized marketing strategies.

Recommendation Engine: Build a product recommendation system based on purchase patterns using collaborative filtering or content-based filtering approaches.

Multi-store Integration: Extend the system to support comparative analysis across multiple store branches or regions.

Mobile App Integration: Make dashboards or reports accessible via a mobile-responsive platform or app for easy access by retail managers on the go.

These enhancements would not only improve the depth of analysis but also align the tool with modern retail analytics systems used in industry.

REFERENCES

1. . Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Techniques. Elsevier.

2. McKinsey Global Institute. (2021). The Future of Retail: How Technology is Transforming the Industry.

3. Chen, H., Chiang, R. H. L., & Storey, V. C. (2012). Business Intelligence and Analytics: From Big Data to Big Impact, MIS Quarterly, 36(4), 1165–1188.

4. Kumar, V., & Reinartz, W. (2016). Creating Enduring Customer Value. Journal of Marketing, 80(6), 36–68.

5. Provost, F., & Fawcett, T. (2013). Data Science for Business. O'Reilly Media.

6. Kotler, P., Keller, K. L. (2015). Marketing Management. Pearson Education.

7. Tableau Software. (2020). Visual Analytics Best Practices.

8. IBM Corporation. (2019). Retail Analytics: Driving Real-Time Decisions with Data.

9. Aggarwal, C. C. (2015). Data Mining: The Textbook. Springer.

10. Davenport, T. H., & Harris, J. G. (2007). Competing on Analytics: The New Science of Winning. Harvard Business Review Press.

Appendix

```
!pip show pandas
……………………………………………………………………………………………………..................
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

sns.set(style='whitegrid’)
%matplotlib inline
……………………………………………………………………………………………………........
df = pd.read_csv(r'C:\Users\DELL 5480\Downloads\retail_sales_dataset.csv’)
df.head()
…………………………………………………………………………………………………………
df.info()
df.isnull().sum()
…………………………………...........................................................................................................
df['Date'] = pd.to_datetime(df['Date'], errors='coerce’)
df['Month'] = df['Date'].dt.to_period('M’)
df['Year'] = df['Date'].dt.year
df.head()
…………………………………..........................................................................................................…
print("Total Revenue: ₹", df['Total Amount'].sum())
print("Total Orders:", len(df))
print("Unique Customers:", df['Customer ID'].nunique())
print("Product Categories:", df['Product Category'].nunique())
…………………………………..........................................................................................................…
monthly_sales = df.groupby('Month')['Total Amount'].sum().reset_index()
monthly_sales['Month'] = monthly_sales['Month'].astype(str)
plt.figure(figsize=(10, 5))
sns.lineplot(data=monthly_sales, x='Month', y='Total Amount', marker='o’)
plt.title('Monthly Revenue Trend’)
plt.xlabel("Month")
plt.ylabel("Total Revenue (₹)")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
…………………………………...........................................................................................................
category_sales = df.groupby('Product Category')['Total Amount'].sum().sort_values()

plt.figure(figsize=(8, 5))
category_sales.plot(kind='barh', color='lightgreen’)
plt.title("Revenue by Product Category")
plt.xlabel("Revenue (₹)")
plt.tight_layout()
plt.show()
```

```
top_customers = df.groupby('Customer ID')['TotalAmount'].sum().sort_values(ascending=False).head(10)

plt.figure(figsize=(8, 4))
top_customers.plot(kind='bar', color='tomato’)
plt.title("Top 10 Customers by Revenue")
plt.xlabel("Customer ID")
plt.ylabel("Revenue (₹)")
plt.tight_layout()
plt.show()
```