Retail Analytics & AI-Powered Sales Forecasting System

1. Introduction
This project focuses on building an end-to-end Retail Analytics and AI-Powered Sales Forecasting System to support data-driven decision-making in retail organizations. The system analyzes historical sales data, identifies performance patterns, segments stores using machine learning, and forecasts future revenue trends.

2. Project Objectives
• Analyze sales performance across stores and product categories
• Identify trends and seasonal patterns
• Segment stores based on performance using clustering
• Forecast future revenue using time-series modeling
• Present insights through interactive dashboards

3. Dataset Description
The dataset contains transactional retail sales data collected between January 2023 and December 2024. It includes attributes such as Date, Store ID, Product Category, Quantity Sold, Unit Price, and Revenue.

4. Tools and Technologies
Microsoft Excel was used for data cleaning, exploratory data analysis, and dashboard creation. Python was used for advanced analytics including K-Means clustering and SARIMAX-based forecasting.

5. Methodology
The project follows a structured analytical workflow as given below:

1. **Data Collection**
   Retail sales data covering multiple stores and product categories was collected for the period January 2023 to December 2024.
2. **Data Cleaning and Preparation (Excel)**
   The raw data was cleaned using Microsoft Excel by removing duplicate records, handling missing values, correcting date formats, and creating derived fields such as Year, Month, Quarter, and Revenue.

3. **Exploratory Data Analysis (Excel)**
   Exploratory analysis was performed using pivot tables and charts to analyze store-wise sales, product category performance, and monthly sales trends. Key patterns and seasonal behaviors were identified at this stage.

4. **Advanced Analytics and Machine Learning (Python)**
   Python was used to perform advanced analytics. Store segmentation was carried out using the K-Means clustering algorithm based on revenue and sales-related features. Sales forecasting was performed using the SARIMAX time-series model to predict future revenue.

5. **Export of Python Results**
   The clustering results and forecasted sales values generated in Python were exported to Excel for further use.

6. **Integration into Excel Dashboard**
   The Python-generated outputs were integrated into the Excel dashboard and visualized alongside historical sales data using charts, KPIs, and slicers.

7. **Dashboard Creation and Insights**
   An interactive dashboard was developed in Excel to present key performance indicators, sales trends, store performance segments, and future sales forecasts, enabling data-driven decision-making.

6. Dashboard and Visualization

An interactive dashboard was developed in Excel to present KPIs, sales trends, category performance, store clusters, and forecasted revenue.

7. Key Insights

• A small number of stores contribute a major share of revenue
• Certain product categories consistently outperform others
• Clear seasonal sales patterns are observed
• Forecast results indicate stable short-term growth

8. Conclusion

The project successfully demonstrates how retail sales data can be transformed into actionable insights using analytics and AI techniques. The combined use of Excel and Python ensures both usability and analytical depth.

9. Future Enhancements

Future improvements may include real-time data integration, store-level forecasting, and deployment using web-based BI platforms.

References

1. Microsoft Corporation. *Microsoft Excel Documentation*.
   Available at: https://support.microsoft.com/excel
   Accessed: 2025.

2. pandas Development Team. *pandas Documentation*.
   Available at: https://pandas.pydata.org/docs/
   Accessed: 2025.

3. scikit-learn Developers. *scikit-learn: Machine Learning in Python*.
   Available at: https://scikit-learn.org/stable/documentation.html
   Accessed: 2025.

4. Statsmodels Development Team. *Statsmodels: Time Series Analysis and SARIMAX Documentation*.
   Available at: https://www.statsmodels.org/stable/index.html
   Accessed: 2025.

5. Hunter, J. D. (2007). *Matplotlib: A 2D Graphics Environment*.
   Computing in Science & Engineering, 9(3), 90–95.
   Available at: https://matplotlib.org/stable/index.html

6. Python Software Foundation. *Python Documentation*.
   Available at: https://docs.python.org/3/
   Accessed: 2025.

Submission Details

Submitted by:
Shashank R Desai
shashankrdesai8803@gmail.com

College / University:
M S Ramaiah University of Applied Sciences

Department:
Data Science & Analytics