Superstore Order Return Analysis & Dashboard Report

1. Abstract

This report details a project focused on analysing and predicting order returns using the sample Superstore dataset. The primary goals were to identify characteristics associated with returned orders and to develop tools for monitoring and potentially mitigating returns. Using Python within a Jupyter Notebook environment, order and return data were loaded, integrated, and cleaned. Exploratory Data Analysis revealed variations in historical return rates across product categories and regions. A Logistic Regression model was trained to predict the probability of an order being returned. Key deliverables include the Python codebase, a CSV file listing orders identified as high-risk based on predicted probability, and guidance for creating an interactive Power BI dashboard for visualizing return trends and patterns. This project provides insights into order returns within the dataset and actionable tools for further investigation and monitoring.

2. Introduction

Product returns significantly impact business profitability and operational efficiency. Understanding the factors driving returns is essential for implementing effective reduction strategies. This project undertook an analysis of the sample Superstore dataset to uncover patterns related to returned orders. The objectives included identifying which segments or product categories exhibit higher return rates, building a predictive model to flag orders likely to be returned, and designing an interactive dashboard for ongoing monitoring. By analysing historical data and predicting future risks, the project aims to provide actionable insights and tools to help manage and reduce order returns in the context of the Superstore dataset.

3. Tools Used

The following tools and technologies were employed throughout this project:

- **Python:** Core language for data processing, analysis, and modelling.
  - *Libraries:* Pandas (data manipulation), NumPy (numerical operations), Scikit-learn (modelling, preprocessing, evaluation), Matplotlib/Seaborn (visualization during EDA).
- **Jupyter Notebook:** Interactive environment for developing and executing the Python code.
- **CSV:** File format for input data (Orders, Returns).
- **Power BI:** Business intelligence tool used for designing the interactive data visualization dashboard (based on project analysis results).

4. Steps Involved in Building the Project

The project execution followed these key phases:

1. **Data Acquisition & Integration (Python):**
   - Loaded the Superstore 'Orders' and 'Returns' data from the provided CSV files.
   - Merged the datasets based on 'Order ID' to create a binary IsReturned flag (1 for returned, 0 otherwise) for each order line. This was essential for identifying the target variable.
2. **Data Cleaning & Feature Engineering (Python):**
   - Converted date columns ('Order Date', 'Ship Date') to the appropriate datetime format.
   - Calculated the DaysToShip feature.
   - Performed basic data cleaning, such as handling missing date values and ensuring data consistency.
3. **Exploratory Data Analysis (EDA) (Python):**
   - Calculated the overall order return rate.
   - Analysed and visualized return rates across different dimensions like 'Category', 'Sub-Category', 'Region', and 'Segment' using bar charts to identify potential patterns or high-return groups.

4. **Predictive Modelling (Python):**
   - Selected relevant features for predicting order returns (excluding 'Manufacturer' due to absence in the specific data file).
   - Pre-processed data: Applied StandardScaler to numerical features and OneHotEncoder to categorical features.
   - Split data into training and testing sets.
   - Trained a Logistic Regression model, incorporating class_weight='balanced' to handle potential data imbalance.
   - Evaluated the model's performance using metrics such as a classification report, confusion matrix, and ROC AUC score.
5. **High-Risk Order Identification (Python):**
   - Used the trained model to predict the return probability for all orders.
   - Generated a CSV file (high_risk_orders.csv) listing orders exceeding a defined probability threshold, flagged as high-risk.
6. **Dashboard Design (Power BI):**
   - Prepared the analysed data (including IsReturned and PredictedReturnProbability flags) for Power BI by exporting it to a CSV file from Python OR provided instructions to load raw data and perform merges/calculations directly in Power BI.
   - Defined necessary DAX measures (e.g., Total Orders, Total Returns, Overall Return Rate).
   - Designed an interactive dashboard layout including KPI cards, charts showing return rates by various dimensions, and slicers for filtering.
   - Incorporated drill-through functionality to allow users to navigate from summary views to more detailed information.

5. Conclusion

This project successfully analysed the Superstore dataset to identify characteristics associated with returned orders and developed a predictive model for return risk. The analysis highlighted variations in return rates across different product categories and regions within the dataset. The Logistic Regression model provides a mechanism to flag potentially high-risk orders, and the generated high_risk_orders.csv offers a direct list for potential follow-up actions.

Furthermore, the outlined Power BI dashboard provides a valuable tool for stakeholders to visually explore return trends, filter data interactively, and drill down into specific areas of interest. While the analysis was limited by the available data (order-level returns, no specific return reasons), the project deliverables offer a solid foundation for data-driven monitoring and potential intervention strategies aimed at understanding and managing returns within the Superstore context.