E■commerce Return Rate Reduction Analysis

Project Report (Concise) — October 27, 2025

Abstract
This project identifies drivers of product returns across categories, geographies and marketing channels. Using data cleaning, exploratory analysis, and a logistic regression model, we quantify return risk and produce a prioritized list of high■risk products. A Power BI dashboard surfaces return risk with drill■through filters for stakeholders to take targeted actions to reduce return rates.

Introduction
Returns reduce profitability and increase operational costs. This analysis aims to (1) measure return rates by category, supplier and region, (2) identify patterns and root causes, and (3) predict the likelihood a given order will be returned so the business can intervene pre■ or post■sale.

Tools Used
• Python (pandas, scikit■learn, matplotlib) — data cleaning, feature engineering, modelling.
• SQL (MySQL / PostgreSQL) — data extraction and aggregation from transactional tables.
• Power BI — interactive dashboarding with drill■through filters and return risk score visualizations.
• CSV — export of high■risk product list for operational use.

Steps Involved in Building the Project
1. Data ingestion: import order, item, product, and return tables; join on order_id / product_id.
2. Data cleaning: handle missing values, normalize category labels, parse dates, and remove duplicates.
3. Exploratory analysis: compute return % by category, supplier, region and marketing channel; visualize trends and seasonality.
4. Feature engineering: create features such as product age, price band, discount rate, days to deliver, customer tenure, and channel flags.
5. Modelling: train a logistic regression to predict probability of return; evaluate with ROC■AUC, precision■recall, and calibration plots.
6. Risk scoring: convert predicted probabilities into a tiered risk score (e.g., Low/Medium/High) and flag top percentiles as high■risk.
7. Dashboarding: build Power BI visuals — heatmaps, bar charts, trend lines, and drill■through to product and order details for root■cause analysis.
8. Deliverables: interactive dashboard, Python codebase (notebooks/scripts), and CSV of high■risk products with actionable columns (product_id, risk_score, return_reason_count).

Conclusion
The combined approach (data analysis + predictive modelling + dashboarding) enables the business to prioritize interventions on high■risk products and channels. Short■term actions include targeted quality checks, improved product descriptions, and adjusted marketing for risky channels. Longer term, use model feedback to inform supplier negotiations and product assortment decisions. Key next steps: run A/B tests for mitigation strategies, retrain model monthly, and monitor dashboard KPIs (return rate, cost per return).

Deliverables
• Power BI interactive dashboard with drill■through filters.
• Python codebase (data cleaning + model training + scoring).
• CSV: high■risk products (for operational follow■up).