

# E-commerce Product Return Prediction Report

## Project Title

Predicting E-Commerce Product Returns Using Logistic Regression and Random Forest with SMOTE Balancing

## Objective

Build a predictive model to identify whether a customer will return a product after purchase. This helps e-commerce businesses reduce return-related costs and improve customer experience.

## Dataset Overview

The dataset contains features related to customer transactions, demographics, and return status.

Key columns include:

- Customer ID
- Purchase Date
- Product Category
- Product Price, Quantity, Total Purchase Amount
- Payment Method, Returns (Target)
- Customer Age, Gender, Churn

## Data Preprocessing

- Cleaned missing values and duplicates.
- Converted Returns to binary: 0 = No return, 1 = Returned.
- Used SMOTE for class imbalance.
- Feature scaling and encoding applied.

## Exploratory Data Analysis

- Return rates by product category visualized.
- Price vs returns trends.
- Age and gender distributions analyzed.
- Found that high-priced products and specific categories had higher return probability.

# E-commerce Product Return Prediction Report

## Models Used

1. Logistic Regression: for baseline classification and interpretability.
2. Random Forest Classifier: for better accuracy, nonlinear patterns, and ensemble performance.

## Model Evaluation

- Metrics: Confusion Matrix, Accuracy, Precision, Recall, F1 Score.
- Logistic Regression: ~85% Accuracy
- Random Forest: ~90% Accuracy
- SMOTE helped improve recall for returned products.

## Conclusion

- Random Forest outperformed Logistic Regression.
- SMOTE balanced the dataset, improving model fairness.
- Predicting returns can optimize e-commerce logistics and operations.

## Future Enhancements

- Include more behavioral data like reviews and browsing history.
- Try advanced models like XGBoost or Deep Learning.
- Deploy in real-time systems to detect risky transactions.

## Tools & Technologies

Python, Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn, SMOTE, Jupyter Notebook