Predicting Profitability in Retail Transactions

Machine Learning Project

A.Y. 24-25

University of Bologna
Digital Transformation Management

Alessandro Astolfi, Francesco Russo

Why Predict Profit?

In retail, profit doesn’t always follow sales.
High sales may hide unprofitable discounts, and small orders can still bring strong margins.
Understanding why some transactions make or lose money helps businesses optimize pricing and promotions.

Can we predict profit and understand what truly drives it?

Project Goal

The goal of our project was to predict the profit of each retail transaction.

From the very beginning, we were curious to explore what really drives profitability — whether it’s discounts, sales volumes, product categories, or shipping times.

Our analysis included data exploration, feature engineering, and model testing, aiming not just to build a good predictor, but to discover **meaningful business insights behind the numbers.**

Dataset Overview

We worked with the **Superstore dataset**, a collection of nearly 10,000 retail transactions described by 21 variables. Each row represents a single product (with its quantity) purchased within a customer order, including detailed information about customers, products, time, and sales performance.

Main Features

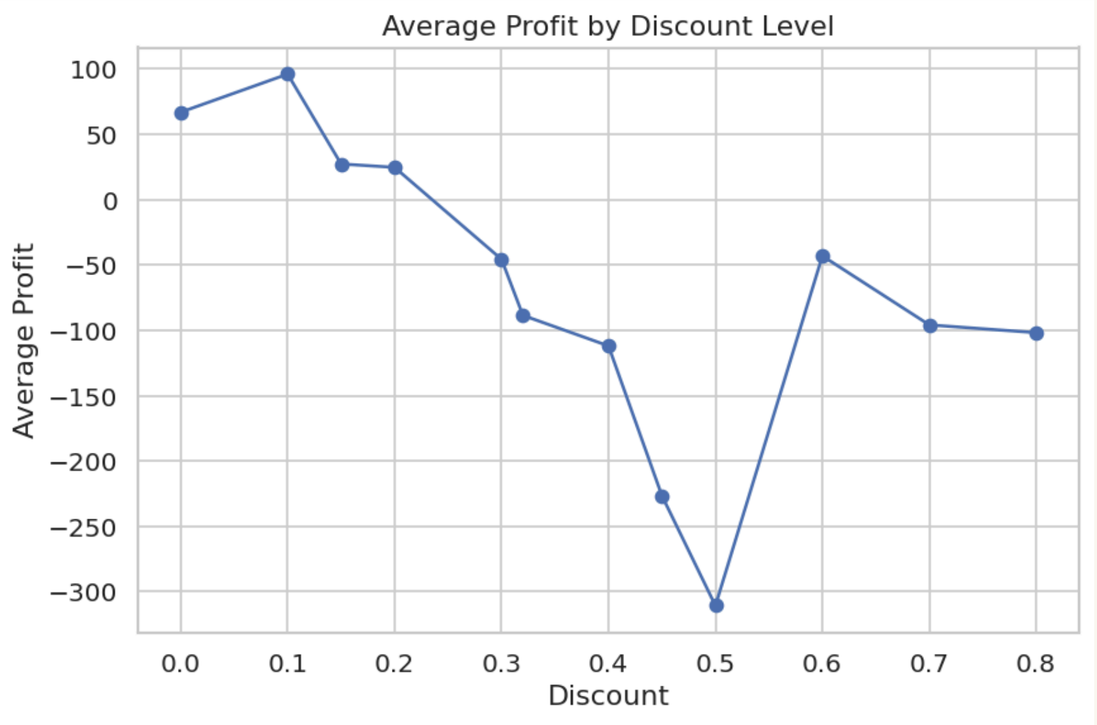
- Sales, Quantity, Discount, Profit → commercial metrics.
- Category, Sub-Category → product-level attributes.
- Segment, Region, State, City → customer & geographic info.
- Order Date, Ship Date, Ship Mode → temporal & logistics data.

Exploratory Data Analysis (EDA)

Before modeling, we conducted an exploratory analysis
to better understand distributions, correlations, and anomalies.

Key insights:

- Profit and Sales are right-skewed — a few large transactions dominate.
- Discount shows a strong negative correlation with Profit.
- **Profits tend to drop sharply for discounts above 20%.**
- Some transactions show extreme losses not explained by business logic.



Data Preparation

Before modeling, we focused on data quality and consistency.

- Verified missing values → none found
- Detected and removed one duplicate record
- Identified one suspicious transaction with a 0.2 discount but very low profit
    - This anomaly suggested the presence of further outliers, which we later analyzed in depth

These checks confirmed that the dataset was reliable, but hinted that **some extreme profit values could distort modeling results**.

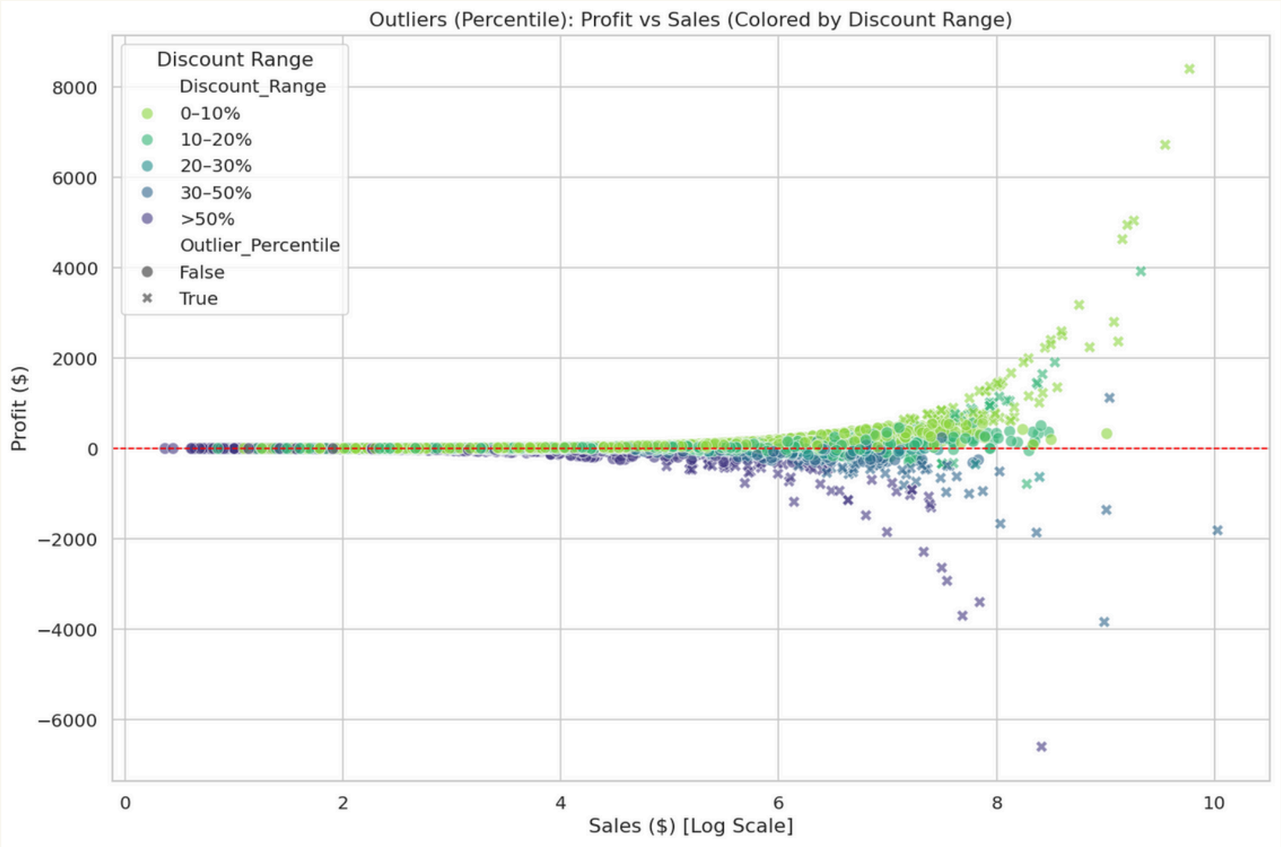
Outlier Detection & Treatment

Outliers were identified through visual inspection and statistical filtering.

- Scatter plots of Profit vs Sales showed extreme observations: some outliers had unusually high or low profits, with no consistent business explanation.

- Applied percentile-based filtering (1st–99th) on Profit: removed ≈2% of records (200 rows), forming the **“Normal Orders” dataset.**

This step reduced noise and improved model stability and reliability.



Modeling Approach: Baselines & Pipeline

We started simple and reproducible: an 80/20 split and a single preprocessing pipeline built with ColumnTransformer

- Linear Regression was our baseline — to test the limits of linearity.
- Random Forest captured non-linear effects and feature interactions.

Data pre-processing: categorical features were one-hot encoded, numerical ones scaled only for linear models.

Metrics: R^2, MAE, RMSE

Early results confirmed strong non-linear relationships and a **slight overfitting** of the Random Forest, which we later mitigated by tuning parameters like *max_depth* and *min_samples_leaf.*

Modeling Approach: Cross-Validation

After cleaning the data and stabilizing results, we moved to a **5-Fold Cross-Validation**, ensuring a more reliable and unbiased performance estimate.

We compared four model families:

- Linear Regression — interpretable baseline
- Random Forest — strong, flexible ensemble
- XGBoost and LightGBM — gradient boosting variants, efficient and accurate

Each model was trained within the same preprocessing pipeline, guaranteeing consistency and avoiding any data leakage.

Model Comparison (Cross-Validation Results)

Model	R^2	RMSE	MAE
Linear Regression	0.546	49.16	28.30
Random Forest	0.821	30.75	12.34
XGBoost	**0.823**	**30.67**	**12.77**
LightGBM	0.817	31.13	12.99

- The results clearly highlighted the superiority of non-linear ensemble models, with both XGBoost and Random Forest achieving R^2 scores above 0.82.
- Linear Regression, while interpretable, struggled to capture the complexity of the relationships driving profit variability.

Hyperparameter Optimization (Random Forest)

To refine the Random Forest model, we applied **RandomizedSearchCV** with 5-Fold Cross-Validation, testing 25 random parameter combinations.

The goal: find a balance between accuracy, stability, and efficiency — avoiding overfitting while keeping high predictive power.

Best configuration found:

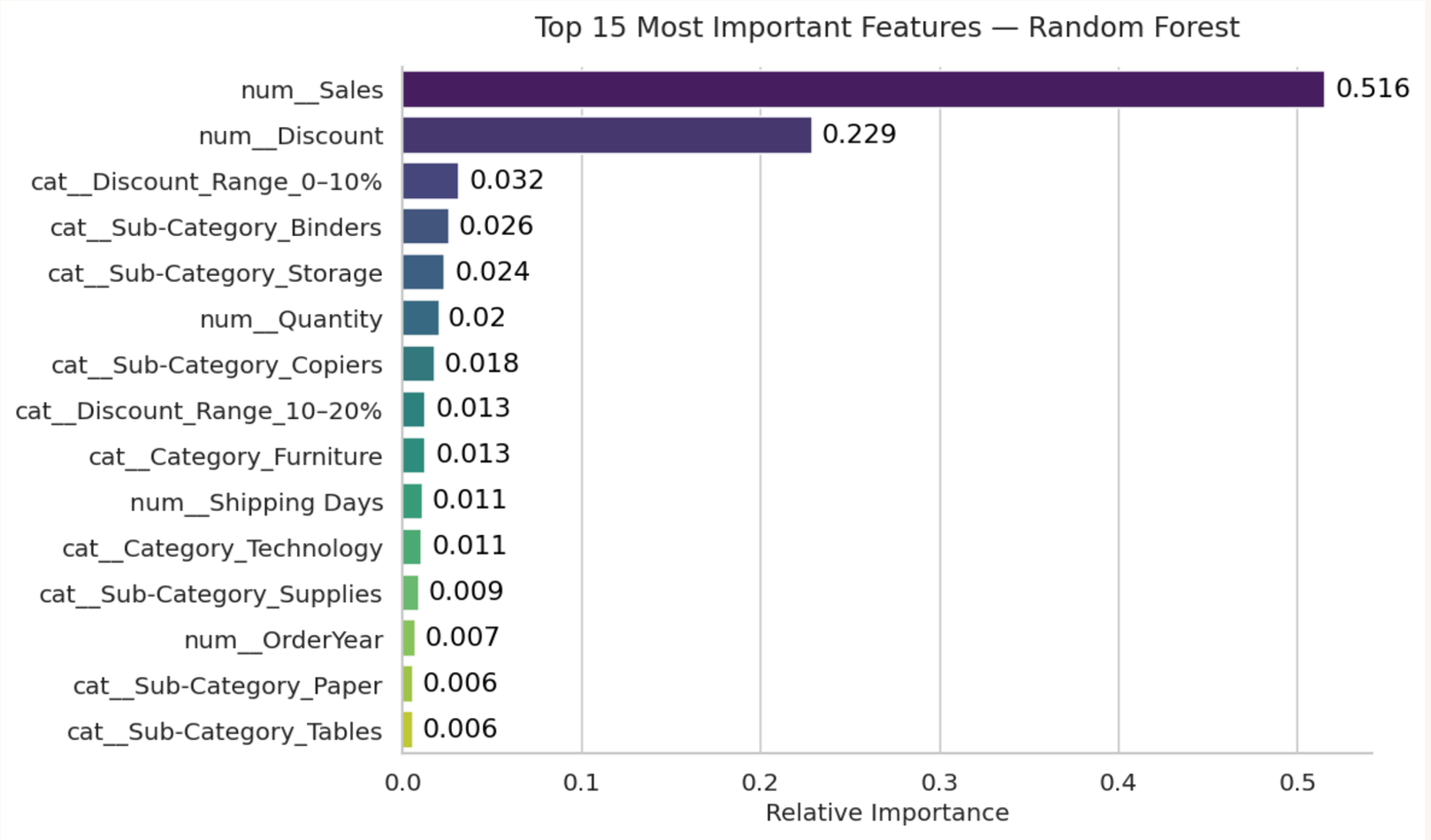
- n_estimators: 228
- max_depth: 29
- min_samples_leaf: 2
- max_features: None

Best cross-validated R^2: ≈ 0.829

Feature Selection

After tuning the Random Forest, we analyzed its feature importance to understand which variables truly drive profitability.

- Features **contributing less than 1%** to total model importance were removed, reducing the feature set from 46 → 11.



Final Model Comparison & Validation

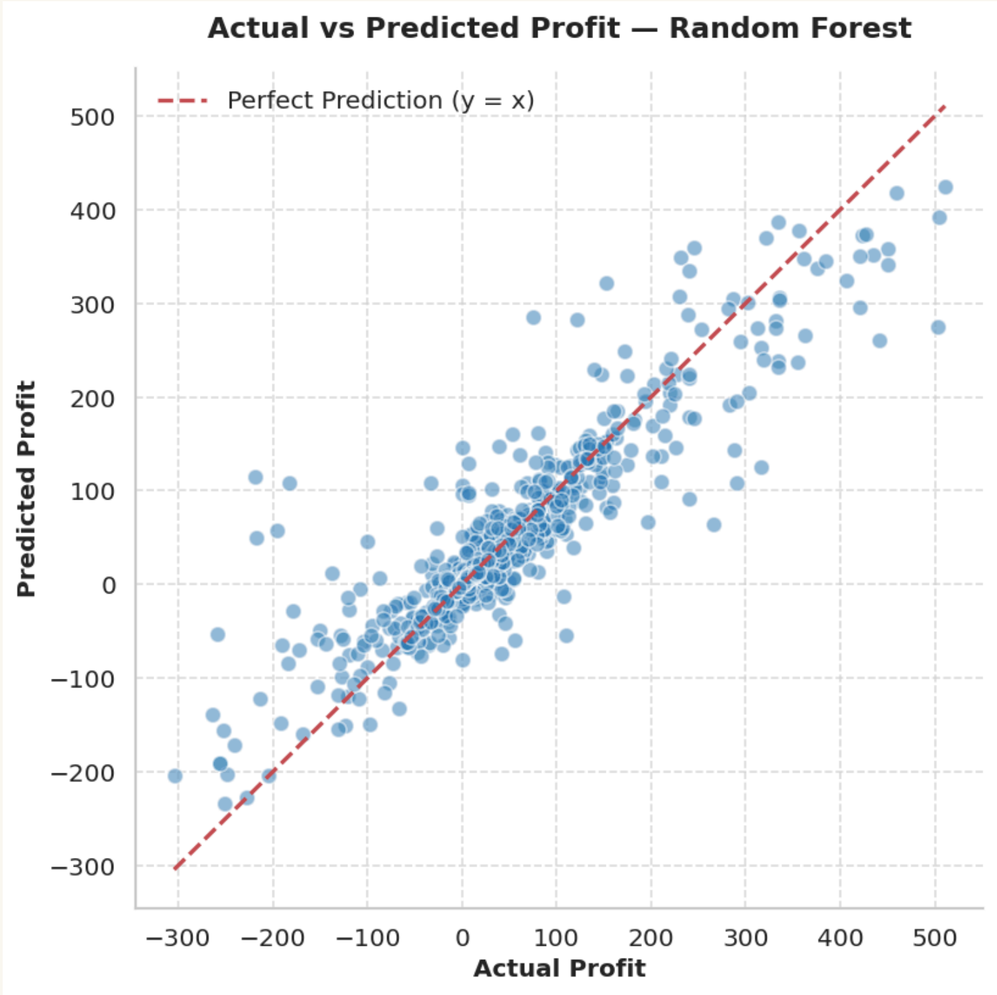
Model	R^2	RMSE	MAE
Random Forest (All Features)	**0.829**	**30.32**	**11.90**
Random Forest (Selected Features)	0.814	31.59	13.25
LightGBM	0.827	30.48	12.89

- The simplified model maintained nearly identical performance: $\Delta R^2 = -0.015$, with minimal increase in MAE and RMSE. This confirms that **most predictive power is concentrated in a small and interpretable subset of variables**.
- All models achieved comparable accuracy, confirming that our manually tuned pipeline was already highly effective, interpretable, and computationally efficient.

Model Evaluation - Optimized Random Forest

The optimized Random Forest Regressor achieves strong predictive accuracy.
Most points align closely along the diagonal, showing that predicted profits follow actual values with minimal bias.

- Slight deviations appear mainly for extreme cases — usually highly discounted or very high-sales transactions — but overall, **the model generalizes well** across normal operations.



Key Insights & Conclusions

Working on this project taught us that predicting profit is not just about accuracy — it's about understanding the dynamics behind profitability.

Here's what we take away from the analysis:

- **Discounts have a breaking point**: Moderate discounts drive sales, but beyond ~20%, profits collapse — a clear "discount cliff".
- **Efficiency pays off**: Faster shipping and smoother logistics consistently lead to higher profits.
- **Timing matters**: Promotions can boost sales but also increase volatility, so good timing is key to keeping margins healthy.
- **Simplicity wins**: Just a handful of features (Sales, Discount, Quantity, Shipping Days) explain most of the profit behavior.

Thank You