"Buy-again"
Product Recommendation Engine through Machine learning using customer price sensitivity

Abstract

With the rise of e-commerce in the last decade, particularly in the retail segment, customers have to choose the products they want from a wide selection which makes customers' shopping experience exhausting. From a service provider's perspective, there is an associated risk of losing the customer. In this project, we discuss how we built a "Buy-again" personalized recommendation engine that monitors customer behavior and provides personalized product recommendations to customers.

The engine is designed to consider customers' frequency, recency, monetary, price sensitivity as well as product prices and discounts. The engine uses information from in-store and online purchases for products purchased by customers at least once for the last two years. After identifying the probability of a customer purchasing a specific product on his/ her next visit, products with highest probabilities are listed in the "Buy-again" section on the retailor website.

Business Objectives

- Reduce check-out time
- ★★★ Increase customer satisfaction
- Long-term impact
  - Increase customer base
  - Increase sales

Overview

50% Of revenue comes from frequent products	100,000+ Customers
500+ Products	3+ Million Visits
$8+ Million Sales	2 Years of data

Methodology

Data Input
- Customer transactions
- Product pricing & substitutions

Data preparation & transformation
- Exploratory data analytics
- Customer price sensitivity (clustering)
- Feature engineering
- Feature selection

Modeling
Standardization → Resampling → Model design → Model training & testing → Model evaluation → Standardization

Model deployment and product prioritization

Features

- Customer - Product Recency
- Customer - Product Frequency
- Customer - Product Monetary
- Product & Substitution Price
- Customer Price Sensitivity

Machine Learning Models

15 Trials
Models & Features

Logistic Lasso Regression	62.1% Recall
Logistic Regression	62.0% Recall
Random Forrest	50.6% Recall

Best Performing Model

Logistic Lasso Regression

Train Recall	63.0%
Test Recall	62.1%
Area Under the Curve (AUC)	68.6%

Tools & Libraries

- python™
- Minitab®
- scikit learn
- Excel

Project Team

Feras Seder
fseder@purdue.edu

Lohita Srinivasan
srinival@purdue.edu

Mukul Sharma
sharm504@purdue.edu

Sai Nithin Reddy Mannepuli
smannepu@purdue.edu

Shreyansh Jain
jain475@purdue.edu

Vinit Vilas Jadhav
jadhavv@purdue.edu

Acknowledgement

Matthew Lanham
lanhamm@purdue.edu

Kirstin Dolick
kdolick@purdue.edu

PURDUE UNIVERSITY®
Krannert School of Management