Intelligent Bundle Generation for Personalized Promotion Optimization in Retail

Submitted in partial fulfillment of the requirements for the degree

of

Master of Technology

by

Vinod Kumar

Roll No: MA23M026

Under the guidance of

Guide: Prof. Satyajit Roy

Co-guide: Dr. Sharadha Ramanan

DEPARTMENT OF MATHEMATICS

INDIAN INSTITUTE OF TECHNOLOGY MADRAS

Declaration

I hereby declare that the work presented in this dissertation entitled **"Intelligent Bundle Generation for Personalized Promotion Optimization in Retail"**, submitted in partial fulfilment of the requirements for the award of the degree of **Master of Technology in Industrial Mathematics & Scientific Computing**, is my own work carried out in the **Department of Mathematics, Indian Institute of Technology Madras**.

This project was conducted under the supervision of **Prof. Satyajit Roy (Guide), Department of Mathematics, IIT Madras**, and **Dr. Sharadha Ramanan (Co-Guide), TCS Research and Innovation, IIT Madras Research Park**, from June 2024. To the best of my knowledge, the content of this dissertation has not been submitted for the award of any other degree or diploma at this or any other institute or university.

Date: May 2025
Place: IIT Madras

Vinod Kumar
M.Tech. in Industrial Mathematics & Scientific Computing
Department of Mathematics
IIT Madras

Certificate

This is to certify that the report titled **"Intelligent Bundle Generation for Personalized Promotion Optimization in Retail"**, submitted by **Vinod Kumar (MA23M026)**, in partial fulfillment of requirement for the award of the degree of Master of Technology in Industrial Mathematics and Scientific Computing, Indian Institute of Technology Madras, is the record work done by him during the academic year 2024-2025 in the Department of Mathematics, IIT Madras, India, under my supervision.

Prof. Satyajit Roy
Department of Mathematics
IIT Madras

Dr. Sharadha Ramanan
TCS Research & Innovation
IITM Research Park, Chennai

Acknowledgement

I take this opportunity to express my profound gratitude to my project guide, **Prof. Satyajit Roy**, Department of Mathematics, IIT Madras, and my co-guide **Dr. Sharadha Ramanan**, TCS Research & Innovation, for their continuous guidance, encouragement, and invaluable insights at every stage of this project.

I am deeply thankful to the **Department of Mathematics, IIT Madras**, and **TCS Research & Innovation** for their collaborative support and for providing me with the opportunity to undertake this industrial project as part of my academic coursework.
My sincere appreciation goes to **Ms. Uma G Maheshwari** (Developer, TCS Research & Innovation), **Ms. Srividhya Sethuraman** (Researcher, TCS Research & Innovation) and **Prabhat Kumar** for their timely assistance, and technical guidance for this project.

Vinod Kumar
MA23M026
Department of Mathematics
IIT Madras

Abstract

This project aims to develop an intelligent decision-support system for generating personalized product bundles and optimizing promotional strategies in the retail industry. The primary objective is to maximize overall sales and revenue by forecasting product demand and modeling cross-price elasticit, how the price or presence of one product affects the demand for others.

The problem is formulated as a **Mixed-Integer Nonlinear Programming (MINLP)** model, which enables both binary decisions (e.g., whether to include a product in a bundle) and continuous decisions (e.g., how much of each product to include). The optimization model incorporates key factors such as baseline demand, product interdependencies, pricing strategies, and real-world business constraints like promotional budgets, minimum bundle sizes, and revenue thresholds.

To support accurate forecasting and dynamic bundle optimization, the system leverages **machine learning techniques** for demand prediction and cross-elasticity estimation. Tools and technologies used include **Python**, **Pandas**, **NumPy**, **Scikit-learn** for modeling, and **GEKKO** for formulating and solving the optimization problem.

For experimentation and validation, **retail transaction datasets** real market simulated sales data were used. This dataset provides insights into customer purchasing behavior, product affinities, and pricing sensitivity, allowing the model to recommend product bundles that align with customer preferences and maximize promotional effect.

The resulting system delivers **data-driven, customer-centric bundle recommendations** that improve the effectiveness of promotional campaigns and drive better business outcomes for retailers.

Contents

1 Introduction and Objectives

In today's fast-changing retail environment, customers expect more than just convenience—they want experiences tailored to their preferences. Traditional fixed product bundles, such as a shampoo and conditioner combo, were once used to encourage customers to buy more. However, these bundles are the same for all customers and fail to consider individual needs. As a result, they often feel generic and lead to lower engagement and fewer purchases.

Today's consumers prefer bundles that reflect their specific interests and shopping habits. For example, one shopper may regularly purchase healthy breakfast items, while another might be buying snacks for children. Offering everyone the same set of products no longer meets the demands of a diverse customer base.

Figure 1: AI-Powered Bundling for Personalized Retail Promotions

To address this shift, we can apply Artificial Intelligence (AI), machine learning, and optimization techniques to understand customer behavior and automatically generate personalized product bundles. These technologies allow businesses to analyze large volumes of transaction data and uncover patterns in how products are purchased together. By identifying such patterns, we can create smarter, more relevant bundles that increase customer satisfaction and drive sales.

This project focuses on designing an intelligent system that combines AI and mathematical optimization to generate product bundles dynamically. Instead of offering static combinations, the system learns from real-world sales data and adapts over time. For instance, if the system finds that many customers often purchase cereal and almond milk together, it can suggest a breakfast bundle that also includes granola or juice. The system evolves as more data is collected, continuously improving its recommendations based on seasonal trends, customer segments, and feedback.

Project Objectives

- **Analyze historical sales and customer behavior:** Study past purchases to identify popular products, frequent buying patterns, and seasonal demand shifts.

- **Discover product affinities using machine learning:** Apply algorithms to find products that are often bought together or influence each other's demand, such as complementary or substitute goods.

- **Generate optimal bundles using advanced optimization:** Develop a bundle generation framework using Mixed-Integer Nonlinear Programming (MINLP), solved through the GEKKO optimization suite with the APOPT solver. The model incorporates business rules (e.g., budget limits, pricing strategies) and product relationships to generate feasible and profitable bundles.

- **Personalize recommendations for individual customers or segments:** Use customer profiles and behavior data to tailor bundles, making the shopping experience more relevant and engaging.

- **Optimize promotional strategies to improve sales performance:** Align bundles with promotional goals such as increasing the average basket size, boosting conversion rates, and attracting new customers.

- **Evaluate system performance with key metrics:** Use performance indicators like average order value, bundle adoption rate, and customer retention to measure the effectiveness of the system and guide improvements.

Traditionally, bundles (like shampoo + conditioner) were the same for everyone. But today, customers expect bundles tailored to their preferences and shopping habits.

This project proposes an AI-powered system that creates personalized product bundles by analyzing customer purchase data. Using machine learning and optimization (like MINLP with GEKKO), it dynamically forms bundles that better match individual customer needs, improve shopping experience, and boost sales and loyalty.

2 Traditional Methods and Literature Survey

2.1 Traditional Promotional Strategies

Traditional promotional strategies in retail primarily include fixed discounting, rule-based bundling, and generic marketing campaigns. Fixed discounting refers to offering the same discount across selected products without tailoring the offer to individual customer behavior or preferences. Rule-based bundling generally relies on historical purchase patterns or expert-defined rules to create bundles, often overlooking customer-specific interests. Generic campaigns, such as broad-based coupons or loyalty rewards, target the mass market rather than distinct user segments. While these methods are simple and widely adopted, they often lead to inefficiencies. The lack of personalization results in reduced customer engagement and conversion, ultimately lowering return on investment (ROI).

2.2 Literature Survey

The development of bundle recommendation systems has significantly evolved with the incorporation of data-driven and machine learning approaches. These newer systems aim to overcome the limitations of traditional strategies by focusing on personalization, dynamic content generation, and alignment with customer behavior.

Sun and Li [3] categorized bundle recommendation techniques into discriminative and generative models. Discriminative methods identify frequent itemsets or use classification models to select appropriate bundles, while generative methods learn from user interaction data to generate personalized bundles. Their work presents a unified framework applicable across industries like retail, entertainment, and gaming.

Wei and Liu [4] introduced a framework for personalized bundle creative generation, which not only selects bundles but also creates associated marketing content such as slogans and templates. Their approach employs a contrastive non-autoregressive model, improving both the quality and speed of generation.

Sun and Yang [5] focused on automatically creating bundles aligned with personalized promotions by leveraging user behavior data. Their technique improves recommendation quality by aligning promotions with user preferences and business objectives.

Zhu and Harrington [6] formalized the Bundle Recommendation Problem (BRP), offering a comprehensive optimization framework that integrates customer preferences, business constraints, and profitability goals for intelligent bundle formation.

Pathak and Gupta [7] applied matrix factorization and collaborative filtering to generate personalized bundles by modeling complex user-product interactions. Their method emphasizes behavioral personalization.

Liu and Fu [8] emphasized understanding customer motives behind purchases to form more meaningful bundles. Their behavioral modeling approach results in recommendations that better resonate with user intent.

Beheshtian Ardakani and Fathian [9] proposed a novel bundling model that integrates customer loyalty and market segmentation. Using evaluation metrics like the silhouette coefficient, support, and confidence, they demonstrated improved effectiveness in direct marketing.

Ettl and Harsha [10] advocated for a data-driven optimization approach using customer interaction data. Their system aims to strike a balance between customer appeal and business profitability.

Uma G et al. [11] proposed a Mixed-Integer Nonlinear Programming (MINLP) approach to optimize retail promotions by accounting for cross-elasticity effects, bundle conflicts, and business constraints. Their method demonstrated superior margin performance over heuristic and standard non-linear solvers using real-world retail data.

These studies collectively highlight the growing importance of personalization, behavioral modeling, and machine learning in bundle recommendation systems. By addressing user intent and business goals simultaneously, these approaches enhance customer satisfaction and promotional effectiveness.

3 Basic Ingredients for Methodology

This section explains the key steps needed to design a smart bundling and promotional strategy. The goal is to make the system personalized for each customer, aware of product demand, and sensitive to pricing.

3.1 Exploratory Data Analysis (EDA) and Preprocessing

This step is important to understand the dataset, find patterns, and get the data ready for modeling.

First, we collected three types of data: transactional data, customer data, and product data. Transactional data includes past purchase records like product IDs, quantities purchased, prices, and timestamps of the purchases. Customer data includes information like age, gender, location, and shopping history. It also includes customer preferences and behavior patterns. Product data contains details like product category, brand name, price history, production cost, and where the product fits in the product hierarchy (for example, department $\rightarrow$ category $\rightarrow$ sub-category).

Next, we cleaned the data. This means we fixed any problems in the dataset. For example, if some values were missing (null), we either removed those rows or filled in the missing values in a smart way. If there were negative prices or duplicated entries, we corrected or removed them. We also made sure all formats were consistent. For example, all date fields followed the same format, and all category names were standardized.

After cleaning, we created new features that add value to the dataset. One such feature is the product affinity score, which tells how likely two products are to be bought together. Another is Customer Lifetime Value (CLV), which predicts how much money a customer is expected to spend over time. We also created demand-related features, which help us understand how product demand changes with price or during different times of the year.

3.2 Customer Segmentation

Segmenting customers helps us create more personalized offers and bundles.

We first used RFM analysis to group customers based on their buying behavior. RFM stands for Recency, Frequency, and Monetary. Recency tells us how recently a customer made a purchase. Frequency tells us how often they buy. Monetary tells us how much money they usually spend. This analysis helps us identify loyal customers, new customers, and customers who are at risk of leaving.

Then we applied clustering techniques to group customers into segments. We used algorithms like K-means, which divides customers into a fixed number of groups based on their similarity. We also used DBSCAN, which is another clustering method that can discover groups of different shapes and sizes. It also helps identify outlier customers who do not fit into any group.

3.3 Interrelated Items

Understanding how products are related helps us create better product bundles—ones that customers are more likely to buy together.

We started by identifying frequent itemsets, which are groups of products that are often purchased together in the same transaction. To do this, we used well-known algorithms like Apriori and FP-Growth. Apriori works by gradually building larger itemsets based on frequent smaller ones, while FP-Growth is a faster alternative that uses a tree structure to efficiently discover frequent patterns in the purchase data.

To go beyond simple co-purchase counts and uncover deeper product relationships, we applied Matrix Factorization. First, we constructed a product co-occurrence matrix, where each cell indicates how often two products were bought together. Matrix Factorization then decomposes this large matrix into smaller latent factors that capture hidden similarities between products.

The result is a set of product embeddings — numeric vectors that represent each product and summarize its relationship with others. These embeddings help us identify products that are similar or often associated with one another, even if they do not frequently appear in the same basket.

3.4 Price Elasticity and Cross Elasticity Modeling

Understanding how prices affect demand is critical for setting optimal prices, planning promotions, and increasing profitability without sacrificing customer satisfaction. In this section, we study how a product's demand responds to changes in its own price (own-price elasticity) and how it is affected by the price of related products (cross-price elasticity).

Own-Price Elasticity

Own-price elasticity measures the percentage change in demand for a product when its own price changes. The formula used is:

$$\epsilon_{ii} = \frac{\Delta d_i}{\Delta p_i} \cdot \frac{p_i}{d_i}$$

Where:

- d_i is the demand for product i
- p_i is the price of product i
- $\frac{\Delta d_i}{\Delta p_i}$ is the rate of change in demand with respect to price

The interpretation of ϵ_{ii} is as follows:

- If $\epsilon_{ii} < -1$, the product is **price elastic**. A small decrease in price leads to a relatively large increase in demand. Example: consider branded breakfast cereals — if the price of Kellogg's Corn Flakes drops by 10%, customers may significantly increase their purchases, or switch from other brands, leading to a demand rise of more than 10%.

- If $-1 < \epsilon_{ii} < 0$, the product is **price inelastic**. A change in price results in a smaller relative change in demand. Example: salt or cooking oil — even a 10% increase in price might not change demand much, because these are daily essentials and there are few substitutes.

In our modeling, we estimate elasticity using historical sales data. By fitting regression models or applying finite difference methods on time series of price and quantity sold, we can approximate how sensitive demand is to price changes for each product.

Cross-Price Elasticity

Cross-price elasticity measures the percentage change in demand for one product (i) when the price of a different product (j) changes. The formula is:

$$\epsilon_{ij} = \frac{\Delta d_i}{\Delta p_j} \cdot \frac{p_j}{d_i}$$

The value of ϵ_{ij} helps us understand the relationship between products:

- If $\epsilon_{ij} > 0$, the products are **substitutes**. An increase in the price of product j will increase the demand for product i.
  Example: If the price of Pepsi increases, customers may switch to Coca-Cola, increasing its demand.

- If $\epsilon_{ij} < 0$, the products are **complements**. An increase in the price of product j reduces the demand for product i.
  Example: If the price of coffee machines increases, the demand for coffee pods may fall, as fewer customers are likely to buy the machine and therefore do not need the pods.

- If $\epsilon_{ij} \approx 0$, the products are **independent**, meaning there is little to no relationship between them.
  Example: A price change in toothpaste is unlikely to affect the demand for cooking oil.

In our system, we use cross-price elasticity insights to decide which products should or should not be bundled together. For instance, bundling substitutes (like two types of detergent) might not make sense unless we are offering a choice. On the other hand, bundling complements (e.g., pasta and pasta sauce) can drive increased sales of both items.

Real-World Use Case in Promotion Design

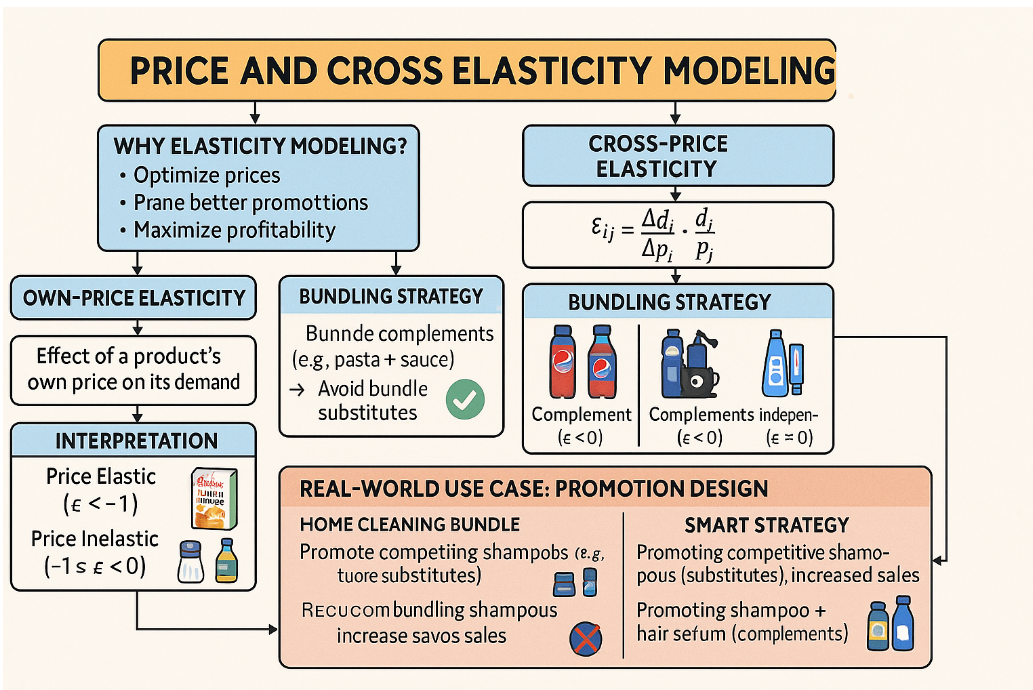


Figure 2: Price and Cross Elasticity

Assume we are setting up a promotion for a home cleaning bundle: floor cleaner, mop, and disinfectant spray. Based on cross-price elasticity estimates, we observe that the demand for mops is highly sensitive to changes in the price of floor cleaners (i.e., they are strong complements). Therefore, reducing the price of the floor cleaner by 15% may significantly boost sales of both mops and disinfectant sprays, improving the total basket size and revenue.

On the contrary, promoting two shampoos from competing brands together may not be effective, as they are substitutes and likely to cannibalize each other's demand. Instead, offering a shampoo with a hair serum (complement) could result in a better response.

By combining both own-price and cross-price elasticity into our demand forecasting and bundling engine, we are able to simulate different pricing scenarios and recommend strategies that increase customer engagement and revenue in a data-driven manner.

3.5 Demand Forecasting

Demand forecasting is essential for ensuring product availability, managing inventory, optimizing supply chains, and designing effective promotional strategies. Accurate forecasts empower businesses to meet customer needs efficiently while avoiding overstocking or stockouts.

Baseline Demand Estimation

We began by estimating the **baseline demand** for each product — the expected sales volume under regular market conditions without any promotional influence or price discounts. This provides a reference point to evaluate the effect of marketing strategies and seasonal events.

To estimate baseline demand, we analyzed historical sales data over several months to identify:

- *Seasonal trends* (e.g., higher sales of ice cream in summer)
- *Weekly patterns* (e.g., increased grocery shopping on weekends)
- *Holiday effects* (e.g., demand spikes during Diwali or Christmas)
- *Product lifecycle behavior* (e.g., new products gaining popularity over time)

We also included features derived from both product and customer metadata such as:

- Product attributes: category, sub-category, brand, regular price
- Customer behavior: purchase frequency, average basket size, preferred time/day of shopping
- Contextual features: time of year, day of the week, promotional calendar

Machine Learning Models

To model and predict demand, we experimented with several supervised machine learning algorithms, including:

- **Linear Regression**: A simple model that assumes a linear relationship between input features and demand. While interpretable, it may underperform for non-linear relationships.
- **Random Forest Regressor**: An ensemble of decision trees that captures non-linear dependencies and interactions between variables. It performs well with minimal feature scaling and handles categorical features effectively.
- **XGBoost Regressor**: A powerful gradient-boosting framework that iteratively builds trees to reduce prediction error. It often yields high accuracy and robust performance on structured data.

We trained the models using historical sales as the target and evaluated their performance using metrics such as RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error). XGBoost consistently outperformed other models in our experiments due to its ability to handle sparse data and non-linearities.

Real-World Examples and Applications

- *Example 1 (Grocery Category)*: For a perishable product like milk, accurate daily demand forecasting is critical. Under-forecasting can lead to lost sales and customer dissatisfaction, while over-forecasting causes waste. Using historical demand and weekday effects, our model predicts milk demand with a $\pm 5\%$ error margin, enabling better procurement and shelf management.
- *Example 2 (Festival Season)*: During Diwali, demand for sweets and gifting items spikes. By incorporating calendar features and past festive sales, our models were able to predict demand spikes up to 3 weeks in advance. This allowed retailers to adjust stock levels and plan marketing campaigns proactively.

- *Example 3 (Price Sensitivity)*: For high-value items like chocolates or energy drinks, demand fluctuates with price changes. By combining price features with elasticity scores, we predicted how demand would shift with different discount levels, helping marketing teams plan optimal promotions.

Business effect
Our demand forecasting framework enabled:

- Inventory planning with higher accuracy, reducing overstocking and wastage
- Targeted promotions based on predicted demand shortfalls or surpluses
- Dynamic pricing simulations based on predicted lift in demand
- Improved coordination between supply chain and marketing teams

In summary, by leveraging historical data and machine learning models, we developed a reliable system to forecast product demand at scale, improving both customer satisfaction and operational efficiency.

3.6 Mixed-Integer Nonlinear Programming (MINLP) and Its Relevance to the project

Mixed-Integer Nonlinear Programming (MINLP) is a powerful mathematical optimization framework that combines the capabilities of both Mixed-Integer Programming (MIP) and Nonlinear Programming (NLP). In an MINLP problem, some decision variables are constrained to be integers (often representing binary decisions like selection or assignment), while others can be continuous. Additionally, the objective function and/or constraints can be nonlinear, making these problems both flexible and challenging to solve.

MINLP models are highly applicable to real-world problems that involve complex decision-making under nonlinear dynamics and discrete choices. In the context of retail promotion and bundle recommendation systems, the use of MINLP is particularly promising due to the multifaceted nature of the problem. Retailers often seek to optimize multiple objectives simultaneously, such as maximizing expected revenue, minimizing cost, and enhancing customer satisfaction, all while adhering to operational and business constraints.

For example, in bundle generation and promotion planning, the decision to include a product in a bundle can be modeled as a binary variable (1 if included, 0 otherwise). At the same time, the expected profit or utility from that bundle may depend nonlinearly on factors such as historical purchase frequencies, price elasticity, cross-product influence, and promotion effect — all of which can be captured through nonlinear functions. The resulting formulation naturally leads to a MINLP problem.

Moreover, customer constraints such as purchase limits, inventory constraints on the supplier side, and minimum/maximum bundle sizes add further layers of complexity, reinforcing the need for an expressive modeling framework like MINLP. This allows for the incorporation of:

- Binary decisions on product inclusion in bundles. - Nonlinear customer behavior models, such as diminishing returns or interaction effects. - Business rules such as price thresholds, exclusivity, or promotion budgets. - Objective functions that blend profit, conversion likelihood, and personalization quality.

Recent studies in intelligent promotion and personalization, such as those incorporating behavioral modeling and personalized marketing, align well with the structure of MINLP. While many data-driven methods like collaborative filtering or neural networks focus on prediction, MINLP provides a complementary approach by optimizing over the decision space — making it a strategic layer on top of predictive analytics.

Despite its strength, solving MINLPs can be computationally intensive due to the non-convex nature and combinatorial explosion of possibilities. Therefore, solvers such as GEKKO and CPLEX are often employed, sometimes with relaxation techniques or heuristic approximations to obtain near-optimal solutions within reasonable time.

In this project, MINLP serves as a bridge between data-driven insights and actionable decision-making for personalized promotions and bundle optimization. By integrating user-specific patterns, product affinities, and business objectives into a unified mathematical model, MINLP enables a structured approach to designing effective and profitable promotion strategies in the retail domain.

4 Methodology and Mathematical formulation

The goal of this methodology is to design an intelligent system for generating personalized product bundles and optimizing promotions in the retail sector. By incorporating the concept of **cross elasticity of demand**, **bundle generation**, and **demand forecasting**, the approach aims to maximize overall sales revenue while respecting key constraints, including budget limits, demand forecasts, and minimum bundle size requirements. The methodology is formulated as a Mixed-Integer Nonlinear Programming (MINLP) problem, which combines both continuous and binary variables.

4.1 Problem Definition and Variables

The problem is defined as the generation of product bundles for personalized promotions in retail, with the objective of maximizing revenue while ensuring that the bundles meet customer demand forecasts and business constraints. The key variables used in this formulation are:

- x_i: A binary variable that indicates whether product i is included in the bundle. $x_i = 1$ if product i is included in the bundle, and $x_i = 0$ otherwise.
- $\delta[i]$: The discount rate for product i
- $P[i]$: The price of product i in the bundle.
- $P_d[i]$: The discounted price for product i
- D_i: The demand for product i, which depends on both the price and the cross-elasticity with other products in the bundle.
- $\hat{D}_i$: The baseline demand for product i without considering any cross-price effects.

- ϵ_{ij}: The cross-price elasticity between products i and j, representing how the demand for product i changes in response to changes in the price of product j.
- C_i: The cost of product i.
- B: The available budget for the bundle.
- y_i: The quantity of product i in the bundle.
- R_{min}: The minimum required revenue from the bundle.
- M_{min}: The minimum number of products required in each bundle.
- L : The list of interrelated items.
- k : The maximum units bundle can have.

4.2 Cross Elasticity and Demand Forecasting

Cross elasticity of demand measures how the demand for one product is affected by the price change of another product. The demand for product i in bundle B is formulated as:

$$D_i = x_i \cdot \hat{D}_i \cdot \left(1 + \sum_{j \neq i} \epsilon_{ij} \cdot \frac{P_j - P_i}{P_i}\right)$$

This equation introduces a nonlinear term in the model, as the demand for a product is influenced by both its own price and the prices of other products in the bundle, which introduces nonlinearity due to the interaction terms $x_i \cdot \epsilon_{ij} \cdot \frac{P_j - P_i}{P_i}$.

Where:

- x_i ensures that the demand for a product is only considered if the product is included in the bundle.
- ϵ_{ij} represents the cross-price elasticity between products i and j, influencing the demand for product i based on the price difference relative to other products in the bundle.

- The term $\frac{P_j - P_i}{P_i}$ captures the percentage change in the price of i relative to j.

This formulation ensures that the demand for each product is influenced not only by its own price but also by the prices of other products included in the bundle, thus capturing the interdependencies between product demands.

4.3 Objective Function: Maximizing Revenue

The objective of this optimization problem is to maximize the total revenue R_B from the bundle, considering the prices and demand for each product. This leads to the following objective function:

$$R_B = \sum_{i \in L} P_i \cdot \left(x_i \cdot \hat{D}_i \cdot \left(1 + \sum_{j \neq i} \epsilon_{ij} \cdot \frac{P[j] - P_d[i]}{P_d[i]} \right) \cdot y_i \right)$$

Discounted price

$$P_d[i] = P[i] \cdot (1 - \delta[i])$$

This objective function is **nonlinear** because of the demand equation, which contains cross-price elasticity terms. The revenue depends on both the prices and the demand, which is influenced by both the individual product prices and their interdependencies (cross-elasticities) with other products in the bundle.

4.4 Constraints

The optimization problem must adhere to the following constraints:

4.4.1 Budget Constraint

The total cost of the products in the bundle must not exceed the available budget B:

$$\sum_{i \in L} C_i \cdot x_i \leq B$$

4.4.2 Demand Limitation

The quantity of each product in the bundle should not exceed the forecasted demand:

$$x_i \cdot y_i \leq \hat{D}_i \quad \forall i$$

4.4.3 Revenue Requirement

The total revenue from the bundle must meet a minimum threshold $R_{\min}$:

$$\sum_{i \in L} P_d[i] \cdot y_i \geq R_{\min}$$

4.4.4 Non-negativity and Binary Constraints

The decision variable x_i must be binary, indicating whether product i is in the bundle (1) or not (0):

$$x_i \in \{0, 1\} \quad \forall i$$

The quantity of each product y_i must be non-negative:

$$k \geq y_i \geq 0 \quad \forall i$$

4.4.5 Bundle Size Constraint

The total number of products in the bundle must meet the minimum required size $M_{\min}$:

$$\sum_{i \in L} x_i \geq M_{\min}$$

4.4.6 Discount Bounds

The discount should be less then maximum discount :

$$0 \leq \delta[i] \leq \delta_{max}$$

4.5 Mathematical Model: MINLP Formulation

Objective Function:

$$\text{Maximize: } R_B = \sum_{i \in L} P_d[i] \cdot \left(x_i \cdot \hat{D}_i \cdot \left(1 + \sum_{j \neq i} \epsilon_{ij} \cdot \frac{P_d[j] - P_d[i]}{P_d[i]} \right) \cdot y_i \right)$$

Subject to:

- **Budget Constraint:**

$$\sum_{i \in L} C_i \cdot x_i \leq B$$

- **Demand Limitation:**

$$x_i \cdot y_i \leq \hat{D}_i \quad \forall i$$

- **Revenue Requirement:**

$$\sum_{i \in L} P_d[i] \cdot y_i \geq R_{\min}$$

- **Non-negativity and Binary Constraints:**

$$x_i \in \{0, 1\} \quad \forall i$$

$$k \geq y_i \geq 0 \quad \forall i$$

- **Bundle Size Constraint:**

$$\sum_{i \in L} x_i \geq M_{\min}$$

- **Discount Bounds**

$$0 \leq \delta[i] \leq \delta_{max}$$

Algorithm 1: Intelligent Bundle Generation with Optimal Discounting using MINLP Framework

Input : ProductList, Total Budget (B), Minimum Revenue Threshold (R_min), Minimum Bundle Size (M_min), Max Discount δ_{max}

Variables: Binary selection variable ($x[i]$),
Quantity variable ($y[i]$),
Discount rate ($\delta[i]$),
Discounted price ($P_d[i] = P[i] \cdot (1 - \delta[i])$),
Demand ($D[i]$)

Output : BestBundle, Quantities, Discounts, ExpectedRevenue

for *each product i in ProductList* **do**
  $x[i] \in \{0, 1\}$
  $y[i] \geq 0$
  $0 \leq \delta[i] \leq \delta_{max}$
  $P_d[i] = P[i] \cdot (1 - \delta[i])$
  if $x[i] == 1$ **then**
    $D[i] = \hat{D[i]} \cdot \left(1 + \sum_{j \neq i} \epsilon[i][j] \cdot \frac{P_d[j] - P_d[i]}{P_d[i]}\right)$
  else
    $D[i] = 0$

Define objective:
  Maximize total revenue: $R\_B = \sum_i P_d[i] \cdot D[i] \cdot y[i]$

Subject to:
1. Budget constraint: $\sum_i C[i] \cdot x[i] \leq B$
2. Demand limitation: $x[i] \cdot y[i] \leq D[i] \quad, \forall i$
3. Revenue requirement: $\sum_i P_d[i] \cdot y[i] \geq R\_min$
4. Bundle size constraint: $\sum_i x[i] \geq M\_min$
5. Discount bounds: $0 \leq \delta[i] \leq \delta_{max}$
6. Binary and non-negativity constraints: $x[i] \in \{0, 1\}, \quad y[i] \geq 0$

return *BestBundle* $= \{i \mid x[i] = 1\}$,
Quantities $= y[i]$ *for each product in BestBundle,*
Discounts $= \delta[i]$ *for each product in BestBundle,*
ExpectedRevenue $= R_B$

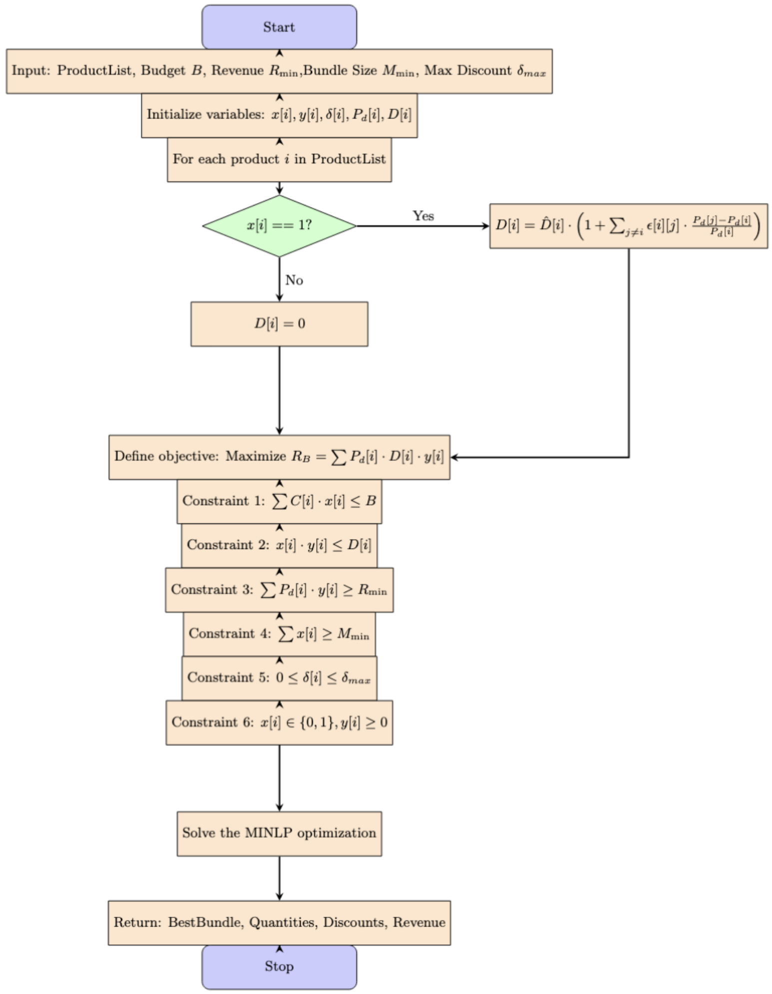


Figure 3: Steps to perform.

4.6 Solution Approach

To solve this MINLP, we use a combination of optimization techniques such as:

- **Branch and Bound** or **Branch and Cut** for handling the binary variables.
- **Nonlinear programming solvers** (e.g., APOPT, IPOPT or KNITRO) for handling the nonlinear demand model.
- **Heuristic methods** (e.g., Genetic Algorithms or Simulated Annealing) for larger-scale problems or when exact methods are computation-

ally expensive.

The solver iterates over potential solutions, considering the interdependencies between product prices, cross elasticities, and the binary selection of products, to find the optimal bundle that maximizes revenue while satisfying the constraints.

4.7 Implementation and Data Sources

The solution is implemented in **Python**, utilizing the **GEKKO optimization suite**, specifically the **APOPT solver**, to handle the Mixed-Integer Nonlinear Programming (MINLP) formulation. GEKKO is particularly well-suited for solving large-scale optimization problems with nonlinear relationships and integer decision variables, making it a powerful tool for this application.

Since direct access to complete retail data is limited, the dataset was **simulated based on realistic assumptions and patterns observed in retail environments**. The simulation was guided by real-world retail scenarios and included the following components:

- **Product prices and costs**: Simulated based on average market prices and typical markup margins used by retailers.
- **Demand forecasts**: Generated using synthetic time-series data that mimics seasonal trends, promotional spikes, and customer behavior variations.
- **Cross elasticity data**: Estimated using assumed customer substitution and complementarity behavior, modeled using econometric-inspired formulas that capture inter-product relationships.

Key assumptions made during the simulation include:

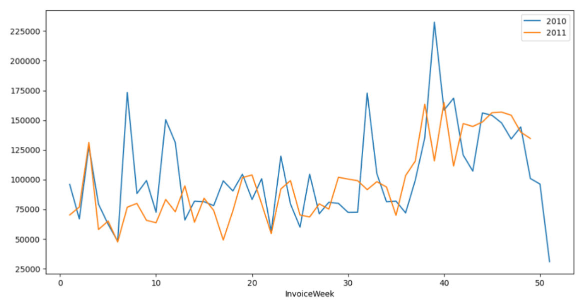
- Fixed product categories with defined price ranges and cost structures.
- Seasonal variations in demand based on assumed time periods (e.g., holidays, weekends).

- Elasticity matrices that reflect realistic interactions among a subset of products (e.g., complementary and substitute effects).
- Budget constraints and promotional limits imposed at the category or store level.

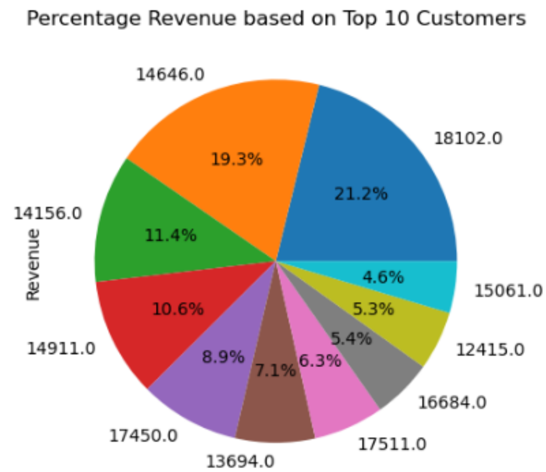
This methodology supports an integrated approach to personalized bundle generation and promotion optimization in the retail domain. By incorporating **cross elasticity**, **demand forecasting**, and **budget constraints**, the MINLP formulation captures complex interdependencies among products and customer purchasing behavior. The APOPT solver effectively handles the nonlinear and integer aspects of the problem, enabling the generation of **cost-effective and revenue-optimizing bundles** under realistic retail constraints.

5 Experiments and Results

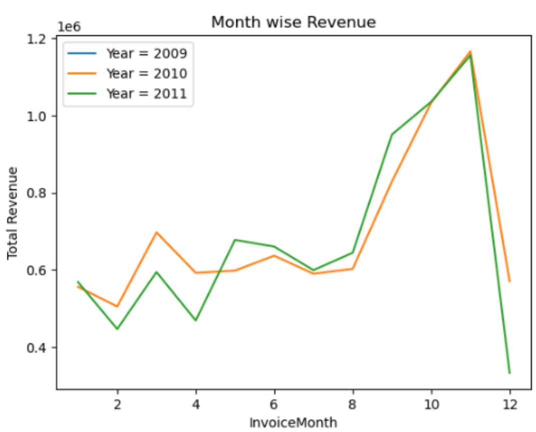
As mentioned in Section 4.7, we obtained the dataset and conducted extensive Exploratory Data Analysis (EDA) to understand the underlying patterns and distributions. Below are ten representative graphs generated during the EDA phase, which highlight trends in customer behavior, product reordering, frequency of purchases, and other key insights.



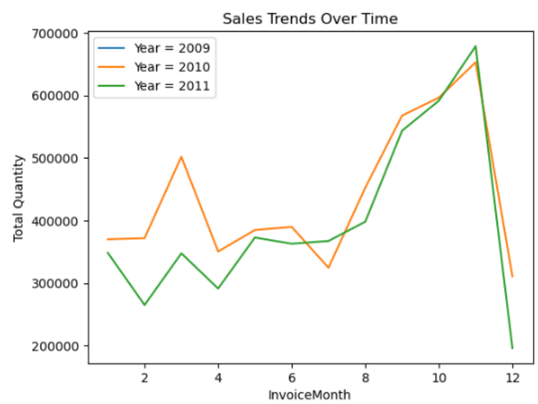
(a) Sales Over Weeks (2 Years)



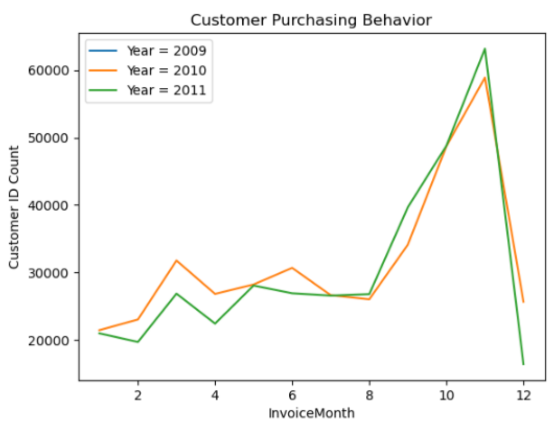
(b) Revenue by Top 10 Customers



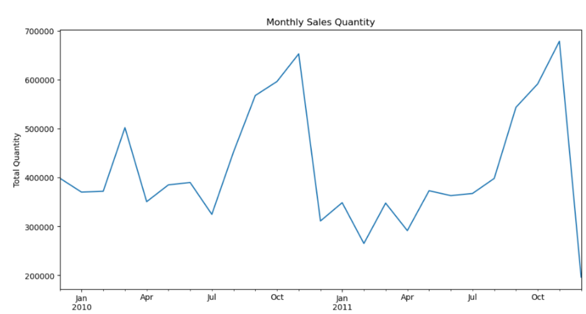
(c) Customer Share by Quantity



(d) Month-wise Revenue (2 Years)

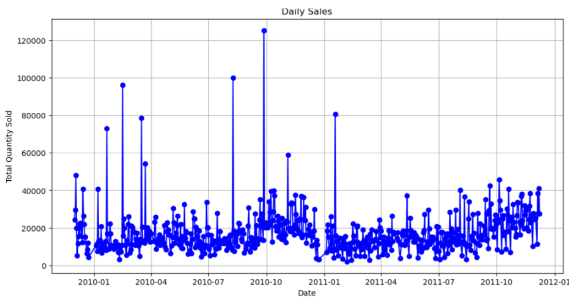


(e) Month-wise Sales Quantity

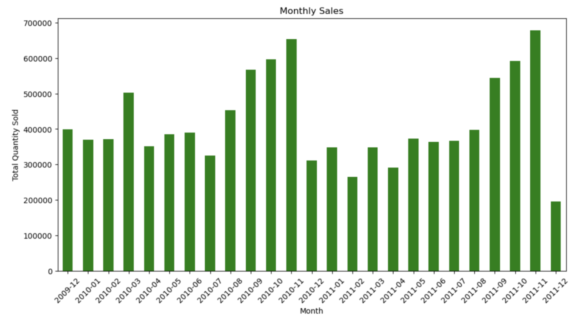


(f) Customers per Month

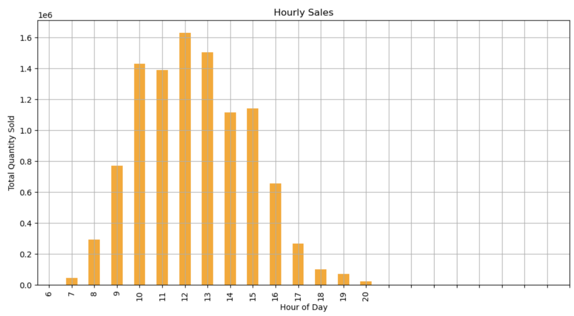
Figure 4: Retail Sales Insights – Trends by Weeks, Customers, and Monthly Revenue



(a) Daily Sales Trends



(b) Monthly Sales (Quantities)



(c) Hourly Sales (Quantities)

Product Name	Actual Sales	Forecasted Sales	Own Price Elasticity
Whole Wheat Loaf	52	43	-1.45
Organic Eggs	98	86	-1.30
Oatmeal Packs	86	92	-1.15
Cheddar Blocks	47	49	-0.98
Granola Boxes	52	55	-1.20
Salted Butter	73	86	-1.40
Greek Yogurts	34	63	-1.10
Almond Milks	37	48	-1.25
Orange Juices	62	49	-1.35
Cornflakes	79	92	-1.18
Soy Milks	65	70	-1.22
Vanilla Yogurts	73	78	-1.05
Ghee Tubs	66	60	-1.00
Mixed Vegetables	79	72	-1.50
Fresh Berries	56	55	-1.28
Free-range Eggs	87	93	-1.10
Seasonal Fruits	89	63	-1.42
Tropical Juice	83	90	-1.33
Skimmed Milk	76	81	-1.26

(d) Price Elasticities

Figure 5: Retail Sales Insights – Daily, Monthly, and Hourly Patterns

Product	Whole Wheat Loaf	Organic Eggs	Oatmeal Packs	Cheddar Blocks	Salted Butter	Greek Yogurts	Almond Milks	Orange Juices	Cornflakes	Farm Veggies	Seasonal Fruits	Soy Milk	Protein Bars	Mixed Vegetables	Fresh Berries	Free-range Eggs	Seasonal Fruits	Tropical Juice	Skimmed Milk
Whole Wheat Loaf	**-1.45**	-0.23	0.15	0.05	0.08	0.06	-0.378	-1.09	-0.99	0.11	0.08	-1.09	0.09	-1.378	0.10	0.06	0.09	-2.99	0.11
Organic Eggs	-1.378	**-1.30**	0.15	0.10	0.18	0.11	-1.09	0.10	0.13	0.09	0.11	0.08	-1.09	0.10	0.09	0.11	0.13	-0.99	0.06
Oatmeal Packs	0.10	0.15	**-1.15**	-1.378	0.10	0.09	0.08	0.11	0.14	-1.09	0.06	-1.378	0.10	0.08	-1.09	0.10	-1.378	0.09	0.11
Cheddar Blocks	0.05	0.10	-1.378	**-0.98**	0.25	0.10	0.06	0.08	0.11	0.06	-1.09	0.08	0.09	0.10	0.15	-1.378	0.10	0.09	-1.09
Salted Butter	0.08	0.18	0.10	0.25	**-1.40**	0.30	0.15	0.20	0.17	0.11	0.10	-1.378	0.14	0.13	0.20	0.18	0.15	-1.378	0.11
Greek Yogurts	0.06	0.11	0.09	0.10	0.30	**-1.10**	0.22	0.14	-1.378	0.09	0.10	0.11	0.10	-1.378	0.10	0.13	0.14	0.10	0.09
Almond Milks	-0.378	-1.09	0.08	0.06	0.15	0.22	**-1.25**	0.18	0.10	-1.378	0.10	0.09	-1.09	0.11	0.10	0.09	0.10	0.11	0.08
Orange Juices	-1.09	0.10	0.11	0.08	0.20	0.14	0.18	**-1.35**	0.13	0.10	0.09	0.11	0.13	0.14	0.11	-1.378	0.15	0.10	0.08
Cornflakes	0.09	0.13	0.14	0.11	0.17	-1.378	0.10	0.13	**-1.18**	0.10	0.09	0.08	0.09	0.08	0.10	0.09	0.10	-1.378	0.11
Farm Veggies	0.11	0.09	-1.09	0.06	0.11	0.09	-1.378	0.10	0.10	**-1.50**	0.15	0.08	-1.09	0.11	0.09	0.08	0.10	-1.09	0.10
Seasonal Fruits	0.08	0.11	0.06	-1.09	0.10	0.10	0.10	0.09	0.09	0.15	**-1.38**	0.09	0.10	0.09	-1.09	0.11	-1.378	0.08	0.09
Soy Milk	-1.09	0.08	-1.378	0.08	-1.378	0.11	0.09	0.11	0.08	0.08	0.09	**-1.12**	0.10	0.09	0.08	0.10	-1.378	0.11	0.09
Protein Bars	0.09	-1.09	0.10	0.09	0.14	0.10	-1.09	0.13	0.09	-1.09	0.10	0.10	**-1.20**	0.13	0.11	0.10	-1.378	0.11	0.08
Mixed Vegetables	0.13	0.10	0.08	0.10	0.13	-1.378	0.11	0.14	0.08	0.11	0.09	0.09	0.13	**-1.22**	0.09	0.11	-1.378	0.08	0.10
Fresh Berries	0.10	0.09	-1.09	0.15	0.20	0.10	0.10	0.11	0.10	0.09	-1.09	0.08	0.11	0.09	**-1.05**	-1.378	0.13	0.08	0.10
Free-range Eggs	0.06	0.11	0.10	-1.378	0.18	0.13	0.09	-1.378	0.09	0.08	0.11	0.10	-1.378	0.11	-1.378	**-1.40**	0.14	0.11	0.09
Seasonal Fruits	0.09	0.13	-1.378	0.10	0.15	0.14	0.10	0.15	0.10	0.10	-1.378	0.11	-1.378	-1.378	0.13	0.14	**-1.10**	0.10	0.11
Tropical Juice	0.08	0.10	-1.09	0.09	-1.378	0.13	0.11	0.10	-1.378	-1.09	0.08	0.11	0.11	0.09	0.08	0.11	0.10	**-1.20**	-1.09
Skimmed Milk	-1.09	0.09	0.08	-1.09	0.11	0.09	0.08	0.09	0.11	0.10	0.09	0.09	0.08	0.10	0.09	-1.378	0.11	-1.09	**-1.10**

Figure 6: Cross Elasticities of Products with Price Elasticity on Diagonal

Based on the behavioral patterns observed, we performed customer segmentation using clustering techniques. This allowed us to group customers into four distinct segments based on their ordering habits, product preferences, and frequency of reorders.

5.1 Results

we implemented the methodology discussed in Section 4.6 separately on each of these customer segments to better capture segment-specific behaviors and improve prediction performance.

Below, we present the results obtained for one customer segment:

Product Name	Selected	Discount %	Revenue	Margin
Whole Wheat Loaf	Selected	22.04%	$689.83	$606.60
Bundle of 4 Organic Eggs + 4 Oatmeal Packs	Selected	44.04%	$753.77	$417.01
Bundle of 2 Cheddar Blocks + 2 Granola Boxes	Not Selected	0.0%	$118.23	$88.67
Bundle of 3 Salted Butter + 4 Greek Yogurts	Selected	29.98%	$694.04	$446.22
Bundle of 4 Herb Butter + 1 Baguette	Not Selected	0.0%	$71.64	$53.73
Bundle of 5 Orange Juices + 5 Almond Milks	Selected	38.77%	$667.81	$395.14
Bundle of 3 Cornflakes + 2 Soy Milks	Selected	20.61%	$623.26	$427.00
Bundle of 3 Vanilla Yogurts + 3 Ghee Tubs	Selected	22.29%	$376.49	$255.36
Bundle of 5 Mixed Vegetables + 2 Fresh Berries	Selected	56.18%	$489.86	$210.40
Bundle of 1 Garlic Butter + 1 Free-range Eggs	Selected	54.39%	$779.50	$352.21
Bundle of 2 Seasonal Fruits + 1 Salted Butter	Selected	59.58%	$670.26	$255.69
Salted Butter	Selected	25.72%	$1007.21	$687.77
Multigrain Cereal	Selected	30.37%	$945.92	$590.87
Cottage Cheese	Not Selected	0.0%	$285.37	$174.88
Brown Eggs	Selected	19.57%	$1563.42	$1279.32
Seasonal Fruits	Selected	23.01%	$1360.35	$398.48
Tropical Juice	Selected	25.15%	$1087.35	$829.38
Skimmed Milk	Selected	22.31%	$587.34	$290.91
Fresh Vegetables	Selected	29.04%	$1728.79	$1537.88
Fruit Yogurt Cups	Selected	9.9%	$831.27	$633.88

Summary Metrics

- **Total Selected Items:** 17
- **Total Revenue:** $15331.68
- **Total Margin:** $9931.39

Figure 7: Model results for one customer segment

The segment-specific modeling approach helped uncover unique trends and model behaviors, which would have been diluted in a combined analysis across all customers.

5.2 Related Codes :

Here you can find the related codes and files LINK

6 Tools and Technologies

- **Programming Languages:** Python, R
- **Libraries & Frameworks:** NumPy, Pandas, Scikit-learn, XGBoost, TensorFlow, PyTorch, Matplotlib, Seaborn, Plotly, GEKKO
- **Optimization Solvers:** GEKKO [1], CPLEX [2]
- **Data Visualization:** Plotly, Matplotlib, Seaborn
- **IDE & Environments:** Jupyter Notebook, VS Code, Google Colab, Command Line Interface (CLI)
- **Documentation & Reporting:** LaTeX, Microsoft Word
- **Operating Systems:** macOS

7 Project Plan and Timeline

Month	**Milestone**
1-2	Data Collection, Manipulation, and Preprocessing
3-4	Literature Survey (Retail domain, ML/DL techniques)
5	Exploration of Retail Data and Initial ML/DL Tasks
6-7	Topic-Specific Literature Survey (MINLP and MILP)
8	Methodology Design and Python Implementation
9	Result Analysis and Evaluation
10	Final Report Writing review

8 Challenges and Future Work

8.1 Challenges

One of the primary challenges in this project was **handling the complexity of retail data**, which involved a vast number of products and a highly diverse customer base. The data posed significant difficulties in terms of preprocessing, integration, and computational handling. Extracting meaningful insights from such a large and intricate dataset required robust data engineering and **careful feature design**.

Another major challenge was **customer segmentation**. Choosing the appropriate **segmentation method** and the **right number of clusters** was critical, as it directly influenced downstream decisions like bundle creation and pricing. Moreover, selecting the correct **evaluation metrics** to validate the quality and stability of these clusters, such as silhouette score, added an additional layer of complexity and experimentation.

A further complication arose in the **calculation of cross-elasticities** between bundled items. Understanding and quantifying how the **demand for one product** is affected by the **pricing or availability of another** required detailed modeling and was **computationally intensive**. This process was crucial for designing **effective product bundles** and optimizing **discount strategies**.

Additionally, aligning business-level constraints such as **total promotional budget** and desired revenue at the **customer segment level** proved to be a **complex optimization task**. It involved balancing **business goals** with **customer preferences** while ensuring fairness and profitability across different segments.

8.2 Future Work

Future development of this project will focus on **improving scalability**. While the current framework performs well on medium-sized datasets, scaling it for enterprise-scale data will require **optimization** and possibly the use of **distributed systems** to handle high-volume transactions and customer interactions in real-time.

Another area of future work is the **generalization** of the proposed methodology to **different types of retail datasets**. By validating the approach on datasets from varied domains such as **electronics**, **apparel**, and **grocery**, we aim to assess its **robustness and adaptability** across contexts.

To enhance usability, we plan to develop a user-friendly interface that allows business users to input key parameters such as budget limits, margin targets, and segment preferences. The system will then generate optimized product bundles and associated **discount percentages** based on the provided inputs.

Incorporating external market signals such as competitor pricing, economic indicators, and marketing campaign calendars into the modeling framework is another future direction. These factors can further refine demand estimation and improve the relevance of the recommendations.

Finally, we aim to build an **automated decision support system** that not only suggests bundles and pricing strategies but also evaluates their post-implementation effectiveness. This will facilitate continuous learning and improvement in promotional planning and customer engagement.

References

[1] Beal, L. T., & Hedengren, J. D. (2020). *GEKKO Optimization Suite*. Available at: `https://github.com/BYU-PRISM/GEKKO`

[2] IBM. (2023). *IBM ILOG CPLEX Optimization Studio*. Available at: `https://www.ibm.com/products/ilog-cplex-optimization-studio`

[3] H. Sun and X. Li, "A Survey on Bundle Recommendation: Methods, Challenges, and Applications," *IEEE Access*, vol. 10, pp. 46827–46847, 2022.

[4] Z. Wei and F. Liu, "Towards Personalized Bundle Creative Generation," *Proceedings of the 30th International Conference on Computational Linguistics*, pp. 1329–1337, 2021.

[5] H. Sun and X. Yang, "Revisiting Bundle Recommendation: Datasets, Tasks, and Challenges," *ACM Computing Surveys*, vol. 54, no. 8, pp. 1–25, 2022.

[6] Y. Zhu and K. Harrington, "Bundle Recommendation in E-commerce: The Bundle Recommendation Problem (BRP) and Its Applications," *Proceedings of the 41st International Conference on Information Systems*, 2020.

[7] R. Pathak and A. Gupta, "Generating and Personalizing Bundle Recommendations Using Matrix Factorization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 12, pp. 5171–5182, 2021.

[8] S. Liu and Y. Fu, "Modeling Buying Motives for Personalized Product Bundling," *Journal of Retailing and Consumer Services*, vol. 56, pp. 102162, 2020.

[9] M. Beheshtian Ardakani and M. Fathian, "A Novel Model for Product Bundling and Direct Marketing in E-commerce," *International Journal of Data Analysis Techniques and Strategies*, vol. 8, no. 1, pp. 45–62, 2016.

[10] M. Ettl and S. Harsha, "A Data-Driven Approach to Personalized Bundle Optimization and Promotion Planning," *Proceedings of the 2020 IEEE International Conference on Data Mining*, pp. 184–193, 2020.

[11] Uma Maheswari G, Srividhya Sethuraman, and Sharadha Ramanan, *Optimizing Retail Promotions with Mixed-Integer Nonlinear Programming and Cross-Elasticity Effects*, International Conference on Optimization and Learning (OLA2025).