Store-Specific Assortments in the Presence of Product Constraints

Mert Çetin
Rotterdam School of Management, Erasmus University, cetin@rsm.nl

Victor Martínez-de-Albéniz
IESE Business School, University of Navarra, valbeniz@iese.edu

Abstract. When allocating products to brick-and-mortar stores, retailers face product availability constraints, requiring them to balance product offerings across locations. While the assortment planning literature is extensive, most optimization models focus on a *single* store, overlooking this balancing challenge. We formulate a store-specific assortment optimization problem for a network under limited product availability. We show that the problem is NP-complete and develop a tractable continuous relaxation under a multinomial logit demand, using mixed quadratic conic programming, which provides an upper bound. Our analysis further allows us to build solution algorithms, for which we obtain performance guarantees. Through numerical experiments, we show that our algorithms achieve near-optimal performance. We subsequently validate our model using real-world data from a large multinational apparel retailer. We estimate joint product-store attractiveness parameters, and find that revenues can potentially improve by up to 30% when compared to current policies.

Key words: assortment planning, retail, product allocation, MNL-choice model, brick-and-mortar

Submitted: Aug. 1, 2025

1. Introduction

Product allocation is a crucial process related to inventory management, in which the supply of products must be matched to demand at points-of-sales. Besides accounting for the possibility of substitution across products within the assortment (Kök et al. 2008), retailers face numerous operational constraints such as capacity constraints or product-specific fixed costs (Rusmevichientong et al. 2010, Feldman and Topaloglu 2015b, Kunnumkal and Martínez-de Albéniz 2019). In the assortment planning literature, most studies focus on a *single* store. In practice, however, large apparel retailers must allocate products across multiple stores, each with unique characteristics. Considering the seasonal nature of fashion retail and the importance of product variety (Caro and Martínez-de Albéniz 2015), it is unlikely that retailers possess sufficient means to send every product to each of their stores. Data from a large fashion apparel retailer reveals that 28%

of their products face supply constraints, preventing the allocation of a complete set of products (i.e., one item from each available size and color) to every store. Furthermore, our partner retailer mandates sending a minimum of three sets once a product is allocated, resulting in 56% of products being constrained for distribution across stores. As such, the importance of the allocation decision is substantial, with significant potential to enhance retailer revenues if the decision process can be improved.

Given this restricted product availability, retailers must meticulously assess the demand potential for each product in various stores, as the sales potential can vary significantly between locations. Surprisingly, during our interviews with the retailer, we discovered that decision-makers do not fully account for this variation when comparing product performance across stores and when making allocation decisions. Instead, they adhere to a rule-book approach, basing their decisions solely on the expected performance of products and the market share of stores, independently. Our experiences with other large retailers suggest that this type of procedure is common in the industry.

In this paper, we present an assortment optimization model for an *entire network* of stores, incorporating heterogeneous demand functions across stores. This approach allows us to capture the combined attractiveness of products and stores within the network. Our objective is to maximize the expected revenue of the store network, subject to product supply constraints – which vary in availability for each product – and store capacity limitations. The novel aspect of our optimization model lies in the product constraints, which couple different stores together, in contrast to conventional models that treat revenues as independent across stores. We allow customer preferences to vary not only across products within a given store but also across different stores for the same product. Additionally, we can account for potential differences in visitor volumes across stores.

First, we demonstrate that the resulting optimization problem is NP-complete, even for MNL demands, on which we focus. The hardness result is obtained by reducing the problem to an instance of the well-known partition problem – considering two stores and one product with equal margins in both stores. To address this computational challenge, we introduce a tractable continuous relaxation of the assortment optimization problem and formulate it as a conic optimization problem. This formulation enables us to leverage commercial solvers to obtain an upper bound for the problem. Theoretically, we prove that the projection of this fractional solution onto the integer space is at least $1/m$ times the optimal integer solution, where m represents the total number of stores in the network. We complement this general bound with a parametric performance guarantee that shows the gap can be very small when product attractiveness is small compared to that of the outside option.

Next, we introduce several heuristic solution algorithms. First, we present an algorithm that identifies a local maximum of the continuous relaxation. This fractional solution is then projected onto the integer space to obtain an approximate solution for the integer problem. In addition, we develop an alternative algorithm that leverages optimality properties to achieve comparable performance to the integer projection method in terms of optimality gaps, while significantly reducing computational time. Furthermore, we conduct an extensive computational study to evaluate the effectiveness of these algorithms across a range of demand structures.

Moreover, we validate our models by applying our methodologies to real data obtained from our retailer partner. We estimate a multinomial logit (MNL) demand function using sales, inventory, and footfall data, which allows us to estimate the sales potential of each product in each store. To circumvent the selection bias created by the product allocation decision made by the retailer (no sales observed if a product is never allocated to a store), we apply a control function approach leveraging the retailer's allocation guidelines as instruments in our estimation. We obtain the revenue information from the sales data. Subsequently, we execute our algorithms using parameters derived from the data. We find that the retailer could have potentially increased its revenues by up to 30% by employing our proposed algorithms. This insight highlights why taking the network into consideration is important: traditional heuristic methods, such as revenue-ordered assortments (Talluri and Van Ryzin 2004) or popular assortments (Ryzin and Mahajan 1999), overlook the interdependence of assortments across different stores.

Our paper makes several new contributions to the broad space of assortment planning. First, we present a unified assortment optimization model that considers the varying performances of products across different stores while being constrained by product volumes, thereby alleviating the general assumption of store independence. Interviews with the managers at our partner retailer suggest that our approach can provide them with a useful framework to support allocation. Second, we develop easy-to-implement heuristic algorithms to efficiently generate near-optimal solutions to the problem. These algorithms are built upon simple yet effective principles, making both our optimization methodologies and heuristics easily applicable; by introducing additional constraints or modifying the objective function, our approach can be easily adapted to different retail settings. Lastly, we validate our analysis using real-world data obtained from our partner retailer. Through our empirical analysis, we demonstrate the monetary benefits of improved product-store allocation strategies.

The rest of the paper is organized as follows. Section 2 provides a concise review of the relevant literature. Problem formulation is given in §3. The analytical results are provided in §4, followed by the computational experiments in §5. After presenting the application to a fashion retail case in §6, we conclude in §7. Mathematical proofs and formulations are included in Appendix A, and complementary analyses to the empirical study are presented in Appendix B.

2. Literature Review

Our work is related to two main streams of literature. First, it falls within the rich literature on assortment planning, a brief overview of which is presented in this section. For a more in-depth review of the literature on assortment planning, we refer to Kök et al. (2008). Anderson et al. (1992) and Train (2009) provide textbook introductions to discrete choice models. Second, we give an introduction to the literature on allocation problems, as we study a similar problem for the product-store context.

2.1. Assortment planning

Assortment optimization involves selecting a subset of products from a larger set to maximize expected revenue, where customer demand is typically modeled using discrete choice models. A foundational result by Talluri and Van Ryzin (2004) shows that under the MNL model, the optimal assortment is a revenue-ordered set – a collection of products with the highest margins up to a certain threshold.

Revenue-ordered assortments are not guaranteed to be optimal once we move beyond the unconstrained MNL setting. Rusmevichientong et al. (2010) study a capacitated version of the problem where the assortment size is limited and show that the revenue-ordered structure is lost. Similarly, Sumida et al. (2021) consider MNL-based models with totally unimodular constraints and show that the problem can be reformulated as a linear program, in which revenue-ordered assortments are no longer optimal. This enables tractable optimization in a wide range of practical assortment and pricing applications. Désir et al. (2022) reformulate the nonlinear MNL revenue objective into a submodular set function, allowing for approximation via greedy algorithms and LP rounding techniques.

When models move beyond the assumptions of basic MNL choice model, computational complexity increases. One such example is the nested MNL, where customers first choose a subset of options, and then make a discrete choice among that subset, instead of a single choice among all options. Davis et al. (2014) show that nested MNL formulation is tractable under certain conditions regarding the choice parameters, but NP-hard in general. Further, they formulate a convex program whose optimal objective value corresponds to an upper bound for their nested MNL assortment optimization. Gallego and Topaloglu (2014) introduce space constraints to assortment optimization under nested MNL type demand assumption, where the problem is NP-hard, and offer a well-performing approximation with performance guarantees. Similarly, Feldman and Topaloglu (2015b) consider capacity constraints while optimizing the assortment under nested MNL, and demonstrate that solving the continuous relaxation of an individual problem provides an upper bound.

Another body of work on MNL-type assortment optimization considers mixtures of MNL, i.e., mixed MNL, to overcome the limitations of assuming homogeneity in customer behavior. These models consider diversified preferences across individual decision-makers. For instance, Rusmevichientong et al. (2014) study the assortment optimization as a mixture of multinomial logit models to accommodate the customer heterogeneity, and show that the problem is NP-complete even only considering two different customer types. Feldman and Topaloglu (2015a) provide a Lagrangian relaxation to efficiently construct a tight upper bound for the mixed MNL model, and find that this relaxation corresponds to an assortment optimization under MNL model with fixed product costs. Moreover, Şen et al. (2018) formulate a capacitated assortment optimization under mixed MNL, and reformulate their model using second-order cone programming, which allows them to obtain the optimal solution through commercial solvers. We use a similar reformulation to solve our relaxation. The continuous relaxation is directly studied in Kunnumkal and Martínez-de Albéniz

(2019), who consider assortment optimization under MNL with product fixed costs, while also extending their results to mixed MNL formulation.

Among these works, Feldman and Topaloglu (2015a) is one of the most closely related to ours in terms of the general problem setting, as each consumer type in their model can be interpreted as representing the customer base of a different store in our framework. However, they focus on identifying a common assortment across all consumer types, whereas we allow for store-specific assortments – effectively personalizing the offering for each store. On the technical side, their model employs an equality constraint that penalizes deviations in product presence or absence across consumer types. In contrast, our model imposes an upper-bound capacity constraint on product availability. This difference leads to fundamentally distinct structural properties and solution approaches in the underlying optimization problems.

Barré et al. (2024), on the other hand, adopts an approach more directly aligned with ours, offering individualized online assortments to customers, comparable to offering store-specific assortments. However, a key technical distinction lies in the constraint structure: they impose a lower bound on product exposure, requiring each product to appear at least a specified number of times across customer views, a covering-type constraint. In contrast, we enforce an upper bound on how often a product can be offered, reflecting production-based capacity, a packing-type constraint. This fundamental difference in problem structures makes their results non-transferable to our setting.

Lastly, the formulation in Caro et al. (2014) may also appear somewhat analogous to ours if one interprets time periods as stores and restricts each product to be allocated to only one period. However, our setting differs in two important ways: first, product capacities can vary across items, and second, we do not assume a systematic decay in product attractiveness. Instead, we model heterogeneous product performance across stores, without any monotonic assumptions.

2.2. Allocation problems

Most of the allocation literature addresses how to statically assign products across locations under known or forecast demand. For example, Caro and Gallien (2010) propose a two-step approach: first forecasting demand, then allocating inventory using a mixed-integer program that accounts for substitution effects. Similarly, Ulu et al. (2012) and Chen et al. (2017) explore approaches to improve allocation effectiveness under demand uncertainty, but without modeling sequential learning. These works generally treat each allocation period independently, without updating preferences or parameters.

A smaller subset of the literature explicitly incorporates learning. Notably, Caro and Gallien (2007) develop a Bayesian learning model for dynamic assortment optimization, where the retailer updates its beliefs about customer preferences over time and adapts the assortment accordingly. Similarly, Gallien et al. (2015) present a phased allocation strategy, tested in a field experiment with a large retailer, showing that learning from early demand signals before committing inventory can significantly improve performance. More recently, Agrawal et al. (2019) propose a dynamic program that estimates MNL parameters iteratively, balancing learning and earning objectives.

3. Problem Formulation

We consider a retailer that is operating a network of m stores. A set of n different products are available at the distribution center. We assume a *finite* supply in the distribution center that *varies* for each product. We have to decide on the assortment of products in *each* of these stores considering the entirety of the store network. We let $J = \{1, 2, ..., n\}$ denote the set of products and $S = \{1, 2, ..., m\}$ the set of stores.

Consider a certain allocation policy, defined by the matrix $\boldsymbol{x}_{\cdot,s}$, where $x_{js} = 1$ when product j is included in the assortment of store s, and $x_{js} = 0$ otherwise. We assume customers make their purchase decision following an additive store-specific utility function, where the revenue yielding this decision can be represented by the function $f_s(\boldsymbol{x}_{\cdot,s})\, \forall s \in S$. Then,

$$Z(\boldsymbol{x}) = \sum_{s \in S} f_s(\boldsymbol{x}_{\cdot,s}) \tag{1}$$

denotes the expected revenue of offering the set of specific assortments $\boldsymbol{x}_{\cdot,s}\ \forall s \in S$. Without the loss of generality, we assume that $f_s(\cdot)$ is a function representing the dependent demand, i.e. a function characterizing the discrete choice of consumers while accounting for the substitutability of alternatives.

Furthermore, for each product j, we assume that the retailer has a total supply quantity at the distribution center, which can be allocated to at most q_j stores. In practice, retailers typically ship a fixed quantity to each store, implying a maximum number of stores that can be served. Additionally, we incorporate store capacity (k_s) constraints, which define the maximum assortment size that can be stocked in each store. For simplicity, we do not consider breadth vs. depth trade-offs, as explored in Martínez-de Albéniz and Kunnumkal (2022), in order to focus exclusively on the inter-store allocation decision.

Hence, the revenue optimization problem can be formulated as

$$\begin{aligned} (OPT) \qquad & \max \quad Z(\boldsymbol{x}) \\ & s.t. \quad \sum_{s \in S} x_{js} \leq q_j, \quad \forall j \in J \\ & \qquad \sum_{j \in J} x_{js} \leq k_s, \quad \forall s \in S \\ & \qquad x_{js} \in \{0, 1\}, \quad \forall j \in J, \quad \forall s \in S. \end{aligned} \tag{2}$$

Note that we focus on a single assortment decision for an entire season, maintaining it as a fixed assignment throughout the season. In other words, we do not consider post-allocation replenishment decisions, as studied in Caro and Gallien (2010), which are based on sales rates. Instead, we concentrate on the substitution effects that a product generates on other products within the store, in the spirit of Kök and Fisher (2007). Our methodology is guided by the operational dynamics of our partner company, where all new-season products are first gathered at a distribution center before before simultaneous allocation to all brick-and-mortar stores. Consequently, our formulation (2) represents a single-period assortment optimization model. Notably, we treat q_j as a fixed parameter, as our partner retailer follows a sequential decision-making process, where the manufacturing team determines production volumes based on aggregate demand, and the allocation team subsequently makes distribution decisions with a fixed product supply. However, in an approach that would explicitly optimize inventory levels, this parameter could be transformed into a decision variable.

Although we present a generic formulation, we later focus on a specific function representing $f_s(\cdot)$ in order to further investigate model properties, and to present an application using real data from our partner company. We consider a multinomial logit (MNL) type demand, i.e., $Z := Z^{MNL}$ in (2), where

$$Z^{MNL}(\boldsymbol{x}) = \sum_{s \in \mathcal{S}} \omega_s \frac{\sum_{j \in J} r_{js} v_{js} x_{js}}{v_{0s} + \sum_{j \in J} v_{js} x_{js}} \tag{3}$$

in which ω_s denote is the market potential of store s in the retailer's store network, r_{js} is the revenue generated with the sale of product j in store s, v_{js} is the preference weight associated with product j in store s, and v_{0s} is the preference weight associated with not making a purchase in store s. Note that we also consider a mixture of MNLs for our computational study in §5 as a more general formulation following Equation (2).

4. Analytical Results

4.1. Tractability

Existing literature suggests that the assortment optimization models following nonlinear dependent demand functions can be notoriously difficult to solve (Kök et al. 2008). Hence, we start our analysis by studying the computational hardness when the objective is given by (3), understanding that problem (2) is at least as hard. In fact, it is sufficient to consider the instance where there are two identical stores $S = \{1, 2\}$ with large store capacities, and n products with equal margins $v_{1s} = v_{2s} = ... = v_{ns}$ for all $s \in S$. We refer to this specific problem as *Two-Store Equal-Margin Assortment Customization*.

Theorem 1. (Tractability). *The* Two-Store Equal-Margin Assortment Customization *is NP-complete.*

Theorem 1 shows that, even when there are only two stores and all products have equal margins, the problem remains NP-complete. The proof follows a reduction of a decision-theoretic version of our problem to the well known NP-complete *Partition* problem.

Despite the computational hardness of the overall problem, we can make meaningful progress by analyzing its structure under certain assumptions. In particular, when the subproblem for each store can be solved exactly and efficiently – for example, under the standard MNL model (Talluri and Van Ryzin 2004), these solutions can be leveraged to construct feasible solutions for the global problem.

This motivates a simple heuristic that generates a set of candidate solutions by solving each stores subproblem independently, subject only to store-level capacity constraints. The resulting solutions are all feasible for the overall problem, as each product is allocated to at most one store. Among these, we can select the one with the highest objective value to serve as the heuristic solution. While straightforward, we show that the objective value achieved by this heuristic (R^H) is guaranteed to be within a factor of m of the global optimum (R^{OPT}), where m is the number of stores. Remarkably, this simple approach performs well compared to more sophisticated upper bounds we develop later, making it both a practical and interpretable benchmark.

Theorem 2. (A simple upper bound). $R^H \geq \frac{1}{m} \cdot R^{OPT}$.

While this bound is coarse, it motivates the development of tractable relaxations that preserve problem structure while enabling scalable solution methods. In the following subsection, we introduce a continuous relaxation approach that forms the basis of an efficient approximation algorithm.

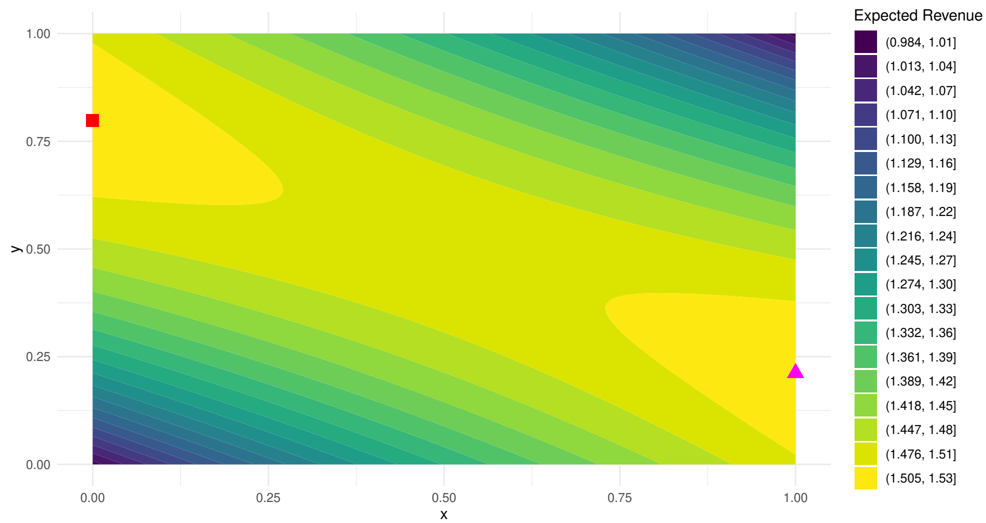
4.2. A Solution Approach Based on a Continuous Relaxation

The literature offers a variety of approaches to address the computational challenges associated with assortment optimization, depending on the underlying model assumptions. The most common approach for solving MNL-based assortment optimization problems is to formulate a mixed-integer linear programming (MILP) model, which can often be solved using commercial solvers (Bront et al. 2009). However, in our formulation, different assortments for different stores are coupled through product capacity constraints, significantly increasing the computational complexity of an MILP formulation (Méndez-Díaz et al. 2014). In particular, our bipartite formulation, spanning both stores and products, limits the applicability of the MILP approach, as large retailers typically manage hundreds of stores and thousands of products simultaneously. To address this challenge, we take an alternative approach and formulate a continuous relaxation of the problem, a method widely employed in the literature (Davis et al. 2014, Kunnumkal and Martínez-de Albéniz 2019). Thus, we present the continuous relaxation of (2):

$$
\begin{aligned}
(FRAC) \qquad \max \quad & Z(\boldsymbol{x}) \\
s.t. \quad & \sum_{s \in S} x_{js} \leq q_j, \quad \forall j \in J \\
& \sum_{j \in J} x_{js} \leq k_s, \quad \forall s \in S \\
& 0 \leq x_{js} \leq 1, \quad \forall j \in J, \quad \forall s \in S.
\end{aligned}
\tag{4}
$$

Our formulation (4) is a maximization problem under totally unimodular constraints. When the objective function is linear, the solution is trivial. Even when the objective function is nonlinear, a linear transformation might be applied to determine the objective value, which can then be used to construct the optimal solution (Davis et al. 2014, Sumida et al. 2021). However, without loss of generality, the objective function $Z(\boldsymbol{x})$ is not necessarily concave, meaning that this conjecture is limited to ensuring a linear feasible space. The results of Davis et al. (2014) regarding linear relaxation yielding integer solutions do not extend

Figure 1: Contour plot of total revenue for two products and two stores following MNL type demand function – 3, with parameters $r_1 = 1$, $r_2 = 2$, $v_{11} = v_{12} = v_{21} = v_{22} = v_{01} = v_{02} = 1$.



to our case due to the coupling of stores through product constraints. As a result, the possibility of interior optimal solutions cannot be ruled out. Even so, the structure of the problem provides a solid foundation for designing a practical solution method. When $f_s(\cdot)$ is continuously differentiable, its partial derivatives can guide the search for improvements. In particular, we can leverage the totally unimodular constraint structure to ensure feasibility while mimicking the gradient ascent principle. Again, this approach does not guarantee convergence to the global optimum, as it may get trapped in suboptimal local maxima. Figure 1 illustrates such a case, where following the direction of the derivative during partial allocation leads to convergence at a local maximum (triangle point), while the global maximum lies elsewhere (square point). Still, this method efficiently identifies locally optimal solutions and can capture interior optima by observing vanishing gradients near such points.

The strength of this approach lies in its simplicity, as it can be easily implemented and modified to accommodate various retailer-specific requirements. To this end, we introduce the Fractional Fill and Swap Algorithm (henceforth, FFS), with its corresponding pseudo-code provided in Appendix A.4. This algorithm begins with empty assortments for all stores and fractionally allocates products following the direction of the first-order derivative. It then computes the first-order derivative matrix for all possible product-store combinations and selects the largest derivative to initiate allocation by assigning a small fraction $0 < e \leq 1$ to the respective product-store pair. This process generates a new first-order derivative matrix, and the algorithm iterates until no further revenue improvements are possible.

Throughout the iterations, feasibility constraints are continuously checked to ensure that only valid allocations are considered. The algorithm enforces store and product capacity constraints while preventing the decision variable from exceeding one, thereby ensuring that no product is allocated beyond its maximum allowable quantity. Specifically, initially, product constraints are loose, making the algorithm initially 'Fill' the stores with partial products. Additionally, the algorithm maintains flexibility by allowing modifications to previously made allocations. When an item has exhausted its availability and the product capacity constraint is tight, the algorithm checks for possible swaps to optimize its distribution, hence a 'Swap' phase. This adjustment ensures that the allocation remains as profitable as possible.

Although the algorithm is computationally straightforward, the choice of step size e significantly influences both solution quality and computational efficiency. A smaller step size enhances convergence but increases computation time, whereas a larger step size accelerates execution at the risk of a suboptimal solution. Moreover, swap operations, particularly those involving reallocation of fully assigned products, substantially increase computation time. However, a near-local maximum can still be achieved by relying solely on fractional ascent steps. To address concerns regarding computational efficiency, we propose a simplified version of the algorithm, referred to as the Fractional Fill Algorithm (henceforth, FF). This alternative omits swap operations and focuses exclusively on fractional ascent, making it particularly useful for practitioners operating under time constraints who prioritize operational efficiency over absolute optimality. Moreover, when $e = 1$, we refer to this method as the Integer Fill Algorithm (henceforth, IF). We distinguish IF as a separate variant due to its significantly faster computation time compared to the fractional version (FF), which results from the larger step size and reflects the inherently discrete nature of the allocation.

Both FFS and FF yield solutions to the relaxed problem, which must then be projected onto the integer space. This projection requires more care than simply rounding fractional values, as naive rounding can lead to infeasible allocations. For example, consider an optimal fractional allocation $x^* = (0.6, 0.6, 0.6, 0.2)$ across four stores with a product capacity constraint $q_j = 2$. Although x^* satisfies the constraint, naive rounding to $(1, 1, 1, 0)$ violates it. Thus, we need a systematic projection method to ensure feasibility. Several integer projection strategies exist. One is random rounding (Raghavan and Tompson 1987); another is greedy rounding, which prioritizes higher-gradient assignments. We evaluate multiple options and adopt a tailored method: for each product j, we select the q_j stores with the highest fractional allocations in $\boldsymbol{x}^*$, forming the set $\mathcal{S}_j = \{s \mid \text{rank}(x_{js}) \leq q_j\}$. We assign these stores a value of 1 and set all others to 0.

4.3. Results under MNL Demand

In this subsection, we examine the properties of our model under an MNL-type objective. This structure provides two key advantages: it offers deeper insights into the thresholds for allocating products to stores and enables the development of a stronger, case-specific performance guarantee for our solution approach.

A conic reformulation. We begin by reformulating the problem to enhance tractability. Traditionally, the non-linearity of MNL-based objectives is addressed through bilinear mixed-integer programming (Bront et al. 2009, Méndez-Díaz et al. 2014). Although our focus is on the continuous relaxation rather than the integer formulation, this bilinearization technique can still be applied. However, it does not fully overcome the computational challenges arising from the bipartite structure. To address this, we follow Şen et al. (2018) and introduce a conic reformulation of problem (4). The final mixed-integer conic formulation is presented in (5)–(14).

$$(CONIC) \qquad \min \quad \sum_{s\in S}\omega_s v_{0s}\bar{r}_s y_s + \sum_{s\in S}\omega_s \sum_{j\in J} v_{js}(\bar{r}_s - r_{js}) z_{js} \tag{5}$$

$$s.t. \quad \sum_{s\in S} x_{js} \leq q_j, \quad \forall j \in J \tag{6}$$

$$\sum_{j\in J} x_{js} \leq k_s, \quad \forall s \in S \tag{7}$$

$$V_s = v_{0s} + \sum_{j\in J} v_{js} x_{js}, \quad \forall s \in S \tag{8}$$

$$z_{js} \geq y_s x_{js}, \quad \forall j \in J, \quad \forall s \in S \tag{9}$$

$$y_s V_s \geq 1, \quad \forall s \in S \tag{10}$$

$$v_{0s} y_s + \sum_{j\in J} v_{js} z_{js} \geq 1, \forall s \in S \tag{11}$$

$$z_{js} \geq 0, \quad \forall j \in J, \quad \forall s \in S \tag{12}$$

$$y_s \geq 0, \quad \forall s \in S \tag{13}$$

$$0 \leq x_{js} \leq 1, \quad \forall j \in J, \quad \forall s \in S. \tag{14}$$

where $y_s = 1/(v_{0s} + \sum_{j\in J} v_{js} x_{js})$, and $z_{js} = x_{js} y_s$ are new decision variables – details on introduction of new variables and parameters are presented in Appendix A.5.

A key difference between our reformulation and that of Şen et al. (2018) lies in the treatment of second-order cone constraints. Our final conic formulation excludes them. In MILP settings, these constraints reduce to equalities when $x_{js} \in \{0, 1\}$, but in the relaxed case $0 \leq x_{js} \leq 1$, they hold only as inequalities, thus becoming redundant.

These constraints primarily define auxiliary variables z_{js} but do not sufficiently capture interactions between decision variables in the relaxed model. To address this, we retain the mixed-quadratic constraints in (9). While commercial solvers often convert such constraints into second-order cones internally, explicitly including them in an MILP with binary variables can mislead the solver and increase computation time due to unnecessary linear algebra operations. This conic formulation enables the use of commercial solvers to compute the global optimum of the relaxation, providing a valid upper bound for both FFS and FF. Beyond improving the formulation, the well-defined structure of the MNL model enables us to strengthen performance guarantees through a parametric upper bound.

In particular, we compare the optimal fractional allocation of a product across the store network to any of its integer projections. Let $\boldsymbol{x}^*$ denote the fractional optimal solution obtained via the conic formulation, and

$z^{\text{FRAC}} = f(\boldsymbol{x}^*)$ the corresponding objective value, z^{RND} the value of a feasible integer solution obtained by rounding $\boldsymbol{x}^*$, and z^{OPT} the value of the true optimal integer solution. By construction, these values satisfy:

$$z^{\text{FRAC}} \geq z^{\text{OPT}} \geq z^{\text{RND}}.$$

The worst-case scenario for our rounding approach occurs when all rounding directions lead to the same integer solution – that is, when no improvement is possible through alternative rounding choices. In this case, although the resulting solution remains feasible, it is maximally distant from the fractional optimum, representing the poorest performance achievable through rounding.

A parametric performance guarantee. To quantify the quality of this rounded integer solution relative to the fractional one, we define the multiplicative degradation factor:

$$\Delta := \frac{z^{\text{FRAC}}}{z^{\text{RND}}}.$$

This implies a relative loss of $\Delta - 1$. Since $z^{\text{RND}} \leq z^{\text{OPT}}$, it follows that

$$\frac{z^{\text{FRAC}}}{z^{\text{OPT}}} \leq \Delta.$$

Therefore, the bound on Δ characterizes both the performance of a specific rounded solution and an upper bound on the integrality gap between the fractional and integer optima.

To make this bound interpretable in terms of problem parameters, we derive a parametric expression for Δ by characterizing the fractional optimal solution and substituting it into the objective function. We first consider the case of rounding the fractional allocations of a single product, i.e., Δ_j. Note that the rest of products may remain fractional at this point; their value does not affect the performance bound. For each product j and store s, we define

$$\beta_{js} := \frac{v_{0s} + \sum_{j \in J} v_{js}}{v_{js}},$$

where the numerator reflects the total attractiveness of store s, and the denominator captures the attractiveness of product j in that store. We formalize this parametric upper bound in Theorem 3.

Theorem 3. (Parametric upper bound). *For any product j,*

$$\Delta_j \leq \frac{1}{q_j}\left(m + \sum_{s \in S} \beta_{js} - \frac{\left(\sum_{s \in S} \sqrt{\beta_{js}(\beta_{js}+1)}\right)^2}{q_j + \sum_{s \in S} \beta_{js}}\right).$$

This parametric upper bound depends on the ratios β_{js} and the product capacity q_j. Its value is driven by both the capacity and the relationship between product attractiveness and the total assortment attractiveness in each store. As q_j increases, the bound becomes tighter. This is intuitive: when $q_j = m$, the product can be allocated to all stores, and the fractional solution exactly matches its integer projection.

We formalize the behavior of this bound in two limiting cases shown in Corollaries 1 and 2.

Corollary 1 (Pessimistic upper bound on Δ_j)**.** *The pessimistic (worst-case) upper bound on Δ_j occurs when $q_j = 1$, yielding $\Delta_j \le m$. When $q_j = m$, the bound depends on the relative size of $\beta_{tot} := \sum_{s \in S} \beta_{js}$:*

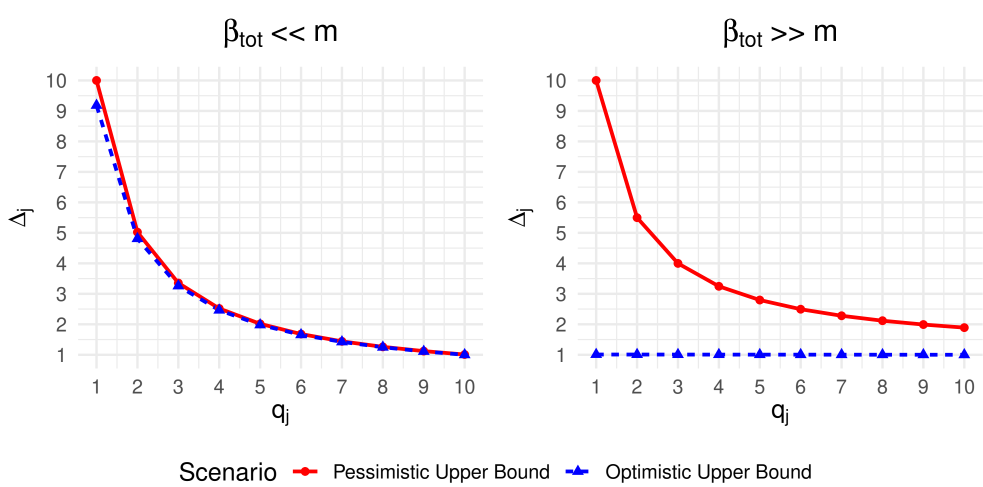
- *If $\beta_{tot} \gg m$, then $\Delta_j \to 2 - \frac{1}{m}$,*
- *If $\beta_{tot} \ll m$, then $\Delta_j \to 1$.*

Corollary 2 (Optimistic upper bound on Δ_j)**.** *The optimistic (best-case) bound on Δ_j occurs when $q_j = m$, in which case $\Delta_j = 1$. When $q_j = 1$, the bound satisfies:*

- *If $\beta_{tot} \gg m$, then $\Delta_j \to 1$,*
- *If $\beta_{tot} \ll m$, then $\Delta_j \to m$.*

The behavior of Δ_j depends strongly on the distribution of β_{js} across stores and its relationship to the product capacity q_j. When attractiveness is concentrated in a few stores and q_j is small, Δ_j approaches the pessimistic (worst-case) upper bound of m, effectively recovering the general bound in Theorem 2. Conversely, when relative attractiveness is more evenly distributed and the product capacity is larger, Δ_j approaches the optimistic (best-case), tending toward 1. We illustrate this behavior in Figure 2 using an example with $m = 10$, varying values of q_j, and different levels of B_{tot}.

Figure 2: Illustration of the pessimistic and optimistic upper bounds on Δ_j as a function of q_j, for $m = 10$ and fixed values of β_{tot} both greater and smaller than m.



A key insight is that, since $\beta_{js} \ge 1/v_{js}$, even the least attractive stores tend to have relatively large β_{js} values. This makes it likely that $\beta_{\text{tot}} \gg m$. Specifically, when the attractiveness of an individual product is small relative to the total assortment attractiveness, β_{js} becomes large, which helps keep the bound tight regardless of the specific margin values. Therefore, in most realistic settings – where assortments are substantial compared to individual product appeal, the range $1 \le \Delta_j \le 2 - \frac{1}{m}$ serves as a strong and reliable estimate for Δ_j.

We now extend the results from single-product rounding to the multi-product case. Let $\boldsymbol{x}^{(k)}$ denote the solution obtained after rounding the k-th fractional component of $\boldsymbol{x}^*$, with $\boldsymbol{x}^{(0)} := \boldsymbol{x}^*$. Suppose we sequentially round all fractional components. Then:

$$f(\boldsymbol{x}^{(0)}) \leq \Delta_1 \cdot f(\boldsymbol{x}^{(1)}),$$
$$f(\boldsymbol{x}^{(1)}) \leq \Delta_2 \cdot f(\boldsymbol{x}^{(2)}),$$
$$f(\boldsymbol{x}^{(2)}) \leq \Delta_3 \cdot f(\boldsymbol{x}^{(3)}),$$
$$\vdots$$
$$f(\boldsymbol{x}^{(k-1)}) \leq \Delta_k \cdot f(\boldsymbol{x}^{(k)}).$$

Let $\mathcal{J}_k$ denote the set of indices corresponding to fractional components in $\boldsymbol{x}^*$, and define $z^{\text{RND}} := f(\boldsymbol{x}^{(k)})$. Then the overall bound is:

$$\frac{z^{\text{FRAC}}}{z^{\text{RND}}} = \Delta \leq \prod_{j \in \mathcal{J}_k} \Delta_j.$$

This establishes a worst-case bound on the overall performance of the rounded integer solution, based on the individual rounding factors Δ_j. In the worst case – where each product incurs the maximum possible degradation – the overall bound could reach $m^{|\mathcal{J}_k|}$. However, in practice, we expect the bound to be significantly tighter. As discussed earlier, real-world instances typically satisfy $\beta_{\text{tot}} \gg m$, reflecting large assortments and relatively low individual product attractiveness. This suggests that $1 \leq \Delta \leq (2 - \frac{1}{m})^{|\mathcal{J}_k|}$ is a more realistic estimate.

The optimal policy is based on product thresholds. In addition to these theoretical insights, the MNL structure facilitates a closed-form representation of the first-order derivative required for implementing FFS, FF, and IF. Assuming $\omega_s = v_{0s} = 1$ for all $s \in S$, we express the gradient as follows:

$$W_{js} := \frac{\partial Z^{MNL}}{\partial x_{js}} = \left(\frac{v_{js}}{1 + \sum_{j \in \mathcal{J}} x_{js} v_{js}} \right) \left(r_{js} - \frac{\sum_{j \in \mathcal{J}} r_{js} x_{js} v_{js}}{1 + \sum_{j \in \mathcal{J}} x_{js} v_{js}} \right). \tag{15}$$

We can rewrite the gradient (15) as a function of the total assortment attractiveness per store (V_s) and expected revenue per store (Π_s) for clear notation as follows:

$$W_{js}(\Pi_s, V_s) = \frac{v_{js}}{V_s} \Big(r_{js} - \Pi_s \Big) \tag{16}$$

where, for every store s, Π_s is the revenue achieved with current allocation $\boldsymbol{x}_{\cdot s}$ and V_s the attractiveness:

$$V_s = 1 + \sum_{j \in \mathcal{J}} x_{js} v_{js} \text{ and } \quad \Pi_s = \frac{\sum_{j \in \mathcal{J}} r_{js} x_{js} v_{js}}{1 + \sum_{j \in \mathcal{J}} x_{js} v_{js}}. \tag{17}$$

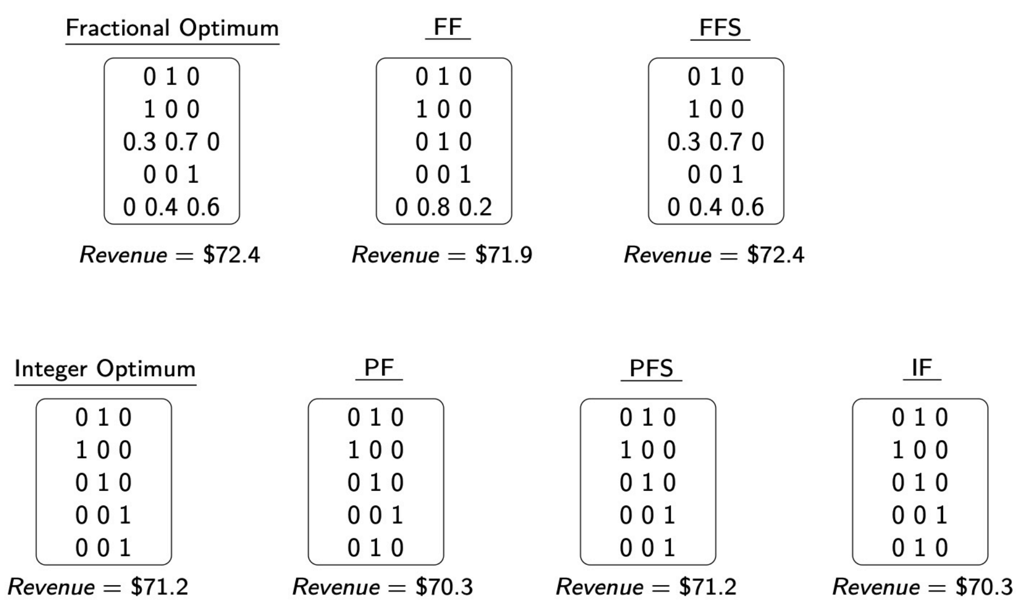
This unique representation of the first-order derivative function allows us to characterize the optimal solution for the integer optimization model (2).

Theorem 4. (Necessary optimality condition). *Under MNL demand, at a local optimum of* (4)*, there exist thresholds* W_j*,* $j = 1, \dots, n$*, such that* $x_{js}^* = 1$ *when* $s \in \{s \mid W_{js} > W_j\}$*, and* $x_{js}^* = 0$ *when* $s \in \{s \mid W_{js} < W_j\}$.

Theorem 4, thus provides a straightforward way to identify the optimal solution: one must calculate the gradient matrix $\boldsymbol{W} = (W_{js})$ of a given assignment matrix $\boldsymbol{x}$, and check the relationship between $\boldsymbol{W}$ and the thresholds for all $s \in S$, and $j \in J$, to confirm optimality.

For a clearer illustration of how our solution algorithms work, we present a numerical example in Figure 3. In this example, we consider an instance with three stores ($m = 3$) with sufficiently large capacities ($k_s \geq n$ for $s \in S = \{1, 2, 3\}$), and five products ($n = 5$), where each product can be allocated to only one store ($q_j = 1$ for $j \in J = \{1, ..., 5\}$). We generate random parameters following $r_{js} \sim U[1, 3]$, $v_{js} \sim U[0, 1]$, $v_{0s} \sim U[5, 10]$, and then run our algorithms. Under each title, we present the corresponding allocation matrix, where columns represent stores, rows represent products, and the expected revenue is displayed at the bottom. We present the optimal solution to (2), obtained by solving the MILP model (40) in Gurobi, as the Integer Optimum, and the solution to the continuous relaxation (4), via the conic formulation (5), as the Fractional Optimum, along with their corresponding revenues.

Figure 3: Numerical example for illustration.



We also report the performance of our approximation methods. FFS recovers the fractional optimum in this instance, although only local optimality is guaranteed in general. For the integer case, the projection of FF (henceforth, PF) and IF yield near-optimal solutions, while the projection of FFS (henceforth, PFS) recovers the integer optimum in this example. Both PF and IF exhibit an optimality gap of approximately 1.2%. A broader evaluation is presented in Section 5.

5. Computational Experiments

In this section, we present a set of computational experiments to test the performance of our algorithms.

5.1. Experimental Setup

We contemplate two different models representing the consumer choice for our computational study. First, we keep the MNL type consumer choice as shown in (3). The literature suggests that MNL function is an effective way of modeling discrete consumer choice as it is computationally efficient and relatively simple to implement (Kök et al. 2008). Second, we consider heterogeneity across consumers and use a mixed MNL demand. For this approach, we offer the following objective function Z^{MIX}:

$$Z^{MIX}(\boldsymbol{x}) := \sum_{s\in S}\omega_s \sum_{c\in\mathcal{C}}\gamma_c \frac{\sum_{j\in J} r_{js} v_{js}^c x_{js}}{v_{0s} + \sum_{j\in J} v_{js}^c x_{js}} \tag{18}$$

where we assume that a customer can be of type $c \in \mathcal{C}$ following the probability γ_c, and the respective set of product-store attractiveness parameters for consumer type c is $v_{js}^c\ \forall c \in \mathcal{C}, s \in S, j \in J$. This model allows us to understand the effect of consumer heterogeneity in our optimization context. Hence, we create scenarios where there are different numbers of consumer classes.

Considering these two different consumer choice models allows us to grasp the underlying mechanisms better, while testing for the computational power of the algorithms we propose. After establishing our base models for the numerical study, we generate a variety of problem instances using randomly drawn parameters. For each problem instance, we derive the revenues by sampling r_{js} from the uniform distribution over $[1,3]$, the store market shares α_s from $U[20,25]$, and we sample v_{0s} from $U[5,10]$. Moreover, we sample the attractiveness parameters v_{js} from $U[0,1]$ for instances following MNL (3). Also, we use the same uniform distribution $U[0,1]$ to sample for v_{js}^c regarding instances following mixed-MNL (18), and we sample the customer type arrival rate γ_c from a $U[0,1]$ probability distribution. Note that we choose not to include store capacity constraints in these computational experiments, as the novel aspect of our model lies in exploring the coupling of stores through product capacity constraints.

The optimization problems are solved with Gurobi 10.0.1 solver on a computer with an Apple M1 processor and 8 GB RAM operating on 64-bit macOS Ventura 13.5.2 through R API. We use the default settings of Gurobi except that we force the solver to take in non-convex expressions to accommodate the mixed quadratic constraints. We also set the time limit is set to 600 seconds. We complete all instance generation, algorithm completion, and optimization problems using R.

5.2. Computational Results

We consider two sets of instances varying in size. First, we fix the number of stores at $m = 2$ and the number of products at $n = 10$, with each product having unit capacity, i.e., $q_j = 1$ for all j. When $m = 2$ and $q_j = 1$, each product has three possible assignment options: store A, store B, or no store. This results in $(m+1)^n = 3^{10} = 59{,}049$ possible allocation scenarios, which can be exhaustively enumerated as an alternative to solving the integer program with commercial solvers.

For these instances (referred to as the $m = 2$ set), we compute all performance measures relative to the optimal integer solution. While feasible for this small setting, this approach becomes intractable for larger instances due to the exponential growth in possible assignments with increasing m.

Then, we calculate all the performance measures for the instances of this size (thence called as instances of $m = 2$), on the basis of the optimal solution for the integer formulation. Note that, this approach would not be practical under any realistically sized instance, since the required computation grows exponentially with the number of stores. We evaluate the performance of the fractional optimum, our previously proposed algorithms (FFS, FF, IF), their corresponding projected versions (PFS, PF), and two benchmark greedy heuristics commonly used in practice. The first is the *Popular Set* heuristic, which assigns products to stores based solely on the product-store attractiveness parameters v_{js}, ranking them in descending order and assigning to the top q_j stores. The second is the *Margin-Ordered* heuristic, which follows the same approach but ranks stores in descending order of the store-specific margins r_{js} for each product. These heuristics are fast and simple, making them widely used by practitioners, and thus serve as important baselines for comparison against our near-optimal methods.

For each scenario (combination of instance size and consumer choice model), we generate 200 random instances. For each instance, we record the computation time (in seconds) and the optimality gap. We then report, for each method, the average and maximum optimality gap, the average runtime, and the percentage of instances in which the optimal solution was found.

We evaluate each algorithm under both the MNL choice model (20) and a mixed MNL model (18) with five consumer types, i.e., $|\mathcal{C}| = 5$. The first set of results, corresponding to the $m = 2$ instances, is presented in Table 1.

Table 1: Performance comparison between MNL and Mixed-MNL models on 200 randomly generated instances with $m = 2$, $n = 10$, and $q_j = 1$.

Solution Method	**MNL**				**Mixed-MNL**			
	Time (s)	**Avg. Gap**	**Max Gap**	**Share Opt.**	**Time (s)**	**Avg. Gap**	**Max Gap**	**Share Opt.**
Fractional Optimum	0.02	-2.41	0	43%				
FF	0.20	-1.79	0	36%	0.45	-2.14	0	67%
PF	0.20	0.01	0.70	80%	0.45	0.65	9	85%
FFS	1.61	-1.94	0	43%	3.56	-2.49	0	67%
PFS	1.61	0	0.55	89%	3.56	0.67	8.78	87%
IF	<0.01	0.28	2.13	27%	<0.01	0.22	2.84	74%
Popular Set	<0.01	3.97	17.22	0%	<0.01	7.61	30.55	0%
Margin-Ordered	<0.01	5.17	25.73	0%	<0.01	4.56	15.81	0%

As expected, computational run times are very small for all instances with $m = 2$. We observe a negative optimality gap for fractional optimum, consistent with its role as an upper bound to the integer problem.

Similarly, FO and FFS also provide upper bounds, with a maximum optimality gap of zero in instances where the relaxation coincides with the integer optimum.

Our integer projection methods perform strongly: PF recovers the exact optimal solution in 80% (85%) of instances, and PFS in 89% (87%), for MNL (mixed-MNL) models. Incorporating swap operations into the local search yields only marginal improvements. The IF algorithm consistently outperforms the greedy dispatch heuristics in both solution quality and runtime. While it does not guarantee optimality, it efficiently delivers high-quality solutions. Overall, all of our proposed methods significantly outperform the greedy heuristics, highlighting the effectiveness of fast, well-structured approximation strategies in this setting.

Our initial choice of $m = 2$ enables accurate comparisons against the true optimum, but it does not reflect realistic retail environments. To address this, we generate a second set of instances using the same parameter distributions, but with $m = 10$, $n = 40$, and $q_j \sim U[1, 9]$. In this larger setting, solving (2) optimally becomes computationally infeasible. Instead, we evaluate performance relative to the solution of the continuous relaxation (4), obtained via its conic formulation, which provides a natural upper bound.

As before, we generate 200 instances for each choice model and report the same set of performance metrics, including optimality gaps and average runtime. The results for the $m = 10$ instances are presented in Table 2.

Table 2: Performance comparison between MNL and Mixed-MNL models on 200 randomly generated instances with $m = 10$, $n = 40$, and $q_j \sim U[1, 9]$.

Solution Method	**MNL**				**Mixed-MNL**			
	Time (s)	**Avg. Gap**	**Max Gap**	**Share Opt.**	**Time (s)**	**Avg. Gap**	**Max Gap**	**Share Opt.**
FF	0.67	0.08	0.10	45%	14.15	0.12	0.37	25%
PF	0.67	0.12	1.21	83%	14.15	0.12	0.42	12%
FFS	18.63	0.08	0.10	47%	198.71	0.03	0.18	32%
PFS	18.63	0.11	1.18	85%	198.71	0.08	0.36	17%
IF	0.01	0.12	0.39	21%	0.16	0.18	0.42	11%
Popular Set	<0.01	4.38	6.20	0%	<0.01	10.19	14.08	0%
Margin-Ordered	<0.01	4.92	10.49	0%	<0.01	4.67	6.81	0%

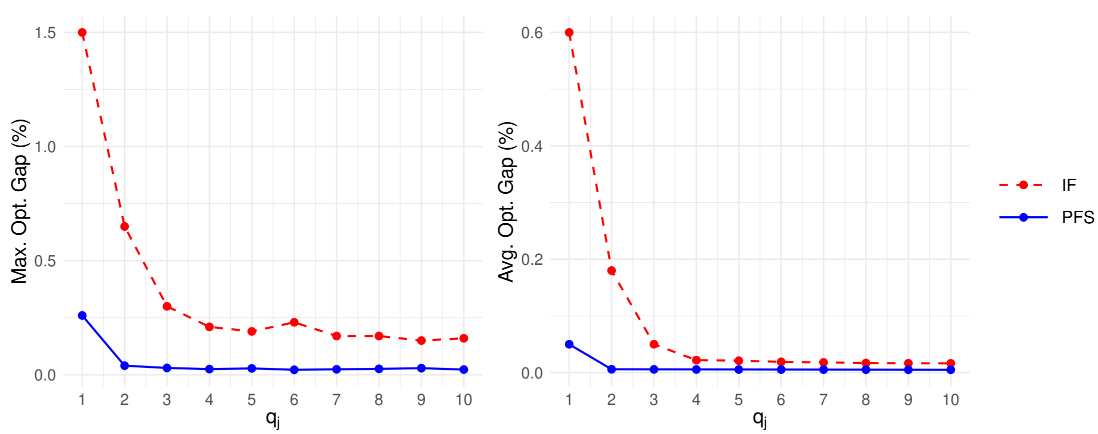
We observe that the average running times increase notably for FF and FFS, while the heuristic methods and IF remain computationally efficient. As expected, the greedy dispatch heuristics perform poorly in terms of optimality gap. In contrast, PF and PFS not only achieve low optimality gaps but also recover the optimal integer solution in majority of the instances. Similarly, IF demonstrates strong performance, with a maximum optimality gap of just 0.42% and an average gap as low as 0.15% across all problem versions.

Despite its high solution quality, IF recovers the exact benchmark solution only in some instances. This highlights that, while IF may not always find the true optimum, it provides a highly effective and scalable approximation for the product-store allocation problem. Notably, the performance of all proposed approximation methods improves significantly for larger instances, compared to the smaller $m = 2$ case.

This trend underscores an important distinction between instance sizes: larger-scale problems exhibit smaller optimality gaps across all approximation methods. This suggests that, in line with our modeling assumptions, larger retailers may benefit more from the proposed algorithms than smaller ones.

Finally, we conduct a sensitivity analysis on q_j to explore the effect of product availability on the performance of the approximation algorithms. We create scenarios where there are 10 stores ($m = 10$) and 40 products ($n = 40$) under MNL type demand assumption. Then, for each $q_j \in \{1, ..., 10\}$, we run 100 instances, and calculate the average and maximum optimality gaps comparing to the upper bound provided by the continuous relaxation. We present the results regarding the IF, and PFS. Particularly, we show the variation in the maximum and average optimality gaps in Figure 4.

Figure 4: Variation of maximum and average optimality gaps with respect to the capacity constraint.



We observe that when q_j, i.e. the product supply increase, both approximation methods perform better. This is due to the fact that the interdependence of assortments across stores decrease with higher supply. In fact, our problem reduces to sum of unbounded single store assortment optimization problems, when there is no availability constraints, i.e, $q_j = 10$, for $j \in J$. PFS always provides a better solution than the IF, yet the gap between them also decreases as product supply increases. Indeed, we observe that the IF performs only slightly worse comparing to the integer projection method, yet requires only a fraction of the computation time.

Our entire computational study demonstrates a significant potential for the IF, especially for the retail practitioners, because it serves as an easily implementable, time-efficient approach that provides near-optimal solutions without any commercial solver license requirements. Hence, we present an application of the IF to fashion retail context using data from a large retailer in Section 6.

6. Application
6.1. A Fashion Retail Case

In this section, we apply our optimization model to the allocation process of a large fashion retailer. Our partner is a multinational apparel company with operations in over 100 countries and several billion euros in annual revenue. We focus here on their operations in one country, where they have a network of more than 300 brick-and-mortar stores. Using the open source database of the tax agency in the respective country, we obtain average income levels in postal code areas where these stores are located as a proxy for the socioeconomic status of their corresponding consumer base. After matching this income information with stores, we end up focusing on 81 stores that we can use for further analysis. We receive store-week-product-level stock and transaction data between February 2022 and September 2022. Specifically, we know for every store, every week and every existing product (product is defined as a combination of model and color, and aggregates all possible sizes; in other words, a product is the aggregation of all SKUs of different sizes of the same model-color), how many units were carried by the store at the beginning of the week, and how many were sold at which specific price. This information allows us to infer which products were sent to each store, i.e., the retailer's allocation decision. Furthermore, we have weekly footfall information per store, i.e. how many potential consumers visited each store during which week, as well as product capacity in terms of how many stores each product was allocated to.

Our objective is twofold. First, we illustrate how our model can be applied in practice, and this entails discussing how one should generate the model parameters before applying the algorithms discussed in §§4-5. We thus focus on a single season, Spring-Summer 2022. Second, once parameters have been estimated, we run our algorithms to generate counterfactual allocations, and estimate the gain compared to the current policy. This will provide the order of magnitude of the model-based potential gain of optimized allocations. Of course, this potential gain should be further validated by a field experiment, which entails significant complications and is out of scope for this paper.

We therefore use the data to estimate the parameters necessary for our optimization model, in Equation (2) where consumers follow an MNL demand, thus the objective function is (3). We must focus on a single category of products to properly account for the the product substitution, e.g., a dress cannot be an alternative to a T-shirt for a consumer. We choose to focus our analysis on one of the largest categories, T-shirts. We repeat the entire analysis presented in this section for dresses category for robustness and report it in Appendix B.1.

As our estimation is bipartite by nature, i.e., the attractiveness parameter spans across products and stores, we aggregate the data on the time dimension. We acknowledge that the weekly variance of the estimation variables, such as inventory and sales, might affect the estimation process. Hence, we present another estimation procedure based on a panel data that spans across products, stores and weeks for robustness purposes, see Appendix B.2.

Our final data spans across 1,038 products and 81 stores, and contains the following variables. $Sales_{js}$ is the number of product j sales observed in store s, and we introduce the variable as $logSales_{js} := log(Sales_{js} + 1)$ to reduce the skewness and heteroscedasticity of the variable (we add one to avoid problems with zeros, although the number of observations with allocation and zero sales is very small). This transformation further helps us to easily interpret the coefficient of the estimation as a percentage.

Obviously, the most important driver of sales is allocation, since the product needs to be at the store to be sold. Defining the allocation using the inventory information is quite trivial as a product must be assigned to each store where we observe a large, positive inventory of that product. Although our partner retailer formally allocates products to a specific set of stores, their unique policy of accepting returns in all of their stores, and reselling those at the point-of-return, creates a mismatch between the *planned* allocation and the *observed* allocation. Particularly, even though we observe positive sales and inventory of a specific product in a store, this does not necessarily mean that the managers allocated that product to that store. This policy constitutes a threat to our analysis, since we have to make sure to account for the intended allocations to compare the current practice with what our algorithms suggest. Hence, we define the binary variable $Allocated_{js}$, to distinguish the *planned* allocation, based on the observed inventory of product j in store s. According to the company, it is safe to assume that an allocation was *planned*, if the inventory level of a product in a store is more than four units at any given week (similar results are obtained when we vary this threshold). Thus, we construct our variable following the same approach.

We are interested in estimating v_{js}, which should only be observable when $Allocated_{js} = 1$. It is not trivial to correctly measure this effect empirically. There might be many factors involved such as location, customer base, or product characteristics. The retailer suggested to include the interaction of product price and average income around the store as a predictor of v_{js}. This seems a practical way of coding the affinity between a product and a store, as higher income consumers are inclined to purchase more expensive products than the rest. Therefore, we introduce P_j, which is a percentile measure of the median sales price of all products under the same category, and I_s – a percentile measure of average income level in the respective postal code areas of store locations as a measure of socioeconomic wealth in the geographic area of store s. Note that we could consider alternative notions of compatibility between stores and products, see details in Appendix B.3.

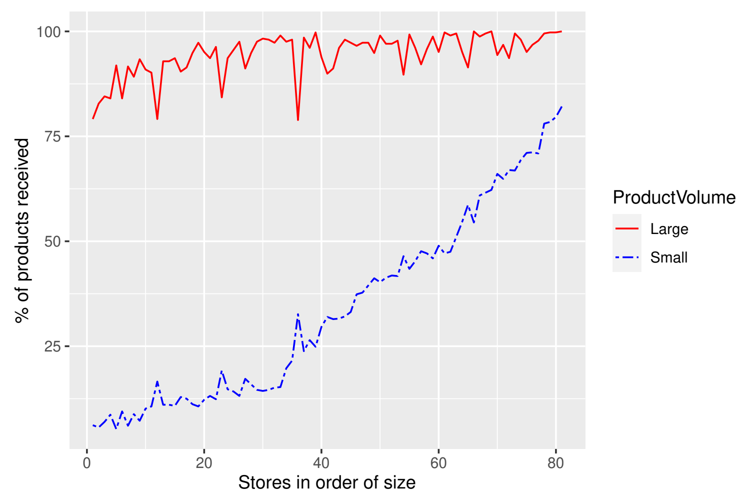
Some descriptive statistics of the final sample are presented in Table 3, along with further information on model parameters that can be directly observed, such as footfall ω_s, price r_{js} or number of stores covered by each product q_j, further discussed in the next subsection.

In addition, we depict in Figure 5 the distribution of products to stores, depending on the relative store size (shown in the x-axis: small stores on the left, large stores on the right). We conjecture that the current practice leads the allocation of products with large volumes to almost all stores, whereas products with limited supply are accumulated in big stores. We can of course see that for both cases, there is significant

Table 3: Descriptive statistics of final data.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
Variables							
logSales_{js}	84,078	0.28	0.83	0.00	0.00	0.00	6.11
logInventory_{js}	84,078	0.31	0.69	0.00	0.00	0.00	4.53
P_j	1,038	0.50	0.29	0.00	0.25	0.75	1.00
I_s	81	0.50	0.28	0.00	0.27	0.73	1.00
Allocated_{js}	84,078	0.49	0.49	0.00	0.00	1.00	1.00
Direct measurements							
Store potential (ω_s)	81	320,464	153,543	70,215	223,613	400,289	864,520
Price (r_{js})	84,078	10.08	5.20	2.69	5.99	12.99	32.00
Product capacity (q_j)	1,038	51	13	13	42	61	81

Figure 5: Distribution of big and small products to stores.



amount of variation, suggesting that there is a non-obvious allocation in place – possibly non-random, implying that we will need to control for potential endogeneity of current allocation decisions, thereby biasing our estimates of v_{js}.

6.2. Generating Model Parameters

After presenting the descriptive information about the data, we move onto describing how to retrieve the parameters of our optimization model. First, we can directly obtain ω_s from the footfall data, since number of visitors in a store is representative of its market potential. Second, we calculate r_{js} as the median sales price of product j in store s, as we can detect the differences of revenue for each specific item across stores. Interestingly, within one country, r_{js} is almost the same in all the stores, which means that in practical terms $r_{js} = r_{js'}$, $s \neq s'$. Third, we calculate q_j as the number of unique stores that had inventory of more than four units of product j at any given time. Descriptive statistics regarding these parameters are presented in Table 3. Unlike the rest of the parameters, there is no trivial way of obtaining the product-store attractiveness parameter v_{js} without a sophisticated estimation. Hence, in §6.3, we build an estimation model for v_{js} within an MNL framework.

Consistent with the literature, we assume that the following function (19) represents the utility that a consumer i obtains from buying j at store s:

$$U_{ijs} := \alpha_j + \alpha'_s + \beta X_{js} + \varepsilon_{ijs} \tag{19}$$

where α_j, and α'_s are product, and store fixed effects respectively, and $\boldsymbol{X_{js}}$ are the covariates of interest, i.e., the product-store interaction discussed in §6.1, and ε_{ijs} is a Gumbel-distributed shock. We aim to capture intrinsic attractiveness of products and stores with α_j and α'_s. We normalize to zero the value of the outside option, i.e., not purchasing anything; in other words, we assume that visiting store s and not buying any product generates a Gumbel utility ε_{i0s}. As a result, $v_{0s} = 1$ and $v_{js} = e^{\alpha_j + \alpha'_s + \beta X_{js}}$.

Although the choice probability is defined at the consumer level, we do not possess individual level data, but we can use the aggregate conversion rate of each product in each store. More specifically, we calculate $Conversion_{js} = Sales_{js}/Footfall_s$. This represents the probability distribution regarding consumer choice for every store in each week. Note that $\sum_{j \in \mathcal{J} \cup \{0\}} Conversion_{js} = 1, \forall s \in \mathcal{S}$, hence the conversion rates in a store is indeed a probability distribution.

Thus, our estimation follows:

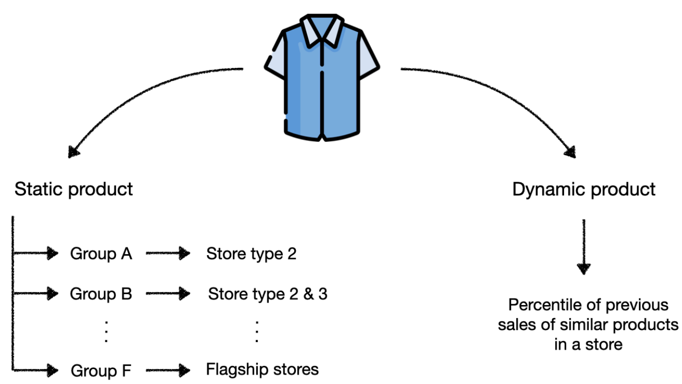
$$\begin{aligned} Conversion_{js} \sim MNL\Big(& \underbrace{\alpha_j + \alpha'_s}_{\text{Product and store heterogeneity}} \\ + & \underbrace{\beta A_{js}}_{\text{Interaction between products and stores}} \\ + & \underbrace{\gamma Allocated_{js}}_{\text{Effect of the planned allocation}} \Big) \end{aligned} \tag{20}$$

where A_{js} is a measure of fitness of matching. In this particular case, $A_{js} := P_j \times I_s$ – see Appendix B.3 for estimations following different proxies for fitness of matching.

6.3. Identification Strategy to Infer v_{js}

There are two major endogeneity concerns regarding our estimation procedure, both of which are associated with the allocation decision. First, planned allocation creates censoring issue in the data since we cannot observe sales of product j in store s if the product was *never* allocated to the store. Note that the particular return policy of the retailer mentioned in §6.1 introduces an exogenous variation to this issue. Although several papers in the literature study demand estimation under censoring, they focus on censoring caused by stock-outs (Jain et al. 2015, Mersereau 2015), which is not applicable for our allocation variable. Also, this is innate to the nature of our problem setting: the allocation problem would not be relevant if every product was to be sold in every store.

Figure 6: Allocation guidelines that are in-use in the partner company.



Second, there is a selection bias that is driven by the allocation decision made by the company. Fortunately, in our interviews with the retailer, we identified that they follow a rule-book approach for allocation rather than formal demand optimization. Although underlying factors to this approach still possesses a threat to our estimation, we argue that these guidelines can be used as instruments for the endogenous allocation variable. We reason this with company guidelines being directly correlated with the planned allocation, whereas independent from MNL-type demand estimation as we introduced in §6.2.

Thankfully, we can eliminate this problem by following the widely used control function approach in the literature (Petrin and Train 2010, Phillips et al. 2015, Boada-Collado and Martínez-de Albéniz 2020). This is a two-step estimation approach. First, the endogenous variable is estimated as a function of a set of exogenous variables and instrumental variables. Second, the residuals obtained in this first estimation are used as a control variable to correct for endogeneity. Most commonly used instrumental variables in inventory planning literature are the lagged variables (e.g. using the inventory levels of previous week to remove endogeneity from the inventory levels of the current week). However, this is not applicable to our setting, since the endogenous variable $Allocated_{js}$ does not vary across time (whether a product is assigned to a store or not). Hence, we use the current allocation guidelines from our partner company as our instruments, which are summarized in Figure 6.

As we can see, the retailer's rule-book approach classifies products in two types. *Static products* are basic products with little seasonality, while *dynamic products* are fashion products following a specific trend or a niche design. This segregation appears within each category, so there are static and dynamic T-shirts, static and dynamic dresses, etc. The allocation strategy for these products entirely depend on this segregation. Thus, company policy is high level, and undermines the effect of matching a product with a specific store.

Particularly, each static product is associated with a certain set of store *types*. This means that these products should not be allocated to stores outside of this set and should be allocated to all stores within this set. Of course, in practice, inventory limitations may make allocation managers deviate from this rule, so the final set of stores that receive the static product may be larger or smaller than the target set. In any case, for these products we can define the binary variable $StoreConsidered_{js}$, equal to one if store s belongs to the target set, and zero otherwise. Since the set of stores is predefined within eight options, it constitutes a valid instrument because it is independent of the actual attractiveness of the product in the specific store.

For dynamic products, the retailer ranks the stores on the basis of previous sales of a set of similar products (e.g. high-waist light-blue jeans, rather than blue jeans). They then allocate the product to stores in the order determined by the rank, starting with highest-selling store, then the second highest, etc. We thus define $StorePreviousPercentile_{js}$ as a continuous variable representing the previous sales of products similar to the *dynamic* product of interest in store s as a percentile. Again, this should be a valid instrument because there is no consideration of demand involved in the allocation of dynamic products.

We acknowledge that our estimation methodology is unusual, and present several reasons here to enlighten the readers. First, we choose the multinomial logit to be able to capture product substitution during our estimation, and for the sake of accuracy with our modeling and analytic findings presented in §3-§4. Following this choice, pursuing a control function approach to tackle endogeneity is the common methodology in the literature (Phillips et al. 2015). Second, we use two multiple first stage regressions to represent the reality in our estimations in the best way possible, following the independent allocation strategies company use for static vs. dynamic products.

It is worth noting that we encounter a methodological dilemma for the first stage estimations. Although econometric theory suggests the use of OLS estimation for the first stage of the control function approach, this possesses a danger to the accuracy of our estimation due to allocation decision being naturally binary. Particularly, using OLS estimation would allow fractional allocations that would not correctly account for actual selection bias. On the other hand, using generalized linear models would make the first stage estimation more accurate, yet there is no proof of accuracy for control function approach following first stage estimations different than OLS. Hence, we opt for the OLS estimation given its stronger theoretical foundations. We thus propose Equation (21) as the first stage estimation model regarding static products, and Equation (22) as the first stage estimation model for dynamic products. Note however that we present the same estimation following a logit first stage in Appendix B.4, for completeness.

$$Allocated_{js} = \lambda_j + \lambda_s^{'} + \mu A_{js} + \nu StoreConsidered_{js} + \varepsilon_{js} \tag{21}$$

$$Allocated_{js} = \lambda_j + \lambda_s^{'} + \mu A_{js} + \nu StorePreviousPercentile_{js} + \varepsilon_{js} \tag{22}$$

We estimate the endogenous variable $Allocated_{js}$ using data that spans across products and stores, hence we control for the corresponding fixed effects (λ_j, $\lambda_s^{'}$), where $A_{js} := P_j \times I_s$. We inform the models on the respective allocation guidelines with; $StoreConsidered_{js}$ for *static* products, and $StorePreviousPercentile_{js}$ for *dynamic* products. Moreover, we include the same interaction variable as Equation (20) to isolate any potential effect of a specific product-store match. We present descriptive statistics regarding the allocation process of products under T-shirts category in Table 22 in Appendix B.6.

Another important condition to ensure the strength of our instrumental variable choice is to check whether there is multicollinearity. We present the zero order correlations for both static and dynamic products in Figure 11 in Appendix B.6. As the highest correlation is 0.19, there is no significant risk of multicollinearity. Although variance inflation factors (VIFs) may not be the most appropriate tool in two-way fixed effects regression, as fixed effects absorb variability, potentially distorting the values, we present VIFs calculated based on residualized (de-meaned) variables in Appendix B.7. All VIFs for the instrumental variables are below 2, confirming that multicollinearity is not a concern.

Table 4: First stage OLS estimation results.

	Dependent variable: Allocated$_{js}$	
	Static Products	*Dynamic Products*
StoreConsidered$_{js}$	0.11^{***}	
	(0.00)	
StorePreviousPercentile$_{js}$		-0.01
		(0.97)
A_{js}	0.14	0.21
	(0.08)	(0.08)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Num. obs.	$14,661$	$69,417$
R^2 (full model)	0.49	0.45

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

We present the results of our first stage regressions following (21)-(22) in Table 4 regarding the products under T-shirts category in spring-summer 2022 collection. Estimation results for (21) are presented in the left column, whereas right column contains the results regarding (22). We find that the exogenous allocation variable for static products ($StoreConsidered_{js}$) is positive and statistically significant, whereas the corresponding variable for dynamic products ($StorePreviousPercentile_{js}$) is insignificant. This suggests that the company adheres to its stated allocation policy for static products, but not for dynamic ones. Interviews with company officials revealed that the allocation process for dynamic products involves multiple employees and coordination across various systems, making it operationally complex. As a result, these decisions are often left to the discretion of individual managers and are thus nearly random.

More interestingly, we observe that the coefficients regarding A_{js} are insignificant for both estimations, which implies that the rule-book allocation approach of the company does not regard the effect of matching a unique product j with a unique store s. This signals a lost opportunity for the company, as it undermines the variance of attractiveness parameter v_{js} across stores.

Following the control function approach, we extract the expected error terms from the first-stage regressions (Table 4) and include them in our second-stage estimation in Equation (20), with results presented in Table 5. Additionally, we report the single-stage MNL estimation results without controlling for the endogeneity of $Allocated_{js}$ in Table 5.

Table 5: Results of single stage MNL estimation and second stage MNL estimation with control function.

	Dependent variable: Conversion$_{js}$	
	Single stage	*Control function*
A_{js}	0.360***	0.253***
	(0.000)	(0.000)
Allocated$_{js}$	4.963***	5.480***
	(0.000)	(0.000)
ExpectedError$_{js}$		-0.523***
		(0.000)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Number of observations	84,078	84,078
Number of variables	1,121	1,122
Log likelihood	-2826628	-2826605
Pseudo R^2	0.11	0.11

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

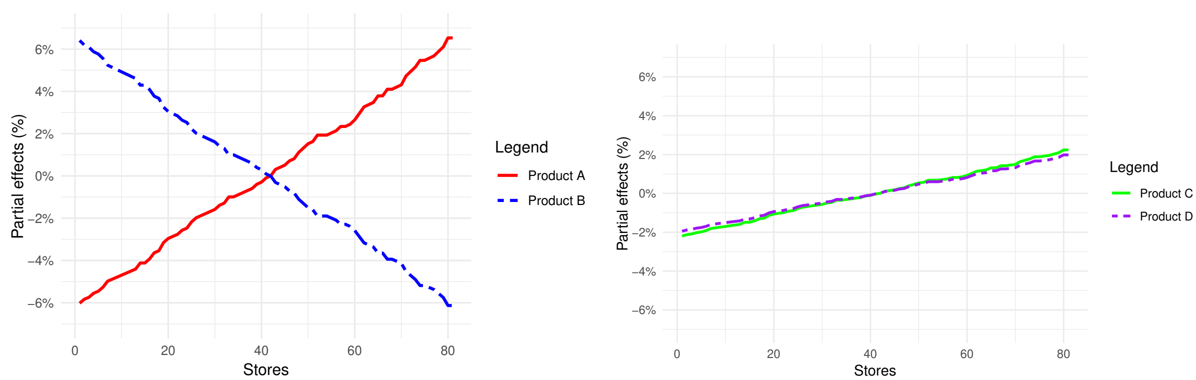
As expected, we observe positive and significant coefficients for $Allocated_{js}$ in both estimations. The large coefficient magnitudes are expected, given that this variable represents the *planned* allocation, which in turn leads to high inventory levels of the specific product at the corresponding store.

Similarly, the negative and significant coefficient for the expected error term from the first-stage estimation is also expected. We conjecture that this reflects product-store allocations that could have been avoided. In other words, this coefficient suggests that the current allocation strategy is suboptimal. In fact, after controling for the endogeneity of $Allocated_{js}$, we see that the corresponding odds ratio for sales potential is $e^{5.480-0.523} \approx 142$, hence suggesting a $\times 142$ increase in odds of a sale when a product is *intentionally* allocated to a store, compared to selling only those items that may be returned, hence available by chance.

Interestingly, we observe positive and significant coefficients for the interaction variable A_{js} in both of the estimation models. This indicates that when products are matched with stores based on their monetary properties, they are more likely to be sold in those stores. Notably, this interaction term remains both statistically and economically significant even when applying the control function approach, despite already being included in the first-stage regressions. This suggests that products perform better post-allocation when they are matched with stores based on their price positioning and the socioeconomic wealth of the stores surrounding area. When a product is allocated to a store with which it aligns economically, the odds of a sale increase by 29% ($e^{0.253} \approx 1.29$). However, since A_{js} shifting from 0 to 1 is rare in practice, this likely overstates the true effect. To provide a more realistic picture, we examine how the partial effect of A_{js} varies across store-product pairs, presenting context-specific contributions instead of relying on the average odds ratio.

To derive these partial effects of A_{js}, we follow the algebra of Cameron and Trivedi (2005), who extend the results of binary logistic regression with fixed effects to the multinomial case. We visualize these partial effects for selected products in Figure 7, where stores are ordered in ascending order of their respective average income level, i.e., I_s.

Figure 7: Variation of partial effect of A_{js} for different products across stores.



(a) Products A and B. **(b)** Products C and D.

In Figure 7a, Product A is the most expensive product in our data set, whereas Product B is the least expensive. As expected, the cheaper product performs better in stores serving lower-income consumers, whereas the expensive product's performance improves as the income level of consumers increases. Additionally, we observe that the maximum observed partial effect on the odds ratio is 6.5%, which is significantly lower than the previously mentioned 29% average effect. This is because the measure A_{js} is unlikely to take extreme values in most cases.

Furthermore, in Figure 7b, we illustrate how these partial effects differ for two products within a similar price range. Although Product C is slightly more expensive than Product D, we observe the same trend: the more expensive product generates a higher partial effect in stores located in wealthier areas. However, the difference in partial effect magnitudes between Products C and D is much smaller than the difference observed between Products A and B. This finding underscores the importance of the specific relationship between product price and the income level of potential consumers in determining sales performance.

6.4. Counterfactual

After establishing the estimation procedure for the attractiveness parameter v_{js}, we are ready to evaluate the impact of our finding in §4. First, we consider the observed allocation, and calculate the corresponding revenue of the company on the basis of our estimated parameters ω_s, r_{js} and q_j as previously explained, and v_{js} using the coefficients following the control function approach, and Equation (20).

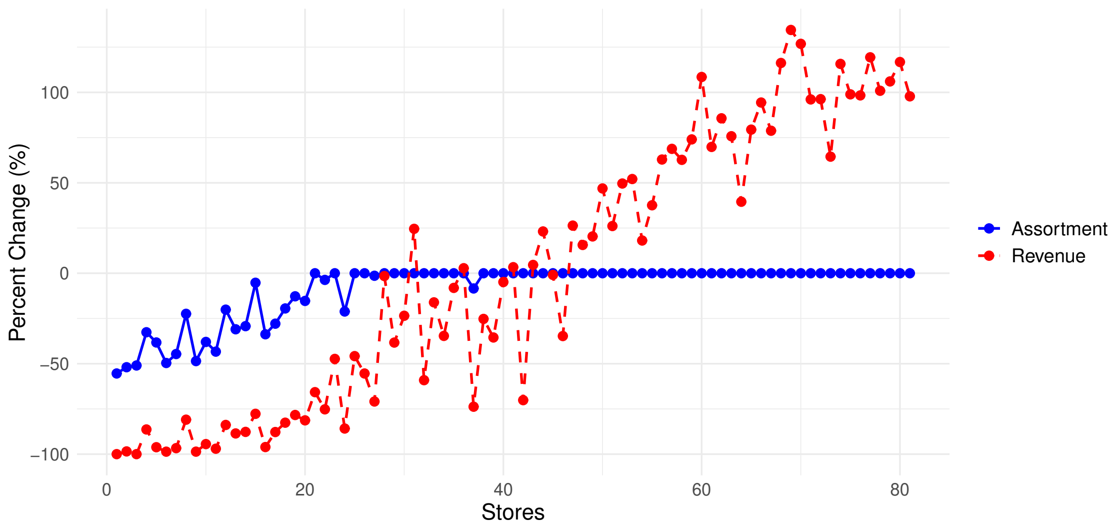
Following the findings from the computational experiment presented in §5, we choose to apply our IF (3) for a realistic approximation, since the run time for solution approaches are expected to be unpractical for the practitioners. Run time is approximately 4 minutes for T-shirts category products. Afterwards, we calculate the revenues using the same set of parameters regarding the proposed allocation. We observe a potential improvement of 30.4% for the T-shirts category, This is a very significant increase in revenues, of the order of tens of millions of euros per year.

Although the importance of our study to practitioners is evident, we conduct a deeper analysis to uncover the underlying mechanisms. In Figure 8, we visualize how assortment size and expected revenue per store changed following the new allocation after the algorithms were applied to T-shirts, with stores presented in increasing order of their observed revenues. We observe that the majority of the improvement occurs in stores that already have high revenue. This outcome is driven by two key factors. First, our estimation of v_{js} parameters favors better-performing stores, as higher-performing stores are more likely to have higher estimated parameters. Second, the greedy nature of our algorithm prioritizes swaps that match higher product performance with higher store performance, effectively aligning product price with the income level of the store's customer base.

Notably, although the algorithm may lead to store starvation in lower-revenue stores, it still boosts overall revenue by providing the same assortment size with a different mix. Our optimization model and algorithm are flexible enough to accommodate store starvation concerns through equality constraints on store capacities and by introducing an additional logical control to ensure full store capacity utilization, respectively. However, in its current form, this analysis also provides valuable insights for retailers regarding store performance.

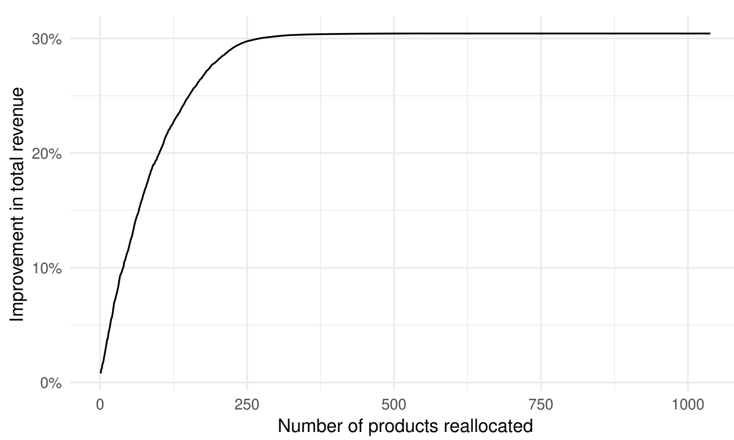
Moreover, the marginal improvement of reallocating one additional product decreases with number of products that have been reallocated already: in Figure 9, we observe that after around 25% of products are reallocated to stores, the gain from further reallocation becomes almost insignificant. There are two underlying reasons for this. First is the greedy nature of our IF. Particularly, the algorithm prioritizes products with higher price margins and higher popularity, hence with each product reallocation, the average

Figure 8: Change in assortment size and revenue per store following the new allocation after the algorithms were applied to T-shirts category.



price margin and popularity of remaining products become less and less. Another reason is innate to the presence of substitution effects, an essential feature of the MNL. When products are reallocated, their performance in the new stores increases. However, this improvement cannibalizes on the performance of the other products that are already in the store. Thus, as better products enter a store, it becomes more challenging to avoid this cannibalization effect. The main takeaway of this is that even a partial reallocation (e.g., only 200 products out of 1,000) that considers product-store matching has potential to significantly increase retailers' revenues.

Figure 9: Improvement in the revenue by number of products reallocated.



7. Conclusion

This paper has introduced a store-specific assortment optimization model for a network of stores with limited product supply. Our formulation incorporates a joint product-store attractiveness parameter to enhance the potential for matching the right products with the right stores. We demonstrate that this problem is NP-complete and analyze its continuous relaxation. We then propose an algorithm that efficiently identifies a local maximum, with performance guarantees. Additionally, we analyze assortment optimization under the multinomial logit (MNL) consumer choice model and leverage its optimality characteristics to develop a time-efficient approximation algorithm.

To evaluate the performance of these algorithms, we conduct an extensive computational study, assessing both optimality gaps and running times. Moreover, we validate our optimization model using real-world data from a multinational apparel retailer. By leveraging company policies to define a robust identification strategy, we estimate the performance of each product across all stores. Our findings reveal that a retailer's revenues can increase revenues by up to 30% when products are allocated to stores based on their price positioning and the socioeconomic characteristics of store locations.

Our approach to assortment planning has broad implications for the retail industry. We show that neglecting the product-store matching specificity represents a lost opportunity for retailers. Our proposed algorithms are easily implementable in practice. Additionally, we find that even partial reallocation of products based on store-specific performance can lead to significant revenue gains.

Our findings open new avenues for future research on store-specific assortments. One promising direction is the integration of operational aspects, such as joint manufacturing and assortment decisions across a store network. Another avenue is to explore the dynamic assortment optimization problem, where products are not only allocated once but can be reallocated among stores over time. Additionally, further empirical research is needed to enhance our understanding of product-store interactions. While we have conducted an in-depth empirical analysis, our study is subject to data limitations, and there are numerous factors influencing store-product relationships. Future research could expand empirical investigations by incorporating alternative estimation procedures and methodologies to provide deeper insights into this critical problem.

Acknowledgements

The authors are immensely grateful to Carles Pérez Guallar for his collaboration and industry insights. They also thank Danny Segev for his valuable input related to Theorem 2 and appreciate the seminar participants at IESE Business School, Erasmus University, Tilburg University, and Frankfurt School of Finance and Management for their insightful comments.

References

Agrawal, S., V. Avadhanula, V. Goyal, and A. Zeevi. 2019. MNL-bandit: A dynamic learning approach to assortment selection. *Operations Research* 67 (5): 1453–1485.

Anderson, S. P., A. De Palma, and J.-F. Thisse. 1992. *Discrete choice theory of product differentiation*. MIT press.

Barré, T., O. E. Housni, and A. Lodi. 2024. Assortment optimization with visibility constraints. In *International Conference on Integer Programming and Combinatorial Optimization*, 124–138. Springer.

Boada-Collado, P., and V. Martínez-de Albéniz. 2020. Estimating and optimizing the impact of inventory on consumer choices in a fashion retail setting. *Manufacturing & service operations management* 22 (3): 582–597.

Bront, J. J. M., I. Méndez-Díaz, and G. Vulcano. 2009. A column generation algorithm for choice-based network revenue management. *Operations research* 57 (3): 769–784.

Cameron, A. C., and P. K. Trivedi. 2005. *Microeconometrics: methods and applications*. Cambridge university press.

Caro, F., and J. Gallien. 2007. Dynamic assortment with demand learning for seasonal consumer goods. *Management science* 53 (2): 276–292.

Caro, F., and J. Gallien. 2010. Inventory management of a fast-fashion retail network. *Operations research* 58 (2): 257–273.

Caro, F., and V. Martínez-de Albéniz. 2015. Fast fashion: Business model overview and research opportunities. *Retail supply chain management: Quantitative models and empirical studies*:237–264.

Caro, F., V. Martínez-de Albéniz, and P. Rusmevichientong. 2014. The assortment packing problem: Multiperiod assortment planning for short-lived products. *Management Science* 60 (11): 2701–2721.

Chen, L., A. J. Mersereau, and Z. Wang. 2017. Optimal merchandise testing with limited inventory. *Operations Research* 65 (4): 968–991.

Davis, J. M., G. Gallego, and H. Topaloglu. 2014. Assortment optimization under variants of the nested logit model. *Operations Research* 62 (2): 250–273.

Désir, A., V. Goyal, and J. Zhang. 2022. Capacitated assortment optimization: Hardness and approximation. *Operations Research* 70 (2): 893–904.

Feldman, J., and H. Topaloglu. 2015a. Bounding optimal expected revenues for assortment optimization under mixtures of multinomial logits. *Production and Operations Management* 24 (10): 1598–1620.

Feldman, J. B., and H. Topaloglu. 2015b. Capacity constraints across nests in assortment optimization under the nested logit model. *Operations Research* 63 (4): 812–822.

Gallego, G., and H. Topaloglu. 2014. Constrained assortment optimization for the nested logit model. *Management Science* 60 (10): 2583–2601.

Gallien, J., A. J. Mersereau, A. Garro, A. D. Mora, and M. N. Vidal. 2015. Initial shipment decisions for new products at Zara. *Operations Research* 63 (2): 269–286.

Jain, A., N. Rudi, and T. Wang. 2015. Demand estimation and ordering under censoring: Stock-out timing is (almost) all you need. *Operations Research* 63 (1): 134–150.

Kök, A. G., and M. L. Fisher. 2007. Demand estimation and assortment optimization under substitution: Methodology and application. *Operations Research* 55 (6): 1001–1021.

Kök, A. G., M. L. Fisher, and R. Vaidyanathan. 2008. Assortment planning: Review of literature and industry practice. *Retail Supply Chain Management*:99–153.

Kunnumkal, S., and V. Martínez-de Albéniz. 2019. Tractable approximations for assortment planning with product costs. *Operations Research* 67 (2): 436–452.

Martínez-de Albéniz, V., and S. Kunnumkal. 2022. A model for integrated inventory and assortment planning. *Management Science* 68 (7): 5049–5067.

Méndez-Díaz, I., J. J. Miranda-Bront, G. Vulcano, and P. Zabala. 2014. A branch-and-cut algorithm for the latent-class logit assortment problem. *Discrete Applied Mathematics* 164:246–263.

Mersereau, A. J. 2015. Demand estimation from censored observations with inventory record inaccuracy. *Manufacturing & Service Operations Management* 17 (3): 335–349.

Petrin, A., and K. Train. 2010. A control function approach to endogeneity in consumer choice models. *Journal of marketing research* 47 (1): 3–13.

Phillips, R., A. S. Şimşek, and G. Van Ryzin. 2015. The effectiveness of field price discretion: Empirical evidence from auto lending. *Management Science* 61 (8): 1741–1759.

Raghavan, P., and C. D. Tompson. 1987. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica* 7 (4): 365–374.

Rusmevichientong, P., Z.-J. M. Shen, and D. B. Shmoys. 2010. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations Research* 58 (6): 1666–1680.

Rusmevichientong, P., D. Shmoys, C. Tong, and H. Topaloglu. 2014. Assortment optimization under the multinomial logit model with random choice parameters. *Production and Operations Management* 23 (11): 2023–2039.

Ryzin, G. v., and S. Mahajan. 1999. On the relationship between inventory costs and variety benefits in retail assortments. *Management Science* 45 (11): 1496–1509.

Şen, A., A. Atamtürk, and P. Kaminsky. 2018. A conic integer optimization approach to the constrained assortment problem under the mixed multinomial logit model. *Operations Research* 66 (4): 994–1003.

Sumida, M., G. Gallego, P. Rusmevichientong, H. Topaloglu, and J. Davis. 2021. Revenue-utility tradeoff in assortment optimization under the multinomial logit model with totally unimodular constraints. *Management Science* 67 (5): 2845–2869.

Talluri, K., and G. Van Ryzin. 2004. Revenue management under a general discrete choice model of consumer behavior. *Management Science* 50 (1): 15–33.

Train, K. E. 2009. *Discrete choice methods with simulation*. Cambridge university press.

Ulu, C., D. Honhon, and A. Alptekinoğlu. 2012. Learning consumer tastes through dynamic assortments. *Operations Research* 60 (4): 833–849.

Appendices

A. Appendix: Proofs and Formulations

A.1. Proof of Theorem 1

Consider two stores $s \in \mathcal{S} = \{1, 2\}$, and n products $j \in \mathcal{J} = \{1, 2, ..., n\}$ with capacities $q_1 = ... = q_n = 1$, and retail prices $r_{1s} = r_{2s} = ... = r_{ns} = 1 \ \forall s \in \mathcal{S}$, where the set of available products are $\mathcal{A}_1$ and $\mathcal{A}_2$ respective to the store indices. We assume market potential of stores are normalized to 1, i.e, $\omega_1 = \omega_2 = 1$; the product-store attractiveness parameters $v_{js} \in \mathbb{Z}_+ \ \forall j, s$; non-purchase attractiveness parameters $v_{01} = v_{02} = v_0 \in \mathbb{Q}_+$; store capacities are finitely sufficiently large $c_1, c_2 > M$; and the target profit $K \in \mathbb{Q}_+$.

We want to check if there exists a partition $\mathcal{A}_1$ and $\mathcal{A}_2$ such that $\mathcal{A}_1 \cup \mathcal{A}_2 = 1, ..., n$ and $\mathcal{A}_1 \cap \mathcal{A}_2 = \emptyset$ such that

$$\frac{\sum_{j \in \mathcal{A}_1} v_{j1}}{v_0 + \sum_{j \in \mathcal{A}_1} v_{j1}} + \frac{\sum_{j \in \mathcal{A}_2} v_{j2}}{v_0 + \sum_{j \in \mathcal{A}_2} v_{j2}} \geq K.$$

Let v_0 be given. We show that *Two-Store Equal-Margin Assortment Customization* is in NP, because we can transform an arbitrary instance of *Partition*, which is a well-known NP-complete problem, to an equivalent *Two-Store Equal-Margin Assortment Customization* problem.

The *Partition* problem is defined as follows.

Consider the set of items indexed by $1, ..., k$ and the size $c_i \in \mathbb{Z}_+$ associated with each item. Is there a subset $S \subseteq \{1, .., k\}$ such $\sum_{i \in S} c_i = \sum_{i \notin S} c_i$?

Let $C = \frac{1}{2} \sum_{i=1}^{n} c_i \in \mathbb{Z}_+$. Since $\sum_{i \in S} c_i = \sum_{i \notin S} c_i$ if and only if $\sum_{i \in S} c_i = C$, we may assume without loss of generality that $C \in \mathbb{Z}_+$.

An instance of *Partition* can be solved by solving the following instance of *Two-Store Equal-Margin Assortment Customization*: define the preference weights as $\{v_{j1} \cup v_{j2}\} = c_i$ for $i = 1, ..., 2n$, and set the target profit as $K = 2C/(v_0 + C)$ for both stores. The *Partition* problem indeed has a solution *if an only if* there exists a partition X_1 and X_2 such that $\frac{\sum_{j \in X_1} v_{j1}}{v_0 + \sum_{j \in X_1} v_{j1}} + \frac{\sum_{j \in X_2} v_{j2}}{v_0 + \sum_{j \in X_2} v_{j2}} \leq \max_{y \in [0, 2T]} \{\frac{y}{v_0 + y} + \frac{2T - y}{v_0 + 2T - y}\} = \frac{2C}{v_0 + C} = K$. This is true because $G(y) = \frac{y}{v_0 + y} + \frac{2C - y}{v_0 + 2C - y}$ is concave in y over the interval $[0, 2C]$ and achieves a unique maximum at $y = C$. $\square$

A.2. Proof of Theorem 2

Let $\boldsymbol{x}^*$ be an optimal allocation matrix for products $j \in \{1, 2, \ldots, n\}$ across stores $s \in \{1, 2, \ldots, m\}$. The objective value of this optimal solution is:

$$\text{OPT} = Z(\boldsymbol{x}^*) = \sum_{s \in S} f_s(\boldsymbol{x}^*_{\cdot, s}).$$

Since there are m stores, the pigeonhole principle ensures that at least one store must contribute at least an average fraction of the total revenue:

$$f_s(\boldsymbol{x}^*_{\cdot,s}) \geq \frac{\text{OPT}}{m}, \quad \text{for some } s \in S.$$

Now, consider solving an unconstrained MNL assortment problem for each store independently. Each of these solutions constitutes a feasible solution to the overall problem since no products are allocated to multiple stores, and each product must be available for at least one store. We define the set of store-specific optimal solutions as:

$$\mathcal{X} = \left\{ \boldsymbol{x}^*_s := \arg \max_{\boldsymbol{x}_{\cdot,\boldsymbol{s}} \in \{0,1\}^{|J|}} f_s(\boldsymbol{x}_{\cdot,\boldsymbol{s}}) \mid s \in S \right\}.$$

We then select the store whose optimal assortment yields the maximum revenue:

$$\boldsymbol{y}^* = \arg \max_{\boldsymbol{x}^*_s \in \mathcal{X}} f_s(\boldsymbol{x}_{\cdot,\boldsymbol{s}}).$$

By the previously established bound from the pigeonhole principle, we have:

$$\frac{Z(\boldsymbol{y}^*)}{\text{OPT}} \geq \frac{1}{m}.$$

Thus, this selection guarantees a $1/m$-approximation of the optimal solution. Since allocating products to a single store represents a potential worst-case scenario for the projection of the fractional allocation, we can conclude that the projection operation itself also maintains a $1/m$-approximation of the optimal solution. □

A.3. Disproof of Convexity under MNL Demand

Let's focus on an instance where there are two stores and two products, and consumers follow MNL type discrete choice. We assume homogeneity across stores regarding their market shares (i.e. $\omega_s = 1$, $\forall s \in S$), and non-purchase attractiveness parameters (i.e. $v_{0s} = 1$, $\forall s \in S$) for ease of notation. We assume both products are limited in quantity to be sent to at most one store, i.e. $q_1 = q_2 = 1$, and store capacities are sufficiently large $c_1, c_2 > M$. Moreover, we assume that the revenue per unit sales of each product is homogeneous across stores, i.e. $r_{11} = r_{12} = r_1$, $r_{21} = r_{22} = r_2$. Also, we consider a positive and unique set of store-product attractiveness parameters $v_{11}, v_{12}, v_{21}, v_{22}$. Lastly, we call that the optimal allocation fractional as $x^*_{11} = x$, $x^*_{12} = 1 - x$, $x^*_{21} = y$, and $x^*_{22} = 1 - y$, where $0 \leq x, y \leq 1$. Then, we can denote the optimal objective function as follows:

$$\begin{aligned} OBJ^* =& \frac{r_1 v_{11} x + r_2 v_{21} y}{v_{01} + v_{11} x + v_{21} y} + \frac{r_1 v_{12}(1-x) + r_2 v_{22}(1-y)}{v_{02} + v_{12}(1-x) + v_{22}(1-y)} \\ =& \Pi_1 + \Pi_2 \end{aligned} \tag{23}$$

We want to check whether that this optimal assignment is a local maximum if the objective function or not, i.e. whether $det(Hessian) > 0$, and $\partial^2 OBJ^*/\partial x^2 < 0$. holds or not. Since we have two decision variables involved – x and y, we disassemble $\frac{\partial OBJ}{\partial x}$, and $\frac{\partial OBJ}{\partial y}$ as follows:

$$\begin{aligned}\frac{\partial OBJ}{\partial x} &= \frac{\partial \Pi_1}{\partial x} + \frac{\partial \Pi_2}{\partial x}\\ \frac{\partial OBJ}{\partial y} &= \frac{\partial \Pi_1}{\partial y} + \frac{\partial \Pi_2}{\partial y}\end{aligned}$$

Then,

$$\begin{aligned}\frac{\partial \Pi_1}{\partial x} &= \frac{r_1 v_{11}}{v_{01} + v_{11}x + v_{21}y} - \frac{v_{11}(r_1 v_{11} x + r_2 v_{21} y)}{(v_{01} + v_{11}x + v_{21}y)^2}\\ &= \frac{r_1 v_{11}}{v_{01} + v_{11}x + v_{21}y} - \frac{v_{11}}{v_{01} + v_{11}x + v_{21}y}\Pi_1\\ &= \frac{v_{11}}{v_{01} + v_{11}x + v_{21}y}(r_1 - \Pi_1) = \frac{v_{11}}{N_1}(r_1 - \Pi_1)\end{aligned}$$

$$\begin{aligned}\frac{\partial \Pi_2}{\partial x} &= \frac{-r_1 v_{12}}{v_{02} + v_{12}(1-x) + v_{21}(1-y)} - \frac{-v_{12}[r_1 v_{12}(1-x) + r_2 v_{21}(1-y)]}{[v_{02} + v_{12}(1-x) + v_{22}(1-y)]^2}\\ &= \frac{-r_1 v_{12}}{v_{02} + v_{12}(1-x) + v_{22}(1-y)} + \frac{v_{12}}{v_{02} + v_{12}(1-x) + v_{22}(1-y)}\Pi_2\\ &= \frac{v_{12}}{v_{02} + v_{12}(1-x) + v_{22}(1-y)}(-r_1 + \Pi_2) = -\frac{v_{12}}{N_2}(r_1 - \Pi_2)\end{aligned}$$

By symmetry:

$$\begin{aligned}\frac{\partial \Pi_1}{\partial y} &= \frac{v_{21}}{v_{01} + v_{11}x + v_{21}y}(r_2 - \Pi_1) = \frac{v_{21}}{N_1}(r_2 - \Pi_1)\\ \frac{\partial \Pi_2}{\partial y} &= \frac{-v_{22}}{v_{02} + v_{12}(1-x) + v_{22}(1-y)}(r_2 - \Pi_2) = -\frac{v_{22}}{N_2}(r_2 - \Pi_2)\end{aligned}$$

Then,

$$\begin{aligned}\frac{\partial OBJ^*}{\partial x} &= \frac{v_{11}}{N_1}(r_1 - \Pi_1) - \frac{v_{12}}{N_2}(r_1 - \Pi_2)\\ \frac{\partial OBJ^*}{\partial y} &= \frac{v_{21}}{N_1}(r_2 - \Pi_1) - \frac{v_{22}}{N_2}(r_2 - \Pi_2)\end{aligned}$$

As we are interested in understanding the concavity around (23), we have to check whether the Hessian matrix is negative semidefinite or not. Thus:

$$\begin{aligned}\frac{\partial^2\Pi_1}{\partial x^2} &= \frac{\partial}{\partial x}\left[\frac{\partial\Pi_1}{\partial x}\right] = \frac{\partial}{\partial x}\left[\frac{v_{11}}{N_1}(r_1-\Pi_1)\right] \\ &= \frac{-v_{11}(\partial\Pi_1/\partial x)}{N_1} - \frac{v_{11}(r_1-\Pi_1)v_{11}}{N_1^2} \\ &= -\left(\frac{\partial\Pi_1}{\partial x}\right)^2 \frac{1}{r_1-\Pi_1} - \frac{\partial\Pi_1}{\partial x}\frac{1}{r_1-\Pi_1}\frac{\partial\Pi_1}{\partial x} \\ &= \left(\frac{\partial\Pi_1}{\partial x}\right)^2\left[\frac{-2}{r_1-\Pi_1}\right]\end{aligned}$$

By symmetry:

$$\frac{\partial^2\Pi_2}{\partial x^2} = -\left(\frac{\partial\Pi_2}{\partial x}\right)^2\left[\frac{-2}{r_1-\Pi_2}\right]$$

Notice that $\partial^2\Pi_1/\partial x^2$ is only strictly negative when $r_1 - \Pi_1 > 0$, which corresponds to the domain where the first-order gradient is positive. Then, we change our focus from the general concavity toward quasiconcavity on the domain of positive first-order gradient, i.e., under the assumptions of $r_1, r_2 > \Pi_1$ and $r_1, r_2 < \Pi_2$ to establish the domain where $\frac{\partial OBJ}{\partial x} > 0$, and $\frac{\partial OBJ}{\partial y} > 0$.

Hence, within the respective domain,

$$\frac{\partial^2 OBJ^*}{\partial x^2} = \frac{\partial^2\Pi_1}{\partial x^2} + \frac{\partial^2\Pi_2}{\partial x^2} < 0$$

And, by symmetry,

$$\frac{\partial^2 OBJ^*}{\partial y^2} < 0.$$

We still need to check whether the Hessian matrix is definite or indefinite. Hence, we move forward with $\frac{\partial^2\Pi}{\partial x\partial y} = \frac{\partial^2\Pi_1}{\partial x\partial y} + \frac{\partial^2\Pi_2}{\partial x\partial y}$:

$$\begin{aligned}\frac{\partial^2\Pi_1}{\partial x\partial y} &= \frac{\partial}{\partial x}\left[\frac{\partial\Pi_1}{\partial y}\right] = \frac{\partial}{\partial x}\left[\frac{v_{21}}{N_1}(r_2-\Pi_1)\right] \\ &= \frac{-v_{21}(\partial\Pi_1/\partial x)}{N_1} - \frac{v_{21}(r_2-\Pi_1)v_{11}}{N_1^2} \\ &= -\frac{\partial\Pi_1}{\partial y}\frac{1}{r_2-\Pi_1}\frac{\partial\Pi_1}{\partial x} - \frac{\partial\Pi_1}{\partial y}\frac{\partial\Pi_1}{\partial x}\frac{1}{r_1-\Pi_1} \\ &= \frac{\partial\Pi_1}{\partial y}\frac{\partial\Pi_1}{\partial x}\left[-\frac{1}{r_2-\Pi_1} - \frac{1}{r_1-\Pi_1}\right]\end{aligned}$$

$$\frac{\partial^2 \Pi_2}{\partial x \partial y} = \frac{\partial}{\partial x}\left[\frac{\partial \Pi_2}{\partial y}\right] = \frac{\partial}{\partial x}\left[\frac{-v_{22}}{N_2}(r_2 - \Pi_2)\right]$$
$$= \frac{v_{22}(\partial \Pi_2/\partial x)}{N_2} - \frac{(-v_{22})(r_2 - \Pi_2)(-v_{12})}{N_2^2}$$
$$= -\frac{\partial \Pi_2}{\partial y}\frac{1}{r_2 - \Pi_2}\frac{\partial \Pi_2}{\partial x} - \frac{\partial \Pi_2}{\partial y}\frac{\partial \Pi_1}{\partial x}\frac{1}{r_1 - \Pi_2}$$
$$= \frac{\partial \Pi_2}{\partial y}\frac{\partial \Pi_2}{\partial x}\left[-\frac{1}{r_2 - \Pi_2} - \frac{1}{r_1 - \Pi_2}\right]$$

Recall that we have to check $det(Hessian) \geq 0$ or not, i.e.

$$\left(\frac{\partial^2 \Pi}{\partial x \partial y}\right)^2 \underbrace{\leq}_{?} \frac{\partial^2 \Pi}{\partial x^2}\frac{\partial^2 \Pi}{\partial y^2}$$

Let's define

$$\left(\frac{\partial^2 \Pi}{\partial x \partial y}\right)^2 = \left(D_{1y}D_{1x}\left[-\frac{1}{r_2 - \Pi_1} - \frac{1}{r_1 - \Pi_1}\right] + D_{2y}D_{2x}\left[-\frac{1}{r_2 - \Pi_2} - \frac{1}{r_1 - \Pi_2}\right]\right)^2$$

where $D_{1x} = \frac{\partial \Pi_1}{\partial x}$, $D_{2x} = \frac{\partial \Pi_2}{\partial x}$, $D_{1y} = \frac{\partial \Pi_1}{\partial y}$, $D_{2y} = \frac{\partial \Pi_2}{\partial y}$.

As the closed form representation for this expression is hard to follow, we move forward with an instance. Thus, we plug in the following instance to disprove by counterexample; $r_1 = 1$, $r_2 = 2$, $v_{11} = v_{12} = v_{21} = v_{22} = v_{01} = v_{02} = 1$, and choose $x = y = 0.5$, which shows $\left(\frac{\partial^2 \Pi}{\partial x \partial y}\right)^2 = 0.5625 > \frac{\partial^2 \Pi}{\partial x^2}\frac{\partial^2 \Pi}{\partial y^2} = 0.3125$. Hence, any critical point regarding (23) is a saddle point. Thus, we can only affirm that FFS converges to a local maximum around the blank assortments under the multinomial logit type demand assumption. □

Furthermore, we illustrate how the expected profits change following $(x, y) \in [0, 1]^2$ in Figure 1 in the main text.

A.4. Pseudo-codes of Proposed Algorithms

Algorithm 1 *Fractional Fill and Swap Algorithm*

Initialize blank assortment matrix $\boldsymbol{x}_{\cdot,\boldsymbol{s}} = \mathbf{0}$
Calculate the corresponding revenue matrix $\mathbf{f_s} = \mathbf{0}$, $\forall s \in S$
Set the auxiliary previous revenue $\mathbf{f_s}^* = -\mathbf{1}, \forall s \in S$
while $\sum_{s \in S}(\mathbf{f_s} - \mathbf{f_s}^*) > \mathbf{0}$ **do**
  Calculate the corresponding first-order gradient matrix $\mathbf{f}_\mathbf{s}^{'} = \frac{\partial}{\partial x_{js}}[f_s(\boldsymbol{x}_{\cdot,\boldsymbol{s}})]$
  Get the product-store tuple $(j^{'}, s^{'}) := \underset{j \in J, s \in S}{\arg\max}\ (\mathbf{f}_\mathbf{s}^{'} | \mathbf{x}_{\cdot,\boldsymbol{s}} < \mathbf{1} \wedge \sum_{\mathbf{s} \in \mathbf{S}} \mathbf{x_{j's}} < \mathbf{q_{j'}} \wedge \sum_{\mathbf{j} \in \mathbf{J}} \mathbf{x_{js'}} < \mathbf{k_{s'}})$
  Add a small fractional amount (e) to the corresponding $x_{j^{'}s^{'}} \subseteq \boldsymbol{x}_{\cdot,\boldsymbol{s}}$
  if $\sum_{s \in S} x_{j's} = q_j$ **then**
    Define $\mathcal{S} = \{s \in S \mid \sum_{j \in J} x_{js'} < k_{s'}\}$
    Define $s_{min} = \underset{j=j^{'}, s \in \mathcal{S}}{\arg\min}\, \mathbf{f}_\mathbf{s}^{'}$ and $s_{max} = \underset{j=j^{'}, s \in \mathcal{S}}{\arg\max}\, \mathbf{f}_\mathbf{s}^{'}$
    Define $\delta = min(x_{j's_{min}}, 1 - x_{j's_{max}}, e)$
    Set $x_{j's_{min}} = x_{j's_{min}} - \delta$ and $x_{j's_{max}} = x_{j's_{min}} + \delta$
  end if
  Define new allocation matrix $\boldsymbol{x}^*_{\cdot,\boldsymbol{s}}$
  Calculate a new gradient matrix $\mathbf{f}^*_\mathbf{s}$ following $\boldsymbol{x}^*_{\cdot,\boldsymbol{s}}$
end while

Algorithm 2 *Fractional Fill Algorithm*

Initialize blank assortment matrix $\boldsymbol{x}_{\cdot,\boldsymbol{s}} = \mathbf{0}$
Calculate the corresponding revenue matrix $\mathbf{f_s} = \mathbf{0}$, $\forall s \in S$
Set the auxiliary previous revenue $\mathbf{f_s}^* = -\mathbf{1}, \forall s \in S$
while $\sum_{s \in S}(\mathbf{f_s} - \mathbf{f_s}^*) > \mathbf{0}$ **do**
  Calculate the corresponding first-order gradient matrix $\mathbf{f}_\mathbf{s}^{'} = \frac{\partial}{\partial x_{js}}[f_s(\boldsymbol{x}_{\cdot,\boldsymbol{s}})]$
  Get the product-store tuple $(j^{'}, s^{'}) := \underset{j \in J, s \in S}{\arg\max}\ (\mathbf{f}_\mathbf{s}^{'} | \mathbf{x}_{\cdot,\boldsymbol{s}} < \mathbf{1} \wedge \sum_{\mathbf{s} \in \mathbf{S}} \mathbf{x_{j's}} < \mathbf{q_{j'}} \wedge \sum_{\mathbf{j} \in \mathbf{J}} \mathbf{x_{js'}} < \mathbf{k_{s'}})$
  Add a small fractional amount (e) to the corresponding $x_{j^{'}s^{'}} \subseteq \boldsymbol{x}_{\cdot,\boldsymbol{s}}$
  Define new allocation matrix $\boldsymbol{x}^*_{\cdot,\boldsymbol{s}}$
  Calculate a new gradient matrix $\mathbf{f}^*_\mathbf{s}$ following $\boldsymbol{x}^*_{\cdot,\boldsymbol{s}}$
end while

Algorithm 3 *Integer Fill Algorithm*

Initialize blank assortment matrix $\boldsymbol{x_{js}} = \mathbf{0}$
Initialize the corresponding matrices $\mathbf{V_s} = \mathbf{1}$, and $\mathbf{\Pi_s} = \mathbf{0}$.
Set the auxiliary previous revenue $\mathbf{\Pi_s}^* = -\mathbf{1}, \forall s \in S$
while $\sum_{s \in S} (\mathbf{\Pi_s} - \mathbf{\Pi_s}^*) > \mathbf{0}$ **do**
    Calculate the corresponding gradient matrix $\boldsymbol{W_{js}} = W_{js}(\mathbf{\Pi_s}, \mathbf{V_s})$.
    Get the product-store tuple $(j^{'}, s^{'}) = \underset{j \in J, s \in S}{\arg\max} (\boldsymbol{W_{js}} | x_{js} = 0 \wedge \sum_{s \in S} x_{j^{'}s} < q_{j^{'}} \wedge \sum_{j \in J} x_{js^{'}} < k_{s^{'}})$,
    Assign $x_{j^{'}s^{'}} = 1$
    Define new allocation matrix $\boldsymbol{x^*_{js}}$
    Calculate the new matrices $\mathbf{V_s}^*$ and $\mathbf{\Pi^*_s}$ following $\boldsymbol{x^*_{js}}$
end while

A.5. Steps of Conic Reformulation of Continuous Relaxation

Let's recall the continuous relaxation for our store-specific assortment optimization model:

$$
\begin{aligned}
(FRAC) \qquad \max \quad & \sum_{s \in \mathcal{S}} \omega_s \frac{\sum_{j \in J} r_{js} v_{js} x_{js}}{v_{0s} + \sum_{j \in J} v_{js} x_{js}} \\
s.t. \quad & \sum_{s \in S} x_{js} \le q_j, \quad \forall j \in J \\
& \sum_{j \in J} x_{js} \le k_s, \quad \forall s \in S \\
& 0 \le x_{js} \le 1, \quad \forall j \in J, \quad \forall s \in S.
\end{aligned}
\tag{24}
$$

First, we introduce $y_s = 1/(v_{0s} + \sum_{j \in J} v_{js} x_{js})$ to our model:

$$
\begin{aligned}
(FRAC^{'}) \qquad \max \quad & \sum_{s \in \mathcal{S}} \omega_s \sum_{j \in J} r_{js} v_{js} y_s x_{js} \\
s.t. \quad & \sum_{s \in S} x_{js} \le q_j, \quad \forall j \in J \\
& \sum_{j \in J} x_{js} \le k_s, \quad \forall s \in S \\
& v_{0s} y_s + \sum_{j \in J} v_{js} y_s x_{js} = 1, \quad \forall s \in S \\
& y_s \ge 0, \quad \forall s \in S \\
& 0 \le x_{js} \le 1, \quad \forall j \in J, \quad \forall s \in S.
\end{aligned}
\tag{25}
$$

We linearize the bilinear terms $y_s x_{js}$ in the formulation using the standard *big-M* approach. We introduce the new continuous variable $z_{js} = y_s x_{js}$, and we use $1/v_{0s}$ as an upper bound (M) on y_s for all $s \in S$. Then, we have the following LP:

$$
\begin{aligned}
(FRAC'') \qquad & \max \quad \sum_{s\in\mathcal{S}} \omega_s \sum_{j\in J} r_{js} v_{js} z_{js} \\
& s.t. \quad \sum_{s\in S} x_{js} \le q_j, \quad \forall j\in J \\
& \sum_{j\in J} x_{js} \le k_s, \quad \forall s\in S \\
& v_{0s} y_s + \sum_{j\in J} v_{js} z_{js} = 1, \quad \forall s\in S \\
& v_{0s}(y_s - z_{js}) \le 1 - x_{js}, \quad \forall j\in J, \quad \forall s\in S \\
& 0 \le z_{js} \le y_s, \quad \forall j\in J, \quad \forall s\in S \\
& v_{0s} z_{js} \le x_{js}, \quad \forall j\in J, \quad \forall s\in S \\
& y_s \ge 0, \quad \forall s\in S \\
& 0 \le x_{js} \le 1, \quad \forall j\in J, \quad \forall s\in S.
\end{aligned} \tag{26}
$$

In order to introduce the conic formulation, we restate the objective as minimization. To do that, we define $\bar{r}_s = max_{j\in J}\, r_{js}\ \forall s\in S$, and rewrite the objective function as:

$$
max \sum_{s\in S} \omega_s \bar{r}_s - \sum_{s\in S} \omega_s \left[\frac{v_{0s}\bar{r}_s + \sum_{j\in J} v_{js}(\bar{r}_s - r_{js}) x_{js}}{v_{0s} + \sum_{j\in J} v_{js} x_{js}} \right] \tag{27}
$$

As the first component in (27) is constant, we can pose the problem as minimizing the second component. Also, since the objective coefficients are nonnegative, it suffices to use only lower bounds on y_s and z_{js} variables, leading to:

$$
\begin{aligned}
(FRAC-CONIC) \qquad & \min \quad \sum_{s\in S} \omega_s v_{0s} \bar{r}_s y_s + \sum_{s\in S} \omega_s \sum_{j\in J} v_{js}(\bar{r}_s - r_{js}) z_{js} && (28) \\
& s.t. \quad \sum_{s\in S} x_{js} \le q_j, \quad \forall j\in J && (29) \\
& \sum_{j\in J} x_{js} \le k_s, \quad \forall s\in S && (30) \\
& z_{js} \ge y_s x_{js}, \quad \forall j\in J, \quad \forall s\in S && (31) \\
& y_s \ge \frac{1}{v_{0s} + \sum_{j\in J} v_{js} x_{js}}, \quad \forall s\in S && (32) \\
& z_{js} \ge 0, \quad \forall j\in J, \quad \forall s\in S && (33) \\
& y_s \ge 0, \quad \forall s\in S && (34) \\
& 0 \le x_{js} \le 1, \quad \forall j\in J, \quad \forall s\in S. && (35)
\end{aligned}
$$

By definition, (31) and (32) are satisfied at equality at an optimal solution. Then, we can define

$$V_s = v_{0s} + \sum_{j \in J} v_{js} x_{js} \tag{36}$$

and observing that $V_s \geq 0 \, \forall s \in S$, we can rewrite constraint (32) in rotated cone form:

$$y_s V_s \geq 1. \tag{37}$$

We also use the constraints

$$v_{0s} y_s + \sum_{j \in J} v_{js} z_{js} \geq 1, \, \forall s \in S \tag{38}$$

to strengthen the continuous relaxation of the formulation following the discussion of Şen et al. (2018). Then, our final conic formulation for the continuous relaxation of the assortment optimization model is:

$$\begin{aligned}
(CONIC) \qquad \min \quad & \sum_{s \in S} \omega_s v_{0s} \bar{r}_s y_s + \sum_{s \in S} \omega_s \sum_{j \in J} v_{js} (\bar{r}_s - r_{js}) z_{js} \\
s.t. \quad & \sum_{s \in S} x_{js} \leq q_j, \quad \forall j \in J \\
& \sum_{j \in J} x_{js} \leq k_s, \quad \forall s \in S \\
& V_s = v_{0s} + \sum_{j \in J} v_{js} x_{js}, \quad \forall s \in S \\
& z_{js} \geq y_s x_{js}, \quad \forall j \in J, \quad \forall s \in S \\
& y_s V_s \geq 1, \quad \forall s \in S \\
& v_{0s} y_s + \sum_{j \in J} v_{js} z_{js} \geq 1, \forall s \in S \\
& z_{js} \geq 0, \quad \forall j \in J, \quad \forall s \in S \\
& y_s \geq 0, \quad \forall s \in S \\
& 0 \leq x_{js} \leq 1, \quad \forall j \in J, \quad \forall s \in S.
\end{aligned} \tag{39}$$

A.6. Mixed Integer Conic Formulation

Under the integer assignment variable assumption, i.e. when $x_{js} \in \{0, 1\}$, we can further improve the conic formulation by adding rotated second-order cone constraints instead of the mixed quadratic constraints. We follow the approach of Şen et al. (2018). Specifically, we introduce $z_{js} V_s \geq x_{js}^2$ instead of (39) as the underlying assumption $x_{js} = x_{js}^2$ holds. Then, the resulting MILP formulation is:

$$
\begin{aligned}
(CONIC) \qquad \min \quad & \sum_{s\in S}\omega_s v_{0s}\bar{r}_s y_s + \sum_{s\in S}\omega_s \sum_{j\in J} v_{js}(\bar{r}_s - r_{js}) z_{js} \\
s.t. \quad & \sum_{s\in S} x_{js} \le q_j, \quad \forall j \in J \\
& \sum_{j\in J} x_{js} \le k_s, \quad \forall s \in S \\
& V_s = v_{0s} + \sum_{j\in J} v_{js} x_{js}, \quad \forall s \in S \\
& z_{js} V_s \ge x_{js}^2, \quad \forall j \in J, \quad \forall s \in S \\
& y_s V_s \ge 1, \quad \forall s \in S \\
& v_{0s} y_s + \sum_{j\in J} v_{js} z_{js} \ge 1, \forall s \in S \\
& z_{js} \ge 0, \quad \forall j \in J, \quad \forall s \in S \\
& y_s \ge 0, \quad \forall s \in S \\
& x_{js} \in \{0,1\}, \quad \forall j \in J, \quad \forall s \in S.
\end{aligned} \tag{40}
$$

A.7. Proof of Theorem 3

Problem Setting. Consider m stores, each with optimal integer assortments $\mathcal{S}_1, \ldots, \mathcal{S}_m$, yielding revenues $\frac{A_1}{B_1}, \ldots, \frac{A_m}{B_m}$. We allocate an additional product with total capacity q, marginal revenues $r_1, \ldots, r_m$, store-product attractiveness parameters $v_1, \ldots, v_m$, and that there are sufficient store capacities for all stores.

Define:

$$D_i := r_i - \frac{A_i}{B_i}, \quad \forall i.$$

Step 1: Fractional Optimal Objective

The fractional objective is:

$$OBJ_{frac} = \sum_{i=1}^{m} \frac{A_i}{B_i} + \underbrace{\sum_{i=1}^{m} \frac{v_i x_i D_i}{B_i + v_i x_i}}_{f(x_1,\ldots,x_m)}$$

following the solution to:

$$\max_{x_i \ge 0,\, x_i \le 1} f(\boldsymbol{x}), \quad \text{subject to} \quad \sum_{i=1}^{m} x_i = q.$$

Equivalently:

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} \frac{D_i x_i}{\beta_i + x_i}, \quad \text{where } \beta_i = \frac{B_i}{v_i}.$$

The Lagrangian is:

$$\mathcal{L}(x, \lambda, \mu) = \sum_{i=1}^{m} \frac{D_i x_i}{\beta_i + x_i} + \lambda\left(\sum_{i=1}^{m} x_i - q\right) + \sum_{i=1}^{m} \mu_i x_i$$

KKT stationarity (assuming all stores received a non-zero allocation, i.e. $\mu_i = 0, \forall i$) gives:

$$\frac{\partial \mathcal{L}}{\partial x_i} = \frac{D_i \beta_i}{(\beta_i + x_i)^2} - \lambda = 0, \quad \forall i.$$

Solving explicitly for x_i gives:

$$x_i^* = \sqrt{\frac{D_i \beta_i}{\lambda}} - \beta_i$$

Enforce $\sum_i x_i = q$ to solve for λ: Solve for $\sqrt{\lambda}$ explicitly:

$$\frac{1}{\sqrt{\lambda}} \sum_{i=1}^{m} \sqrt{D_i \beta_i} = q + \sum_{i=1}^{m} \beta_i \quad \Rightarrow \quad \sqrt{\lambda} = \frac{\sum_{i=1}^{m} \sqrt{D_i \beta_i}}{q + \sum_{i=1}^{m} \beta_i}.$$

Thus,

$$\lambda^* = \frac{\left(\sum_{i=1}^{m} \sqrt{D_i \beta_i}\right)^2}{\left(q + \sum_{i=1}^{m} \beta_i\right)^2}.$$

and

$$x_i^* = \frac{\sqrt{D_i \beta_i} \left(q + \sum_{j=1}^{m} \beta_j\right)}{\sum_{j=1}^{m} \sqrt{D_j \beta_j}} - \beta_i$$

which corresponds to:

$$\begin{aligned} f(\boldsymbol{x}^*) &= \sum_{i=1}^{m} \frac{D_i \sqrt{D_i \beta_i} \left(q + \sum_{j=1}^{m} \beta_j\right) - D_i \beta_i \sum_{j=1}^{m} \sqrt{D_j \beta_j}}{\sqrt{D_i \beta_i} \left(q + \sum_{j=1}^{m} \beta_j\right)} \\ &= \sum_{i=1}^{m} D_i - \frac{\left(\sum_{j=1}^{m} \sqrt{D_j \beta_j}\right)^2}{q + \sum_{j=1}^{m} \beta_j} \end{aligned}$$

Step 2: Integer Optimal Objective

The integer objective selects the q stores with largest

$$\lambda_i = \frac{D_i}{\beta_i + 1}.$$

$$OBJ_{int} = \sum_{i=1}^{m} \frac{A_i}{B_i} + \sum_{i \in \{q \text{ stores with largest } \lambda_i\}} \lambda_i.$$

Step 3: Define the rounding bound Δ

The ratio $\dfrac{OBJ_{frac}}{OBJ_{int}}$ reaches its worst case when $\sum_{i=1}^{m} \dfrac{A_i}{B_i} = 0$, and the ratio $\Delta := \dfrac{f(x^*)}{OBJ_{int}}$ attains its maximum, which occurs when all λ_i are equal — see the proof below.

Proof. Consider increasing the margin of store k:

$$\lambda_k \to \lambda_k + \epsilon \quad \Rightarrow \quad D_k \to D_k + \epsilon(\beta_k + 1)$$

while other stores' margins remain unchanged.

Case 1: $k \notin \{q \text{ stores with largest } \lambda_i\}$

- Integer objective remains: $OBJ'_{int} = OBJ_{int}$
- Fractional objective increases:

$$f(\boldsymbol{x}_\epsilon^*) \geq f(\boldsymbol{x}^*) + \mu \quad \text{for some } \mu > 0$$

  as the function $f(x_i)$ is strictly increasing and concave over $x_i \geq 0$. $f_i'(x_i) = \frac{D_i\beta_i}{(\beta_i+x_i)^2} > 0$ and $f_i''(x_i) = -\frac{2D_i\beta_i}{(\beta_i+x_i)^3} < 0$
- Ratio becomes:

$$\Delta' = \frac{f(\boldsymbol{x}^*) + \mu}{OBJ_{int}} > \Delta$$

Case 2: $k \in \{q \text{ stores with largest } \lambda_i\}$

- Integer objective increases: $OBJ'_{int} = OBJ_{int} + \epsilon$
- New fractional objective must be at least as good as the previous optimal solution under perturbation. Hence, we can cap the fractional objective change:

$$\Delta f^* = \epsilon \cdot \frac{(\beta_k + 1)x_k}{\beta_k + x_k} < \epsilon$$

- Ratio becomes:

$$\Delta' = \frac{f(\boldsymbol{x}^*) + \Delta f^*}{OBJ_{int} + \epsilon} < \frac{f(\boldsymbol{x}^*)}{OBJ_{int}} = \Delta$$

Hence, if $\exists i, j$ such that $\lambda_i \neq \lambda_j$, then increasing λ_k can either increase or decrease Δ, depending on whether k is among the q stores with the largest λ_i. Therefore, Δ cannot be maximized under unequal λ_i, since any non-uniform distribution admits a perturbation that increases Δ. Only when all λ_i are equal is no such improving perturbation possible. Thus, the maximum ratio is achieved when all margins are equal. □

Step 4: Bound Δ Under Parametric Conditions

$$\Delta = \frac{f(\boldsymbol{x}^*)}{OBJ_{int}} = \left[\sum_{i=1}^{m} D_i - \frac{\left(\sum_{j=1}^{m}\sqrt{D_j\beta_j}\right)^2}{q + \sum_{j=1}^{m}\beta_j}\right] \cdot \frac{1}{q\lambda}$$

Given $\lambda = \frac{D_i}{\beta_i+1}$ is constant ($\lambda_i = \lambda$), we substitute $D_i = \lambda(\beta_i + 1)$ into the original expression for Δ:

$$\Delta = \left[\lambda\sum_{i=1}^{m}(\beta_i+1) - \frac{\lambda\left(\sum_{j=1}^{m}\sqrt{\beta_j(\beta_j+1)}\right)^2}{q + \sum_{j=1}^{m}\beta_j}\right] \cdot \frac{1}{q\lambda} = \frac{1}{q}\left[\left(m + \sum_{i=1}^{m}\beta_i\right) - \frac{\left(\sum_{j=1}^{m}\sqrt{\beta_j(\beta_j+1)}\right)^2}{q + \sum_{j=1}^{m}\beta_j}\right]$$

The λ terms cancel, yielding an expression purely in terms of β_i and q. Hence, we obtain the parametric upper bound:

$$\Delta \leq f(\beta_1, \ldots, \beta_m) = \frac{1}{q}\left(m + \sum_{i=1}^{m}\beta_i - \frac{\left(\sum_{i=1}^{m}\sqrt{\beta_i(\beta_i+1)}\right)^2}{q + \sum_{i=1}^{m}\beta_i}\right).$$

□

A.8. Proofs of Corollaries 1 and 2

To establish a pessimistic (worst-case) upper bound on Δ, we consider the following optimization problem:

$$\max_{\beta_i} \sum_{i=1}^{m} \sqrt{\beta_i(\beta_i+1)} \quad \text{subject to} \quad \sum_{i=1}^{m} \beta_i = \beta_{\text{tot}},$$

where β_{tot} is the parameter representing the sum of all β_i $\forall i$.

Since $\sqrt{\beta_i(\beta_i+1)}$ is increasing in β_i, the maximum occurs when all of β_{tot} is allocated to a single term:

$$\begin{aligned} f(\beta_{\text{tot}}, 0, \ldots, 0) &= \frac{1}{q}\left(m + \beta_{\text{tot}} - \frac{\beta_{\text{tot}}(\beta_{\text{tot}}+1)}{q+\beta_{\text{tot}}}\right) \\ &= \frac{mq + (m+q-1)\beta_{\text{tot}}}{q(q+\beta_{\text{tot}})} \end{aligned}$$

When $q = 1$, we recover the bound m. For $q = m$, the bound can be analyzed in two regimes: if $\beta_{\text{tot}} \gg m$, the bound approximates $2 - \frac{1}{m}$; if $\beta_{\text{tot}} \ll m$, the bound approaches 1. In general, for fixed β_{tot}, the bound decreases as q increases.

□

Similarly, to determine an optimistic (best-case) upper bound on Δ, we consider:

$$\min_{\beta_i} \sum_{i=1}^{m} \sqrt{\beta_i(\beta_i+1)} \quad \text{subject to} \quad \sum_{i=1}^{m} \beta_i = \beta_{\text{tot}}.$$

The minimum occurs when β_i is evenly distributed among all m terms:

$$\begin{aligned} f\left(\frac{\beta_{\text{tot}}}{m}, \ldots, \frac{\beta_{\text{tot}}}{m}\right) &= \frac{1}{q}\left(m + \beta_{\text{tot}} - \frac{m^2 \frac{\beta_{\text{tot}}}{m}\left(\frac{\beta_{\text{tot}}}{m}+1\right)}{q+\beta_{\text{tot}}}\right) \\ &= \frac{m+\beta_{\text{tot}}}{q+\beta_{\text{tot}}}. \end{aligned}$$

When $q = m$, the bound equals 1, which is intuitive: when there is enough budget to allocate to all stores, the fractional and integer solutions coincide. When $q = 1$, the bound matches the upper-bound result: it approaches m as $\beta_{\text{tot}} \ll m$ and converges to 1 as β_{tot} increases.

□

A.9. Closed-Form Functions under Mixed-MNL Demand

We present the total expected revenue under mixed-MNL type consumer choice as follows:

$$\sum_{s\in S} \alpha_s \sum_{c\in \mathcal{C}} \gamma_c \frac{\sum_{j\in J} r_{js} v_{js}^c x_{js}}{v_{0s} + \sum_{j\in J} v_{js}^c x_{js}} \tag{41}$$

Then, the profit regarding one type of consumer in one store is:

$$\Pi_s^c(x_{js}) = \gamma_c \frac{\sum_{j\in J} r_{js} v_{js}^c x_{js}}{v_{0s} + \sum_{j\in J} v_{js}^c x_{js}} \tag{42}$$

We can define

$$V_s^c(x_{js}) = v_{0s} + \sum_{j\in J} v_{js}^c x_{js} \tag{43}$$

Then, the first-order gradient function of (18) is:

$$W_{js}^c(x_{js}) = \frac{v_{js}^c}{V_s^c}(\alpha_s \gamma_c r_{js} - \Pi_s^c). \tag{44}$$

A.10. Conic Formulations under Mixed-MNL Demand

Following the discussion in A.5, the mixed quadratic conic formulation for the continuous relaxation of the assortment optimization model under mixed MNL type consumer choice is:

$$\begin{aligned}
(Mixed-CONIC) \qquad \min \quad & \sum_{s\in S} \omega_s v_{0s}^c \bar{r}_s \sum_{c\in\mathcal{C}} \gamma_c y_s^c + \sum_{s\in S} \omega_s \sum_{c\in\mathcal{C}} \gamma_c \sum_{j\in J} v_{js}^c (\bar{r}_s - r_{js}) z_{js}^c \\
s.t. \quad & \sum_{s\in S} x_{js} \le q_j, \quad \forall j \in J \\
& \sum_{j\in J} x_{js} \le k_s, \quad \forall s \in S \\
& V_s^c = v_{0s}^c + \sum_{j\in J} v_{js}^c x_{js}, \quad \forall s \in S, \quad \forall c \in \mathcal{C} \\
& z_{js}^c \ge y_s^c x_{js}, \quad \forall j \in J, \quad \forall s \in S, \quad \forall c \in \mathcal{C} \\
& y_s^c V_s^c \ge 1, \quad \forall s \in S., \quad \forall c \in \mathcal{C} \\
& v_{0s}^c y_s^c + \sum_{j\in J} v_{js}^c z_{js} \ge 1, \forall s \in S, \quad \forall c \in \mathcal{C} \\
& z_{js}^c \ge 0, \quad \forall j \in J, \quad \forall s \in S, \quad \forall c \in \mathcal{C} \\
& y_s^c \ge 0, \quad \forall s \in S, \quad \forall c \in \mathcal{C} \\
& 0 \le x_{js} \le 1, \quad \forall j \in J, \quad \forall s \in S.
\end{aligned} \tag{45}$$

Mixed Integer Conic Formulation under Mixed MNL Demand:

$$
\begin{aligned}
(Mixed-MILP) \qquad \min \quad & \sum_{s\in S}\omega_s v_{0s}^c \bar{r}_s \sum_{c\in\mathcal{C}}\gamma_c y_s^c + \sum_{s\in S}\omega_s \sum_{c\in\mathcal{C}}\gamma_c \sum_{j\in J} v_{js}^c(\bar{r}_s - r_{js}) z_{js}^c \\
s.t. \quad & \sum_{s\in S} x_{js} \le q_j, \quad \forall j\in J \\
& \sum_{j\in J} x_{js} \le k_s, \quad \forall s\in S \\
& V_s^c = v_{0s}^c + \sum_{j\in J} v_{js}^c x_{js}, \quad \forall s\in S, \quad \forall c\in\mathcal{C} \\
& z_{js}^c V_s^c \ge x_{js}^2, \quad \forall j\in J, \quad \forall s\in S, \quad \forall c\in\mathcal{C} \\
& y_s^c V_s^c \ge 1, \quad \forall s\in S., \quad \forall c\in\mathcal{C} \\
& v_{0s}^c y_s^c + \sum_{j\in J} v_{js}^c z_{js} \ge 1, \forall s\in S, \quad \forall c\in\mathcal{C} \\
& z_{js}^c \ge 0, \quad \forall j\in J, \quad \forall s\in S, \quad \forall c\in\mathcal{C} \\
& y_s^c \ge 0, \quad \forall s\in S, \quad \forall c\in\mathcal{C} \\
& 0 \le x_{js} \le 1, \quad \forall j\in J, \quad \forall s\in S.
\end{aligned} \tag{46}
$$

B. Appendix: Complementary Analyses

B.1. Robustness of Counterfactual Analysis

In this subsection, we repeat the same analysis we follow for T-shirts category in Section 6 for the products of dresses category. We present the descriptive statistics regarding the dresses in Table 6.

Table 6: Descriptive statistics of data regarding the dresses.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
Variables							
logSales$_{js}$	72,252	0.29	0.83	0.00	0.00	0.00	5.36
logInventory$_{js}$	72,252	0.33	0.70	0.00	0.00	0.00	4.86
P_j	892	0.50	0.29	0.01	0.26	0.75	1.00
I_s	81	0.48	0.28	0.00	0.26	0.69	1.00
Allocated$_{js}$	72,252	0.68	0.46	0.00	0.00	1.00	1.00
Direct measurements							
Store potential (ω_s)	81	320,464	153,543	70,215	223,613	400,289	864,520
Price (r_{js})	72,252	26.77	13.75	4.99	17.99	32.95	193.49
Product capacity (q_j)	892	56	9	20	48	65	79

Descriptive statistics are comparable to those presented in Table 3. We move forward with the first stage estimations, where we follow Equations (21) and 22. We present the results in Table 7.

Similar to the results of the main estimation in Table 4, we obtain insignificant estimation coefficients for A_{js}, which means the current policy does not regard the product-store matching for this category of

Table 7: First stage OLS estimation results for dresses category.

	Dependent variable: Allocated$_{js}$	
	Static Products	*Dynamic Products*
StoreConsidered$_{js}$	0.11^{***}	
	(0.00)	
StorePreviousPercentile$_{js}$		0.53^{***}
		(0.07)
A_{js}	0.04	0.11
	(0.07)	(0.05)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Num. obs.	$27,378$	$44,874$
R^2 (full model)	0.47	0.48

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

products as well. Also, we get positive and significant coefficients for the allocation policy variables whose magnitudes are also very close to those presented in Table 4.

We follow Equation (20) for the second stage estimation and present the results of both single stage and two-stages estimations in Table 8.

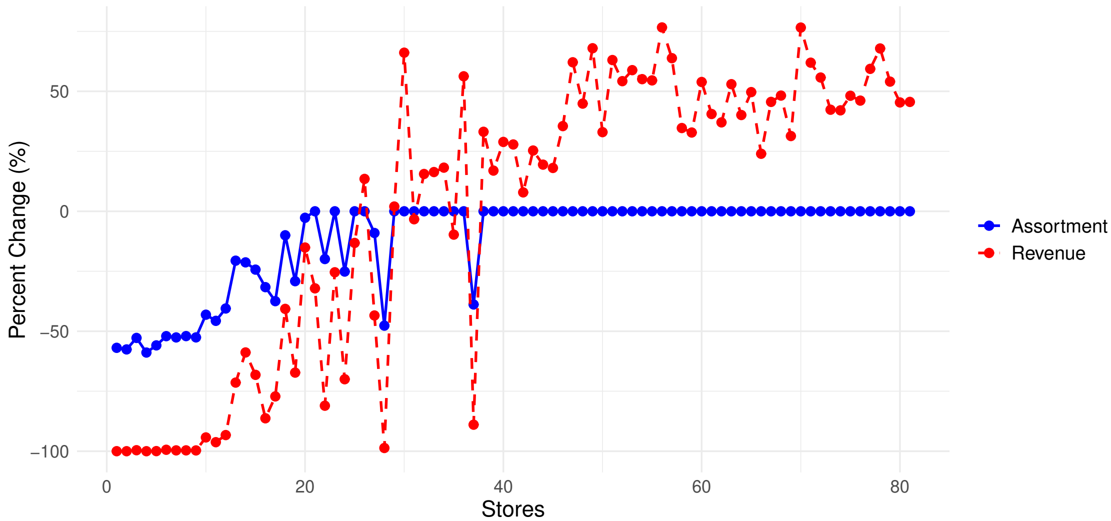
Table 8: Second stage MNL estimation results regarding dresses category.

	Dependent variable: Conversion$_{js}$	
	Single stage	1^{st}*stage OLS*
A_{js}	0.437***	0.289***
	(0.001)	(0.001)
Allocated$_{js}$	3.510***	4.947***
	(0.000)	(0.000)
Residual$_{js}$		-1.443***
		(0.001)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Number of observations	72,252	72,252
Number of variables	975	976
Log likelihood	-2469108	-2468906
Pseudo R^2	0.11	0.11

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

All of the estimation coefficients in Table 8 follow the same significance and sign with the main estimation presented in Table 5. Moreover, the magnitudes of these coefficients are very similar, which strengthens the main estimation results.

Figure 10: Change in assortment size and revenue per store following the new allocation after the algorithms were applied to dresses.



Then, when we apply our IF to products of dresses category, estimating the respective v_{js} parameters using the results in Table 8, and observe a potential improvement of 15.7% in total expected revenue. The difference in the magnitude of improvement between T-shirts and dresses is primarily due to the retailer's unusual return policy, which is more prevalent for dresses. This policy inadvertently leads to more unintentional but effective product-store matches within the category.

We observe that the majority of the improvement occurs in stores that already have high revenue. This outcome is driven by two key factors. First, our estimation of v_{js} parameters favors better-performing stores, as higher-performing stores are more likely to have higher estimated parameters. Second, the greedy nature of our algorithm prioritizes swaps that match higher product performance with higher store performance, effectively aligning product price with the income level of the store's customer base.

Moreover, we repeat the visualization of how assortment size and expected revenue per store changes after the reallocation of dresses in Figure 10.

We observe the same trend, where the majority of the improvement occurs in stores that already have high revenue, and there is some store starvation around low-revenue stores. This effect seems to be more pronounced for dresses, which we attribute to the greater heterogeneity in product performance within this category.

B.2. Estimation based on panel data analysis

Following the discussion in section 6, we replicate our estimation using panel data that spans across stores, products and weeks. On the top of Table 3, we present the descriptive statistics of panel data specific variables in Table 9.

Table 9: Descriptive statistics of panel specific variables.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
logSales$_{jst}$	2,460,060	0.03	0.19	0.00	0.00	0.00	3.85
logInventory$_{jst}$	2,460,060	0.23	0.60	0.00	0.00	0.00	4.71

Similar to our previous analysis, we assume consumers make a discrete choice between products and stores along with a non-purchase option. However, for this analysis, we further assume that they repeat this discrete-choice for each week. Particularly, the function (47) represents the utility of that a consumer i obtains from buying product j at store s during week t:

$$U_{ijst} := \alpha_j + \alpha_s^{'} + \alpha_t^{''} + \boldsymbol{\beta X_{jst}} + \varepsilon_{ijst}. \tag{47}$$

where α_j, α_s', and $\alpha_t^{''}$ are product, store, and time fixed effects respectively, $\boldsymbol{X_{jst}}$ are the covariates of interest and ε_{ijst} is a Gumbel-distributed shock. We aim to capture intrinsic attractiveness of products and stores with α_j and α_s', whereas $\alpha_t^{''}$ captures seasonality. We normalize to zero the value of the outside option, i.e., visiting store s at week t, and not buying any product generates a Gumbel utility ε_{i0st}. As a result, the choice probability of a consumer i purchasing product j from store s can be written as:

$$\mathbb{P}(Purchase_{ijst}) = \frac{e^{\alpha_j + \alpha_s' + \alpha_t^{''} + \boldsymbol{\beta X_{jst}}}}{1 + \sum_{k \in A_{st}} e^{\alpha_k + \alpha_s' + \alpha_t^{''} + \boldsymbol{\beta X_{kst}}}}, \tag{48}$$

where A_{st} represents the assortment set in store s at time t.

As we do not possess individual level data, we perform an aggregate analysis where we focus on the conversion rate regarding each product in each store at each week. More specifically, we calculate $Conversion_{jst} = Sales_{jst}/Footfall_{st}$, to represent the probability distribution regarding consumer choice for every store in each week. Note that $\sum_{j \in \mathcal{J} \cup \{0\}} Conversion_{jst} = 1, \forall s \in \mathcal{S}, t \in \mathcal{T}$, hence the weekly conversion rates in a store are indeed a probability distributions.

Thus, our estimation follows:

$$\begin{aligned} Conversion_{jst} \sim MNL\Big(& \underbrace{\alpha_j + \alpha_s'}_{\text{Product and store heterogeneity}} + \underbrace{\alpha_t^{''}}_{Seasonality} \\ & + \underbrace{\beta A_{js}}_{\text{Interaction between product and store types}} \\ & + \underbrace{\gamma \, logInventory_{jst}}_{Servicelevelprovided} \Big) \end{aligned} \tag{49}$$

where $A_{js} := P_j \times I_s$ following the previous discussion. As the literature shows that the inventory levels effect the consumer choice (Boada-Collado and Martínez-de Albéniz 2020), we introduce

Table 10: First stage OLS estimation results for the panel data estimation.

	Dependent variable: logInventory$_{jst}$	
	Static Products	*Dynamic Products*
StoreConsidered$_{js}$	0.20^{***}	
	(0.05)	
StorePreviousPercentile$_{js}$		0.09
		(0.20)
A_{js}	0.27	0.20
	(0.12)	(0.16)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Week FEs	Yes	Yes
Num. obs.	$428,970$	$2,031,090$
R^2	0.31	0.27

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

$logInventory_{jst} := log(Inventory_{jst} + 1)$, where $Inventory_{jst}$ is the inventory level of product j at store s at week t. This variable accounts for the provided service level of different products in different stores as a consequence of the replenishment decisions, hence accounting for the allocation decision.

Before estimating MNL demand regarding consumer choice, we have to introduce new first stage estimation equations as the threats to estimation that we discussed in section 6 persist for this setting. For the sake of coherency, we use the same panel data dimensions for the first stage estimation. Under this setting, $logInventory_{jst}$ becomes our endogenous variable as the no inventory would directly mean no sales. Hence, we introduce (50) and (51), where we control for seasonality using $\alpha_t^{''}$, alongside previously introduced α_j and $\alpha_s^{'}$.

$$logInventory_{jst} = \alpha_j + \alpha_s^{'} + \alpha_t^{''} + StoreConsidered_{js} + A_{js} + \varepsilon_{js} \tag{50}$$

$$logInventory_{jst} = \alpha_j + \alpha_s^{'} + \alpha_t^{''} + StorePreviousPercentile_{js} + A_{js} + \varepsilon_{js} \tag{51}$$

We present the first stage estimation results regarding (50) and (51) in Table 10.

We observe these results follow the same sign and significance with our main estimation model. Furthermore, we present the second stage demand estimation results in Table 11.

We observe that main results are coherent with our estimation presented in section 6, and we find a positive and statistically significant coefficient for weekly inventory. This estimation is coherent with the literature (Boada-Collado and Martínez-de Albéniz 2020) regarding both estimations. More importantly,

Table 11: Second stage MNL estimation results for the panel data estimation.

	Dependent variable: Conversion$_{jst}$	
	Single stage	1^{st}*stage OLS*
A_{js}	0.462***	0.358***
	(0.002)	(0.002)
logInventory$_{jst}$	0.832***	1.564***
	(0.000)	(0.000)
ExpectedError$_{js}$		-0.724***
		(0.000)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Date FEs	Yes	Yes
Number of observations	2,460,060	2,460,060
Number of variables	1,152	1,153
Log likelihood	-1670618	-1670619
Pseudo R^2	0.02	0.02

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

we observe a positive and significant coefficient for A_{js} similar to our main estimation presented in Table (5). However, we find that the average marginal effect of this matching is much higher than what the main estimation suggests. We conjecture that once the effect of seasonality was considered with the panel data estimation, products perform much better post-allocation in the stores that follow similar monetary characteristics.

B.3. Analyses on alternative notions of compatibility between stores and products

We repeat our empirical analyses by introducing different definitions for A_{js} as mentioned in section 6. For all the alternatives we introduce in this subsection, we follow the same first stage estimations (21) and (22), along with the same second stage estimation (20).

Particularly, we define three alternative definitions to capture the compatibility of stores and products. First, we define $A^{'}_{js} := V_j : I_s$, where we keep the previously introduced measure of socioeconomic wealth – I_s, and we introduce V_j as the popularity of product j. More specifically, V_j is a percentile measure of total production volume of the respective product. We present the respective first stage estimation results in Table 12, and the second stage estimation results in Table 13.

We are able to replicate our results in terms of the sign and the significance of our estimation coefficients, although the magnitudes of this estimation are expectantly different. This means that popular products (manufactured in large volume) perform better in stores located in socioeconomically richer areas. This is an interesting result, as basic products are usually produced in large volumes, and they tend to be less expensive. We conjecture this with the fact that potential higher income consumers are more likely to purchase in general. Also, estimation coefficient of $A^{'}_{js}$ is smaller in magnitude than A_{js} – see Table 5,

Table 12: First stage OLS estimation results regarding $A^{'}_{js}$.

	Dependent variable: Allocated$_{js}$	
	Static Products	*Dynamic Products*
StoreConsidered$_{js}$	0.11^{***}	
	(0.01)	
StorePreviousPercentile$_{js}$		-0.01
		(0.12)
$A^{'}_{js}$	0.09	0.01
	(0.04)	(0.02)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Num. obs.	$14,661$	$69,417$
R^2 (full model)	0.49	0.45

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Table 13: Second stage MNL estimation results regarding $A^{'}_{js}$.

	Dependent variable: Conversion$_{js}$	
	Single stage	1^{st}*stage OLS*
$A^{'}_{js}$	0.151^{***}	0.143^{***}
	(0.004)	(0.004)
Allocated$_{js}$	5.079^{***}	5.471^{***}
	(0.000)	(0.000)
Residual$_{js}$		-0.506^{***}
		(0.001)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Number of observations	84,078	84,078
Number of variables	1,121	1,122
Log likelihood	-2826704	-2826640
Pseudo R^2	0.11	0.11

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

which suggests that doing the matching between products and stores monetary measures of both dimensions is economically more significant.

Second, we introduce $A^{''}_{js} := P_j : F_s$, where we keep the previous measure of product price – P_j, and we define F_s as the market share of store s. Particularly, F_s is a percentile measure of total footfall observed in store s throughout the time horizon. We present the results of the first stage of this estimation in Table 14, and Table 15 shows the results of the second stage estimation.

Our results show that there is a *negative* and significant coefficient for $A^{''}_{js}$. Then, expensive products perform worse post-allocation in popular stores. This is an expected result given the fast-fashion retail

Table 14: First stage OLS estimation results regarding A''_{js}.

	Dependent variable: Allocated$_{js}$	
	Static Products	*Dynamic Products*
StoreConsidered$_{js}$	0.11^{***}	
	(0.00)	
StorePreviousPercentile$_{js}$		-0.01
		(0.13)
A''_{js}	0.02	0.13
	(0.07)	(0.11)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Num. obs.	$14,661$	$69,417$
R^2 (full model)	0.48	0.45

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Table 15: Second stage MNL estimation results regarding A''_{js}.

	Dependent variable: Conversion$_{js}$	
	Single stage	1^{st}*stage OLS*
A''_{js}	-0.01***	-0.07***
	(0.000)	(0.001)
Allocated$_{js}$	4.965***	5.479***
	(0.000)	(0.000)
ExpectedError$_{js}$		-0.513***
		(0.000)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Number of observations	84,078	84,078
Number of variables	1,121	1,122
Log likelihood	-2826670	-2826648
Pseudo R^2	0.11	0.11

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

context. Large stores with high visitor traffic are usually located in busy areas, and their main target consumer base are likely to be the average income consumers rather than high income consumers. Also, note that in terms of magnitude of the estimation coefficient, A''_{js} has a smaller economical significance compared to A_{js}.

Lastly, we estimate using $A^*_{js} := V_j : F_s$, where V_j is the measure of product popularity and F_s is the measure of store market share as explained earlier in this section. We present the first stage estimation results in Table 16, and the results of the second stage estimation in Table 17.

Table 16: First stage OLS estimation results regarding A^*_{js}.

	Dependent variable: Allocated$_{js}$	
	Static Products	*Dynamic Products*
StoreConsidered$_{js}$	0.11^{***}	
	(0.03)	
StorePreviousPercentile$_{js}$		-0.00
		(0.13)
A^*_{js}	-0.03	-0.03
	(0.04)	(0.03)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Num. obs.	$14,661$	$69,417$
R^2 (full model)	0.48	0.45

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Table 17: Second stage estimation results regarding A^*_{js}.

	Dependent variable: Conversion$_{js}$	
	Single stage	1^{st} *stage OLS*
A^*_{js}	-0.067***	-0.046***
	(0.005)	(0.001)
Allocated$_{js}$	4.985***	5.486***
	(0.000)	(0.000)
ExpectedError$_{js}$		-0.523***
		(0.000)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Number of observations	84,078	84,078
Number of variables	1,121	1,122
Log likelihood	-2826687	-2826646
Pseudo R^2	0.11	0.11

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Similar to the analysis with A''_{js}, our results suggest that popular products perform worse post-allocation in popular stores. Although this is an unexpected finding, it aligns with the main research objective of this paper. Products should be matched with stores where they have the highest potential to improve sales probability and revenue. This does not necessarily mean allocating popular products to popular stores, as is common industry practice. Instead, greater potential lies in exploring more complex dependencies between stores and products, such as price positioning and income levels as we examine in Section 6.

Table 18: First stage logit estimation results.

	Dependent variable: Allocated$_{js}$	
	Static Products	*Dynamic Products*
StoreConsidered$_{js}$	0.80^{***}	
	(0.14)	
StorePreviousPercentile$_{js}$		-0.18
		(0.43)
A_{js}	1.60^{***}	1.69^{***}
	(0.32)	(0.13)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Num. obs.	$14,661$	$69,417$
Deviance	10588.81	54384.69
Log Likelihood	-5326.23	-27192.35

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

B.4. First stage estimation as *logit*

To accommodate our dilemma regarding the choice of estimation methodology for the first stage regressions, we repeat the analysis in §6.3 using logit regressions (52)-(53).

$$Allocated_{js} \sim LOGIT(\alpha_j + \alpha_s^{'} + A_{js} + StoreConsidered_{js}) \tag{52}$$

$$Allocated_{js} \sim LOGIT(\alpha_j + \alpha_s^{'} + A_{js} + StorePreviousPercentile_{js}) \tag{53}$$

We present the results of the estimation in Table 18.

Similar to the OLS results, our analysis confirms that the company follows its stated policy for static products, as indicated by the positive and significant coefficient of $StoreConsidered_{js}$, and does not follow its policy for dynamic products.

Surprisingly, we observe a significant effect for the interaction variable A_{js} in this case, suggesting that the current allocation policy considers our matching variable. However, given that this effect was insignificant in our main model and across multiple robustness checks, it is difficult to draw definitive conclusions from this finding. Furthermore, we present the second-stage estimation in Table 19, where we find that the interaction variable remains positive and significant, reinforcing our previous findings that the matching variable increases sales probability post-allocation. More importantly, we show that not only do both the mathematically-proven OLS and the better-fit logit estimation follow the same trend, but we also replicate the exact same estimated coefficient for the matching variable A_{js} in both approaches. After this robustness check, we feel more confident using the OLS estimation in our main analysis, given its strong mathematical foundation as an estimation technique.

Table 19: Results of second stage MNL estimation with control function, where the first stage is a *logit*.

	Dependent variable: Conversion$_{js}$
A_{js}	0.253***
	(0.000)
Allocated$_{js}$	5.479***
	(0.000)
ExpectedError$_{js}$	-0.522***
	(0.000)
Product FEs	Yes
Store FEs	Yes
Number of observations	84,078
Number of variables	1,122
Log likelihood	-2826605
Pseudo R^2	0.11

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

B.5. Estimation using inventory instead of allocation

Following the discussion in section 6, we repeat our estimation where we drop the allocation variable and introduce the inventory variable instead. Note that this should account for the unusual company policy of returns that creates *unintended* product-store allocations. We follow the exact variable definitions introduced in section 6, and offer (54) - (55) as the first stage OLS estimations, whereas (56) represents the second stage estimation.

$$logInventory_{js} = \alpha_j + \alpha_s^{'} + StoreConsidered_{js} + A_{js} + \varepsilon_{js} \tag{54}$$

$$logInventory_{js} = \alpha_j + \alpha_s^{'} + StorePreviousPercentile_{js} + A_{js} + \varepsilon_{js} \tag{55}$$

$$Conversion_{js} \sim MNL(\alpha_j + \alpha_s' + \beta A_{js} + \gamma logInventory_{js}) \tag{56}$$

We present the estimation results of the first stage estimations in Table 20, and the results of second stage estimation in Table 21.

Our results replicate in terms of sign and significance of our estimation coefficients regarding all independent variables. Furthermore, the magnitude of our two-stage estimation is coherent with what has been found in the literature (Boada-Collado and Martínez-de Albéniz 2020).

Table 20: First stage OLS estimation results with $logInventory_{js}$.

	Dependent variable: logInventory$_{js}$	
	Static Products	*Dynamic Products*
StoreConsidered$_{js}$	0.21^{***}	
	(0.04)	
StorePreviousPercentile$_{js}$		0.01
		(0.17)
A_{js}	0.17	0.19
	(0.07)	(0.08)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Num. obs.	$14,661$	$69,417$
R^2	0.44	0.40

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Table 21: Second stage estimation with $logInventory_{js}$.

	Dependent variable: Conversion$_{js}$	
	Single stage	1^{st}*stage OLS*
A_{js}	0.489***	0.308***
	(0.001)	(0.001)
logInventory$_{js}$	0.729***	1.681***
	(0.000)	(0.000)
ExpectedError$_{js}$		-0.953***
		(0.000)
Product FEs	Yes	Yes
Store FEs	Yes	Yes
Number of observations	84,078	84,078
Number of variables	1,121	1,122
Log likelihood	-2857387	-2857338
Pseudo R^2	0.10	0.10

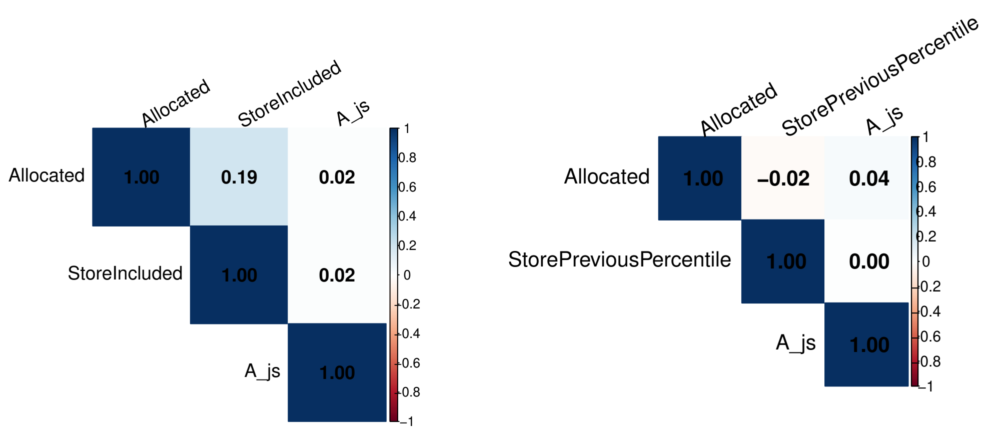
$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

B.6. Descriptive Statistics for First Stage Data

Table 22: Descriptive statistics of variables used in first stage estimations.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
Static Products							
$Allocated_{js}$	14,661	0.434	0.496	0	0	1	1
$StoreConsidered_{js}$	14,661	0.499	0.500	0	0	1	1
P_j	14,661	0.451	0.293	0.003	0.222	0.672	0.999
I_s	14,661	0.499	0.284	0.008	0.270	0.730	1.000
Dynamic Products							
$Allocated_{js}$	69,417	0.507	0.499	0	0	1	1
$StorePreviousPercentile_{js}$	69,417	0.503	0.290	0.012	0.250	0.753	1.000
P_j	69,417	0.511	0.286	0.001	0.260	0.758	1.00
I_s	69,417	0.499	0.284	0.008	0.270	0.730	1.000

Figure 11: Zero order correlations for dynamic and static products.



(a) Static products.

(b) Dynamic products.

B.7. VIFs from residualized first stage estimations

Table 23: VIFs from residualized first stage estimations.

Static Products – Estimation (21)

Variable	VIF	Tolerance (1/VIF)
StoreIncluded$_{js}$	1.005	0.995
A_{js}	1.005	0.995

Dynamic Products – Estimation (22)

Variable	VIF	Tolerance (1/VIF)
StorePreviousPercentile$_{js}$	1.001	0.998
A_{js}	1.001	0.998