Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

A Nonparametric Joint Assortment and Price Choice Model

Srikanth Jagabathula, Paat Rusmevichientong

informs
http://pubsonline.informs.org/journal/mnsc/

MANAGEMENT SCIENCE
Vol. 63, No. 9, September 2017, pp. 3128–3145
ISSN 0025-1909 (print), ISSN 1526-5501 (online)

A Nonparametric Joint Assortment and Price Choice Model

Srikanth Jagabathula,[a] Paat Rusmevichientong[b]

[a] Stern School of Business, New York University, New York, New York 10012; [b] Marshall School of Business, University of Southern California, Los Angeles, California 90089

Contact: sjagabat@stern.nyu.edu (SJ); rusmevic@marshall.usc.edu (PR)

Received: August 14, 2013
Revised: March 13, 2015; October 22, 2015
Accepted: February 4, 2016
Published Online in Articles in Advance: July 22, 2016

https://doi.org/10.1287/mnsc.2016.2491

Abstract. The selection of products and prices offered by a firm significantly impacts its profits. Existing approaches do not provide flexible models that capture the joint effect of assortment and price. We propose a nonparametric framework in which each customer is represented by a particular price threshold and a particular preference list over the alternatives. The customers follow a two-stage choice process; they consider the set of products with prices less than the threshold and choose the most preferred product from the set considered. We develop a tractable nonparametric expectation maximization (EM) algorithm to fit the model to the aggregate transaction data and design an efficient algorithm to determine the profit-maximizing combination of offer set and price. We also identify classes of pricing structures of increasing complexity, which determine the computational complexity of the estimation and decision problems. Our pricing structures are naturally expressed as business constraints, allowing a manager to trade off pricing flexibility with computational burden.

History: Accepted by Yinyu Ye, optimization.
Funding: This research was supported in part by the National Science Foundation [Grants CMMI-1454310, CMMI-1157569, and CMMI-1433396].
Supplemental Material: The online appendix is available at https://doi.org/10.1287/mnsc.2016.2491.

Keywords: nonparametric choice models • joint assortment and price optimization • EM algorithm • transaction data

1. Introduction

What products to carry and what prices to charge are important decisions faced by a firm. These decisions influence the purchase behavior of customers, and, therefore, the firm's revenues and gross profits. To effectively optimize its offerings, a firm must understand how its product and price offerings *jointly* impact consumer demand. The existing literature in the areas of marketing and operations proposes the following general procedure: (i) *model* the impact of assortment and price on consumer demand, (ii) *fit* the model to data on product availability and sales transactions, and (iii) *optimize* the assortment or price or the joint assortment and price offering using the demand predictions from the fitted model. The biggest challenge in executing the above procedure is to design a model that faithfully captures the underlying choice patterns *and* allows for tractable algorithms to fit and optimize.

Most existing proposals are based on parametric choice models, which specify particular functional forms relating product attributes (such as price) to utility values and choice probabilities. Parametric models are parsimonious and therefore computationally tractable. However, the choice structures must be prespecified, increasing the risk of model misspecification and leading to inaccuracies in decision making. The alternative is to adopt a nonparametric approach, which removes the need for explicit specification of choice structures and instead "learns" the appropriate structure from data.[1] However, the lack of parsimonious structures makes model estimation, and *particularly* optimization, computationally difficult. Existing work (e.g., Haensel and Koole 2011, Farias et al. 2013, van Ryzin and Vulcano 2014) proposes computationally efficient techniques to estimate model parameters when data consist of only assortment changes and not price changes. The literature is, however, silent on capturing the joint assortment and price changes and solving the joint assortment and price decision problem using nonparametric approaches.

The key contribution of this paper is a nonparametric approach for joint assortment and price optimization. Despite its flexibility, our model allows for both tractable estimation *and* optimization. Our framework allows managers to make a continuous trade-off between decision complexity and computational burden. We follow the general *model*, *fit*, and *optimize* procedure described above. We focus on a canonical retailer selling a universe of n products from a specific category or subcategory. There are frequent changes in prices and more frequent changes in the offer sets, either due to stock-outs or deliberate screening, which is common in the online environment. Product features, other than prices, remain fixed. The retailer has collected historical data in the form of sales transactions, product availabilities, and offered prices. The retailer must utilize available data to determine the assortment and price combination that maximizes the expected revenue. The above setting is broad and includes the classical revenue management setting for airlines, hotels, and cruises, in which the available bookings and prices change frequently.

1.1. Overview of Our Approach

1.1.1. Model. We extend the prevailing nonparametric rank-based choice model to capture the impact of price changes on demand. In a rank-based choice model, customers make choices from an offer set according to a preference list so that if the most preferred option is unavailable, they go down the list to pick an available option, as long as it is preferred over the no-purchase option. We extend the rank-based model by supposing that customers follow a two-stage procedure. In the first stage, the customer forms a *consideration set* by selecting the subset of products whose prices are less than or equal to a *price threshold*. In the second stage, she chooses the most preferred product from the chosen consideration set. Existing literature allows the customer to form a consideration set by applying broader threshold-based screening rules comprising of attributes other than price, but we focus on the price-based threshold because in our setting only prices change. Of course, neither the ranked list nor the price threshold of the individual customer is observed, so we describe them using probability mass functions (PMFs): the population is described by (a) the *threshold PMF* that describes a distribution over all possible price thresholds and (b) the *preference PMF* that describes a distribution over all possible ranked lists.

Because customer preferences are influenced by product prices, the preference PMF will be a function of price. But allowing the preference PMF to depend arbitrarily on price results in an intractable model that cannot extrapolate demand to new prices. To address this, we suppose that the domain of price vectors partitions into a "small" number K of partitions such that the preference PMF λ_k is the same for all the price vectors in partition k. In effect, this assumption supposes that price thresholds capture the immediate effects of price changes, while the ranked lists capture the residual effects. The assumption is also supported by empirical evidence from Gilbride and Allenby (2004). It results in tractable estimation because only a finite number of threshold and preference PMFs need to be estimated from data.

1.1.2. Fit. We train the model parameters on historical data that consist of sales transactions, product availabilities, and offered prices. The estimation consists of two steps: (a) clustering the price vectors to result in a partitioning of the price domain and (b) running the expectation maximization (EM) algorithm to estimate the threshold and preference PMFs for each partition separately. We use a general-purpose clustering algorithm to partition the training price vectors into K parts. We tune K using cross-validation. Given the partitioning, we fit a model to each partition by maximizing the log-likelihood function. Because the log-likelihood function is in general hard to maximize, we adopt the EM algorithm. The EM algorithm treats the price thresholds of individual observations as the latent variables and iteratively alternates between inferring the latent variables (from observed data and previous PMFs) and generating new PMF estimates from the inferred latent variables. The sequence of estimates produced can be shown to converge to a stationary point of the log-likelihood function. The key challenge in running the EM algorithm is carrying out the resulting M-step, which involves solving a convex program with $n!$ (n factorial) number of variables. Practically, we carry out the M-step by reducing it to a rank-aggregation problem and using existing heuristics (see Ali and Meilă 2012) to obtain solutions with good empirical performance. Theoretically, we propose a dynamic programming (DP) formulation to show that the solution can be obtained efficiently (in polynomial time) if we restrict the feasible prices to a structured class.

1.1.3. Optimize. Finally, we optimize the estimated model to determine the revenue-maximizing combination of assortment and prices. The decision problem is NP-hard in the strong sense (Aggarwal et al. 2004, Rusmevichientong et al. 2006). We address this challenge by proposing an approximation algorithm with a provable performance guarantee based on a DP formulation. Under appropriate technical conditions, we show that the DP admits a polynomial-time approximation scheme (PTAS). For any prespecified $\epsilon > 0$, the PTAS can determine a price vector whose revenue is within $1-\epsilon$ of the optimal revenue, using computing time that is polynomial in the number of products, for a fixed ϵ.

We analyze our method both theoretically and empirically. Theoretically, we identify pricing structures that lend tractability to both the estimation and the optimization problems. We call these the d-sorted pricing structures, the simplest of which ($d = 0$) results in product prices that respect a prespecified reference ordering.[2] These structures have received little attention in the literature. They allow us to isolate the source of complexity in solving the estimation and optimization problems. Furthermore, they are *not* esoteric mathematical structures, but naturally map to a firm's business constraints. Empirically, we test our methods both on real-world and synthetic sales transactions data. The tests with the real-world transactions from the IRI Academic Data Set demonstrate the *predictive* accuracy of our methods: an average of 26% improvement over the benchmark latent-class multinomial logit (LC-MNL) model on a chi-square metric, which measures the relative error in predicting market shares. The tests with the synthetic data demonstrate the *decision* accuracy: an average of 11% higher revenue extracted against the LC-MNL benchmark.

1.2. Literature Review

We position our work as part of the new stream of literature on choice modeling techniques designed specifically for applications in operations; see, for example, Farias et al. (2013), van Ryzin and Vulcano (2014), Alptekinoğlu and Semple (2016), and Blanchet et al. (2016). This body of work is characterized by its emphasis on the prediction accuracy rather than on the explanatory power of the models, because accurate decision making requires accurate predictions rather than accurately modeling the underlying choice process. Indeed, the objectives of producing accurate predictions and accurate modeling of underlying choice process are not equivalent (see Ebbes et al. 2011). Furthermore, our work may be viewed as a step toward extending the framework of discrete models of choice in operations to account for behavioral heuristics, such as simple screening rules (see Ben-Akiva et al. 1999). Our work has connections to the literature on choice modeling in both operations and marketing. Because our model is designed for operational decision making, we focus predominantly on the work in operations, with a brief discussion on connections with marketing at the end.

In operations, rank-based choice models have been used to model demand in the context of airline revenue management and retail operations. Traditional approaches had assumed independent demands for each product. However, if products are close substitutes and their availability changes over time, the demand for each product becomes a function of the entire offer set. As a result, interest has shifted over the last two decades from independent demand models to choice-based demand models. In airline revenue management, the work by Belobaba and Hopperstad (1999), Ratliff et al. (2008), and Vulcano et al. (2010) has demonstrated improvements in revenues from the incorporation of choice models. Retail applications of rank-based choice models have been pioneered by Mahajan and van Ryzin (1998), who considered a single-period stochastic inventory model in which customers substitute products within the assortment as inventory is depleted. The above approaches mostly focus on assortment effects on demand.

On the other hand, the research that accounts for price effects on demand has mostly focused on optimization for a given parametric choice model. Hanson and Martin (1996) showed that the profit function under the logit choice model is not jointly concave in the price vector. However, there is a one-to-one correspondence between the prices and the market shares, and it has been shown that the profit function is concave in market shares by Song and Xue (2007) and Dong et al. (2009) for the MNL model and by Li and Huh (2011) for the nested logit (NL) model. Li and Huh (2011) assumed that the price sensitivity parameters are the same across all products within the same nest. More recently, Gallego and Wang (2014) relaxed this assumption and obtained a characterization of the optimal prices.

The above body of work either ignores price effects, focusing only on assortment effects, or adopts parametric models of price. Our work differs from the above literature in considering a nonparametric approach to model the joint effect of assortment and price. Furthermore, most of the existing work focuses on optimization issues, with less emphasis on estimation. Our work addresses both estimation (from readily available transaction data) and optimization, resulting in an end-to-end solution.

Nonparametric approaches to choice modeling have been gaining traction within operations as a result of the increased availability of data. The prevailing nonparametric model describes customer purchase behavior using a general PMF on ranked lists of all the alternatives, including the no-purchase option. In the context of this general model, Farias et al. (2013) proposed tractable procedures to predict expected revenues as a function of offer sets using historical sales transaction data. In van Ryzin and Vulcano (2014), the authors proposed a "market discovery" algorithm, which constructs a preference PMF from sales data by iteratively finding ranked lists that increase the data log-likelihood value. Jagabathula (2016) considers the problem of finding the revenue-maximizing offer set under a general rank-based choice model. They propose a general local search algorithm and show that the algorithm converges to the optimal solution in several important special cases. Finally, Honhon et al. (2012) proposed efficient algorithms for assortment optimization for interesting structures of the general rank-based model. The above nonparametric approaches focus only on assortment effects of demand and do not model price effects. We extend the nonparametric rank-based demand model to also capture price effects and then solve the joint assortment and price optimization problem.

In marketing, our model has material connections to the literature on consideration sets. The idea that consumers use a two-stage decision process in which they first evaluate products to form a smaller relevant or consideration set and then make the purchase decision from this consideration set has been studied and empirically established in extensive work; see, for example, Howard and Sheth (1969), Urban (1975), Belonax and Mittlestaedt (1978), Parkinson and Reilly (1979), Alba and Chattopadhyay (1985), and Hauser and Wernerfelt (1990). In particular, there are strong modeling connections in our work to Gilbride and Allenby (2004), who demonstrated that a two-stage model in which customers pick a consideration set and then choose from the consideration set according to a discrete choice model has better in-sample and out-of-sample fit to conjoint study data on cameras, when compared to a discrete choice model alone.

2. Problem Formulation

In this section, we provide a precise description of our model and the problem formulation. We model the aggregate demand that a firm receives in response to the assortments and prices it offers. We consider a universe of n products, represented by the set $\mathcal{N} = \{a_1, a_2, \ldots, a_n\}$, and suppose that the assortment is drawn from the product universe. Let a_0 denote the no-purchase or the outside option. Customers arrive sequentially to the firm and decide to either purchase one of the offered products or leave without making a purchase, in which case we say that the customer chooses the no-purchase option a_0. We allow for stock-outs and price changes so that the offer set and prices seen by different customers may be different.

We suppose that customers use a two-stage model for making their purchase decision. In the first stage, a customer forms a *consideration set* of products by selecting the subset of offered products whose prices are less than or equal to a price threshold b. In the second stage, the customer chooses from the consideration set the most preferred product according to a preference list of the products. This model accounts for the phenomenon that customers use decision heuristics to simplify complicated decision tasks. Much of the existing work in marketing has provided empirical evidence for such a two-stage model; see, for example, Gilbride and Allenby (2004) and the references therein. Several screening rules have been studied in the existing literature. Commonly used screening rules are threshold based, resulting in the inclusion of products whose attribute values pass prespecified thresholds. Because the only attribute that changes in our setting is price, we focus on the price-based screening rule. Without loss of generality, we suppose that the latent thresholds b belong to a finite set $\mathcal{B}$.

The population-level model is described by PMFs over price thresholds and preferences. Suppose products are offered at a price vector $\mathbf{p} = (p_a\colon a \in \mathcal{N})$, with p_a denoting the price of product a and $p_{a_0} = 0$. Then, the population is described by a threshold PMF g: $\mathcal{B} \to [0,1]$ and a preference PMF $\lambda_{\mathbf{p}}$: $\mathcal{S}_{n+1} \to [0,1]$, where $\mathcal{S}_{n+1}$ denotes the collection of all preference lists of the $n+1$ products $\mathcal{N} \cup \{a_0\}$. Each preference list $\boldsymbol{\sigma} = (\sigma_0, \sigma_1, \ldots, \sigma_n) \in \mathcal{S}_{n+1}$ specifies a rank ordering of the products, where σ_i is the product ranked at position i. For any product a, let $\sigma^{-1}(a)$ denote its preference rank. We suppose that lower ranked products are preferred to higher ranked products. When offered a subset $S \subseteq \mathcal{N}$ of products, the customer samples a threshold $b \in \mathcal{B}$ according to the threshold PMF g and a preference list $\boldsymbol{\sigma}$ according to the preference PMF $\lambda_{\mathbf{p}}$ and chooses to purchase the most preferred product from the set of products whose prices are less than or equal to b; that is, the customer's selection is given by $\arg\min_a \{\sigma^{-1}(a)\colon a \in S \cup \{a_0\},\, p_a \le b\}$. Then, the choice probability $\theta_a(S, \mathbf{p})$ of choosing product $a \in S \cup \{a_0\}$ from the offer set S at price vector $\mathbf{p}$ is

$$\theta_a(S, \mathbf{p}) = \sum_{b \in \mathcal{B}} g_b \mathbb{P}_{\lambda_{\mathbf{p}}}\big(a \mid \{q \in S \cup \{a_0\}\colon p_q \le b\}\big),$$

where g_b denotes the probability of sampling threshold b under threshold PMF g, and $\mathbb{P}_{\lambda_{\mathbf{p}}}(a \mid C)$ denotes the probability that a preference list sampled according to $\lambda_{\mathbf{p}}$ results in the purchase of a from subset C:

$$\mathbb{P}_{\lambda_{\mathbf{p}}}(a \mid C) = \sum_{\boldsymbol{\sigma}} \lambda_{\mathbf{p}}(\boldsymbol{\sigma}) \mathbb{I}[\boldsymbol{\sigma}, a, C],$$
$$\text{where } \mathbb{I}[\boldsymbol{\sigma}, a, C] = \mathbb{I}_{\{\sigma^{-1}(a) < \sigma^{-1}(q),\ \forall q \in C \cup \{a_0\},\, q \neq a\}},$$

with $\mathbb{I}_{\{A\}}$ denoting the indicator variable that takes the value 1 whenever the event A is true and 0 otherwise. Note that $\mathbb{I}[\boldsymbol{\sigma}, a, C]$ is 1 if and only if product a has the smallest rank among all products in $C \cup \{a_0\}$.

The above two-stage model is very general and requires further definitions to be estimable from data. In particular, allowing the mapping $\mathbf{p} \mapsto \lambda_{\mathbf{p}}$ to be arbitrary results in a model that is not tractable and cannot extrapolate demand to new prices. To impose additional structure, we note that the model has two levers to capture the dependence of choice on prices: (a) the threshold b, which restricts the inclusion of a product in the consideration set, and (b) the preference PMF $\lambda_{\mathbf{p}}$, which captures any residual impact that price has on choice behavior. We suppose that this residual impact changes slowly with price. More precisely, we assume that the domain of the price vectors $\mathbb{R}_+^n$ is partitioned into K different regions, with $\mathbb{R}_+^n = \mathcal{P}_1 \uplus \mathcal{P}_2 \uplus \cdots \uplus \mathcal{P}_K$ and $\mathcal{P}_k \cap \mathcal{P}_{k'} = \varnothing$ for $k \neq k'$. Each partition $\mathcal{P}_k$ is characterized by a preference PMF λ_k such that $\theta_a(S, \mathbf{p}) = \sum_{b \in \mathcal{B}} g_b \mathbb{P}_{\lambda_k}(a \mid \{q \in S \cup \{a_0\}\colon p_q \le b\})$ for all $\mathbf{p} \in \mathcal{P}_k$. In other words, two price vectors that are "close enough" are assumed to result in the same preference PMF, suggesting that residual price impacts are slow to change with price. The complexity of the model increases linearly with the number of partitions K. As the amount of data increases, the number of partitions K may be increased, enriching the complexity of our model. Thus, the above partitioning eliminates the need for parametric assumptions on how $\lambda_{\mathbf{p}}$ varies with $\mathbf{p}$, resulting in a nonparametric dependence of $\lambda_{\mathbf{p}}$ on $\mathbf{p}$.

There is also empirical evidence to suggest that K is small. Specifically, our model of latent consideration sets is closely related to the work of Gilbride and Allenby (2004), who considered a two-stage model of latent consideration sets and a multinomial probit model for preference lists. They fit this model to data collected from a conjoint study on customer preferences for a new camera format. They showed that the two-stage model has better in-sample and out-of-sample fit when compared to a discrete choice model alone. They also found strong evidence for the threshold-based screening rule. Gilbride and Allenby (2004, p. 400) stated, "Our results indicate that once the choice set is formed, the price and body style do not play a role in the final decision."[3] Based on this evidence, it is reasonable to assume that consideration set formation captures the primary effect of prices on choice. Any residual impact of price on choice is small, so that allowing the preference PMF to change slowly with prices should still result in good quality approximations while keeping the number of partitions K relatively small. Our numerical study provides empirical evidence supporting the quality of our approximations.

2.1. Discussion of Model Assumptions

2.1.1. Relation to Existing Approaches in Operations.

Our two-stage model class extends the rank-based choice model proposed in Farias et al. (2013). The rank-based choice model in uses only rank orderings of products as model primitives. It is very general, subsumes the random utility maximization class of models and removes the need for potentially unrealistic, structural assumptions about how utilities are generated. However, despite its flexibility in capturing a wide range of choice behaviors, it cannot predict choices when prices or other product features vary. Such predictions require explicit modeling of how the rank orderings vary in response to variation in product prices. Utility-based models—such as the popular MNL, NL, and LC-MNL models—capture the dependence of choices on prices by modeling product utilities as functions of product prices. Such an approach results in models that are either parsimonious and tractable (such as the MNL and NL models) or flexible (such as the LC-MNL model) but *not both*. Our approach is to retain flexibility by operating in the space of rank orderings of products and to introduce structure via the layer of consideration sets. This approach allows us to retain the appealing properties of rank-based choice models as well as leverage existing methods to estimate such models.

2.1.2. Relation to the Literature on Consideration Sets.

The idea that consumers use a two-stage decision process in which they first evaluate products to form a smaller consideration set and then make the purchase decision from this consideration set has been studied and empirically verified through extensive work in marketing (see the references cited at the end of Section 1.2). Our price-based screening rule belongs to the class of conjunctive decision rules, which have been studied and empirically established in the marketing literature (see Hauser 2014). A conjunctive rule specifies that a customer considers a product only if the attribute values of *all* the attributes are above (or below) certain acceptable thresholds (and thus conjunctive). The latent price threshold in our model corresponds to the acceptable level for price. Because we only allow product prices to vary (with all the other attributes remaining fixed), products that may not be acceptable due to other attributes can equivalently be assumed to be less preferred than the no-purchase option. Therefore, our screening rule captures general conjunctive screening rules.

2.1.3. Operational Tractability.

Finally, our modeling assumptions are motivated by the desire to strike a balance between the flexibility of the models and their tractability. However, it is not immediately clear if the high dimensional distribution λ over preference lists (whose dimension scales with $n! \sim (n/e)^n$ where n is the number of products) lends itself to tractable estimation and optimization. Surprisingly, we identify a structure in the data that allows us to obtain a continuous trade-off between the "complexity" of the data and the tractability of the estimation and optimization problems. To the best of our knowledge, our work is the first to provide theoretical guarantees for rank-based choice models that operate directly in the space of distributions over preference lists.

2.2. Data Model

We assume data that are available to us in the form of a sequence of choices made by the customers in response to particular offer sets and prices. Formally, we assume that we are given choices of T customers in the form of tuples $\mathsf{Data} = \{(c_1, S_1, \mathbf{p}_1), (c_2, S_2, \mathbf{p}_2), \ldots, (c_T, S_T, \mathbf{p}_T)\}$, where $c_t \in S_t \cup \{a_0\}$ is the product chosen by customer t when offered products in subset S_t at prices $\mathbf{p}_t = (p_{at}: a \in \mathcal{N})$. We assume that $p_{at} = +\infty$ for any $a \notin S_t$. As the price vectors also contain information about the products that are not offered (because we set their prices to infinity), we sometimes simply write $\mathsf{Data} = \{(c_1, \mathbf{p}_1), (c_2, \mathbf{p}_2), \ldots, (c_T, \mathbf{p}_T)\}$ for brevity of notation.

The type of data that we assume is typically readily available in practice either in the form of purchase transactions or aggregated market shares. Because we are assuming that each transaction corresponds to a different customer, aggregated market share data can be readily transformed into the form above by creating a dummy customer for each product purchase (where the number of product purchases is determined by multiplying the market share by the market size). In the data specification above, we assume that we also observe the selection of the no-purchase option or, equivalently, the size of the market. This is a standard assumption, and one can adopt any of the several demand untruncation methods proposed in the literature (see Haensel and Koole 2011 and the references therein) to deal with the data censoring issue. We discuss one such adaptation in Online Appendix F.

2.3. Overview of Research Questions

The biggest challenge with our modeling framework is its computational tractability. To address this challenge, we ask the following two questions: (1) How can we tractably fit the model to transaction data and then determine the optimal joint assortment and pricing decision? (2) How does the computational complexity depend on the complexity of the pricing structure? Answering the first question makes our modeling framework *operational* by providing algorithms to estimate parameters and then determine the optimal decision. Answering the second question allows a manager to make a continuous trade-off between the computational burden and the flexibility afforded by complex pricing structures.

We answer the above questions in three steps; we (a) develop a general purpose estimation methodology to fit our model to historical transactions, (b) identify tractable pricing structures and relate the complexity of the pricing structure to the computational burden of estimation, and (c) extend the tractability of the identified pricing structures to joint assortment and price optimization.

2.3.1. Estimation Methodology. We use the method of maximum likelihood estimation (MLE) to estimate our model parameters. Our model is described by a partition of the domain of the price vectors and the parameters λ_k and g_k for each partition. Joint estimation of the partitions and model parameters is challenging. In fact, even when the partitions are given, estimating model parameters is an NP-complete problem (see Proposition 4.1). As a result, we focus the estimation section on estimating model parameters when partitioning is given. We discuss techniques to embed our estimation methodology into a partitioning scheme in the numerical experiments in Section 6. With a given partitioning, model parameters can be estimated separately for each partition. We focus on an arbitrary partition and drop the partition subscripts from λ and g to simplify notation.

Assuming that each data point is generated from an independent draw from the model, the log likelihood of the data is given by

$$\mathscr{L}(g, \lambda \mid \text{Data}) = \sum_{t=1}^{T} \log\left(\sum_{b \in \mathscr{B}} g_b \mathbb{P}_\lambda(c_t \mid \{a \in S_t \cup \{a_0\}: p_a \le b\})\right).$$

We select as estimates, the parameters $\hat{g}$ and $\hat{\lambda}$ that maximize the log-likelihood function:

$$\hat{g}, \hat{\lambda} \in \arg\max_{g,\lambda} \mathscr{L}(g, \lambda \mid \text{Data}) = \arg\max_{g,\lambda} \sum_{t=1}^{T} \log\left(\sum_{b \in \mathscr{B}} g_b \mathbb{P}_\lambda(c_t \mid \{a \in S_t \{a_0\}: p_a \le b\})\right). \quad (1)$$

We face several challenges in solving the MLE optimization problem in (1). We discuss these issues in detail in Section 3.

2.3.2. Tractable Pricing Structures. The computational complexity of our estimation procedure is dominated by the complexity of solving a high-dimensional linear program (LP) with $O(n!)$ variables. We show that the LP reduces to the rank aggregation problem of determining a ranking σ that minimizes a linear cost function, which is NP-complete (see Proposition 4.1). However, what is not clear is what the source of complexity is. Intuitively, we expect the complexity of estimation to depend on the complexity of the data. But how does one characterize the complexity of the data? We answer this question by identifying classes $\mathcal{P}_d$ of pricing structures, termed the d-sorted pricing structures, where d is an integer taking values between 0 and n. These pricing structures are nested: $\mathcal{P}_0 \subset \mathcal{P}_1 \subset \mathcal{P}_2 \subset \cdots \subset \mathcal{P}_{n-1} \subset \mathcal{P}_n = \mathbb{R}_+^n$, so that d captures the complexity of the pricing structure. Assuming that all the training prices belong to $\mathcal{P}_d$, we show that the complexity of solving the LP is polynomial in n for a fixed d. We discuss the details in Section 4.

2.3.3. Operational Tractability. After estimating the parameters of the model, we consider the canonical operational problem of determining the combination of offer set and price vector that maximizes the expected revenue, corresponding to the following optimization problem:

$$\max_{S \subseteq \mathcal{N}, \mathbf{p} \in \mathbb{R}_+^n} R(S, \mathbf{p}) = \max_{S \subseteq \mathcal{N}, \mathbf{p} \in \mathbb{R}_+^n} \sum_{a \in S} p_a \theta_a(S, \mathbf{p}). \quad (2)$$

The decision problem in (2) is computationally challenging to solve because it not only requires searching over all possible subsets, but the revenue function is also globally nonconcave in prices even for a fixed subset S. The above problem, in general, is NP-hard. Surprisingly, we show that if we restrict the prices to the d-sorted family $\mathcal{P}_d$ for a fixed d, then there exists a polynomial-time approximation scheme for finding the optimal solution. We discuss the details of the decision problem in Section 5.

3. Estimation Methodology

As mentioned, we estimate the model parameters by solving the MLE problem in (1). This problem is in general nonconcave and involves a factorial number of variables, making it computationally challenging to solve to optimality. As a result, we settle for finding a local maximum or a stationary point.

To find a stationary point, we use the EM metaheuristic of Dempster et al. (1977). The EM heuristic is commonly used to simplify the maximization of log-likelihood functions when some of the variables are latent, but the log-likelihood function becomes globally concave when the latent variables are observed. In our setting, the price thresholds are latent, but the log-likelihood function becomes globally concave when the price thresholds are observed; see Lemma 3.1 below, proved in Online Appendix A.1.

Lemma 3.1. *Let $\mathscr{C} = \{(a, A)\colon a = c_t$ and $A = S_t^b$ for some $b \in \mathscr{B}$ and $1 \le t \le T\}$. The complete data log-likelihood function is concave and separable in the parameters g and λ, and is given by*

$$\mathscr{L}_C = \sum_{b \in \mathscr{B}} m_b \log g_b + \sum_{(a,A) \in \mathscr{C}} \gamma_{a,A} \log \mathbb{P}_\lambda(a \mid A), \tag{3}$$

where m_b denotes the number of customers in the data with threshold b and $\gamma_{a,A}$ denotes the number of customers t with the choice and consideration set pair (a, A).

However, because the price thresholds, and hence $(m_b\colon b \in \mathscr{B})$ and $(\gamma_{a,A}\colon (a, A) \in \mathscr{C})$, are not observed, the complete data log-likelihood function in (3) cannot be directly optimized. The EM method deals with this issue as follows. It starts with arbitrary initial estimates of the model parameters $\hat{g}$ and $\hat{\lambda}$ and computes the conditional expected values $\mathbb{E}[\mathscr{L}_C \mid \hat{g}, \hat{\lambda}]$ (the E-step). It then maximizes the resulting conditional likelihood function to generate new estimates of the model parameters (the M-step). The two steps are carried out iteratively with the model parameter estimates generated in each step used as inputs for the next step. The iterations are carried out until convergence. Lemma 3.2 shows that the E-step can be performed efficiently.

Lemma 3.2 (Conditional Expectation for the E-Step). *Given model parameter estimates $\hat{g}$ and $\hat{\lambda}$, the conditional expectation of the complete data log-likelihood function is*

$$\mathbb{E}[\mathscr{L} \mid \hat{g}, \hat{\lambda}] = \sum_{b \in \mathscr{B}} \hat{m}_b \log g_b + \sum_{(a,A) \in \mathscr{C}} \hat{\gamma}_{a,A} \log \mathbb{P}_\lambda(a \mid A), \tag{4}$$

where $\hat{m}_b$ is the expected number of customers with latent threshold b, and $\hat{\gamma}_{a,A}$ is the expected number of customers with the choice and consideration set pair (a, A), given by

$$\hat{m}_b = \sum_{t=1}^{T} h_t(b) \quad \text{and} \quad \hat{\gamma}_{a,A} = \sum_{t=1}^{T} \sum_{b \in \mathscr{B}} h_t(b) \mathbb{1}_{\{a = c_t, A = S_t^b\}}, \tag{5}$$

where $h_t(b)$ is the probability that customer t has latent threshold b conditioned on her observation $(c_t, S_t, \mathbf{p}_t)$ and is given by $h_t(b) = \hat{g}_b \mathbb{P}_{\hat{\lambda}}(c_t \mid S_t^b) / \sum_{b' \in \mathscr{B}} \hat{g}_{b'} \mathbb{P}_{\hat{\lambda}}(c_t \mid S_t^{b'})$.

Based on the result in Lemma 3.2, the EM procedure may be summarized as follows:

Step 0 [Initialization]. Set $\hat{g}$ and $\hat{\lambda}$ to arbitrary feasible initial values.

Step 1 [E-step]. Compute the following estimates using (5):

$\hat{m}_b$ = expected no. of customers with latent threshold b for $b \in \mathscr{B}$, and

$\hat{\gamma}_{a,A}$ = expected no. of customers with choice, a consideration set tuple (a, A) for $(a, A) \in \mathscr{C}$.

Step 2 [M-step]. Generate new estimates $\mathbf{g}^*$ and λ^* as follows:

$$\mathbf{g}^* = \underset{\mathbf{g} \ge 0\colon \sum_{b \in \mathscr{B}} g_b = 1}{\arg\max} \sum_{b \in \mathscr{B}} \hat{m}_b \log g_b = \left(\frac{\hat{m}_b}{\sum_{q \in \mathscr{B}} \hat{m}_q} \colon b \in \mathscr{B} \right),$$

$$\lambda^* = \arg\max \left\{ \sum_{(a,A) \in \mathscr{C}} \hat{\gamma}_{a,A} \log \mathbb{P}_\lambda(a \mid A)\colon \sum_{\sigma \in \mathcal{S}_{n+1}} \lambda(\sigma) = 1,\ \lambda(\sigma) \ge 0,\ \forall\, \sigma \right\}.$$

Step 3 [Check stopping condition]. If the stopping condition is not met, then $\hat{g} \leftarrow g^*$ and $\hat{\lambda} \leftarrow \lambda^*$ and go to Step 1. Otherwise, terminate with the output g^* and λ^*. Several stopping conditions are appropriate; we discuss the issue in Section 3.3.

The E-step in the above algorithm can be carried out efficiently because it only relies on $O(nT)$ choice probabilities, $\mathbb{P}_{\hat{\lambda}}(a \mid A)$ for all $(a, A) \in \mathscr{C}$, as opposed to the entire $(n+1)!$ dimensional distribution $\hat{\lambda}\colon \mathcal{S}_{n+1} \to [0, 1]$. However, carrying out the M-step requires solving the following optimization problem:

$$\max \left\{ \sum_{(a,A) \in \mathscr{C}} \hat{\gamma}_{a,A} \log \mathbb{P}_\lambda(a \mid A)\colon \sum_{\sigma \in \mathcal{S}_{n+1}} \lambda(\sigma) = 1,\ \lambda(\sigma) \ge 0,\ \forall\, \sigma \right\}. \tag{M-step}$$

This problem presents two key challenges: (a) multiplicity of optimal solutions and (b) factorial number of variables. We discuss how we address each of these challenges next.

3.1. Multiplicity of Optimal Solutions: Reparametrization of the M-step

The optimization problem M-step has multiple optimal solutions because the available data are not sufficient to allow for point identification of the model parameters. This multiplicity is a result of the flexibility we have afforded our model and stems from the fundamental nonidentifiability of distributions over rankings from choice data. To see this, let λ^* be an optimal solution and consider choice probabilities $y^*_{a,A} \stackrel{\text{def}}{=} \mathbb{P}_{\lambda^*}(a \mid A)$. Sher et al. (2011) show that for $n \geq 4$, there are multiple PMFs over preference lists that are consistent with *any* given collection of choice probabilities. A rough proof follows from the observation that while the choice probabilities impose $O(2^n)$ degrees of freedom, the underlying distribution over preference lists has $O(n!) = O(2^{n \log n})$ degrees of freedom.

We overcome the multiplicity issue through a reparametrization of the M-step that provides a compact description of the set of optimal solutions. Because we only need the inputs $\hat{g}$ and the choice probabilities $\mathbb{P}_{\hat{\lambda}}(a \mid A)$ for all $(a, A) \in \mathscr{C}$ to carry out each EM iteration, we consider the following reformulation:

$$\max\left\{ \sum_{(a,A)\in\mathscr{C}} \hat{\gamma}_{a,A} \log y_{a,A} \colon y_{a,A} = \mathbb{P}_\lambda(a \mid A),\ \forall (a,A) \in \mathscr{C}, \sum_{\sigma \in \mathscr{S}_{n+1}} \lambda(\sigma) = 1,\ \lambda(\sigma) \geq 0,\ \forall \sigma \right\}. \quad (6)$$

The optimal solution $\mathbf{y}^*$ encapsulates *all* the optimal solutions λ^* to the M-step, $\{\lambda\colon y^*_{a,A} = \mathbb{P}_\lambda(a \mid A),\ \forall (a, A) \in \mathscr{C}\}$, and hence is sufficient to carry out the E-step in the next iteration. Depending on the subsequent prediction or decision problem, a specific distribution λ can be chosen from the identified set.

3.2. Factorial Number of Variables: Obtaining an Improving Solution for M-step

With the reformulation above, the M-step optimization problem can be written more succinctly as

$$\max_{y \in Q_\mathscr{C}} f(\mathbf{y}) \stackrel{\text{def}}{=} \sum_{(a,A)\in\mathscr{C}} \hat{\gamma}_{a,A} \log y_{a,A}, \quad (7)$$

where the polytope $Q_\mathscr{C}$ is defined as

$$Q_\mathscr{C} \stackrel{\text{def}}{=} \left\{ \mathbf{y} \in [0,1]^{|\mathscr{C}|}\colon y_{a,A} = \mathbb{P}_\lambda(a \mid A),\ \forall (a,A) \in \mathscr{C}, \sum_{\sigma \in \mathscr{S}_{n+1}} \lambda(\sigma) = 1,\ \lambda(\sigma) \geq 0,\ \forall \sigma \right\}. \quad (8)$$

The presence of a factorial number of variables makes it computationally challenging to solve (7), despite it being a concave maximization problem. We overcome this issue by accomplishing the simpler task of obtaining an improving solution $\mathbf{y}^*$, whose objective value is greater than or equal to the objective value at the existing estimate $\hat{\mathbf{y}}$. This relaxed variant of the EM algorithm, which relies only on finding an improving solution to the M-step, is called the generalized EM algorithm and can be shown to converge to a stationary point of the log-likelihood function.

To find an improving solution, we find a surrogate function that is easier to optimize. Because linear functions are generally tractable, we use a local linear approximation of the objective function in (7) as the surrogate. To simplify notation, let $f(\mathbf{y})$ denote the objective function $\sum_{(a,A)\in\mathscr{C}} \hat{\gamma}_{a,A} \log y_{a,A}$ of the optimization problem in (7). Then, we can establish the following result:

Proposition 3.1 (Certificate of Optimality for the M-step). *Let $\hat{\mathbf{y}} \in Q_\mathscr{C}$ be given, and let $\mathbf{x}^*$ denote an optimal solution to the linear program*

$$\max_{\mathbf{x} \in Q_\mathscr{C}} \sum_{(a,A)\in\mathscr{C}} c_{a,A} x_{a,A}, \quad \text{(M-step LP)}$$

where $c_{a,A} = \hat{\gamma}_{a,A}/\hat{y}_{a,A}$ for all $(a,A) \in \mathscr{C}$. If

$$\sum_{(a,A)\in\mathscr{C}} c_{a,A} x^*_{a,A} \leq T,$$

then $\hat{\mathbf{y}}$ is the optimal solution to (7). On the other hand, if $\sum_{(a,A)\in\mathscr{C}} c_{a,A} x^*_{a,A} > T$, then there exists an $\alpha \in (0,1)$ such that for $\mathbf{y} = \alpha\hat{\mathbf{y}} + (1-\alpha)\mathbf{x}^*$, we have $f(\mathbf{y}) > f(\hat{\mathbf{y}})$. Such an improving solution $\mathbf{y}$ may be found using a one-dimensional line search

$$\max_{\alpha \in [0,1]} f(\alpha\hat{\mathbf{y}} + (1-\alpha)\mathbf{x}^*),$$

which can be done efficiently because the function $\alpha \mapsto f(\alpha\hat{\mathbf{y}} + (1-\alpha)\mathbf{x}^*)$ is strictly concave on $[0,1]$.

Proposition 3.1 provides a certificate of optimality for the optimization problem in (7). The certificate requires solving the M-step LP. If the optimal value of the LP is less than or equal to the total number of customers T, then $\hat{\mathbf{y}}$ is the optimal solution to (7), and therefore we cannot find a solution that increases the objective value of (7) beyond that at $\hat{\mathbf{y}}$. On the other hand, if the optimal objective value turns out to be strictly bigger than T, then there always exists a solution $\mathbf{y}$ that strictly increases the objective value $f(\cdot)$ beyond the current solution $\hat{\mathbf{y}}$. Such an improving solution may be found using a one-dimensional line search $\max_{\alpha\in[0,1]} f(\alpha\hat{\mathbf{y}} + (1-\alpha)\mathbf{x}^*)$, which can be done efficiently because $\alpha \mapsto f(\alpha\hat{\mathbf{y}} + (1-\alpha)\mathbf{x}^*)$ is strictly concave on $[0,1]$; see, for example, Boyd and Vandenberghe (2004). We note that although the M-step LP in Proposition 3.1 involves $n!$ variables $\{\lambda(\sigma)\colon \sigma \in \mathscr{S}_{n+1}\}$, the feasible polytope $Q_\mathscr{C}$ has many interesting properties that can be exploited for efficient computation. We defer these issues to Section 4.

3.3. Putting Everything Together

We have the following convergence result for the above EM algorithm:

Theorem 3.1 (Convergence of Parameter Estimates). *For all k,*

$$\mathscr{L}(g^{(k)}, \mathbf{y}^{(k)}) < \mathscr{L}(g^{(k+1)}, \mathbf{y}^{(k+1)}),$$

and thus, the sequence of the log-likelihoods associated with the parameter estimates generated by the EM algorithm converges to a value corresponding to a stationary point of the log-likelihood function.

The proof is standard and follows from standard EM machinery (Dempster et al. 1977), so we omit the details. Online Appendix A.4 presents a formal description of the EM algorithm. We make a few remarks about the implementation:

• *Stopping criteria.* We terminate the algorithm when the increase in the log-likelihood value is within a prespecified tolerance parameter.

• *Initialization.* The initialization can be arbitrary as long as the resulting choice probabilities $y_{a,A}$ are strictly positive for all $(a,A) \in \mathscr{C}$. We describe a greedy initialization algorithm in Online Appendix D (see Algorithm 1) that initializes $\lambda^{(0)}$ to have a support of size $n+1$ and strictly positive choice probabilities for all $(a,A) \in \mathscr{C}$.

• *Uniqueness of the solution to* (6). The optimization problem in (6) has a unique solution if $\hat{\gamma}_{a,A} > 0$ for all $(a,A) \in \mathscr{C}$ because $\log(\cdot)$ is strictly concave. It can be seen from Lemma 3.2 that $\hat{\gamma}_{a,A} > 0$ for all $(a,A) \in \mathscr{C}$ if $\hat{g}_b > 0$ for all $b \in \mathscr{B}$. We assume that the EM method is started with an initial estimate g that has a positive probability mass over all the thresholds in $\mathscr{B}$. This ensures that $\hat{g}_b > 0$ for all $b \in \mathscr{B}$ in all the iterations, so that $\hat{\gamma}_{a,A} > 0$ for all $(a,A) \in \mathscr{C}$.

• *Extension to data with unobserved choice of the no-purchase option*. The EM algorithm in this section assumes that we observe the choice of the no-purchase option in the data set. However, in many settings, the sales transaction data do not record the choice of the no-purchase option. We extend our proposed EM method to handle the missing observations of the choice of the no-purchase option by using the approach of Vulcano and van Ryzin (van Ryzin and Vulcano 2013). The details of the extension are given in Online Appendix F.

Additional implementation details are discussed in Section 6 on numerical studies.

4. An Efficient Solution of the M-step LP and d-Sorted Pricing Structures

In this section, we focus on efficiently solving the M-step LP in Proposition 3.1. For that, we identify classes of pricing structures of increasing complexity and relate the complexity of the pricing structure to the computational complexity of solving the M-step LP

$$\max_{\mathbf{x} \in Q_{\mathscr{C}}} \sum_{(a,A) \in \mathscr{C}} c_{a,A} x_{a,A},$$

where the polytope $Q_{\mathscr{C}}$ is given in (8). The optimal solution of the above LP will occur at an extreme point. Because $Q_{\mathscr{C}}$, by definition, is the convex hull of the $(n+1)!$ points $\mathbf{e}_\sigma \in \{0,1\}^{|\mathscr{C}|}$ defined as $e_{\sigma,a,A} = \mathbb{1}[\sigma,a,A]$, every extreme point must be equal to $\mathbf{e}_\sigma$ for some $\sigma \in \mathcal{S}_{n+1}$. As a result, solving the M-step LP is equivalent to solving the following optimization problem:

$$\max_{\sigma \in \mathcal{S}_{n+1}} \sum_{(a,A) \in \mathscr{C}} c_{a,A} \mathbb{1}[\sigma,a,A]. \tag{9}$$

The optimization problem above has the following intuitive interpretation: for each choice and consideration set pair (a,A), the weight $c_{a,A}$ quantifies the importance of that pair. Our goal is to find the ranking that is "most consistent" with the pairs (a,A) according to weights $c_{a,A}$.

The optimization problem in (9) is similar to the linear program obtained in recent work by van Ryzin and Vulcano (2014) when fitting distributions over rankings using the maximum likelihood method. It is also similar to the problems that have appeared in the existing machine learning literature (see Meliă et al. 2007, Ali and Meliă 2012) in the context of *rank aggregation*, where the goal is to find the ranking that is most consistent with a given set of rankings, under appropriate definitions of consistency. Particularly, as shown in Proposition 4.1, a special case of (9) is the popular *Kemeny optimization problem*, in which the objective is to find a single ranking that minimizes the average distance from a given collection of total orderings/rankings over n items.

Proposition 4.1 (Hardness of Solving the M-step LP). *The* M-step LP *is equivalent to*

$$\max_{\sigma \in \mathcal{S}_{n+1}} \sum_{(a,A) \in \mathscr{C}} c_{a,A} \mathbb{1}[\sigma,a,A],$$

and it is NP-complete. Kemeny optimization, however, is NP-hard (*Ali and Meliă* 2012), *which implies that* (9) *is also NP-hard.*

Proposition 4.1 is proved in Online Appendix B. Whereas Kemeny optimization has been well studied in the literature, the M-step LP itself has received little attention. To the best of our knowledge, there has been no study of the source of complexity in solving the M-step LP. Our goal is to understand the *source of complexity* and understand whether there are special and practically relevant cases that can be solved efficiently. We discuss this next.

4.1. The d-Sorted Pricing Structure

To characterize the complexity of solving the M-step LP, we introduce the class of d-sorted pricing structures. To formally define these pricing structures, consider a specific ordering $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_n)$ of the products $\{a_1, a_2, \ldots, a_n\}$, and for all i, let $\tau^{-1}(a_i) \in \{1, 2, \ldots, n\}$ denote the rank of product a_i under $\boldsymbol{\tau}$. Define the collection of price vectors $\mathcal{P}_d$ as follows:

$$\mathcal{P}_d = \big\{\mathbf{p} \in \mathbb{R}_+^n \colon |\pi_{\mathbf{p}}^{-1}(a_i) - \tau^{-1}(a_i)| \le d, \quad \forall\, i \text{ such that } p_i \neq +\infty\big\}, \tag{10}$$

where d is an integer such that $0 \le d \le n$ and $\pi_{\mathbf{p}}$ represent the price ordering of the products according to price vector $\mathbf{p}$ so that $p_{\pi_{\mathbf{p}}(1)} \le p_{\pi_{\mathbf{p}}(2)} \le \cdots \le p_{\pi_{\mathbf{p}}(n)}$, with $\pi_{\mathbf{p}}^{-1}(a_i)$ denoting the rank of product indexed i

and $\pi_{\mathbf{p}}(i)$ denoting the product that is ranked at rank i under $\mathbf{p}$. Also, recall that we assume $\mathbf{p}_i = \infty$ if product i is not offered. By definition,

$$\mathcal{P}_0 \subset \mathcal{P}_1 \subset \mathcal{P}_2 \subset \cdots \subset \mathcal{P}_{n-1} \subset \mathcal{P}_n = \mathbb{R}^n_+.$$

When $d = 0$, $\mathcal{P}_0$ is the set of price vectors that have exactly the same ordering; that is,

$$\mathcal{P}_0 = \left\{\mathbf{p} \in \mathbb{R}^n_+ : p_{\tau_1} \le p_{\tau_2} \le \cdots \le p_{\tau_n}\right\}.$$

This type of constraint ($d = 0$) on the price vectors is quite popular and has been shown to result in tractable estimation and optimization problems (cf. Rusmevichientong et al. 2006, Aggarwal et al. 2004). For $d \ge 1$, the set $\mathcal{P}_d$ can be thought of as a generalization of a strict sorting constraint with the price ranks of the products allowed to deviate from their corresponding reference ranks by no more than d. When $d = n$, it is clear that $\mathcal{P}_d$ admits all possible price vectors so that $\mathcal{P}_n = \mathbb{R}^n_+$. Thus, $\mathcal{P}_d$ defines collections of price vectors of increasing complexity.

In practice, d-sorted pricing structures with small values of d arise when the firm selects a base price ordering $\boldsymbol{\tau}$ and offers prices that are generally consistent with the base ordering but are allowed to deviate by an amount d. Such sorted price structures are generally reasonable when the products are vertically differentiated by brand, price, quality, etc. They also arise from a firm's business constraints, such as premium brands always being priced above nonpremium ones. To our knowledge, these pricing structures have received little attention in the literature. But, as we show below, the class of price vectors $\mathcal{P}_d$ possesses structure that we can exploit to solve both the estimation and optimization problem efficiently with the computational complexity that is polynomial in n for a fixed d but exponential in d. When d is small, our algorithms are guaranteed to be efficient. For the real-world data set used in our numerical experiments in Section 6, we observed that the product prices in the transaction data possess the d-sorted pricing structure with values of d around 3 or 4, depending on the product category.

To state our result for estimation, suppose that all the offered price vectors belong to $\mathcal{P}_d$ for a particular reference ordering $\boldsymbol{\tau}$. Define the collection of tuples

$$\mathcal{C}_d = \big\{(a, A)\colon a \in A, A = \{a'\colon p_{a'} \le b\},\ \text{for some } b \in \mathcal{B}, \mathbf{p} \in \mathcal{P}_d\big\}, \tag{11}$$

comprising of all possible choice and consideration set combinations when the offered price vectors belong to $\mathcal{P}_d$. It follows from the definition of $\mathcal{P}_d$ that $\mathcal{C}_0 \subseteq \mathcal{C}_1 \subseteq \cdots \subseteq \mathcal{C}_n$ and $\mathcal{C}_n$ is the set of all possible subsets of the n products. With this definition, we can establish that the rank aggregation problem (9), and consequently the M-step LP, can be solved efficiently in n for a fixed d. Specifically, we have the following result.

Proposition 4.2 (Efficient Solution of the M-step LP Through DP). *Consider the following rank aggregation problem*:

$$\max_{\sigma \in \mathcal{S}_{n+1}} \sum_{(a, A) \in \mathcal{C}} c_{a, A} \mathbb{1}[\boldsymbol{\sigma}, a, A].$$

If $\mathcal{C} \subseteq \mathcal{C}_d$, then the problem can be solved via a DP in $O(n^3 4^d |\mathcal{C}|)$ operations.

Note that, by definition, $|\mathcal{C}| = O(nT)$, and thus, for small d, the M-step LP can be solved efficiently. The detailed proof of Proposition 4.2 is given in Online Appendix B.2. We provide a brief proof sketch here for the special case where $d = 0$. The same argument can be extended for general d.

Proof Sketch of Proposition 4.2 for $d = 0$. Without loss of generality, assume that $\boldsymbol{\tau}$ is the identity ordering, so $\mathcal{P}_0 = \{\mathbf{p}\colon p_1 \le \cdots \le p_n\}$. With the constraint that all the training price vectors are in $\mathcal{P}_0$, it can be seen that $\mathcal{C} = \{(i, \{a_0, \ldots, a_j\})\colon 0 \le i \le j \le n\}$. To simplify notation, let c_{ij} denote $c_{a_i, A}$ for $A = \{a_0, \ldots, a_j\}$. Also, for any ranking $\boldsymbol{\sigma}$, let σ_{ij} denote the indicator variable taking the value of 1 if and only if $\sigma^{-1}(a_i) < \sigma^{-1}(a_k)$ for all $0 \le k \le j, k \ne i$. Finally, let $c_{00} = 0$. With this notation, (9) becomes

$$\max_{\sigma \in \mathcal{S}_{n+1}} \sum_{j=0}^{n} \sum_{i=0}^{j} c_{ij} \sigma_{ij}.$$

We now propose a DP to recursively construct the optimal preference list. For any ranking $\boldsymbol{\sigma} = (\sigma_0, \sigma_1, \ldots, \sigma_n)$, let σ_r denote the product ranked at position r. The objective in the above optimization problem can be shown to be $\sum_{r=0}^{n} \sum_{j=\sigma_r}^{n} c_{ij} \sigma_{\sigma_r j}$. Now, because σ_r is at position r, $\sigma_{\sigma_r j}$ is 1 if and only if none of the products ranked below r are part of the set $\{a_0, \ldots, a_j\}$. We thus have

$$\sum_{j=0}^{n} \sum_{i=0}^{j} c_{ij} \sigma_{ij} = \sum_{r=0}^{n} \sum_{j=\sigma_r}^{\min\{\sigma_0, \sigma_1, \ldots, \sigma_{r-1}\}-1} c_{\sigma_r j}.$$

The above expression of the objective function suggests a DP in which we construct the optimal ranking sequentially from the least preferred product to the most preferred product. To facilitate the formulation of the DP, we represent each ranking $\boldsymbol{\sigma}$ as the collection of tuples $(\sigma_0, \xi_0), (\sigma_1, \xi_1), \ldots, (\sigma_n, \xi_n)$, where, as before, σ_r denotes the product ranked at position r by $\boldsymbol{\sigma}$ and $\xi_r = \min\{\sigma_0, \sigma_1, \ldots, \sigma_{r-1}\}$ for any $r \ge 1$. For completeness, we set $\xi_0 = n + 1$. With the above representation of $\boldsymbol{\sigma}$, the objective function can be written as $\sum_{r=0}^{n} \sum_{j=\sigma_r}^{\xi_r - 1} c_{\sigma_r j}$.

It is clear from the above expression that the choice of σ_r is *only* influenced by ξ_r, which is determined by the products ranked at the previous $r - 1$ positions and will influence the choice of products to be ranked at positions beyond r *only* through ξ_{r+1}. As a result, we define a DP value function at stage r with the state variable ξ_r. Specifically, we define for any $0 < r \le n$ and $0 \le \xi \le n - r + 1$ the value functions

$$V_r(\xi) = \begin{cases} \max_{\sigma_r} \left[\sum_{j=\sigma_r}^{\xi-1} c_{\sigma_r j} + V_{r+1}(\min\{\sigma_r, \xi\}) \right] \\ \quad \text{if } \xi < n - r + 1, \\ \max_{0 \le \sigma_r \le \xi - 1} \left[\sum_{j=\sigma_r}^{\xi-1} c_{\sigma_r j} + V_{r+1}(\sigma_r) \right] \\ \quad \text{if } \xi = n - r + 1, \end{cases}$$

with the boundary conditions $V_{n+1}(\xi) = 0$ for all $1 \le \xi \le n$ and $V_0(n+1) = \max_{0 \le \sigma_0 \le n} [\sum_{j=\sigma_0}^{n} c_{\sigma_0 j} + V_1(\sigma_0)]$. Note that we need to compute a total of $O(n)$ value functions, each of dimension $O(n)$. Computing each value function requires us to search over $O(n)$ functions to find the maximum value, and computing each of those values requires $O(n)$ computations. Therefore, the running time of the above DP is $O(n^4)$. It can be shown (see Proposition 4.2) that solving the DP results in a feasible solution. The above argument can be extended to the case when all the price vectors belong to the set $\mathcal{P}_d$. Details are provided in Online Appendix B.2.

5. Operational Tractability: Joint Assortment and Price Optimization

We now focus on making the decision: determining the offer set and price combination to maximize revenues (that is, solving $\max_{S \subseteq \mathcal{N},\, \mathbf{p} \in \mathbb{R}_+^n} R(S, \mathbf{p})$). Using the convention that the prices of products not offered are set to $+\infty$ or simply a value larger than $\max_{b \in \mathcal{B}} b$, we drop the dependence on S and rewrite the optimization problem as $\max_{\mathbf{p} \in \mathbb{R}_+^n} R(\mathbf{p})$. Recall from Section 2 that the domain $\mathbb{R}_+^n$ is partitioned into K disjoint regions, with $\mathbb{R}_+^n = \mathcal{P}^1 \cup \mathcal{P}^2 \cup \cdots \cup \mathcal{P}^K$ and $\mathcal{P}^k \cap \mathcal{P}^{k'} = \emptyset$ if $k \ne k'$. Therefore, the optimization problem is equivalent to $\max_{k=1,\ldots,K} \max_{\mathbf{p} \in \mathcal{P}^k} R(\mathbf{p})$.

The decision problem is NP-complete in the strong sense even for the simplest case in which $\mathcal{B} = \{1, 2\}$ and $K = 1$ (Rusmevichientong et al. 2006). Therefore, we constrain the prices to be d-ordered, so the optimization problem that we wish to solve is

$$\max_{k=1,\ldots,K} \max_{\mathbf{p} \in \mathcal{P}^k \cap \mathcal{P}_d} R(\mathbf{p}).$$

For a fixed d, we will show that the optimization problem admits a PTAS.[4] We first describe the PTAS for the single region case ($K = 1$) to facilitate exposition and highlight the key algorithmic techniques. We then extend the PTAS to the general ($K > 1$) case.

5.1. PTAS for a Single Region ($K = 1$)

We assume that d is fixed and the domain is not partitioned. So, our objective is to solve the following problem:

$$Z^* = \max_{\mathbf{p} \in \mathcal{P}_d} R(\mathbf{p}),$$

where $\mathcal{P}_d$ denotes the set of d-sorted prices in $\mathbb{R}_+^n$. The objective can be simplified as follows:

$$\begin{aligned} R(\mathbf{p}) &\overset{\text{def}}{=} \sum_{i=1}^{n} p_i \theta_i(\mathbf{p}) = \sum_{i=1}^{n} p_i \sum_{b \in \mathcal{B}} g_b \mathbb{P}_\lambda(i \mid \{a\colon p_a \le b\}) \\ &= \sum_{i=1}^{n} p_i \sum_{b \in \mathcal{B}} g_b \sum_{\boldsymbol{\sigma} \in \mathcal{S}_{n+1}} \lambda(\boldsymbol{\sigma}) \mathbb{1}[\boldsymbol{\sigma}, i, \{a\colon p_a \le b\}] \\ &= \sum_{(b, \boldsymbol{\sigma}) \in \mathcal{B} \times \mathcal{S}_{n+1}} w(b, \boldsymbol{\sigma}) \sum_{i=1}^{n} p_i \mathbb{1}[\boldsymbol{\sigma}, i, \{a\colon p_a \le b\}], \end{aligned}$$

where $w(b, \boldsymbol{\sigma}) \overset{\text{def}}{=} g_b \cdot \lambda(\boldsymbol{\sigma})$. We can interpret each pair $(b, \boldsymbol{\sigma})$ as a customer type, which comprises $w(b, \boldsymbol{\sigma})$ proportion of the population and has price threshold b and preference ordering $\boldsymbol{\sigma}$. The term $\sum_{i=1}^{n} p_i \mathbb{1}[\boldsymbol{\sigma}, i, \{a\colon p_a \le b\}]$ is the revenue from customer type $(b, \boldsymbol{\sigma})$ under price vector $\mathbf{p}$; note that $\{a\colon p_a \le b\}$ always includes the no-purchase option a_0 because $p_{a_0} = 0 \le b$ for all $b \ge 0$. Throughout this section, we assume without loss of generality that $R(\mathbf{p})$ can be computed in $O(1)$ operations for every price vector $\mathbf{p} \in \mathbb{R}_+^n$. Because $\mathcal{B}$ is finite, we assume, by scaling if necessary, that $\max_{b \in \mathcal{B}} b < 1$, so the price thresholds are strictly less than one. As a result of this scaling, setting the product price to 1 is equivalent to removing it from the offer set. The main result of this section is stated in the following theorem.

Theorem 5.1 (PTAS). *For any $\epsilon \in (0, 1)$, there exists an algorithm $\mathcal{A}_\epsilon$ that generates a price vector $\mathbf{p}_\epsilon \in \mathcal{P}_d$ such that*

$$R(\mathbf{p}_\epsilon) \ge (1 - \epsilon) Z^*,$$

and the running time of algorithm $\mathcal{A}_\epsilon$ is $O(n^3 \times 4^{2d} \times k(\epsilon) \times n^{4(d+1)k(\epsilon)} \times \log(1/b_{\min}))$, where

$$b_{\min} = \min_{b \in \mathcal{B}} b \quad \text{and} \quad k(\epsilon) = \left[\frac{4}{\epsilon} + \frac{4}{\epsilon} \ln \frac{1}{\epsilon} \right]^{1 + 1/\ln(1/\epsilon)}.$$

We note that Aggarwal et al. (2004) developed a PTAS for the special case of $d = 0$, but their development does not extend to general d. Therefore, our development below involves new solution techniques. We provide a sketch of the proof of Theorem 5.1 and defer the details to Online Appendix C. The proof proceeds in three steps:

1. *Discretize the price domain.* We restrict attention to the discrete grid of prices $\mathsf{Dom}_\alpha \overset{\text{def}}{=} \{\alpha^s\colon s \in \mathbb{Z}_+\} = \{1, \alpha, \alpha^2, \ldots\}$, where $\mathbb{Z}_+$ is the set of nonnegative integers and $\alpha \in (0, 1)$. We show that restriction to Dom_α results in an α-approximate solution, i.e, $\max_{\mathbf{p} \in \mathcal{P}_{d,\alpha}} R(\mathbf{p}) \ge \alpha Z^*$, where $\mathcal{P}_{d,\alpha} \overset{\text{def}}{=} \{\mathbf{p} \in \mathcal{P}_d\colon p_i \in \mathsf{Dom}_\alpha, \forall\, i\}$. See Lemma C.1 in Online Appendix C.

2. *Relax the revenue function.* The revenue function $R(\mathbf{p})$ is not directly amenable to optimization because the decision of a customer to purchase product i depends on the prices of all the other products. To limit

this dependence, we relax the revenue function to allow a customer to purchase multiple products in each purchase instance. Specifically, given integer parameter k, define $R^{\alpha,k}(\mathbf{p}) \stackrel{\text{def}}{=} \sum_{(b,\sigma)} w(b,\sigma) \sum_{i=1}^{n} \mathbb{1}[\sigma, i, \{\ell\colon p_\ell \le b \text{ and } p_\ell \le p_i/\alpha^k\}]$, so that a customer previously purchasing product i may now also purchase lower-priced products that are at least k price levels apart. In other words, the purchased subset $\{j_1, j_2, \dots\}$ are such that $p_{j_\ell} \le p_i \cdot \alpha^{(\ell-1)(k+1)}$ for any $\ell \ge 1$. With this relaxation, it is easy to see that a customer generating revenue of p_i previously now generates at least p_i and no more than $p_i \cdot (1 + \alpha^{k+1} + \alpha^{2(k+1)} + \cdots) = p_i/(1-\alpha^{k+1})$. It thus follows that $R(\mathbf{p}) \le R^{\alpha,k}(\mathbf{p}) \le R(\mathbf{p})/(1-\alpha^{k+1})$; see Lemma C.2 in Online Appendix C. Combining this with the approximation from the above step, we can show that if $\hat{\mathbf{p}}$ is the optimal solution to $\max_{\mathbf{p}\in\mathcal{P}_{d,\alpha}} R^{\alpha,k}(\mathbf{p})$, then $Z^* \ge R(\hat{\mathbf{p}}) \ge \alpha(1-\alpha^{k+1})Z^*$. See Proposition C.1 in Online Appendix C.

3. *Optimize the relaxed revenue function.* The above discretization of prices and relaxation of the revenue function limits the dependence of the decision to purchase product i to only products whose prices are no more than k levels larger than p_i. We exploit this fact to formulate the optimization of the relaxed revenue function as a dynamic program. We elaborate this step further below.

The key step in the proof is to show that the relaxed revenue function can be optimized efficiently over $\mathcal{P}_{d,\alpha}$. Specifically, we can establish the following result.

Proposition 5.1 (DP for the Relaxed Problem). *The problem $\max_{\mathbf{p}\in\mathcal{P}_{d,\alpha}} R^{\alpha,k}(\mathbf{p})$ can be solved via a DP with a running time of*

$$O\big(n^3 \times 4^{2d} \times n^{4(k+1)(d+1)} \times \big(\log(1/b_{\min})/\log(1/\alpha)\big)\big).$$

The details of the proof of Proposition 5.1 are deferred to Online Appendix C; we provide an overview here. First, because it is never optimal to price a product below the minimum possible price threshold, we restrict product prices to $\{1, \alpha, \dots, \alpha^H\}$; here, α^H is the largest price that is less than or equal to the smallest price threshold, i.e., $\alpha^H = \max\{\alpha^s\colon \alpha^s \le b_{\min}\}$ and $b_{\min} = \min_{b\in\mathcal{B}} b$. It follows from the definition that $H = O(\log(1/b_{\min})/\log(1/\alpha))$. Given this, we represent each price vector $\mathbf{p}$ using the equivalent *subset representation*: $(A_H, A_{H-1}, \dots, A_1, A_0)$, where $A_s \stackrel{\text{def}}{=} \{i\colon p_i = \alpha^s\}$ consists of products priced at α^s. The optimization problem now reduces to determining the optimal partitioning of the n products into subsets $(A_H, A_{H-1}, \dots, A_1, A_0)$. To simplify notation, we let $A_{[\ell_1:\,\ell_2]}$ denote the tuple $(A_{\ell_1}, A_{\ell_1-1}, \dots, A_{\ell_2})$ for any $\ell_1 \ge \ell_2$.

We determine the optimal partitioning in a sequential fashion as follows: for each $0 \le s \le H$, we fix the sets $A_{[H:\,s-k+1]}$ and consider the subproblem of determining the maximum revenue $J_s^*(A_{[H:\,s-k+1]})$—under the relaxed revenue function—that can be obtained from *only* the products with prices in $A_{[s:0]}$. Because every product is priced at at least α^H, we can show that the optimal prices can be obtained by solving the optimization problem at $s = H$: $\max_{A_{[H:\,H-k+1]}} J_H^*(A_{[H:\,H-k+1]})$. Therefore, it is sufficient to show that we can determine $J_s^*(A_{[H:\,s-k+1]})$, for all possible s and $A_{[H:\,s-k+1]}$, and optimize over $A_{[H:\,H-k+1]}$ efficiently.

We determine $J_s^*(\cdot)$ by formulating it as the DP:

$$J_s^*(A_{[H:\,s-k+1]}) = \max_{A_{s-k}\in\mathcal{D}} \big(\alpha^s G_s(A_{[H:\,s-k]}) + J_{s-1}^*(A_{[H:\,s-k]})\big), \quad (12)$$

where $G_s(\cdot)$ is the number of customers who purchase products in A_s, and $\mathcal{D}$ appropriately restricts A_{s-k} so that $A_{[H:0]}$ is d-sorted.

To understand the recursion, let $W_s(A_{[H:0]})$ denote the revenue under the relaxed revenue function from the products in $A_{[s:0]}$. Then, we have, by definition, $J_s^*(A_{[H:\,s-k+1]}) \stackrel{\text{def}}{=} \max_{A_{[s-k:0]}\in\mathcal{D}'(A_{[H:\,s-k+1]})} W_s(A_{[H:0]})$, where $\mathcal{D}'(A_{[H:\,s-k+1]})$ denotes the domain of $A_{[s-k:0]}$ given $A_{[H:\,s-k+1]}$ such that $A_{[H:0]}$ is d-sorted. Now, the revenue $W_s(\cdot)$ can be decomposed into the revenue from the purchase of products in A_s and the revenue from products in $A_{[s-1:0]}$; i.e., we can write $W_s(A_{[H:0]}) = \alpha^s G_s(A_{[H:0]}) + W_{s-1}(A_{[H:0]})$, where $G_s(A_{[H:0]})$ is the number of customers who purchase products in A_s. Under the relaxed revenue function, the customers who purchase products in A_i have a consideration set that is a subset of the $A_{[H:\,i-k]}$ products with prices less than or equal to k levels above the price of i. Therefore, $G_s(\cdot)$ will only depend on $A_{[H:\,s-k]}$. It now follows that

$$W_s(A_{[H:0]}) = \alpha^s G_s(A_{[H:\,s-k]}) + W_{s-1}(A_{[H:0]}),$$

which implies that

$$\begin{aligned}
&\max_{A_{[s-k:0]}\in\mathcal{D}'(A_{[H:\,s-k+1]})} W_s(A_{[H:0]})\\
&= \max_{A_{[s-k:0]}\in\mathcal{D}'(A_{[H:\,s-k+1]})} \big\{\alpha^s G_s(A_{[H:\,s-k]}) + W_{s-1}(A_{[H:0]})\big\}\\
&= \max_{A_{s-k}\in\mathcal{D}} \Big\{\alpha^s G_s(A_{[H:\,s-k]}) + \max_{A_{[s-k-1:0]}\in\mathcal{D}(A_{[H:\,s-k]})} W_{s-1}(A_{[H:0]})\Big\},
\end{aligned}$$

where the last equality follows from decomposing the maximization over $A_{[s-k:0]}$ into our maximization over A_{s-k} and inner maximization over $A_{[s-k-1:0]}$ with the domains constrained appropriately. The DP recursion (12) now follows from the definition of J_s^*.

We then show that the DP can be solved efficiently by first arguing that the effective state space of each value function $J_s^*(\cdot)$ is small (see Lemma C.5 in Online Appendix C), consisting of $O(n^3 \times 4^{2d} \times n^{4k(d+1)})$ distinct values. We then exploit the d-sorted price structure to show that the number of distinct subsets A_s is at most $O(n^{4(d+1)})$. It then follows from the DP recursion that given $J_{s-1}^*(\cdot)$, the value function $J_s^*(\cdot)$ can be computed in $O(n^3 \times 4^{2d} \times n^{4k(d+1)} \times n^{4(d+1)}) = O(n^3 \times 4^{2d} \times n^{4(k+1)(d+1)})$, where the last term arises because the computation for each state requires maximization over $n^{4(d+1)}$ sets. Finally, because finding the optimal solution requires computing H value functions, the total complexity scales as $O(n^3 \times 4^{2d} \times n^{4(k+1)(d+1)} \times H) = O(n^3 \times 4^{2d} \times n^{4(k+1)(d+1)} \times (\log(1/b_{\min})/\log(1/\alpha)))$. We argue that this term dominates the maximization over $A_{[H:H-k+1]}$, establishing the result.

Using the above result and choosing appropriate values for α and k, such that $\alpha(1-\alpha^{k+1}) \geq 1-\varepsilon$, establishes the result of Theorem 5.1.

5.2. Extension to Multiple Regions

We now extend the PTAS for $K=1$ to the general case with $K>1$ partitions. For the general case, we solve K subproblems: $\max_{\mathbf{p}\in\mathcal{P}^k\cap\mathcal{P}_d} R(\mathbf{p})$ for each $1 \leq k \leq K$. Letting $\mathbf{p}_k^*$ denote the solution for the kth subproblem, we obtain the global optimum $\mathbf{p}^*$ as $\arg\max_{k=1,\ldots,K} R(\mathbf{p}_k^*)$. We solve each subproblem approximately using the ideas described for the $K=1$ case. It is then clear that $\arg\max_{k=1,\ldots,K} R(\hat{\mathbf{p}}_k)$ provides the desired approximation to $\mathbf{p}^*$. Therefore, we focus this section on solving a subproblem.

The complexity of solving $\max_{\mathbf{p}\in\mathcal{P}^k\cap\mathcal{P}_d} R(\mathbf{p})$ depends on the shape of $\mathcal{P}^k$. Arbitrary shapes can make the problem intractable. To avoid this, we focus on regions described by box constraints: $\mathcal{P}^k = \{\mathbf{p}\colon a_i \leq p_i \leq u_i, \forall\, i\}$. We first discretize the price and maximize over the "rounded" domain $\mathcal{P}^k_\alpha \cap \mathcal{P}_{d,\alpha}$, where $\mathcal{P}_{d,\alpha}$ is as defined above and $\mathcal{P}^k_\alpha = \{\mathbf{p}\colon p_i \in \mathsf{Dom}_\alpha, \alpha^{L_i} \leq p_i \leq \alpha^{U_i}, \forall\, i\}$, with $\alpha^{L_i} = \min\{\alpha^s\colon a_i \leq \alpha^s\}$ and $\alpha^{U_i} = \max\{\alpha^s\colon \alpha^s \leq u_i\}$. As shown in Online Appendix G, by considering the relaxed revenue function and adapting the DP technique from Proposition 5.1 in Section 5.1, we can obtain a PTAS for the optimization problem $\max_{\mathbf{p}\in\mathcal{P}^k_\alpha\cap\mathcal{P}_d} R(\mathbf{p})$. Then, under some technical conditions, we can show that the solution of the problem $\max_{\mathbf{p}\in\mathcal{P}^k_\alpha\cap\mathcal{P}_d} R(\mathbf{p})$ provides a good approximation to the original problem $\max_{\mathbf{p}\in\mathcal{P}^k\cap\mathcal{P}_d} R(\mathbf{p})$. The details are beyond the scope of this research and we do not pursue them here.

6. Numerical Study

We carried out two numerical studies to test our methods. The first study tests the *predictive* accuracy of our model on real-world sales transaction data from the IRI Academic Data Set (see Bronnenberg et al. 2008). The second study tests the *decision* accuracy of our model on synthetic transaction data. The first test accomplishes two objectives: (a) it demonstrates the application of our methods in a real-world setting, and (b) it pits our method in a horse race against the popular benchmark, the LC-MNL model. We find that our method obtains an average of 26% improvement over the benchmark on a chi-square metric, which measures the relative error in predicting market shares. The second test demonstrates that our method can increase revenues by 11% compared to the LC-MNL benchmark by improving the joint assortment and pricing decision. We used synthetic data because we needed the ground-truth model to compare the "true" performance of the decisions. Next, we describe the details of the studies.

6.1. Predictive Accuracy: Case Study with the IRI Academic Data Set

The IRI Academic Data Set is a publicly available data set containing real-world purchase transactions of consumer packaged goods for chains of grocery and drug stores. The data consist of weekly sales transactions aggregated over all the customers. We focused on the transactions of three categories for the first two weeks in the year 2011: laundry, yogurt, and coffee. Each transaction contains the following information: the week and store of purchase, the universal product code (UPC) of the purchased product, quantity purchased, price paid, and an indicator of whether the product was on price or display promotion. The data set for the first two weeks contained a total of approximately 220 K, 544 K, and 374 K transactions from 1,272 stores for laundry, yogurt, and coffee, respectively.

We processed the raw transactions to obtain products, prices, and offer sets, as described next. First, we dealt with data sparsity by aggregating the purchased items by vendors to obtain products. Each purchased item is identified in the data set by its collapsed UPC, which is a 13-digit-long code with digits 4 to 8 (five digits) denoting the vendor. There are a totals of 75, 90, and 290 vendors in the first two weeks of purchases for the yogurt, laundry, and coffee categories, respectively. Because no-purchase sales are not observed, we made the assumption that the entire market is reasonably captured by all the stores and the vendors in the data set. Furthermore, the focus is on the revenues from the top nine vendors, so the remaining vendors comprise the "rest of the market." Therefore, we treated the top nine vendors as "products" and aggregated the remaining vendors into the "outside good" or the no-purchase option.

Then, we determined the context of offer set and prices for each purchase instance. Each combination of store and week results in an offer set and price vector combination. We inferred the offer set to be the union of all the products purchased during the particular week at the particular store. We set the purchase price of a product equal to the weighted average of the prices of the different UPCs that the product comprises, where the weight of each price was equal to the corresponding observed sales at the particular store and week combination.

Our preprocessing resulted in a total of 2,470 offer set and price vector combinations for each of the yogurt, laundry, and coffee categories.

6.1.1. Model Fit. We compared the predictive accuracy of our model against the popular LC-MNL benchmark on the above data set. We briefly describe how each of the models were fitted to the provided training data.

Benchmark LC-MNL model. The L-class LC-MNL model assumes that customers belong to one of L classes, for some nonnegative integer L, and customers in class ℓ make choices according to a single-class MNL model with intercept vector $\boldsymbol{\mu}_\ell$ and price coefficient β_ℓ. A customer has a probability of α_ℓ of belonging to class ℓ, where $\alpha_\ell \geq 0$ for all ℓ and $\sum_{\ell=1}^{L} \alpha_\ell = 1$. With these assumptions, the probability that a customer in class ℓ purchases product c from offer set and price combination $(S, \mathbf{p})$ is equal to $\exp(\mu_{\ell c} - \beta_\ell p_c)/(1 + \sum_{j \in S} \exp(\mu_{\ell j} - \beta_\ell p_j))$. We estimated the model parameters by solving the following maximum-likelihood problem:

$$\max_{\boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\alpha}} \sum_{t=1}^{T} \log\left(\sum_{\ell=1}^{L} \alpha_\ell \frac{\exp(\mu_{\ell c_t} - \beta_\ell p_{t, c_t})}{1 + \sum_{j \in S_t} \exp(\mu_{\ell j} - \beta_\ell p_{tj})} \right).$$

The above optimization problem is, in general, hard to solve (see Train 2008). We used the EM algorithm described in Train (2009) to find a stationary point. We picked the LC-MNL as our benchmark because it can approximate any random utility choice model arbitrarily closely as the number of latent classes L increases Train (2008).

Our nonparametric joint assortment and price (JAP) model. We fitted our model using the EM algorithm described in Section 3.3. We discuss the implementation details of two core steps: (a) partitioning the training price vectors into K segments and (b) solving the M-step LP.

We clustered the training price vectors into K segments using the popular k-means algorithm and trained our nonparametric model separately for each segment. We set $K = 10$ using cross-validation. We observed that the predictive performance of the model was robust to the choice of K, as long as it was within a reasonable range. As mentioned above, we fitted a model to each cluster separately. For prediction, we mapped each test price vector to its closest cluster (as measured by the distance to the cluster centroid) and used the model trained on the corresponding cluster.

We solved the M-step LP by implementing a popular local search (LS) heuristic, instead of the DP discussed in Section 4, to emphasize the ease of practical implementation of our method. Our implementation demonstrates that even approximation implementations of the M-step LP yield accurate predictions. The LS heuristic starts with a random permutation and moves to the neighbor that results in the maximum improvement in the objective function. The moves are repeated until either a local optimum is reached, i.e., none of the moves result in an improvement in the objective function, or we hit a prespecified limit on the number of moves. We used a limit of 10 moves in our experiments. Several natural definitions for the neighborhood of the permutations are possible. We define the neighborhood of a permutation as the collection of rankings that are obtained by swapping the positions of two products in the permutation at a time; the neighborhood therefore consists of $O(n^2)$ rankings. The LS heuristic with this definition of neighborhood has been shown to find good approximations to the optimal solution (see Schalekamp and van Zuylen 2009, Ali and Meliă 2012) for the problem of Kemeny optimization, which is a special case of M-step LP (as shown in Online Appendix B). The LS heuristic also has other desirable properties such as using a fixed amount of memory and ease of coding. The precise implementation details of the LS heuristic are present in Online Appendix D.

6.1.2. Experiments and Results. We carried out a two-fold cross-validation in which we randomly partitioned the offer set and price combinations into two parts of roughly equal sizes, trained the models on one part and tested them on the other part, and repeated the process with the train and test sets interchanged. For each model, we measured the predictive accuracies in terms of two popular metrics, mean absolute percentage error (MAPE) and chi-square prediction error (χ2PE), defined as follows: for each model $\in$ {LC-MCL, JAP}, we compute

$$\text{MAPE}^{\text{model}} = \frac{1}{\sum_{(S,\mathbf{p}) \in \mathcal{T}} |S|} \sum_{(S,\mathbf{p}) \in \mathcal{T}} \sum_{a \in S} \frac{|\hat{\theta}_a^{\text{model}}(S, \mathbf{p}) - \theta_a^{\text{actual}}(S, \mathbf{p})|}{\theta_a^{\text{actual}}(S, \mathbf{p})},$$

where $\hat{\theta}_a^{\text{model}}(S, \mathbf{p})$ is the *predicted* probability, under the fitted model, that a will be purchased when offered as part of the offer set S at price vector $\mathbf{p}$, whereas $\hat{\theta}_a^{\text{actual}}(S, \mathbf{p})$ is the empirical probability computed from the observations in the test set. In the above expression, $\mathcal{T}$ denotes the collection of offer set and price combinations in the test set.

Similarly, for each model $\in$ {LC-MCL, JAP}, we compute

$$\chi\text{2PE}^{\text{model}} = \frac{1}{\sum_{(S,\mathbf{p}) \in \mathcal{T}} |S|} \sum_{(S,\mathbf{p}) \in \mathcal{T}} \sum_{a \in S} \frac{(\hat{n}_a^{\text{model}}(S, \mathbf{p}) - n_a^{\text{actual}}(S, \mathbf{p}))^2}{0.5 + n_a^{\text{actual}}(S, \mathbf{p})},$$

where $\hat{n}_a^{\text{model}}(S, \mathbf{p})$ is the *predicted* number of purchases, under the fitted model, of product a when offered as part of the offer set S at price vector $\mathbf{p}$, and $n_a^{\text{actual}}(S, \mathbf{p})$ is the actual number of observed purchases in the test set. We computed the predicted number of purchases by multiplying the predicted choice probability $\hat{\theta}_a^{\text{model}}(S, \mathbf{p})$ with the number of customers who were offered $S, \mathbf{p}$ in the test set. The χ2PE metric is similar to the popular chi-square measure of goodness of fit

Table 1. Improvements in the Predictive Accuracy of Our Nonparametric JAP Model Against the LC-MNL Benchmark

Improvements over LC-MNL under each metric (%)	Laundry	Yogurt	Coffee	Avg. across three product categories
MAPE (%)	23.6	12.2	14.2	16.67
χ2PE (%)	37.2	21.1	21.9	26.73

Note. All the numbers are statistically significant at 5%.

of the form $(O-E)^2/E$, where O refers to the observed value and E refers to the expected or predicted value.

Table 1 reports, for each product category, the *percentage improvements* in the MAPE and χ2PE metrics under our nonparametric JAP model relative to the LC-MNL model, defined as

$$\frac{\text{MAPE}^{\text{LC-MNL}} - \text{MAPE}^{\text{JAP}}}{\text{MAPE}^{\text{LC-MNL}}} \quad \text{and} \quad \frac{\chi 2\text{PE}^{\text{LC-MNL}} - \chi 2\text{PE}^{\text{JAP}}}{\chi 2\text{PE}^{\text{LC-MNL}}}.$$

Higher numbers are better. It is evident from the results that our method significantly outperforms the benchmark method across both metrics. In particular, we notice an average of 16% improvement under MAPE and 26% improvement for χ2PE metrics.

6.2. Decision Accuracy: Simulation Study

We now describe the results from our simulation study, which compares the expected revenues obtained from the assortment and price decisions under our method and the benchmark LC-MNL model. The results demonstrate that on difficult ground-truth model instances, our model on average extracts 11.5% more revenues from the market than the benchmark method. The broad experimental setup is as follows:

1. Pick an instance of the LC-MNL model class as the ground-truth model.
2. Use the ground-truth model to generate transaction data for a collection of assortment and price combinations. The generated data are representative of data collected in practice.
3. Fit the simulated transaction data to our model and a benchmark LC-MNL (true) model.
4. Optimize both the fitted models to determine the joint assortment and price decisions.
5. Compare the ground-truth revenues from the two decisions to determine the average increase in revenues.

The above experimental procedure pits the decisions from our method to the decisions from the true model. We expect the true model to perform better *if sufficient data are available*. However, in practice, customers exhibit diverse and complex choice behaviors, and available data are limited. For such cases, it is no longer clear if fitting the true model will in fact yield the best performance. Our results demonstrate that for complex ground-truth models, our model provides better approximations than fitting the true model when available data are limited.

6.2.1. Ground-Truth Models Generated. We randomly generated instances of the ground-truth model from the L-class LC-MNL model class. The number of products was $n = 9$ (excluding the no-purchase option). We considered $L = 5, 10, 15, 20$ latent classes. Product prices were chosen from the 21 levels in the set $\mathcal{P} = \{0.5, 0.525, 0.55, \ldots, 0.975, 1\}$, starting from 0.5 in increments of 0.025 up to 1. For each value of L, we randomly sampled 50 instances as follows:

(1) The price of each product was sampled uniformly at random from the discrete set $\mathcal{P}$.

(2) The mixing weight of each class was sampled uniformly at random from the interval $[0,1]$ and then normalized so that the mixing weights sum to 1.

(3) For each $1 \le \ell \le L$, the consideration set C_ℓ, consisting of up to five products, was sampled uniformly at random.

(4) An extra class with the consideration set $\{1, 2, \ldots, n\} \setminus \bigcup_{k=1}^{K} C_k$, consisting of products not covered by any of the other consideration sets, if any, was added.

(5) The following was done for each class:

(a) For each of the products in the consideration set corresponding to the class, an intercept term α was sampled uniformly at random from the interval $[-4, 1]$, a price coefficient β was sampled uniformly at random from the interval $[-3, -2]$, and the parameter v was set to $\exp(\alpha + \beta p)$, where p was the price of the product.

(b) The parameter v of the remaining product was set to zero.

These models are computationally hard to estimate and optimize (see Désir and Goyal 2014).

6.2.2. Synthetic Transaction Data Generated. From each instance of the ground-truth model described above, we generated synthetic transaction data as follows: we generated 30 price vectors by sampling the price of each product independently and uniformly at random from the discrete set $\mathcal{P}$. For each price vector, we generated 1,000 offer sets by sampling subsets of sizes between two and eight uniformly at random. For each of the resulting 30,000 offer set and price combinations, we randomly sampled a product choice according to the ground-truth model instance. The resulting data are of the form $(c_1, \mathbf{p}_1, S_1)$, $(c_2, \mathbf{p}_2, S_2)$, $\ldots, (c_T, \mathbf{p}_T, S_T)$, where $T = 30{,}000$. The above sampling mechanism mimics realistic settings in which prices change at a slower rate, but offer sets change at a faster rate because of stockouts, deliberate scarcity, or web page limitations.

6.2.3. Experiments Conducted. For each ground-truth model instance, we fitted our nonparametric JAP and the benchmark LC-MNL models to the synthetic transaction data. We solved the following MLE problem to fit the benchmark LC-MNL model with k classes:

$$\max_{\mu,\beta,\alpha} \sum_{t=1}^{T} \log\left(\sum_{\ell=1}^{L} \alpha_\ell \frac{\exp(\mu_{\ell,c_t} - \beta_{\ell,c_t} p_{t,c_t})}{1 + \sum_{j\in S_t} \exp(\mu_j - \beta_{\ell,j} p_{tj})}\right).$$

The above optimization problem is, in general, hard to solve (see Train 2008). We used the EM algorithm described in Train (2009) to find a stationary point. We tuned the number of classes through cross-validation. To fit our model, we used the algorithm described in Section 3.3 and estimated parameters g and λ with a single partition.

For each of the fitted models, we computed estimates of the optimal joint assortment and price decision. We estimated the optimal decisions under the benchmark LC-MNL model and the nonparametric methods by solving the MILPs LC-MNL Joint Opt and Nonparametric Joint Opt, respectively. The MILPs are described in Online Appendix E. We solved the MILPs with a time limit of 40 s using Gurobi Optimizer version 6.0.2 on a computer with a 3.5 GHz Intel Core i5 processor, 16 GB of RAM, and the Mac OSX Yosemite operating system. The MILPs may not be solved to optimality within the provided time limit, in which case we used the best solution obtained by Gurobi as the estimate of the optimal decision.

6.2.4. Results and Discussion. For each number of latent classes $L \in \{5, 10, 15, 20\}$, we generated 50 ground-truth model instances. For each ground-truth model instance $q = 1, 2, \ldots, 50$, we computed the optimal assortment and price decisions $(S^{(q)}_{\mathsf{JAP}}, \mathbf{p}^{(q)}_{\mathsf{JAP}})$ and $(S^{(q)}_{\mathsf{LC\text{-}MNL}}, \mathbf{p}^{(q)}_{\mathsf{LC\text{-}MNL}})$ respectively under the JAP and LC-MNL models fitted to the transaction data generated from the qth ground-truth model instance. We then evaluated the *percentage increase* in the (true) revenue extracted from using our JAP model versus the benchmark LC-MNL model:

$$\mathsf{Diff}^{(q)} = \frac{R^{(q),\text{true}}(S^{(q)}_{\mathsf{JAP}}, \mathbf{p}^{(q)}_{\mathsf{JAP}}) - R^{(q),\text{true}}(S^{(q)}_{\mathsf{LC\text{-}MNL}}, \mathbf{p}^{(q)}_{\mathsf{LC\text{-}MNL}})}{R^{(q),\text{true}}(S^{(q)}_{\mathsf{LC\text{-}MNL}}, \mathbf{p}^{(q)}_{\mathsf{LC\text{-}MNL}})},$$

where $R^{(q),\text{true}}(\cdot,\cdot)$ denotes the true revenue function associated with the qth instance.

Table 2 reports, for each number of latent classes L, the average percentage increase in the revenues extracted from the decision under our model relative to the decision under the fitted LC-MNL model; that is, $(1/50)\sum_{q=1}^{50} \mathsf{Diff}^{(q)}$. The results illustrate that, on average, our method can extract 11.5% more revenues from the market than the benchmark method.

Table 2. Improvements in the Revenue Under the Decision Computed from Our Nonparametric JAP Model Against the LC-MNL Benchmark

No. of latent classes (L)	5	10	15	20
Improvement (%)	11.23	12.31	11.86	10.53

Note. All numbers are statistically significant at 5%.

Note that our experiments pit our model against the true model. We attribute the poor performance of the benchmark to the presence of different consideration sets for different customer segments. The training data are not sufficient for the estimation procedure to drive the parameter v to zero for products not in the consideration set. Our model, on the other hand, is designed to be flexible to capture complex choice patterns.

7. Conclusions

Motivated by the inflexibility of existing models to capture the joint effect of assortments and prices, we proposed a tractable, nonparametric joint assortment and price choice model. Our approach is data driven, makes few structural assumptions, and is designed to improve the accuracy of revenue predictions. Surprisingly, the model also allows for tractable estimation *and* tractable optimization. The key technical contribution of our work is the identification of classes of pricing structures of increasing complexity. We then related the complexity of the pricing structure with the computational burden of carrying out estimation and optimization. Our characterization allows us to establish theoretical guarantees for our estimation algorithm and design a PTAS for the joint assortment and price optimization problem.

Our work opens the door for many exciting future research directions. The core of our model is based on a two-stage choice process, one in which a customer first forms a consideration set of relevant products and then chooses from the consideration set. Existing work in marketing provides empirical support for such a two-stage process in which customers adopt screening heuristics to form consideration sets. Our work has shown the potential gains in predictive accuracy that can be obtained from the two-stage choice models. Exploring the additional flexibility afforded by consideration sets to obtain tractable, nonparametric models is an exciting future direction.

Another key aspect of our work is our ability to provide guarantees for solving the M-step LP. As discussed in Section 4, the M-step LP generalizes the popular Kemeny optimization problem. Although it has been shown that the Kemeny optimization problem is NP-hard, very little work has been done on understanding the source of complexity to find tractable subproblems. The d-sorted price characterization we obtain is one of the few general structures that has allowed for isolation of the source of complexity to arrive at algorithms with provable guarantees. Further exploration of the d-sorted price structures can allow us to obtain principled heuristics that have so far remained unexplored for the very important Kemeny optimization problem. Finally, it is surprising that the d-sorted pricing structures also allow us to design a PTAS for the joint assortment and price optimization problem. Taking the key intuitions behind the PTAS to design scalable optimization algorithms for practical-sized problems can have a huge practical impact.

Acknowledgments
The authors thank department editor Yinyu Ye, the associate editor, and three anonymous referees for their detailed and thoughtful comments, which greatly improved this paper. They also thank Huseyin Topaloglu and Yayi Zou for pointing out a technical issue in one of the proofs in a previous version of this manuscript, which led to a cleaner proof for the result that the joint assortment and price decision problem admits a polynomial-time approximation scheme.

Endnotes
[1] Indeed, as businesses increasingly move online, such data-driven approaches are essential for businesses to take advantage of the dynamic environments, complex demand patterns, and associated wealth of data offered by online environments.

[2] Such sorted pricing structures are generally reasonable when products are vertically differentiated by brand, size, quality, etc.

[3] The authors use “choice set” for “consideration set.”

[4] The strong NP-completeness result eliminates the possibility of a fully polynomial-time approximation scheme for general d. Thus, PTAS is the best that we can hope for.

References

Aggarwal G, Feder T, Motwani R, Zhu A (2004) Algorithms for multi-product pricing. Díaz J, Karhumäki J, Lepistö A, Sannella D, eds. *Proceedings of the 31st International Colloquium on Automata, Languages and Programming (ICALP)*, Lecture Notes in Computer Science, Vol. 3142 (Springer, New York), 72–83.

Alba JW, Chattopadhyay A (1985) Effects of context and post-category cues on recall of competing brands. *J. Marketing Res.* 22(3):340–349.

Ali A, Meilă M (2012) Experiments with Kemeny ranking: What works when? *Math. Soc. Sci.* 64(1):28–40.

Alptekinoğlu A, Semple JH (2016) The exponomial choice model: A new alternative for assortment and price optimization. *Oper. Res.* 64(1):79–93.

Belobaba PP, Hopperstad C (1999) Boeing/MIT simulation study: PODS results update. 1999 *AGIFORS Reservations and Yield Management Study Group Sympos., London*, April, 27–30.

Belonax JA, Mittlestaedt RA (1978) Evoked set size as a function of number of choice criterion and information variability. Hunt HK, ed. *Advances in consumer Research*, Vol. 5 (Association for Consumer Research, Ann Arbor, MI), 48–51.

Ben-Akiva M, McFadden D, Gärling T, Gopinath D, Walker J, Bolduc D, Börsch-Supan A, et al. (1999) Extended framework for modeling choice behavior. *Marketing Lett.* 10(3): 187–203.

Blanchet J, Gallego G, Goyal V (2016) A Markov chain approximation to choice modeling. *Oper. Res.* 64(4):886–905.

Boyd S, Vandenberghe L (2004) *Convex Optimization* (Cambridge University Press, Cambridge, United Kingdom).

Bronnenberg B, Kruger M, Mela C (2008) The IRI academic data set. *Management Sci.* 27(4):745–748.

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. Ser. B* (*Methodological*) 39(1):1–38.

Désir A, Goyal V (2014) Near-optimal algorithms for capacity constrained assortment optimization. Working paper, Columbia University, New York.

Dong L, Kouvelis P, Tian Z (2009) Dynamic pricing and inventory control of substitute products. *Manufacturing Service Oper. Management* 11(2):317–339.

Ebbes P, Papies D, Van Heerde HJ (2011) The sense and non-sense of holdout sample validation in the presence of endogeneity. *Marketing Sci.* 30(6):1115–1122.

Farias V, Jagabathula S, Shah D (2013) A new approach to modeling choice with limited data. *Management Sci.* 59(2): 305–322.

Gallego G, Wang R (2014) Multiproduct price optimization and competition under the nested logit model with product-differentiated price sensitivities. *Oper. Res.* 62(2):450–461.

Gilbride TJ, Allenby GM (2004) A choice model with conjunctive, disjunctive, and compensatory screening rules. *Marketing Sci.* 23(3):391–406.

Haensel A, Koole G (2011) Estimating unconstrained demand rate functions using customer choice sets. *J. Revenue Pricing Management* 10(5):438–454.

Hanson W, Martin K (1996) Optimizing multinomial logit profit functions. *Management Sci.* 42(7):992–1003.

Hauser JR (2014) Consideration-set heuristics. *J. Bus. Res.* 67(8): 1688–1699.

Hauser JR, Wernerfelt B (1990) An evaluation cost model of consideration sets. *J. Consumer Res.* 16(4):393–408.

Honhon D, Jonnalagedda S, Pan X (2012) Optimal algorithms for assortment selection under ranking-based consumer choice models. *Manufacturing Service Oper. Management* 12(2): 279–289.

Howard JA, Sheth JN (1969) *The Theory of Buyer Behavior* (John Wiley, New York).

Jagabathula S (2016) Assortment optimization under general choice. Working paper, New York University, New York.

Li H, Huh WT (2011) Pricing multiple products with the multinomial logit and nested logit models: Concavity and implications. *Manufacturing Service Oper. Management* 13(4):549–563.

Mahajan S, van Ryzin GJ (1998) Retail inventories and consumer choice. Tayur S, Ganeshan R, Magazine M, eds. *Quantitative Models for Supply Chain Management* (Kluwer Academic Publishers, New York), 491–551.

Meliă M, Phadnis K, Patterson A, Bilmes JA (2007) Consensus ranking under the exponential model. Parr R, van der Gaag L, eds. *Proc. 23rd Conf. Uncertainty in Artificial Intelligence, Vancouver*, 285–294.

Parkinson TL, Reilly M (1979) An information processing approach to evoked set formation. Wilke WL, ed. *Advances in Consumer Research*, Vol. 6 (Association for Consumer Research, Ann Arbor, MI), 227–231.

Ratliff RM, Rao V, Narayan CP, Yellepeddi K (2008) A multi-flight recapture heuristic for estimating unconstrained demand from airline bookings. *J. Revenue Pricing Management* 7(2):153–171.

Rusmevichientong P, Van Roy B, Glynn PW (2006) A nonparametric approach to multiproduct pricing. *Oper. Res.* 54(1):82–98.

Schalekamp F, van Zuylen A (2009) Rank aggregation: Together we’re strong. Finocchi I, Hershberger J, eds. *Proc. Eleventh Workshop on Algorithm Engineering and Experiments (ALENEX), New York*, 38–51.

Sher I, Fox JT, Kim KI, Bajari P (2011) Partial identification of heterogeneity in preference orderings over discrete choices. Technical report, National Bureau of Economic Research, Cambridge, MA.

Song JS, Xue Z (2007) Demand management and inventory control for substitutable products. Working paper, Duke University, Durham, NC.

Train K (2008) EM algorithms for nonparametric estimation of mixing distributions. *J. Choice Modelling* 1(1):40–69.

Train K (2009) *Discrete Choice Methods with Simulation*, 2nd ed. (Cambridge University Press, Cambridge, United Kingdom).

Urban GL (1975) Perceptor: A model for product positioning. *Management Sci.* 21(8):858–871.

van Ryzin G, Vulcano G (2013) An expectation-maximization algorithm to estimate a general class of nonparametric choice models. Working paper, Stern School of Business, New York University, New York.

van Ryzin G, Vulcano G (2014) A market discovery algorithm to estimate a general class of nonparametric choice models. *Management Sci.* 61(2):281–300.

Vulcano G, van Ryzin G, Chaar W (2010) OM practice—Choice-based revenue management: An empirical study of estimation and optimization. *Manufacturing Service Oper. Management* 12(3):371–392.