Probabilistic Choice Models for Product Pricing using Reservation Prices *

R. Shioda L. Tunçel B. Hui

February 1, 2007

Abstract

We consider revenue management models for pricing a product line with several customer segments, working under the assumption that every customer's product choice is determined entirely by their reservation price. We model the customer choice behavior by several probabilistic choice models and formulate the problems as mixed-integer programming problems. We study special properties of these formulations and compare the resulting optimal prices of the different probabilistic choice models. We also explore some heuristics and valid inequalities to improve the running time of the mixed-integer programming problems. We illustrate the computational results of our models on real and generated customer data taken from a company in the tourism sector.

C&O Research Report: CORR 2007-02
Department of Combinatorics and Optimization
University of Waterloo
Waterloo, ON, Canada

*All authors are from the Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON, Canada. Research supported in part by Discovery Grants from NSERC, a research grant from Air Canada Vacations and by a Collaborative Research and Development Grant from NSERC.

1 Introduction

One of the key revenue management challenges for a company is to determine the "right" price for each of their product line. Generally speaking, a company wants to set the prices to maximize their total profit. The challenge arises from the complex relationship between the product prices and the total profit. For example, how do the prices affect the demand for each product? In cases where multiple products are offered by the company, the price and demand for a product cannot be considered in isolation from the other products. That is, the company must take into account the fact that in addition to competitors' products and prices, a customer's decision to purchase a product can be swayed by the relative prices of similar products offered by the same company. Thus, prices need to be set not only to "beat" the competitors products but also to avoid "cannibalizing" the company's own product line. For example, if there are high margin and low margin products, setting the price of the latter too low may decrease demand for the high margin product, thus resulting in lower profit.

In this paper, we study several different models for product pricing from a mathematical programming perspective. The models differ from one another according to different assumptions on customer purchasing behavior. Before we discuss the details of our approach, let us first give a brief overview of relevant work in this area.

1.1 Background

There are many variations on product pricing models depending on the setting. For example, there is the single-product, multi-customer setting, which is primarily concerned with what price to offer to different customer segments. Airline revenue management is one of the most popular examples in this context, where business travelers, leisure travelers and budget travelers are offered different prices for the same flight, depending on the lead time of purchase and additional options (e.g., partially refundable tickets). An alternative framework is the multi-product, multi-customer setting where every customer is offered the same price for a given product, but different customer segments have varying preferences. This is more of a combinatorial problem where given the customer preference information, the prices need to be set to maximize total revenue. We will focus on the second type of problem in this paper.

In general, suppose a company has m different products and market analysis tells them that there are n distinct customer segments, where customers of the same segment behave the "same". A key revenue management problem is to determine optimal prices for each product to maximize total revenue, given the customer choice behavior. There are multitudes of models for customer choice behavior [9], but this paper focuses solely on those based on reservation prices.

Let R_{ij} denote the *reservation price* of segment i for product j, $i = 1, \ldots, n$, $j = 1, \ldots, m$, which reflects how much customers of segment i are *willing* and *able* to spend on product j. R_{ij} is not only the dollar amount that product j is worth to customers in segment i, but it also reflects how much they are able to pay for it. For example, if a customer segment believes that a

7 day vacation to St. Lucia is worth \$2,000, but they can only afford \$1,000 for a vacation, then their reservation price for St. Lucia is \$1,000. We assume that reservation prices are the same for every customer in a given segment and each segment pays the same price for each product. Customer choice models based on reservation prices assume that customer purchasing behavior can be fully determined by their reservation price and the price of products. Without loss of generality, we make the following assumption:

Assumption 1.1. R_{ij} is a nonnegative integer for all $i = 1, \ldots, n$ and $j = 1, \ldots, m$.

If the price of product j is set to $\$\pi_j$, $\pi_j \geq 0$, then the *surplus* of segment i for product j is the difference between the reservation price and the price, i.e., $R_{ij} - \pi_j$. It is often assumed that a segment will only consider purchasing a product with nonnegative utility, i.e.,

Assumption 1.2. If segment i buys product j, then $R_{ij} - \pi_j \geq 0$, $i = 1, \ldots, m$, $j = 1, \ldots, m$.

Even in a reservation price framework, there are several different models for customer choice behavior in the literature. In [2, 3], the authors proposed a pricing model that maximizes profits with the assumption that each customer segment only buys the product with the maximum surplus if the surplus is nonnegative. This model is often referred to as the *maximum utility* or *envy-free pricing* model. In this model, each segment buys at most one product. The authors modeled the problem as a non-convex, nonlinear mixed-integer programming problem and solved the problem using a variety of heuristic approaches.

In [6], the authors examined a Share-of-Surplus Choice Model in which the probability that a segment will choose a product is the ratio of its surplus versus the total surplus for the segment across all products with nonnegative surplus. They proposed a heuristic which involves decomposing the problem into hypercubes and used a simulated annealing algorithm to find the best hypercube. Solutions found by the heuristic for problems with sizes up to 5 products and 10 segments were shown to be near-optimal.

Another approach of pricing multiple products is to consider the problem of bundle pricing [4]. It is the problem of determining whether it is more profitable to offer some of the products together as a package or individually, and what prices should be assigned to the bundles or individual products to maximize profit. The authors formulated the bundle pricing problem as a mixed integer linear programming problem using a disjunctive programming technique [1].

Some research has been done on partitioning customers into segments by the probability that they would buy each product. In [5], the authors proposed a segmentation approach that groups the customers according to their reservation prices and price sensitivity. The probability of a segment choosing a product j is modeled as a multinomial logit model with the segment's reservation price, price sensitivity, and the price of the product j as parameters. Unlike their model, we do not consider price sensitivity in this paper as a criterion when we partition customers into segments and we assume that all segments react to price changes in the same way.

In this paper, we assume that the reservation prices for each customer segment and product are given. Given different models of customer purchasing behavior, we aim to formulate and solve the corresponding revenue maximization problem as a mixed-integer programming problem. In the Appendix, we discuss how we performed the customer segmentation and estimated the reservation prices from customer purchase orders of a Canadian company in the tourism sector.

1.2 Probabilistic Choice Models

In this section, we will introduce the general framework of probabilistic customer choice models that determines the probability that customer segment i will purchase product j, $i = 1, \ldots, n$, $j = 1, \ldots, m$.

Let β_{ij} be binary decision variables where

$$\beta_{ij} := \begin{cases} 1, & \text{if and only if the surplus of product } j \text{ is nonnegative for segment } i, j, \\ 0, & \text{otherwise.} \end{cases}$$

i.e., $\beta_{ij} = 1$ if and only if $R_{ij} - \pi_j \geq 0$ and $\beta_{ij} = 0$ if and only if $R_{ij} - \pi_j < 0$, where, again, π_j is the decision variable for the price of product j. This relationship can be naively modelled by:

$$\begin{aligned} (R_{ij} - \pi_j)\beta_{ij} &\geq 0, \\ (R_{ij} - \pi_j)(1 - \beta_{ij}) &\leq 0, \\ (R_{ij} - \pi_j + 1) &\leq (R_{ij} - \min_i R_{ij} + 1)\beta_{ij}, \end{aligned}$$

for $i = 1, \ldots, n$ and $j = 1, \ldots, m$ (the third inequality is valid under Assumption 1.1). To linearize the above inequalities, we can use a disjunctive programming trick. Let p_{ij} be an auxiliary variable where $p_{ij} = \pi_j \beta_{ij}$, i.e,

$$p_{ij} := \begin{cases} \pi_j, & \text{if } \beta_{ij} = 1, \\ 0, & \text{otherwise.} \end{cases}$$

This relationship can be modeled by the following set of linear inequalities:

$$\begin{aligned} p_{ij} &\geq 0, \\ p_{ij} &\leq \pi_j, \\ p_{ij} &\leq R_{ij}\beta_{ij}, \\ p_{ij} &\geq \pi_j - (\max_{i=1,\ldots,n} R_{ij} + 1)(1 - \beta_{ij}), \end{aligned}$$

for $i = 1, \ldots, n$ and $j = 1, \ldots, m$. The first two inequalities set $p_{ij} = 0$ when $\beta_{ij} = 0$, and the last two inequalities set $p_{ij} = \pi_j$ when $\beta_{ij} = 0$. R_{ij} is a valid upperbound for p_{ij} since if $p_{ij} > R_{ij}$, then $\beta_{ij} = 0$ and thus $p_{ij} = 0$. Also, $\max_{i=1,\ldots,n} R_{ij} + 1$ is a valid upperbound for π_j since no segment will buy product j if $\pi_j > R_{ij}$ for all $i = 1, \ldots, n$.

Here $\boldsymbol{\pi}, \boldsymbol{\beta}$, and $\boldsymbol{p}$ are vectors of π_j, β_{ij} and p_{ij}, respectively; let P be the following polyhedron:

$$
\begin{aligned}
P = \{(\boldsymbol{\pi}, \boldsymbol{\beta}, \boldsymbol{p}) : \quad & R_{ij}\beta_{ij} - p_{ij} \geq 0, && i = 1, \ldots, n, j = 1, \ldots, m, \\
& R_{ij}(1 - \beta_{ij}) - \pi_j \leq 0, && i = 1, \ldots, n, j = 1, \ldots, m, \\
& p_{ij} \leq \pi_j, && i = 1, \ldots, n, j = 1, \ldots, m, \\
& p_{ij} \geq \pi_j - (\max_{i=1,\ldots,n} R_{ij} + 1)(1 - \beta_{ij}), && i = 1, \ldots, n, j = 1, \ldots, m, \\
& R_{ij} - \pi_j + 1 \leq (R_{ij} - \min_i R_{ij} + 1)\beta_{ij}, && i = 1, \ldots, n, j = 1, \ldots, m, \\
& p_{ij} \geq 0, \pi_j \geq 0, && i = 1, \ldots, n, j = 1, \ldots, m\}.
\end{aligned}
\tag{1}
$$

Thus, to model the condition in Assumption 1.2, we need to set prices π_j and β_{ij} such that $\beta \in \{0, 1\}$ and $(\boldsymbol{\pi}, \boldsymbol{\beta}, \boldsymbol{p}) \in P$.

There are ambiguities regarding the choices between multiple products with nonnegative utility. Given all the products with nonnegative surplus, which products would the customer buy? Are there some products they are more likely to buy than others? In a probabilistic choice framework, we need to determine the probability Pr_{ij} that segment i buys product j. Let N_i be the number of customers in segment i. Then the expected revenue for the company is

$$\sum_{i=1}^{n} N_i E[\text{revenue earned from segment } i] = \sum_{i=1}^{n} N_i \sum_{j=1}^{m} \pi_j Pr_{ij}.$$

In our revenue management problem, we can interpret Pr_{ij} as the fraction of customers of segment i that buys product j, i.e., the expected revenue is

$$\sum_{j=1}^{m} \pi_j E[\text{number of customers in segment } i \text{ that buys product } j] = \sum_{j=1}^{m} \pi_j \sum_{i=1}^{n} N_i Pr_{ij}.$$

Furthermore, let Pr_{ij} be positive if and only if the surplus of product j is nonnegative for segment i.

Thus, the expected revenue maximization problem is:

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \sum_{j=1}^{m} N_i \pi_j Pr_{ij}, \\
\text{s.t.} \quad & Pr_{ij} > 0 \Leftrightarrow \beta_{ij} = 1, \quad i = 1, \ldots, n; j = 1, \ldots, m, \\
& Pr_{ij} = 0 \Leftrightarrow \beta_{ij} = 0, \quad i = 1, \ldots, n; j = 1, \ldots, m, \\
& (\boldsymbol{\pi}, \boldsymbol{\beta}, \boldsymbol{p}) \in P, \\
& \beta_{ij} \in \{0, 1\}, \qquad i = 1, \ldots, n; j = 1, \ldots, m.
\end{aligned}
\tag{2}
$$

All the probabilistic choice models explored in this paper are based on the optimization problem (2). What differentiates the models is how Pr_{ij} is defined.

One of the most popular probabilistic choice models in the marketing literature may be the *multinomial logit* (MNL) model,

$$Pr_{ij} := \frac{e^{v_{ij}}}{\sum_k e^{v_{ik}}},$$

where v_{ij} represent the utility or desirability of the product j to segment i. Clearly, there are wide variations in how this v_{ij} is modeled as well. The main motive for the exponential is to allow v_{ij} to take on any real value. For example, an alternative model is to have

$$Pr_{ij} := \frac{v_{ij}}{\sum_k v_{ik}},$$

but we would then require $v_{ij} \geq 0$ and $\sum_k v_{ik} > 0$, which could be easily addressed in many cases.

In this paper, we examine several probabilistic choice models from a mathematical programming perspective. Depending on how Pr_{ij} is modeled, we can formulate the optimization problem (2) as a convex mixed-integer programming problem (MIP). In Section 2, we assume that Pr_{ij} is uniform across all products with nonnegative surplus. We call this model the *Uniform Model*. In Section 3, we modify the Uniform Model so that customers are more likely to purchase products with higher reservation prices. We call this model the *Weighted Uniform Model*. In Section 4, we explore mathematical programming formulations of the Share-of-Surplus Model proposed in [6], including an MIP formulation for the case with restricted prices. Section 5 explores the *Price Sensitive Model* where Pr_{ij} decreases as the price of product j increases. We then discuss properties of the optimal solutions on particular data sets (Section 6) and compare the optimal prices π_j and variables β_{ij} of the different models (Section 7). We also consider enhancements to the models, including heuristics to determine good feasible solutions quickly (Section 8) and valid inequalities to speed up the solution time of the MIPs (Section 9). In Section 10, we show how we can incorporate product capacity limits and product costs into the models. We illustrate some computational results of our models in Section 11 and conclude and discuss future work in Section 12.

Note that proofs for all theorems and lemmas are in Appendix A.

1.3 Notation

Following are common parameters and notation used throughout the paper:

n	number of segments,
m	number of products,
N_i	size of segment i,
R_{ij}	reservation price of segment i for product j,
$\overline{R}_j$	$:= \max_i\{R_{ij}\}$,
$\underline{R}_j$	$:= \min_i\{R_{ij}\}$,
$\widetilde{R}_i$	$:= \max_j\{R_{ij}\}$,
π_j	price of product j (decision variable),
β_{ij}	equals 1 iff $R_{ij} - \pi_j \geq 0$, equals 0 otherwise (decision variable),
P	polyhedron (1).

2 Uniform Model

A very simple model of customer choice behavior is to assume that each segment chooses products with a uniform distribution across all products with nonnegative surplus. We call this model the *Uniform Model*.

2.1 The Formulation

Let β_{ij} be as before. Then in the Uniform Model, the probability that the customer segment i buys product j is

$$Pr_{ij} := \begin{cases} 0, & \text{if } \sum_{j=1}^{m} \beta_{ij} = 0, \\ \frac{\beta_{ij}}{\sum_{k=1}^{m} \beta_{ik}}, & \text{otherwise.} \end{cases}$$

Under this assumption, the expected revenue is $\sum_{i=1}^{n} N_i t_i$ where

$$t_i := \begin{cases} \sum_{j=1}^{m} \pi_j \frac{\beta_{ij}}{\sum_{k=1}^{m} \beta_{ik}} = \frac{\sum_{j=1}^{m} p_{ij}}{\sum_{k=1}^{m} \beta_{ik}}, & \text{if } \sum_{j=1}^{m} \beta_{ij} \neq 0, \\ 0, & \text{otherwise,} \end{cases}$$

where p_{ij} is the auxiliary variable in Section 1.2 such that $p_{ij} := \pi_j \beta_{ij}$. Thus, t_i corresponds to the average price that segment i pays. We reformulate the problem to

$$\begin{array}{lll} \max & \sum_{i=1}^{n} N_i t_i & \\ \text{s.t.} & \sum_{j=1}^{m} \beta_{ij} t_i \leq \sum_{j=0}^{m} p_{ij}, & \forall i, \\ & t_i \leq \widetilde{R}_i \sum_{j=1}^{m} \beta_{ij}, & \forall i, \\ & (\boldsymbol{p}, \boldsymbol{\pi}, \boldsymbol{\beta}) \in P, & \\ & \beta_{ij} \in \{0, 1\}, & \forall i, j. \end{array}$$

Let us introduce yet another auxiliary variable a_{ij} such that $a_{ij} = t_i \beta_{ij}$, i.e., $a_{ij} = t_i$ if $\beta_{ij} = 1$ and $a_{ij} = 0$ otherwise. Then the above formulation can be converted to a linear mixed-integer programming problem

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} N_i t_i, && (3)\\
\text{s.t.} \quad & \textstyle\sum_{j=1}^{m} a_{ij} \le \sum_{j=1}^{m} p_{ij}, && \forall i,\\
& t_i \le \widetilde{R}_i \textstyle\sum_j \beta_{ij}, && \forall i,\\
& a_{ij} \le \widetilde{R}_i \beta_{ij}, && \forall i, \forall j,\\
& a_{ij} \le t_i, && \forall i, \forall j,\\
& a_{ij} \ge t_i - \widetilde{R}_i (1 - \beta_{ij}), && \forall i, \forall j,\\
& (\boldsymbol{p}, \boldsymbol{\pi}, \boldsymbol{\beta}) \in P, && \\
& \beta_{ij} \in \{0, 1\}, && \forall i, j.
\end{aligned}
$$

Theorem 2.1. *Let π_j^*, $j = 1, \dots, m$, be the optimal prices of the Uniform Model. Then, for every product k that is bought, π_k^* equals R_{ik} for some $i = 1, \dots, n$. In particular, let the vectors $\boldsymbol{\pi}^*$ and $\boldsymbol{\beta}^*$ be optimal for Problem (3). If $\sum_{i=1}^{n} \beta_{ik} \ge 1$, then*

$$\pi_k^* = \min_{i:\beta_{ik}^*=1} \{R_{ij}\}.$$

If $\sum_{i=1}^{n} \beta_{ik} = 0$, then $\pi_k^* = \overline{R}_j + 1$.

2.2 Alternative Formulation

Theorem 2.1 motivates the following alternate approach to formulating the Uniform Model. Let us introduce a dummy customer segment, segment 0, where $R_{0j} := \overline{R}_j + 1$ and $N_0 := 0$, and a binary decision variable x_{ij} where:

$$x_{ij} := \begin{cases} 1, & \text{if segment } i \text{ has the smallest reservation price out of all} \\ & \text{segments with nonnegative surplus for product } j, \\ 0, & \text{otherwise.} \end{cases}$$

With the constraint $\sum_{i=0}^{n} x_{ij} = 1$ for all products j, we get

$$\pi_j = \sum_{i=0}^{n} R_{ij} x_{ij}, \qquad \beta_{ij} = \sum_{l:R_{lj} \le R_{ij}} x_{lj}.$$

Thus, the continuous variables p_{ij} and π_j can be eliminated. Using the x_{ij} variables, the objective function of the Uniform Model is

$$\sum_{i=0}^{n} N_i \sum_{j=1}^{m} \left(\sum_{i=0}^{n} R_{ij} x_{ij} \right) \left(\frac{\sum_{l:R_{lj} \le R_{ij}} x_{lj}}{\sum_{k=1}^{m} \sum_{l:R_{lk} \le R_{ik}} x_{lk}} \right) = \sum_{i=0}^{n} N_i \left(\frac{\sum_{j=1}^{m} \sum_{l:R_{lj} \le R_{ij}} R_{lj} x_{lj}}{\sum_{k=1}^{m} \sum_{l:R_{lk} \le R_{ik}} x_{lk}} \right)$$

where the equality follows from $\sum_{i=0}^n x_{ij} = 1, \forall j$, $x_{ij}^2 = x_{ij}$, and $x_{ij}x_{lj} = 0$ for $i \neq l$. Model (3) is thus equivalent to

$$
\begin{array}{llr}
\max & \displaystyle\sum_{i=1}^{n} N_i t_i, & \\
\text{s.t.} & \sum_{i=0}^n x_{ij} = 1, & \forall j, \\
& \sum_{j=1}^m a_{ij} \leq \sum_{j=1}^m \sum_{l:R_{lj}\leq R_{ij}} R_{lj}x_{lj}, & \forall i, \\
& a_{ij} \leq t_i, & \forall i,j, \\
& a_{ij} \leq \widetilde{R}_i \sum_{l:R_{lj}\leq R_{ij}} x_{lj}, & \forall i,j, \\
& a_{ij} \geq t_i - \widetilde{R}_i \sum_{l:R_{lj}> R_{ij}} x_{lj}, & \forall i,j, \\
& t_i \leq \widetilde{R}_i \sum_{j=1}^m \sum_{l:R_{lj}\leq R_{ij}} x_{lj}, & \forall i, \\
& t_i \geq 0, & \forall l,i, \\
& a_{ij} \geq 0, & \forall i,j, \\
& x_{ij} \in \{0,1\}, & \forall i,j.
\end{array}
\tag{4}
$$

2.3 Strength of the Two Formulations

Aside from computational experimentation, we wish to compare the relative "strength" of the original and alternative mixed-integer programming formulations of the Uniform Model. Namely, let us compare the strength of the LP relaxation of formulations (3) and (4).

Let F_1 be the feasible region of the LP relaxation of (3) and let F_2 be that of (4). We compare both formulations on the same data n , m and R_{ij}, $i = 1, \ldots, n$, $j = 1, \ldots, m$. However, note that we add a dummy customer segment 0 in the alternate formulation (4).

Let $\Pi_t(F_k)$ be the projection of the set F_k, $k = 1, 2$, onto the variables t_i, $i = 1, \ldots, n$, i.e.,

$$\Pi_t(F_1) := \{\boldsymbol{t} : \exists(\boldsymbol{\beta}, \boldsymbol{\pi}, \boldsymbol{p}, \boldsymbol{a}) \text{ such that } (\boldsymbol{t}, \boldsymbol{\beta}, \boldsymbol{\pi}, \boldsymbol{p}, \boldsymbol{a}) \in F_1\}$$

and

$$\Pi_t(F_2) := \{\boldsymbol{t} : \exists(t_0, \boldsymbol{x}, \boldsymbol{a}) \text{ such that } (t_0, \boldsymbol{t}, \boldsymbol{x}, \boldsymbol{a}) \in F_2\}$$

where $\boldsymbol{t}$ is the vector of t_i's , $i = 1, \ldots, n$, $\boldsymbol{a}$ is the vector of a_{ij}'s, $i = 1, \ldots, n$, $j = 1, \ldots, m$, $\boldsymbol{\beta}$ is the vector of β_{ij}'s, $i = 1, \ldots, n$, $j = 1, \ldots, m$, $\boldsymbol{\pi}$ is the vector of π_j's, $j = 1, \ldots, m$, $\boldsymbol{p}$ is the vector of p_{ij}'s, $i = 1, \ldots, n$, $j = 1, \ldots, m$, and $\boldsymbol{x}$ is the vector of x_{ij}'s, $i = 1, \ldots, n$, $j = 1, \ldots, m$. The following lemma shows that $\Pi_t(F_2)$ is strictly contained inside $\Pi_t(F_1)$, implying that the optimal objective value of the LP relaxation of (4) is less than or equal to that of (3) in every instance.

Lemma 2.1. $\Pi_t(F_2) \subset \Pi_t(F_1)$ *and the inclusion is strict.*

To test the empirical running time of the Uniform Model, we will use the alternative MIP formulation (4) instead of (3). Section 11 illustrates the running time of the Uniform Model on problem instances of various sizes.

2.4 A Pure 0-1 Formulation

It turns out that the Uniform Model can also be formulated as a pure 0-1 optimization problem. For $k = 0, \ldots, m$, let

$$y_{ik} := \begin{cases} 1, & \text{if segment } i \text{ has exactly } k \text{ products with nonnegative surplus,} \\ 0, & \text{otherwise.} \end{cases}$$

Then, the probability that segment i buys product j is $\sum_{k=1}^{m} \frac{1}{k}\beta_{ij} y_{ik}$ and the Uniform Model can be modeled by the following 0-1 programming problem:

$$\begin{aligned}
\max \quad & \sum_{i=1}^{n} N_i \sum_{j=1}^{m} \sum_{l:R_{lj} \le R_{ij}} R_{lj} \sum_{k=1}^{m} \frac{1}{k} z_{ljik}, && (5) \\
\text{s.t.} \quad & \sum_{i=1}^{n} x_{ij} = 1, && j = 1, \ldots, m, \\
& \sum_{k=0}^{m} y_{ik} = 1, && i = 1, \ldots, n, \\
& \sum_{k=1}^{m} \sum_{j=1}^{m} \sum_{l:R_{l,j} \le R_{i,j}} \frac{1}{k} z_{l,j,i,k} = 1 - y_{i,0}, && i = 1, \ldots, n, \\
& \sum_{j=1}^{m} \sum_{l:R_{l,j} \le R_{ij}} x_{lj} = \sum_{k=0}^{m} k y_{ik}, && i, \ldots, m, \\
& z_{l,j,i,k} \le x_{l,j}, && \forall i, \forall j, k = 1, \ldots, m; l : R_{l,j} \le R_{i,j}, \\
& z_{l,j,i,k} \le y_{i,k}, && \forall i, \forall j, k = 1, \ldots, m; l : R_{l,j} \le R_{i,j}, \\
& z_{l,j,i,k} \ge x_{l,j} + y_{i,k} - 1, && \forall i, \forall j, k = 1, \ldots, m; l : R_{l,j} \le R_{i,j}, \\
& x_{i,j} \in \{0, 1\}, && \forall i, \forall j, \\
& y_{i,k} \in \{0, 1\}, && \forall i, k = 0, \ldots, m, \\
& 0 \le z_{l,j,i,k} \le 1, && \forall i, \forall j, k = 1, \ldots, m; l : R_{l,j} \le R_{i,j}.
\end{aligned}$$

Preliminary Computational Results

To compare the empirical performance of the pure 0-1 formulation (5) and the previous mixed 0-1 formulation (4), we randomly generated multiple instances of reservations prices R_{ij} for

			Uniform Alternate Formulation (4)				Uniform Pure 0-1 Model (5)			
n	m	v	LP optval	SM itn	nodes	time	LP optval	SM itn	nodes	time
4	4	1	2304.79	58	3	0.018	2564.71	206	0	1.320
		2	3447.79	17	0	0.007	**3404.00**	78	0	0.070
		3	333.60	60	0	0.008	**333.00**	62	0	0.050
		4	**3005.67**	25	0	0.002	3060.92	64	0	0.060
		5	3294.81	31	0	0.007	3360.95	103	0	0.090
4	10	1	382.54	157	42	0.065	406.42	4132	83	16.460
		2	381.85	142	3	0.059	398.19	1109	26	365.750
		3	358.60	107	13	0.056	397.36	1859	40	11.860
		4	355.97	105	0	0.037	389.98	496	0	5.230
		5	394.23	90	0	0.029	402.74	267	0	0.630
10	4	1	744.71	196	12	0.106	802.93	4106	52	28.620
		2	845.80	266	35	0.110	856.12	1195	17	803.000
		3	799.50	259	31	0.117	850.95	5320	68	31.880
		4	809.58	159	0	0.033	856.85	972	3	15.600
		5	883.05	99	0	0.031	925.44	1111	21	15.070
10	10	1	985.58	359	36	0.424	997.40	6105	57	240.610
		2	991.44	253	8	0.150	1008.53	5123	45	183.780
		3	1016.35	269	0	0.137	1021.94	1016	0	84.340
		4	825.48	18666	2630	2.762	872.92	139656	1138	1849.010
		5	1014.14	357	19	0.161	1021.50	1309	12	121.720

Table 1: Comparison of Uniform Model formulations (4) and (5) in terms of the objective value of their linear programming relaxation ("LP optval"), total number of dual simplex iterations ("SM itn"), total number of branch-and-bound nodes ("nodes"), and total CPU seconds required to find a provable optimal solution ("time"). n is the number of customer segments, m is the number of products, and v is a label of the problem instance. Bolded LP optval correspond to the integer optimal value.

various n and m. For each (n, m), five random instances were generated. Both models were run with default parameter settings of CPLEX 9.1 and the results are shown in Table 1.

These results clearly show that the mixed-integer formulation (4) is far superior to the pure 0-1 formulation (5) in terms of total running time. This is not surprising since the latter formulation involves significantly more variables, thus the per node computation time is expected to be longer. However, it may be surprising that in almost all cases, the pure 0-1 formulation has a weaker LP relaxation than the mixed-integer formulation and requires more branch-and-bound nodes.

These preliminary computational results may indicate that there is no merit in studying the pure 0-1 formulation. However, since the constraints for (5) can be represented by 0-1 knapsack constraints, there may be strong cover inequalities that can be generated from them. Furthermore, these inequalities can be projected down to the space of x_{ij} variables in the alternate mixed-integer formulation (4). We further explore this idea in Section 9.

3 Weighted Uniform Model

In this section, we modify the Uniform Model of Section 2 so that customers are more likely to purchase a product with higher reservation price. This model, which we call the *Weighted Uniform Model*, is inspired by the multinomial-logit (MNL) model discussed in Section 1.2. Let $v_{ij} = R_{ij}$, but only consider products with nonnegative surplus. Let

$$Pr_{ij} := \begin{cases} 0, & \text{if } \sum_{j=1}^{m} R_{ij}\beta_{ij} = 0, \\ \dfrac{u(R_{ij})\beta_{ij}}{\sum_{k=1}^{m} u(R_{ik})\beta_{ik}}, & \text{otherwise,} \end{cases}$$

where $u(\cdot)$ is a monotonically increasing function of R_{ij}. Thus, with this definition of Pr_{ij}, out of all products with nonnegative surplus, a customer is more likely to buy a product with higher reservation price. In the marketing literature, $u(x) = \exp(x)$ is a common function for the MNL model since $u(x) > 0$ for all $x \in \mathbb{R}, x < \infty$. However, since from Assumption 1.1 $R_{ij} \geq 0, \forall i, j$, we define $u(x) = x$, i.e.,

$$Pr_{ij} := \frac{R_{ij}\beta_{ij}}{\sum_{k=1}^{m} R_{ik}\beta_{ik}}, \quad \text{if } \sum_{j=1}^{m} R_{ij}\beta_{ij} \geq 1.$$

3.1 The Formulation

Analogous to Model (3), the corresponding expected revenue maximizing problem is

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} N_i t_i, && && (6)\\
\text{s.t.} \quad & \textstyle\sum_{j=1}^{m} R_{ij} a_{ij} \le \sum_{j=1}^{m} R_{ij} p_{ij}, && \forall i,\\
& t_i \le \widetilde{R}_i \textstyle\sum_{j=1}^{m} R_{ij} \beta_{ij}, && \forall i,\\
& a_{ij} \le \widetilde{R}_i \beta_{ij}, && \forall i, \forall j,\\
& a_{ij} \le t_i, && \forall i, \forall j,\\
& a_{ij} \ge t_i - \widetilde{R}_i (1 - \beta_{ij}), && \forall i, \forall j,\\
& (\boldsymbol{p}, \boldsymbol{\pi}, \boldsymbol{\beta}) \in P,\\
& \beta_{ij} \in \{0, 1\}, && \forall i, j.
\end{aligned}
$$

3.2 Alternative Formulation

Analogous to the alternate formulation of the Uniform Model in Section 2.2, the Weighted Uniform Model has an alternate formulation using the variables x_{ij}:

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} N_i t_i, && && (7)\\
\text{s.t.} \quad & \textstyle\sum_{i=0}^{n} x_{ij} = 1, && \forall j,\\
& \textstyle\sum_{j=1}^{m} R_{ij} a_{ij} \le \sum_{j=1}^{m} \sum_{l: R_{lj} \le R_{ij}} R_{ij} R_{lj} x_{lj}, && \forall i,\\
& a_{ij} \le t_i, && \forall i, j,\\
& a_{ij} \le \widetilde{R}_i \textstyle\sum_{l: R_{lj} \le R_{ij}} x_{lj}, && \forall i, j,\\
& a_{ij} \ge t_i - \widetilde{R}_i \textstyle\sum_{l: R_{lj} > R_{ij}} x_{lj}, && \forall i, j,\\
& t_i \le \widetilde{R}_i \textstyle\sum_{j=1}^{m} \sum_{l: R_{lj} \le R_{ij}} x_{lj}, && \forall i,\\
& t_i \ge 0, && \forall l, i,\\
& a_{ij} \ge 0, && \forall i, j,\\
& x_{ij} \in \{0, 1\}, && \forall i, j.
\end{aligned}
$$

Like the Uniform Model, this alternate formulation results in a stronger integer programming formulation. Section 11 illustrates the running time of the Weighted Uniform Model (7) on problem instances of various sizes.

4 Share-of-Surplus Model

It seems realistic to assume that the probability of a customer buying a product is related to the surplus. A similar scenario is when a customer prefers buying the product that has the most discount at the moment, rather than picking a product randomly or preferring the product with the highest reservation price. We want a model where higher the surplus, higher the fraction of the segment that buys that product. That is, the probability that a customer buys a product depends on the customer's reservation price as well as the price of the product. A monotonically increasing function is needed to describe the relationship between the probability and the surplus. The *Share-of-Surplus Choice Model* [6] is a form of a probabilistic choice model where the probability that a segment will choose a product is the ratio of its surplus versus the total surplus for the segment across all products with nonnegative surplus.

4.1 The Formulation

In this model, the probability that segment i buys product j is given by:

$$Pr_{ij} := \frac{(R_{ij} - \pi_j)\beta_{ij}}{\sum_k (R_{ik} - \pi_k)\beta_{ik}}.$$

For the moment, let us assume that $\sum_k (R_{ik} - \pi_k)\beta_{ik} > 0$ for all $i = 1, \ldots, n$ for notational simplicity. We will relax this assumption in Section 4.2.

With the above definition, $Pr_{ij} = 0$ if $R_{ij} = \pi_j$, which may not be desirable. To ensure that the probability Pr_{ij} is strictly positive when $R_{ij} = \pi_j$, we may define the probability as follows:

$$Pr^*_{ij} := \frac{(R_{ij} - \pi_j + c)\beta_{ij}}{\sum_k (R_{ik} - \pi_k + c)\beta_{ik}}, \tag{8}$$

where c is a small positive constant. For the sake of simplicity of presentation, we will use the first definition of the probability throughout the rest of this section. Note that this differs from the standard MNL model since we do not consider negative surplus products.

The expected revenue given by this model is

$$\sum_{i=1}^{n} \sum_{j=1}^{m} N_i \pi_j \left(\frac{(R_{ij} - \pi_j)\beta_{ij}}{\sum_k (R_{ik} - \pi_k)\beta_{ik}} \right).$$

We can model this Share-of-Surplus Choice Model as the following nonlinear mixed-integer programming model:

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} \sum_{j=1}^{m} N_i \pi_j \left(\frac{(R_{ij} - \pi_j)\beta_{ij}}{\sum_k (R_{ik} - \pi_k)\beta_{ik}} \right) && (9) \\ \text{s.t.} \quad & (\boldsymbol{\pi}, \boldsymbol{\beta}, \boldsymbol{p}) \in P, \\ & \beta_{ij} \in \{0, 1\}, && i = 1, \ldots, n; j = 1, \ldots, m, \end{aligned}$$

where P is the polyhedron defined in Section 1.2.

The objective function can further be reformulated to a sum of ratios, where the numerator is a concave quadratic and the denominator is linear:

$$\max \sum_{i=1}^{n}\sum_{j=1}^{m} N_i \pi_j \left(\frac{(R_{ij}-\pi_j)\beta_{ij}}{\sum_k (R_{ik}-\pi_k)\beta_{ik}}\right) \quad \Leftrightarrow \quad \max \sum_{i=1}^{n}\sum_{j=1}^{m} N_i \left(\frac{R_{ij}p_{ij}-p_{ij}^2}{\sum_k R_{ik}\beta_{ik}-p_{ik}}\right).$$

Thus, Model (9) can be formulated as the following mixed-integer fractional programming problem with linear constraints:

$$\begin{aligned} \max \quad & \sum_{i=1}^{n}\sum_{j=1}^{m} N_i \left(\frac{R_{ij}p_{ij}-p_{ij}^2}{\sum_k R_{ik}\beta_{ik}-p_{ik}}\right), && \\ \text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{\pi},\boldsymbol{\beta}) \in P, && \\ & \beta_{ij} \in \{0,1\}, && \forall i,j. \end{aligned} \tag{10}$$

The Formulation (10) is a non-convex optimization problem. Unfortunately, there is no apparent convex relaxation of this formulation that yields a tight relaxation. In the next section, we find a mixed-integer programming formulation that approximates the Share-of-Surplus model by restricting the prices.

4.2 Restricted Prices

Unlike the Uniform and Weighted Uniform Models, the computation of optimal prices, given β_{ij}'s, is not immediate for the Share-of-Surplus model. Define $B_i = \{j : \beta_{ij} = 1\}$. Then the optimal prices is the solution to

$$\begin{aligned} \max \quad & \sum_{i=1}^{n}\sum_{j\in B_i} N_i \pi_j \left(\frac{(R_{ij}-\pi_j)}{\sum_{k\in B_i}(R_{ik}-\pi_k)}\right) && \\ \text{s.t.} \quad & R_{ij}-\pi_j \geq 0, && \forall i, j \in B_i, \\ & R_{ij}-\pi_j < 0, && \forall i, j \notin B_i, \\ & \pi_j \geq 0, && \forall j. \end{aligned} \tag{11}$$

If β_{ij} equals one for at least one segment, then we know that

$$\pi_j \in (\max_{i:\beta_{ij}=0} R_{ij}, \min_{i:\beta_{ij}=1} R_{ij}].$$

Suppose product l is bought by at least one segment and its price is increased by $\epsilon > 0$ such that β_{ij}'s do not change. Define $S_j = \{i : \beta_{ij} = 1\}$. Then the change in the objective value is:

$$\sum_{i\in S_l} N_i \left(\frac{\sum_{j\in B_i\setminus\{l\}} \pi_j(R_{ij}-\pi_j) + (\pi_l+\epsilon)(R_{il}-(\pi_l+\epsilon))}{(\sum_{k\in B_i\setminus\{l\}}(R_{ik}-\pi_k)) + (R_{il}-(\pi_l+\epsilon))} - \frac{\sum_{j\in B_i}\pi_j(R_{ij}-\pi_j)}{\sum_{k\in B_i}(R_{ik}-\pi_k)}\right)$$

$$= \sum_{i \in S_l} \epsilon N_i \left(\frac{(R_{il} - (\pi_l + \epsilon)) \sum_{j \in B_i}(R_{ij} - \pi_j) + \sum_{j \in B_i}(\pi_j - \pi_l)(R_{ij} - \pi_j)}{(\sum_{k \in B_i}(R_{ik} - \pi_k))(\sum_{k \in B_i}(R_{ik} - \pi_k) - \epsilon)} \right) \tag{12}$$

Increasing the price of product l by ϵ would result in an increased objective value if (12) is positive. The β_{ij}'s do not change after the price increase, which implies that $R_{il} \geq \pi_l + \epsilon$. Therefore, all the terms in (12) are nonnegative except perhaps $(\pi_j - \pi_l)$. Thus, we can expect (12) to be positive if π_l is relatively low compared to other prices. Intuitively, this means that if π_l is low enough relative to other prices, then we want to raise π_l so that the surplus of product j decreases, hence decreasing the probability that the customers will buy this low-priced product. On the other hand, if π_l is high enough relative to other prices, we want to decrease π_l so that the probability that the customers will buy this expensive product increases, thus generating more revenue.

Suppose we restrict π_j to be equal to $\min_{i:\beta_{ij}=1} R_{ij}$, just as in the Uniform and Weighted Uniform Models. Then the Share-of-Surplus Model can be modeled as a mixed-integer linear programming model. Again, let x_{ij} equal 1 if segment i has the smallest reservation price out of all segments with nonnegative surplus for product j; 0 otherwise. Again, we introduce a dummy segment 0 with $R_{0j} := \overline{R}_j, \forall j$, $N_0 := 0$ and add the constraint $\sum_{i=0}^{n} x_{ij} = 1$. As before, $\beta_{ij} = \sum_{l:R_{lj} \leq R_{ij}} x_{ij}$ and let us restrict π_j to equal $\sum_i R_{ij} x_{ij}$. Then the objective function of the Share-of-Surplus Model is:

$$\sum_{i=0}^{n} \sum_{j=1}^{m} N_i \pi_j \left(\frac{(R_{ij} - \pi_j)\beta_{ij}}{\sum_k (R_{ik} - \pi_k)\beta_{ik}} \right) = \sum_i N_i \left(\frac{\sum_j \sum_{l:R_{lj} \leq R_{ij}} R_{lj}(R_{ij} - R_{lj}) x_{lj}}{\sum_k \sum_{l:R_{lk} \leq R_{ik}} (R_{ik} - R_{lk}) x_{lk}} \right).$$

Let us now relax the assumption that the denominator $\sum_{k=1}^{m} \sum_{l:R_{lk} \leq R_{ik}} (R_{ik} - R_{lk}) x_{lk} > 0$ for all i. Define:

$$t_i := \begin{cases} \frac{\sum_j \sum_{l:R_{lj} \leq R_{ij}} R_{lj}(R_{ij} - R_{lj}) x_{lj}}{\sum_k \sum_{l:R_{lk} \leq R_{ik}} (R_{ik} - R_{lk}) x_{lk}}, & \text{if } \sum_k \sum_{l:R_{lk} \leq R_{ik}} (R_{ik} - R_{lk}) x_{lk} \neq 0, \\ 0, & \text{otherwise.} \end{cases}$$

Let us introduce an auxiliary continuous variable u_{lij} where $u_{lij} := t_i x_{lj}$ for all segments l, i and products j where $R_{lj} \leq R_{ij}$. Then we can formulate the problem as a linear mixed-integer programming problem:

$$\begin{aligned}
\max \quad & \sum_{i=1}^{n} N_i t_i, && (13)\\
\text{s.t.} \quad & \textstyle\sum_{i=0}^{n} x_{ij} = 1, && \forall j,\\
& \textstyle\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} (R_{ij} - R_{lj}) u_{lij} \le \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} R_{lj}(R_{ij} - R_{lj}) x_{lj}, && \forall i,\\
& u_{lij} \le t_i, && \forall l,i,j, R_{lj} \le R_{ij},\\
& u_{lij} \le \widetilde{R}_i x_{lj}, && \forall l,i,j, R_{lj} \le R_{ij},\\
& u_{lij} \ge t_i - \widetilde{R}_i (1 - x_{lj}), && \forall l,i,j, R_{lj} \le R_{ij},\\
& t_i \le \widetilde{R}_i \textstyle\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} (R_{ij} - R_{lj}) x_{lj}, && \forall i,\\
& t_i \ge 0, && \forall l, i,\\
& u_{lij} \ge 0, && \forall l,i,j, R_{lj} \le R_{ij},\\
& x_{ij} \in \{0, 1\}, && \forall i, j.
\end{aligned}$$

If we use the probability Pr^*_{ij} (8) instead, then the objective function is:

$$\sum_i N_i \left(\frac{\sum_j \sum_{l:R_{lj}\le R_{ij}} R_{lj}(R_{ij} - R_{lj} + c) x_{lj}}{\sum_k \sum_{l:R_{lk}\le R_{ik}} (R_{ik} - R_{lk} + c) x_{lk}} \right).$$

Then the problem can be formulated as follows:

$$\max \quad \sum_{i=1}^{n} N_i t_i, \tag{14}$$

$$\text{s.t.} \quad \textstyle\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} (R_{ij} - R_{lj} + c) u_{lij} \le \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} R_{lj}(R_{ij} - R_{lj} + c) x_{lj}, \quad \forall i,$$

$$\begin{aligned}
& \textstyle\sum_{i=0}^{n} x_{ij} = 1, && \forall j,\\
& u_{lij} \le t_i, && \forall l,i,j, R_{lj} \le R_{ij},\\
& u_{lij} \le \widetilde{R}_i x_{lj}, && \forall l,i,j, R_{lj} \le R_{ij},\\
& u_{lij} \ge t_i - \widetilde{R}_i (1 - x_{lj}), && \forall l,i,j, R_{lj} \le R_{ij},\\
& t_i \le \widetilde{R}_i \textstyle\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} (R_{ij} - R_{lj} + c) x_{lj}, && \forall i,\\
& t_i \ge 0, && \forall i,\\
& u_{lij} \ge 0, && \forall l,i,j, R_{lj} \le R_{ij},\\
& x_{ij} \in \{0, 1\}, && \forall i, j.
\end{aligned}$$

If $c < 1$, then we need to replace the constraint

$$t_i \le \widetilde{R}_i \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} (R_{ij} - R_{lj} + c) x_{lj}, \forall i$$

by

$$t_i \leq \frac{1}{c}\widetilde{R}_i \sum_{j=1}^{m} \sum_{l:R_{lj} \leq R_{ij}} (R_{ij} - R_{lj} + c)x_{lj}, \forall i$$

so that the right-hand-side is at least $\widetilde{R}_i$ whenever the summation is non-zero.

The constant c used in the formulation is assumed to be small enough such that the difference between Pr^*_{ij} and Pr_{ij} is almost negligible but that the probability is positive when the surplus is nonnegative. The examination of the effect of the value of c on the problem and the determination of the ideal value for the constant are left for future work.

From experiments, we found that the total computation time of the Share-of-Surplus Model with restricted prices (13) is significantly longer than that of the Uniform and the Weighted Uniform Models. We would like to explore other ways to formulate it or perhaps find cuts in order to decrease the solution time. We may also want to investigate other monotonically increasing functions to describe the probability which would perhaps lead to formulations that are easier to solve. The experimental results are discussed further in Section 11.

5 Price Sensitive Model

A common economic assumption is that as the price of a product decreases, the demand increases. In this section, we discuss a probabilistic choice model where the probability of a customer buying a particular product with nonnegative surplus is inversely proportional to the price of the product.

5.1 The Formulation

Again, let p_{ij} be the auxiliary variable where $p_{ij} := \pi_j \beta_{ij}$. Consider the probability of customer segment i buying product j as defined below:

$$Pr_{ij} := \begin{cases} 0, & \text{if } \beta_{ij} = 0 \text{ (Case 0)}, \\ 1, & \text{if } \beta_{ij} = 1, \sum_k \beta_{ik} = 1 \text{ (Case 1)}, \\ \frac{1}{\sum_k \beta_{ik} - 1}\left(\beta_{ij} - \frac{p_{ij}}{\sum_k p_{ik}}\right), & \text{otherwise (Case 2)}. \end{cases}$$

In this model, $Pr_{ij} = 0$ if product j has a negative surplus for segment i (Case 0), $Pr_{ij} = 1$ if product j is the only product with nonnegative surplus (Case 1), and if there are multiple products with nonnegative surplus (Case 2), Pr_{ij} is inversely proportional to the price of those products. Thus, we call this model the *Price Sensitive Model*. With some reformulation, the expected revenue maximization problem corresponding to this model can be formulated as a second-order cone programming problem with integer variables.

In this model, the expected revenue from segment i, Rev_i, is

$$Rev_i := \begin{cases} 0, & \text{if } \sum_j p_{ij} = 0, \\ \left(\frac{\sum_j p_{ij}}{\sum_j \beta_{ij} - 1 + z_i} - \frac{\sum_j p_{ij}^2}{(\sum_j \beta_{ij} - 1 + z_i)(\sum_k p_{ik})} \right) + (\sum_j p_{ij}) z_i, & \text{otherwise.} \end{cases}$$

Let s_i be an auxiliary variable where $s_i := (\sum_j p_{ij}) z_i$, which we know is a relationship that can be modeled by linear constraints. Also let

$$t_i := \begin{cases} 0, & \text{if } \sum_j p_{ij} = 0, \\ \frac{\sum_j p_{ij}}{\sum_j \beta_{ij} - 1 + z_i} - \frac{\sum_j p_{ij}^2}{(\sum_j \beta_{ij} - 1 + z_i)(\sum_k p_{ik})}, & \text{otherwise.} \end{cases}$$

Then the expected revenue maximization problem corresponding to the Price Sensitive Model is:

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} N_i t_i + \sum_{i=1}^{n} N_i s_i, && (15) \\ \text{s.t.} \quad & \textstyle\sum_j p_{ij}^2 \le (\sum_j p_{ij})^2 - t_i(\sum_j \beta_{ij} - 1 + z_i)(\sum_j p_{ij}), && \forall i, \\ & t_i \le \textstyle\sum_j p_{ij}, && \forall i, \\ & s_i \le \textstyle\sum_j p_{ij}, && \forall i, \\ & s_i \le \textstyle\sum_j R_{ij} z_i, && \forall i, \\ & \textstyle\sum_j \beta_{ij} \le z_i + m(1 - z_i), && \forall i, \\ & z_i \ge \beta_{ij} - \textstyle\sum_{k \ne j} \beta_{ik}, && \forall i, \forall j, \\ & (\boldsymbol{p}, \boldsymbol{\pi}, \boldsymbol{\beta}) \in P, && \\ & \beta_{ij} \in \{0, 1\}, && \forall i, j, \\ & z_i \in \{0, 1\}, s_i \ge 0, && \forall i, j, \end{aligned}$$

where P is the polyhedron (1) defined in Section 1.2.

We need to reformulate the first set of constraints to make the continuous relaxation of (15) a convex programming problem. Let us look at the first set of constraints:

$$\sum_j p_{ij}^2 \le (\sum_j p_{ij})^2 - t_i(\sum_j \beta_{ij} - 1 + z_i)(\sum_j p_{ij}), \quad \forall i. \tag{16}$$

If $t_i > 0$ then $z_i = 0$ and if $z_i = 1$ then $t_i = 0$. Thus, we can eliminate the z_i term from the above inequality if we include the constraint

$$t_i \le \widetilde{R}_i(1 - z_i).$$

Also, let b_{ij} be auxiliary variables where $b_{ij} := t_i \beta_{ij}$. Again, such relations can be modeled by linear constraints. Then, (16) becomes

$$\sum_j p_{ij}^2 \le (\sum_j p_{ij})(\sum_j p_{ij} - \sum_j b_{ij} + t_i), \quad \forall i.$$

Let us further introduce auxiliary variables x_i and y_i such that:

$$x_i + y_i = \sum_j p_{ij} - \sum_j b_{ij} + t_i, \qquad \forall i,$$

$$x_i - y_i = \sum_j p_{ij}, \qquad \forall i.$$

Thus, the constraint becomes

$$\sum_j p_{ij}^2 \leq (x_i + y_i)(x_i - y_i) = x_i^2 - y_i^2$$

Then (16) can be represented by the second-order cone constraints and linear inequalities shown below:

$$\begin{aligned} \sqrt{\textstyle\sum_j p_{ij}^2 + y_i^2} &\leq x_i, && \forall i, \\ x_i + y_i &= \textstyle\sum_j p_{ij} - \sum_j b_{ij} + t_i, && \forall i, \\ x_i - y_i &= \textstyle\sum_j p_{ij}, && \forall i, \\ t_i &\leq \widetilde{R}_i(1 - z_i), && \forall i, \forall j, \\ b_{ij} &\leq \widetilde{R}_i \beta_{ij}, && \forall i, \\ b_{ij} &\leq t_i, && \forall i, \forall j, \\ b_{ij} &\geq t_i - \widetilde{R}_i(1 - \beta_{ij}), && \forall i, \forall j. \end{aligned} \tag{17}$$

The formulation of the Price Sensitive Model becomes:

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} N_i t_i + \sum_{i=1}^{n} N_i s_i, \\ \text{s.t.} \quad & \sqrt{\textstyle\sum_j p_{ij}^2 + y_i^2} \leq x_i, && \forall i, \\ & x_i + y_i = \textstyle\sum_j p_{ij} - \sum_j b_{ij} + t_i, && \forall i, \\ & x_i - y_i = \textstyle\sum_j p_{ij}, && \forall i, \\ & t_i \leq \widetilde{R}_i(1 - z_i), && \forall i, \forall j, \\ & b_{ij} \leq \widetilde{R}_i \beta_{ij}, && \forall i, \\ & b_{ij} \leq t_i, && \forall i, \forall j, \\ & b_{ij} \geq t_i - \widetilde{R}_i(1 - \beta_{ij}), && \forall i, \forall j, \\ & t_i \leq \textstyle\sum_j p_{ij}, && \forall i, \\ & s_i \leq \textstyle\sum_j p_{ij}, && \forall i, \\ & s_i \leq \textstyle\sum_j R_{ij} z_i, && \forall i, \\ & \textstyle\sum_j \beta_{ij} \leq z_i + m(1 - z_i), && \forall i, \\ & z_i \geq \beta_{ij} - \textstyle\sum_{k \neq j} \beta_{ik}, && \forall i, \forall j, \\ & (\boldsymbol{p}, \boldsymbol{\pi}, \boldsymbol{\beta}) \in P, \\ & \beta_{ij} \in \{0, 1\}, && \forall i, j, \\ & z_i \in \{0, 1\}, s_i \geq 0, b_{ij} \geq 0, && \forall i, j. \end{aligned} \tag{18}$$

We can easily eliminate the variables x_i's or y_i's from the above formulation, but we kept them in the above formulation to illustrate the second order cone constraint in a canonical form.

Some preliminary computational results for the Price Sensitive Model are illustrated in Section 11.

6 Special Properties

In this section, we discuss properties of the optimal solutions of our models for data sets with special characteristics.

Lemma 6.1. *Suppose $n \leq m$ and for every segment i, we can find a unique product $p(i)$ such that $R_{ip(i)} = \max_j R_{ij}$. Further suppose that for each of such product $p(i)$, segment i is the unique segment such that $R_{ip(i)} = max_k R_{kp(i)}$.*

Let $J := \{j : j = p(i) \text{ for some segment } i \neq 0\}$. Then in an optimal solution,

$$\beta_{ij} := \begin{cases} 1, & \text{if } j = p(i), \\ 0, & \text{otherwise.} \end{cases}$$

In the alternative formulations, an optimal solution is

$$x_{ij} := \begin{cases} 1, & \text{if } j = p(i), \\ 1, & \text{if } i = 0 \text{ and } j \notin J, \\ 0, & \text{otherwise,} \end{cases}$$

where segment 0 is the dummy segment.

The following lemmas apply to the Uniform Model, the Weighted Uniform Model , and the Share-of-Surplus Model with restricted prices.

Lemma 6.2. *If the optimal values for the x (or β) variables are known, then the optimal prices can be determined. Furthermore, if the optimal prices are known, then the optimal values for the x (or β) variables can be determined.*

Lemma 6.3. *Suppose R_{st} is the maximum reservation price over all segments and products and only one pair of segment and product has that reservation price. Then in any optimal solution, segment s buys product t.*

7 Comparisons

In this section, we compare the optimal solution, in terms of the prices π_j's and β_{ij}'s, of the different models. We notice in most examples, the four models have the same optimal solutions. Of the ones where they have different optimal solutions, usually the Uniform Model, the

Weighted Uniform model and the Price Sensitive model have the same optimal solution, while the Share-of-Surplus Model has a different optimal solution.

Table 2 shows optimal solutions of the models for three small test cases to illustrate the differences in the models. Each sub-table corresponds to a different set of reservations prices. The only difference between the inputs of Test 1 and Test 2 is R_{21}. All the models have the same optimal solution for Test 1, but the Share-of-Surplus Model has a different optimal solution from the other models in Test 2.

Let us consider why the Share-of-Surplus Model has a different optimal solution in Test 2. Clearly, $\beta_{11} = 1$ in an optimal solution in all the models (by Lemma 6.3). If we have $\beta_{12} = 1$ and $\beta_{21} = 1$, then we get more revenue from segment 2. In the Uniform, Weighted Uniform, and Price Sensitive models, this solution gives a higher objective value since $\pi_2 = R_{12}$ is quite high and the probability of segment 1 buying product 2, Pr_{12}, is high enough so that the decrease in revenue from segment 1 is small compared to the revenue from segment 2. Pr_{12} is approximately 0.5, 0.44, and 0.36 in the Uniform, Weighted Uniform, and Price Sensitive Models respectively. It is different with the Share-of-Surplus Model, however, because the surplus of segment 1 for product 1 ($R_{11} - R_{21} = 5$) is relatively high. The probability of segment 1 buying the lower priced product, Pr_{11}, is quite high at 0.86, so the decrease in revenue from segment 1 is greater than the gain in revenue from segment 2. Therefore, the optimal solution in the Share-of-Surplus Model is simply $\beta_{11} = 1$ and all other β's are zero.

Compared to Test 2, the surplus ($R_{11} - R_{21}$) is smaller in Test 1 and also $\pi_1 = R_{21}$ is higher. Therefore, with $\beta_{12} = 1$ and $\beta_{21} = 1$, the decrease in revenue from segment 1 (\$1.75) is smaller than the gain in revenue from segment 2 (\$7) in the Share-of-Surplus Model.

In Test 3, the Weighted Uniform Model has a different optimal solution than the other models. In the other three models, segments 1 and 2 only buy product 1 and segment 3 does not buy any products. This is because the reservation prices of segment 3 are relatively low. If segment 3 buys any product, the revenue from segment 1 and 2 will decrease significantly because of the lower prices and the decrease in revenue cannot be compensated by the extra revenue from segment 3. However, this is not the case in the Weighted Uniform Model. Recall that in the Weighted Uniform Model, the probability of segment i buying product j is proportional to R_{ij}. For both segments 1 and 2, the reservation prices for product 1 are much greater than the reservation prices for product 2. R_{11} and R_{21} are almost double R_{12} and R_{22}, respectively. Therefore, when the price of product 2 is \$22, the probability of segments 1 and 2 buying product 1 at a high price is much greater than the probability of those segments buying product 2. The extra revenue from segment 3 overcompensates the small loss in revenue from the other segments.

We also compare the models' optimal solutions on random data with 5 segments and 5 products in which the reservation prices are uniformly generated from a specified range. The difference in the optimal prices are shown in Tables 16 and 17 in Appendix C. We let 'U', 'W', 'S', and 'P' represent the Uniform, Weighted Uniform, Share-of-Surplus, and Price Sensitive Models respectively. For example, the column "U - W" shows the difference in the optimal prices of

Test 1	$\boldsymbol{R} = \begin{bmatrix} 9 & 8 & 3 \\ 7 & 3 & 2 \\ 1 & 1 & 1 \end{bmatrix}$		
	Price π	β	Obj Value
Uniform	7 8 4	1 1 0 1 0 0 0 0 0	14.50
Share of Surplus	7 8 4	1 1 0 1 0 0 0 0 0	14.25
Weighted Uniform	7 8 4	1 1 0 1 0 0 0 0 0	14.47
Price Sensitive	7 8 4	1 1 0 1 0 0 0 0 0	14.47

Test 2	$\boldsymbol{R} = \begin{bmatrix} 9 & 8 & 3 \\ 4 & 3 & 2 \\ 1 & 1 & 1 \end{bmatrix}$		
	Price π	β	Obj Value
Uniform	4 8 4	1 1 0 1 0 0 0 0 0	10
Share of Surplus	9 9 4	1 0 0 0 0 0 0 0 0	9
Weighted Uniform	4 8 4	1 1 0 1 0 0 0 0 0	9.8824
Price Sensitive	4 8 4	1 1 0 1 0 0 0 0 0	9.3333

Test 3	$\boldsymbol{R} = \begin{bmatrix} 49 & 28 & 27 \\ 46 & 25 & 23 \\ 24 & 22 & 21 \end{bmatrix}$		
	Price π	β	Obj Value
Uniform	46 29 28	1 0 0 1 0 0 0 0 0	92
Share-of-Surplus	46 29 28	1 0 0 1 0 0 0 0 0	92
Weighted Uniform	46 22 28	1 1 0 1 1 0 0 1 0	96.822
Price Sensitive	46 29 28	1 0 0 1 0 0 0 0 0	92

Table 2: Optimal Prices π_j and β_{ij} of all four models on three toy examples. The matrix $\boldsymbol{R}$ corresponds to the reservation prices where the rows correspond to the customer segments and the columns correspond to the products. The column "Price π" corresponds to the optimal prices, the "β" corresponds to the optimal β_{ij} and "Obj Value" corresponds to the optimal objective value.

the Uniform Model and the Weighted Uniform Model. Suppose π_j^1 are the optimal prices of one model and π_j^2 are those of another model. Then the entry in the table is $\sum_{j=1}^{m} |\pi_j^1 - \pi_j^2|$.

The Uniform and the Weighted Uniform Models have the same optimal prices for all these problem instances, probably because it is unlikely in the random data to have the reservation prices for one product to be much larger than those of another product as in Test 3 (Table 2). These two models have the same optimal prices as the Price Sensitive Model except in only two of the problem instances. The same optimal prices (hence, the same optimal β's) imply that the Uniform Model may not be as naïve as it seems since in most cases, it gives the same solutions as the two other more realistic models. However, the Share-of-Surplus Model appears to behave in a special way with results different from the other three models in more cases.

Tables 18 and 19 show the differences in the optimal values of each pair of the models. For example, the column "U - W" is the optimal value of the Uniform Model minus the optimal value of the Weighted Uniform Model. The differences in the optimal values of the Uniform, the Weighted Uniform, and the Price Sensitive Models are quite small in many problem instances, but the Share-of-Surplus Model gives smaller optimal values than the other three models in most cases (the columns "U - S," "W - S," and "P - S" have positive and relatively large entries). It is most likely because the probability for a segment to buy a lower-priced product is usually higher in the Share-of-Surplus Model than in the other three models.

8 Heuristics

As we will see in Section 11, CPLEX takes significant time just to find a feasible solution for larger problems. Fortunately, we can easily find a feasible mixed-integer solution for the formulations of all our models. Thus, we can provide the solver a "good" starting feasible solution in hopes of decreasing the solution time.

8.1 Heuristic 0

One possible strategy, which we call *Heuristic 0*, is to set $\beta_{ij*} = 1$ for each segment i where $R_{ij*} = \max_j R_{ij}$. The other β variables are set accordingly to ensure feasibility. The pseudo-code is presented in Appendix B, Algorithm 1. For some very special data sets, this heuristic is guaranteed to deliver the optimal solution.

Lemma 8.1. *Suppose the conditions are the same as those stated in Lemma 6.1. That is, for every segment i, we can find a unique product $p(i)$ such that $R_{ip(i)} = max_j R_{ij}$, and for each of such product $p(i)$, segment i is the unique segment such that $R_{ip(i)} = max_k R_{kp(i)}$. Then Heuristic 0 gives an optimal solution.*

However, Heuristic 0 may not yield a strong solution in general. For the rest of this section, we discuss a few simple techniques for improving on the feasible solution found by Heuristic 0.

8.2 Heuristic 1

After running Heuristic 0, we select a product k that is bought by at least one customer segment, and let l be the segment with the lowest reservation price that buys product k. We consider the change in the objective value if the segment does not buy product k anymore and perhaps buys another product q that it does not currently buy (i.e., β_{lq} currently equals to 0). This can be thought of as swapping β_{lk} with β_{lq}. We select the option that increases the objective value the most and modify the β variables accordingly. That is, segment l either does not buy product k anymore, or it buys another product instead of product k. If none of the options increases the objective value, we make no changes. We repeat until no swaps can be made to increase the objective value. This algorithm terminates because there are finitely many possible values for the β's and the objective value strictly increases after each swap. The pseudo-code is shown in Appendix B Algorithm 2 and the *swap* subroutine is shown in Algorithm 3.

The order in which we select the products to be examined affects the final solution that will be given by the heuristic. The goal is to use an order that maximizes the total increase in the objective value. In this heuristic, we sort the products by the price and examine the products in the order of the lowest price to the highest price. If we make a change in any iteration, we sort the products again since the prices may change, and start with the lowest-priced product again. The heuristic stops when no changes can be made after examining all the products consecutively from the lowest price to the highest price.

This simple heuristic can be used to find a feasible integral solution for any of the models. The only part that needs to be changed is how the objective value is calculated. The version shown here makes use of β, but it can be easily modified to use the x variables as in the alternative formulation.

8.3 Heuristic 2

Heuristic 1 can be modified to have a polynomial runtime if the price of the product that we examine is non-decreasing in each iteration. From experiments of Heuristic 1, we noticed that if a swap can be made when product k at price π_k is selected, it is very unlikely that a swap can be made for a product at a price lower than π_k in subsequent iterations. Therefore, we would expect the results to be similar if we do not examine products with lower prices again.

Heuristic 2 is the same as Heuristic 1 but the products are selected in a different order. After a customer is swapped out of product k with price π_k before the swap, only products with prices at least π_k are examined. The price of product k increases after a swap, so it will be examined again if there are still customers buying product k. If a new product s is bought and if its new price π_s^{new} is less than π_k, then product s will never be examined. If a product cannot be swapped to increase the objective value, then it will not be examined again. The pseudo-code is presented in Appendix B (Algorithm 4).

Let $O(f(n,m))$ be the runtime to calculate the increase in objective value if segment l does not buy product k anymore or if segment l buys product s instead of product k, where n is the number of customer segments and m is the number of products. Clearly, $f(n,m)$ is polynomial in n and m, since the runtime to calculate the objective value is polynomial.

Lemma 8.2. *The runtime of Heuristic 2 is polynomial.*

8.4 Heuristic 3

Heuristic 3 is a hybrid between Heuristic 1 and Heuristic 2. It examines the products in the same way, but after a swap in which segment l buys product s instead of product k and $\pi_s^{new} < \pi_k$ (equivalently, $R_{ls} < R_{lk}$), it would examine all the products with prices $\geq \pi_s^{new}$. That is, the price of the products that it examines decreases only if a product has a lower price after a swap. The pseudo-code is presented in Appendix B (Algorithm 5).

It is not yet clear if this heuristic has an exponential worst-case runtime. However, experimental results shows that it has a similar runtime as Heuristic 2 and the resulting objective value is usually better (Tables 12, 13, and 14).

8.5 Comparison of the Heuristics

Tables 12, 13, 14, and 15 in Appendix C show the results of the three heuristics with problem instances of different sizes as inputs. The column "n" is the number of segments and "m" is the number of products in the problem instance.

Tables 12 and 13 show the initial objective value found before any swaps (i.e., Heuristic 0), and the number of swaps performed, the number of CPU seconds required and the final objective value found by each heuristic. The objective values are rounded to the nearest integer. Tables 14 and 15 show the difference in time required and the final objective value for each pair of the heuristics. For example, the "Heur. 1 – Heur. 2" columns show the time and objective value of Heuristic 2 subtracted from the time and objective value of Heuristic 1, respectively.

All of the heuristics terminate in a very short time. The time required for Heuristic 1 to terminate increases significantly as the problem size increases. The objective values found are better than or at least as good as the ones found by the other two heuristics, except in one problem instance (when n = 60, m = 20) where Heuristic 2 has a better solution. Experimental results show that Heuristic 3 has a similar runtime as Heuristic 2 and the resulting objective value is usually better. We can see from Tables 14 and 15 that Heuristic 3 found a lower objective value than Heuristic 2 in one problem instance only (when n = 60, m = 20).

The effect of using a starting solution found by the heuristics for the Uniform Model is explored in Section 11.

9 Valid Inequalities

To further improve the solution time for the mixed-integer programming models, we considered several mixed-integer cuts for the various choice models.

9.1 Convex Quadratic Valid Inequalities

In the original Uniform Model (3), the variable a_{ij} were introduced to convexify the bilinear inequalities:

$$\sum_{j=1}^{m} t_i\beta_{ij} \le \sum_{j=1}^{m} p_{ij}, \quad \forall i. \tag{19}$$

We wish to include a convex constraint in the mixed-integer programming formulation that is implied by the above inequalities and some valid convex inequalities.

Let M_i be a positive number (as small as possible) such that $t_i^2 \le M_i$, for every feasible solution $(t_1, \ldots, t_n, \beta_{11}, \ldots, \beta_{nm}, p_{11}, \ldots, p_{nm})$ of the mixed integer programming problem. Also, note that $\beta_{ij}^2 \le \beta_{ij}$. Combining these relations together yields the following set of valid inequalities:

$$a_i t_i^2 + b_i \sum_{j=1}^{m} (\beta_{ij}^2 - \beta_{ij}) + \sum_{j=1}^{m} t_i\beta_{ij} - \sum_{j=1}^{m} p_{ij} \le a_i M_i, \quad i = 1, \ldots, n, \tag{20}$$

where a_i and b_i are nonnegative constants. With appropriate values of a_i and b_i, the above set of quadratic inequalities would represent a convex region.

Lemma 9.1. *The function* $f(t, \beta_1, \ldots, \beta_m, p_1, \ldots, p_m) = at^2 + b\sum_{j=1}^{m}(\beta_j^2 - \beta_j) + \sum_{j=1}^{m} t\beta_j - \sum_{j=1}^{m} p_j$ *is a convex function iff* $a > 0$*,* $b > 0$ *and* $ab \ge \frac{m}{4}$*.*

Next, we generalize the above construction to allow different coefficients b_j for the inequalities $\beta_{ij}^2 \le \beta_{ij}$. Let $\boldsymbol{b}$ denote the vector $(b_1, b_2, \ldots, b_m)^T$ and let $\boldsymbol{B}$ denote the $m \times m$ diagonal matrix with entries $b_1, b_2, \ldots, b_m$ on the diagonal.

Lemma 9.2. *The function* $F(t, \beta_1, \ldots, \beta_m, p_1, \ldots, p_m) = at^2 + \sum_{j=1}^{m} b_j(\beta_j^2 - \beta_j) + \sum_{j=1}^{m} t\beta_j - \sum_{j=1}^{m} p_j$ *is a convex function iff* $\boldsymbol{b} > 0$ *and* $a \ge \sum_{j=1}^{m} \frac{1}{4b_j}$*.*

Corollary 9.1. *Let* M_i *be as above, and* $b_1 > 0, b_2 > 0, \ldots, b_m > 0$*, and* $a \ge \sum_{j=1}^{m} \frac{1}{4b_j}$ *be given. Then the inequality*

$$at_i^2 + \sum_{j=1}^{m} [b_j\beta_{ij}^2 + (t_i - b_j)\beta_{ij} - p_{ij}] \le aM_i$$

is a valid convex quadratic inequality for the feasible region of the mixed integer programming problem.

In the alternate formulation of the Uniform Model (4) with x_{ij} variables instead of β_{ij}'s, the bilinear constraint corresponding to (19) is

$$\sum_{j=1}^{m} t_i \sum_{l:R_{lj}\le R_{ij}} x_{ij} \le \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} R_{lj}x_{ij}, \quad \forall i.$$

As before, we add a times $t_i^2 \le M_i$ and b_{lj} times $x_{lj}^2 \le x_{lj}$ for all j and l such that $R_{lj} \le R_{ij}$ to get a valid convex quadratic inequality for (4).

Corollary 9.2. *Let M_i be as above, and $b_{l1} > 0, b_{l2} > 0, \ldots, b_{lm} > 0$ for l such that $R_{lj} \le R_{ij}$, and $a \ge \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} \frac{1}{4b_{lj}}$ be given. Then the inequality*

$$at_i^2 + \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} [b_{lj}x_{lj}^2 + (t_i - b_{lj} - R_{lj})x_{lj}] \le aM_i$$

is a valid convex quadratic inequality for the feasible region of the mixed integer programming problem. If $b = b_{l1} = b_{l2} = \cdots = b_{lm}$, then we need $ab \ge \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} \frac{1}{4}$.

Clearly, the tighter the upperbound M_i for t_i^2, the stronger the valid convex quadratic inequality. One approach to generate such M_i would be to optimize t_i over the current convex relaxation and square the result. However, such upperbounds for t_i^2 may not be effective. Instead of going after a constant M_i let us consider another upperbound for t_i^2, allowing M_i to be a linear function of the existing variables. For the alternative formulation of the Uniform Model (4), we know that if $\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} x_{lj} \ge 1$ then $t_i = \frac{\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} R_{lj}x_{lj}}{\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} x_{lj}}$ and

$$t_i^2 = \frac{(\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} R_{lj}x_{lj})^2}{(\sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} x_{lj})^2} \le \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} R_{lj}^2 x_{lj},$$

where we used the fact that t_i will be the square of the average of certain R_{ij} values (depending on x_{ij}); clearly, such a value is at most the square of the maximum, which is at most the sum of squares of all such R_{ij} involved in the average. Thus,

Corollary 9.3. *Let $b_{l1} > 0, \ldots, b_{lm} > 0$ for l such that $R_{lj} \le R_{ij}$, and $a \ge \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} \frac{1}{4b_{lj}}$ be given. Then the inequality*

$$at_i^2 + \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} [b_{lj}x_{lj}^2 + (t_i - b_{lj} - R_{lj} - aR_{lj}^2)x_{lj}] \le 0 \tag{21}$$

is a valid convex quadratic inequality for the feasible region of the mixed integer programming problem. If $b = b_{l1} = b_{l2} = \cdots = b_{lm}$, then we need $ab \ge \sum_{j=1}^{m} \sum_{l:R_{lj}\le R_{ij}} \frac{1}{4}$.

We tested the efficacy of the convex quadratic inequality (21) on the Uniform Model (4), where the reservations prices were randomly generated from a uniform distribution. Table 3 illustrates the optimal value of the mixed-integer programming problem (MIP), the optimal value of the corresponding LP relaxation (LP), and the optimal value of the quadratically constrained problem (QCP) resulting from adding the inequalities (21) to the LP relaxation. We see that these inequalities are indeed cuts since the optimal solution of the LP violates them in most instances.

There are four anomalies in Table 3, namely, the instances (n, m, v) with (10, 40, 5), (10, 60, 1), (10, 60, 4) and (10, 60, 5). For each of these instances, the objective value of the QCP relaxation is *strictly less* than the optimal objective value of the MIP. The convex quadratic inequalities were indeed valid for the MIP optimal solution. However, we determined that CPLEX's barrier method returned a suboptimal solution for these QCPs. When observing the details of the CPLEX run, we saw that in each of these four instances the initial solution of the QCP relaxation had primal infeasibility in the order of 10^{10} and dual infeasibility in order of 10^3. CPLEX stopped after 30 to 40 barrier iterations, declaring the current primal solution, with complementarity gap around 10^{-8}, dual infeasibility about 10^{-5} and primal infeasibility around 10^2, as "primal optimal" with "no dual solution available".

While these preliminary experiments show the potential usefulness of our convex cuts, it is also clear that to use them in a robust and effective manner, one needs to work with QCP or second-order cone (SOCP) algorithms that generate dual feasible solutions and use the dual objective function value in the related SOCP-IP computations. We leave the design of such specialized interior-point algorithms to future work.

9.2 Knapsack Covers

The pure 0-1 formulation (5) shown in Section 2.4 may not be as strong as the mixed-integer formulation (4). However, we may be able to exploit the vast amount of work done in developing strong valid inequalities for pure 0-1 programming problems for formulation (5).

One obvious family of valid inequalities are the knapsack covers [8]. From (5), we have the constraints

$$\sum_{j=1}^{m} \beta_{ij} = \sum_{k=0}^{m} k y_{ik}, \quad i = 1, \dots, n,$$

(where we substituted $\beta_{ij} := \sum_{l: R_{l,j} \le R_{ij}} x_{lj}$ purely for notational ease) and

$$\sum_{k=0}^{m} y_{ik} = 1, \quad i = 1, \dots, n.$$

From these, for a given i and k, we get:

$$\sum_{j=1}^{m} \beta_{ij} \le \sum_{l=0}^{k} k y_{il} + \sum_{l=k+1}^{m} m y_{il} = \sum_{l=0}^{k} k y_{il} + \sum_{l=k+1}^{m} m y_{il} + m - m \sum_{k=0}^{m} y_{ik}$$

n	m	v	MIP	LP	With Cut
10	10	1	8980.67	9020.83	9007.12
		2	7576.50	8136.93	8076.50
		3	8656.75	8814.38	8799.53
		4	8767.67	8956.24	8950.98
		5	7369.68	7977.77	7860.84
10	20	1	9658.00	9691.25	9690.69
		2	9373.00	9433.28	9430.34
		3	9276.83	9424.39	9423.05
		4	8603.58	8970.78	8939.18
		5	9473.00	9534.28	9532.12
10	40	1	9777.50	9782.75	9782.05
		2	9798.67	9839.42	9837.89
		3	9788.50	9806.82	9805.72
		4	9592.50	9654.70	9653.06
		5	9771.00	9771.00	9770.49
10	60	1	9836.00	9836.00	9833.83
		2	9836.00	9868.33	9866.21
		3	9860.50	9868.95	9868.08
		4	9865.00	9865.00	9863.50
		5	9854.00	9854.00	9852.26

Table 3: Uniform Model with the convex quadratic valid inequalities (21). n is the number of customer segments, m is the number of products, and v is a label of the problem instance. The column "MIP" is the optimal objective value (4), the column "LP" is the optimal objective value of the LP relaxation of (4), and the column "With Cut" is the optimal objective value of the continuous relaxation of (4) with the convex quadratic inequality (21).

$$\Rightarrow \sum_{j=1}^{m} \beta_{ij} + (m-k)\sum_{l=0}^{k} y_{il} \leq m,$$

where the last inequality is a knapsack constraint (note that $\sum_{l=0}^{k} y_{il} \in \{0,1\}$ in the integer solution so we can treat the term as a 0-1 variable). For a given i and k, let P_{ik} be a subset of $k+1$ products , i.e, $P_{ik} \subseteq \{1,\ldots,m\}$, $|P_{ik}| = k+1$. Thus, the corresponding knapsack cover inequality is

$$\sum_{j\in P_{ik}} \beta_{ij} + \sum_{l=0}^{k} y_{il} \leq k+1. \tag{22}$$

Given a fractional solution to (5), separating (22) can be done in polynomial time. Given x_{ij}'s , and thus β_{ij}'s, we rank β_{ij} for each i, $i = 1,\ldots,n$. For each k, let $P^*_{ik} = \{j :$

β_{ij} is one of the k^{th} largest β_{ij}'s$, j = 1, \ldots, m\}$. Thus, for each i and k, the corresponding cover inequality is violated by the current solution if and only if $\sum_{j \in P_{ik}^*} \beta_{ij} + \sum_{l=0}^{k} y_{il} > k + 1$.

We can also incorporate all of the inequalities (22) into (5) with only polynomial numbers of additional constraints and variables.

Lemma 9.3. *Given i and k, there exists β_{ij}, $j = 1, \ldots, m$ and y_{il}, $l = 0, \ldots, k$ satisfying (22) for all $P_{ik} \subseteq \{1, \ldots, m\}$, $|P_{ik}| = k+1$ if and only if there exists q and p_j, $j = 1, \ldots, m$ such that*

$$\begin{aligned} (k+1)q + \sum_{j=1}^{m} p_j + \sum_{l=0}^{k} y_{il} &\leq k+1, \\ q + p_j &\geq \beta_{ij}, \quad j = 1, \ldots, m, \\ p_j &\geq 0, \quad j = 1, \ldots, m. \end{aligned}$$

Thus, we can either iteratively separate the knapsack cover inequalities, or from Lemma 9.3, add the following constraints to (5):

$$\begin{aligned} (k+1)q_{ik} + \sum_{j=1}^{m} p_{i,j,k} + \sum_{l=0}^{k} y_{il} \leq k+1, &\quad i = 1, \ldots, n; k = 0, \ldots, m, \\ q_{ik} + p_{ijk} \geq \beta_{ij}, &\quad j = 1, \ldots, m; i = 1, \ldots, n; k = 0, \ldots, m, \\ p_{ijk} \geq 0, &\quad j = 1, \ldots, m; i = 1, \ldots, n; k = 0, \ldots, m. \end{aligned} \tag{23}$$

Table 4 illustrates that these knapsack covers (22) are indeed cuts. It compares formulation (5) with and without the cover inequalities (23) in terms of the objective value of their linear programming relaxation on the same randomly generated instances shown in Section 2.4. However, even with all the knapsack cover inequalities, the Pure 0-1 formulation is still weaker than the mixed-integer formulation (4) in most cases.

These knapsack cover inequalities (22) can also be used to generate valid inequalities for the mixed-integer programming formulation (4).

Lemma 9.4. *Suppose $\bar{x}_{ij}$ is a fractional solution of (4) and let $\bar{\beta}_{ij} = \sum_{l: R_{lj} \leq R_{ij}} \bar{x}_{lj}$. For a given i, $i = 1, \ldots, n$, if there are no y_{ik}'s that satisfies*

$$\begin{aligned} \textstyle\sum_{k=0}^{m} y_{ik} &= 1, \\ \textstyle\sum_{k=0}^{m} k y_{ik} &= \textstyle\sum_{j=1}^{m} \bar{\beta}_{ij}, \\ \textstyle\sum_{l=0}^{k} y_{il} &\leq k + 1 - \textstyle\sum_{j \in P_{ik}^*} \bar{\beta}_{ij}, \quad k = 0, \ldots, m \end{aligned} \tag{24}$$

where $P_{ik}^* = \{j : \bar{\beta}_{ij}$ is one of the k largest $\bar{\beta}_{ij}, j = 1, \ldots, m\}$, then

$$\sum_{j=1}^{m} v\beta_{ij} + \sum_{j \in P_{ik}^*} w_k \beta_{ij} \leq \sum_{k=0}^{m} (k+1) w_k \tag{25}$$

n	m	v	MIP (4)	Pure 0-1 (5) without (23)	(5) with (23)
4	4	1	2304.79	2564.71	2399.63
		2	3447.79	**3404.00**	**3404.00**
		3	333.60	**333.00**	**333.00**
		4	**3005.67**	3060.92	**3005.67**
		5	3294.81	3360.95	3271.48
4	10	1	382.54	406.42	390.50
		2	381.85	398.19	391.36
		3	358.60	397.36	373.89
		4	355.97	389.98	365.59
		5	394.23	402.74	384.18
10	4	1	744.71	802.93	799.31
		2	845.80	856.12	853.60
		3	799.50	850.95	848.58
		4	809.58	856.85	842.16
		5	883.05	925.44	911.67
10	10	1	985.58	997.40	990.70
		2	991.44	1008.53	1003.15
		3	1016.35	1021.94	1016.75
		4	825.48	872.92	864.30
		5	1014.14	1021.50	1013.01

Table 4: Objective values of the LP relaxation for the mixed-integer programming formulation (4) and Pure 0-1 formulation with and without the Knapsack Cover inequalities (22), where n is the number of customer segments, m is the number of products, and v is a label of the problem instance. LP objective values in bold corresponds to the IP optimal value.

is a valid inequality for (4) that cuts off $\bar{x}_{ij}$*, where*

$$\begin{aligned} u + kv + \textstyle\sum_{l=0}^{k} w_k \geq 0, & \qquad k = 0, \ldots, m, \\ u + \textstyle\sum_{j=1}^{m} \bar{\beta}_{ij} v + \left(k + 1 - \textstyle\sum_{j \in P_{ik}^*} \bar{\beta}_{ij}\right) w_k < 0, & \\ w_k \geq 0, & \qquad k = 0, \ldots, m, \end{aligned}$$

for some u.

10 Product Capacity and Cost

In all of our discussions thus far, we have assumed that there are no capacity limits nor costs for our products. Clearly, this is not a realistic assumption in many applications. In this section, we discuss how we can incorporate capacity limits and product costs into some of our customer choice models.

10.1 Product Capacity

Product capacity limits are crucial constraints for products such as airline seats and hotel rooms. Certain consumer choice models handle capacity constraints easily, whereas it poses a challenge to others. We present this extension for the Uniform Model, the Weighted Uniform Model, and the Share-of-Surplus Model with restricted prices. We were not able to incorporate the capacity constraint in the Price Sensitive Model while maintaining the convexity of the continuous relaxation. In all of the following subsections, we assume that the company can sell up to Cap_j units of product j, $Cap_j \geq 0$, $j = 1, \ldots, m$.

Uniform and Weighted Uniform Model

Capacity constraints can be incorporated to the mixed-integer formulations of the Uniform Model and the Weighted Uniform Model with some additional variables. We discuss the formulation for the Uniform Model only, since it extends easily to the Weighted Uniform Model.

In the Uniform Model, the expected number of customers that buy product j is $\sum_i N_i \frac{\beta_{ij}}{\sum_k \beta_{ik}}$ if $\sum_k \beta_{ik} \geq 1$ and is 0 if $\sum_k \beta_{ik} = 0$. Let B_{ij} be an auxiliary variable such that $B_{ij} := \frac{\beta_{ij}}{\sum_k \beta_{ik}}$ if $\sum_k \beta_{ik} \geq 1$ and is 0 if $\sum_k \beta_{ik} = 0$, i.e., the fraction of customers from segment i buying product j, Pr_{ij}. Thus, $\beta_{ij} = B_{ij} \sum_k \beta_{ik}$. Let $b_{ijk} := B_{ij}\beta_{ik}$. The capacity constraint can be represented by the following set of linear constraints:

$$
\begin{array}{rl}
\sum_i N_i B_{ij} \leq Cap_j, & \forall j, \\
\beta_{ij} = \sum_k b_{ijk}, & \forall i, \forall j, \\
b_{ijk} \leq \beta_{ik}, & \forall i, \forall j, \forall k, \\
b_{ijk} \geq B_{ij} - (1 - \beta_{ik}), & \forall i, \forall j, \forall k, \\
b_{ijk} \leq B_{ij}, & \forall i, \forall j, \forall k, \\
b_{ijk} \geq 0, & \forall i, \forall j, \forall k.
\end{array}
\tag{26}
$$

The above constraints can also be represented by x_{ij} variables of Section 2.2 instead of the β_{ij} variables.

Share-of-Surplus Model

For the Share-of-Surplus Model with restricted prices (13), the expected number of customers that buy product j is $\sum_i N_i \left(\frac{\sum_{l:R_{lj} \le R_{ij}} (R_{ij} - R_{lj}) x_{lj}}{\sum_k (\sum_{l:R_{lk} \le R_{ik}} (R_{ik} - R_{lk}) x_{lk})} \right)$ if $\sum_k [\sum_{l:R_{lk} \le R_{ik}} (R_{ik} - R_{lk}) x_{lk}] \neq 0$. Let B_{ij} be an auxiliary variable such that

$$B_{ij} := \begin{cases} \left(\frac{\sum_{l:R_{lj} \le R_{ij}} (R_{ij} - R_{lj}) x_{lj}}{\sum_k (\sum_{l:R_{lk} \le R_{ik}} (R_{ik} - R_{lk}) x_{lk})} \right), & \text{if } \sum_k [\sum_{l:R_{lk} \le R_{ik}} (R_{ik} - R_{lk}) x_{lk}] \neq 0 \\ 0, & \text{otherwise.} \end{cases}$$

Again, B_{ij} is the fraction of customers from segment i buying product j, or Pr_{ij}. Thus,

$$\sum_{l:R_{lj} \le R_{ij}} (R_{ij} - R_{lj}) x_{lj} = B_{ij} \sum_k [\sum_{l:R_{lk} \le R_{ik}} (R_{ik} - R_{lk}) x_{lk}].$$

Let $b_{ijlk} := B_{ij} x_{lk}$. Just as before, the capacity constraint can be represented by the following set of linear constraints:

$$\begin{array}{cl} \sum_i N_i B_{ij} \le Cap_j, & \forall j, \\ \sum_{l:R_{lj} \le R_{ij}} (R_{ij} - R_{lj}) x_{lj} = \sum_k [\sum_{l:R_{lk} \le R_{ik}} (R_{ik} - R_{lk}) b_{ijlk}] & \forall i, \forall j, \\ b_{ijlk} \le x_{lk}, & \forall i, \forall j, \forall l, \forall k, \\ b_{ijlk} \ge B_{ij} - (1 - x_{lk}), & \forall i, \forall j, \forall l, \forall k, \\ b_{ijlk} \le B_{ij}, & \forall i, \forall j, \forall l, \forall k, \\ b_{ijlk} \ge 0, & \forall i, \forall j, \forall l, \forall k. \end{array} \tag{27}$$

Risk Products

In some cases, companies may want to penalize against under-shooting a capacity. For example, if there is a large fixed cost or initial investment for product j, the company may sacrifice revenue and decrease its price to ensure that all of the product is sold. We call such products *risk* products. For these products, we may add a penalty for under-shooting in the objective, i.e., given a user-defined penalty coefficient $w_j > 0$ for under-selling product j, we modify the objective to

$$\sum_{i=1}^{n} N_i \sum_{j=1}^{m} \pi_j Pr_{ij} - \sum_{j=1}^{m} w_j (Cap_j - \sum_{i=1}^{n} N_i B_{ij})$$

or

$$\sum_{i=1}^{n} \sum_{j=1}^{m} N_i (\pi_j Pr_{ij} + w_j B_{ij})$$

where B_{ij} is as before.

From a profit optimization point of view, it is sub-optimal to forcibly sell unprofitable products. Such a policy implies that the company is overstocked with these risk products, i.e., Cap_j is too large. In some cases, we may want to treat Cap_j as a variable. For example, in the travel industry, the product procurement division will seek out contracts with hotels to secure certain numbers of rooms for a given time period. However, if that travel destination is not profitable for the company, they may be better off securing very few rooms or not securing any rooms at all. In all of our models, making Cap_j a variable will not affect the linearity of the constraints. Also, there will most likely be an upperbound for Cap_j for all $j = 1, \ldots, m$. If procuring a unit of product j costs v_j, then the objective function can be modified to:

$$\sum_{i=1}^{n} N_i \sum_{j=1}^{m} \pi_j Pr_{ij} - \sum_{j=1}^{m} v_j Cap_j.$$

By determining the optimal value for Cap_j, it should no longer be necessary for the company to penalize under-selling of products[1].

10.2 Product Cost

Suppose each product j has a variable cost of c_j per unit. In the objective function, we want to subtract c_j multiplied by the expected number of customers that buy product j. For all the probabilistic choice models discussed in this paper, the objective function becomes

$$\sum_{i} N_i \sum_{j} (\pi_j - c_j) Pr_{ij}$$

where Pr_{ij} is the probability that the customer segment i buys product j. This is equivalent to lowering all the reservation prices of product j by c_j in all of the models except the Price Sensitive Model. We can also easily incorporate fixed costs for products and capacities (resources) with additional constraints and 0-1 variables.

11 Computational Results

This section illustrates the empirical performances of the Uniform, Weighted Uniform, Share-of-Surplus and Price Sensitive Models on randomly generated and real data sets. The randomly generated reservation prices are generated from a uniform distribution. For each n (number of segments) and m (number of products) pair, five instances were generated. The real data are subsets of reservation prices estimated from actual booking orders of a travel company (our procedure in estimating reservation prices are discussed in Appendix D). There is one instance for each n and m pair. The same data set were used for the Uniform, Weighted Uniform and

[1]It is possible that a company may procure large quantities of a currently non-profitable product to increase their long-term market share. We will not consider such long-term marketing strategy in this paper.

the Share-of-Surplus Model, however the Price Sensitive Model was tested on smaller data sets due to its significantly longer computation time.

The models were run with default parameter settings of CPLEX 9.1 and a time limit of two hours (7200 CPU seconds) unless indicated otherwise. They were run on a machine with four processors and 8 gigabyte of RAM, with at most one process running at a time on each processor.

The Uniform, Weighted Uniform and the Share-of-Surplus Model all began with the solution of Heuristic 1 of Section 8. For every problem instance, the heuristic took at most one CPU second to solve. Although Heuristic 1 found strong feasible integer solutions, it did not significantly improve the total solution time and the total branch-and-bound nodes explored by CPLEX. Thus, even when starting out with a good integer solution, proving optimality was a difficult task for many of these problems.

The tables show the number of segments n, the number of products m, total number of dual simplex iterations ("# of SimplxItns"), total number of branch-and-bound nodes visited ("# of Nodes"), the relative optimality gap when CPLEX was terminated ("FinalGap"), total CPU seconds ("Time"), and the relative gap between the objective value of the best integer solution found by CPLEX versus that of Heuristic 1 ("Heuristic Gap"). For the randomly generated data, the geometric mean over the five instances is illustrated. The column "Provably Optimal" indicates the percentage of the five instances solved to provable optimality within the two hour time limit. For the real data, where there is just one instance for each (n,m) pair, the column "CPLEX Status" indicates whether the problem was solved to provable optimality ("Optimal") or not ("Feasible") within the two hour time limit.

11.1 Uniform Model

We use the alternative formulation (4) of the Uniform Model for all experiments. In this formulation, there are $2nm + n$ variables (mn of them are binary), $2n + m + 3nm$ rows and $m(n + 1)(2n + 1) + 6nm + n$ non-zeros.

Table 5 shows the results of the Uniform Models on the randomly generated data sets. CPLEX solves problems with $n \leq 10$ within seconds to provable optimality, but requires significantly longer time for larger problems. However, even for those problems that CPLEX could not solve to provable optimality within two hours, the optimality gap at termination is very low (largest at 5.58%). Thus, proving optimality seems to be the main difficulty though it succeeds in finding good integer solutions. The heuristic also yields good initial solutions, especially for smaller problems (i.e., $n \leq 20$). For $n = 60$ and $n = 100$, it appears as though the heuristic gives better solution for problems with larger m, but this may be because CPLEX is unable to find better integer solutions for these larger problems.

It is interesting to note that the value of n drives the solution time of the problem. For example, problems with $(n, m) = (10, 100)$ was solved to optimality in 3.98 CPU seconds on average, whereas none of the problems with $(n, m) = (100, 10)$ was solved to provable optimality within two hours. Clearly, the number of variables, number of rows and especially the number of non-zeros differ between the two problem types. For $(n, m) = (10, 100)$, there 2010 variables, 3120 rows, and 29110 non-zeros, whereas $(n, m) = (100, 10)$ has 2100 variables, 3210 rows, and 209110 non-zeros. Perhaps it is the number of non-zeros in the MIP that really drives the solution time.

The heuristic also appears to perform better, in terms of the heuristic gap, in problems where $n \leq m$. For example, the heuristic gaps for $(n, m) = (60, 2)$ and $(n, m) = (100, 2)$ are 20.34% and 50.27%, respectively. However, the heuristic finds the optimal solution for all instances with $n = 2$. This is probably due to properties described in Lemmas 6.1 and 8.1. With randomly generated data and $n \ll m$, the special properties described in the lemmas are more likely to occur.

Table 6 illustrates the results of the Uniform Model on the real data set. It appears that these problems are harder to solve than those with random data. For example, three of the problems with $n = 10$ for the real data did not solve to provable optimality, whereas all the problems with $n = 10$ solved within 4 CPU seconds for the random data. The final optimality gaps are also significantly worse than that of the random data of similar sizes. This is also true for the Heuristic Gap. Thus, it is clear that the relative values of R_{ij}'s are critical in the solution time of these formulations, not just n and m.

11.2 Weighted Uniform Model

We use the alternative formulation (7) of the Weighted Uniform Model for all experiments. The number of variables, number of rows and number of non-zeros are the same as that for the Uniform Model with alternate formulation, namely, there are $2nm + n$ variables (mn of them are binary), $2n + m + 3nm$ rows and $m(n + 1)(2n + 1) + 6nm + n$ non-zeros.

Given that the formulation of the Weighted Uniform Model is very similar to that of the Uniform Model, it is no surprise that the computational results are also very similar. Table 7 shows the results of the Weighted Uniform Model on the randomly generated data set. Again, for most instances, the performance metrics are very similar to that of the Uniform Model. Table 8 illustrates the results on the real data. Just as in the Uniform Model, the optimality gap, running time and the heuristic gap are not as good as with the randomly generated data.

11.3 Share-of-Surplus Model

We use the restricted price formulation of the Share-of-Surplus Model (14) where $c = 1$. In this formulation, there are $\frac{1}{2}mn(n+1) + nm + n$ variables (nm of them being binary), $\frac{3}{2}mn(n+1) + 2n + m$ rows, and $m(n + 1)(5n + 1) + n$ non-zeros.

Table 9 illustrates the results of the Share-of-Surplus model on the randomly generated data.

The running time for this model is significantly longer than that of the Uniform and Weighted Uniform models, especially for large n. This is not completely surprising since the Share-of-Surplus formulation has more variables than the other models (e.g., for $(n, m) = (20, 10)$ the Uniform model has 420 variables, 650 rows and 9830 non-zeros, whereas the Share-of-Surplus model has 2310 variables, 6350 rows and 21230 non-zeros).

However, it was still surprising to find that for several problems, the LP relaxation could not be solved within the two hour time limit. For $(n, m) = (20, 100)$ and $(n, m) = (100, 5)$, the LP relaxation of two out of the five instances could not be solved. For $(n, m) = (60, 10)$, the LP relaxation of three out of the five instances could not be solved. The "Final Gap" value presented for these (n, m) pairs is the geometric mean of the optimality gap over the instances whose LP relaxation was solved. For $(n, m) = (60, 20)$, $(60, 60)$, $(60, 100)$, $(100, 10)$, $(100, 20)$, $(100, 60)$, $(100, 100)$, the LP relaxation of none of the five instances could be solved. These are not terribly large problems – $(n, m) = (20, 100)$ has 23100 variables, 63140 rows, and 212120 non-zeros. Thus, there must be some structural properties that make these LPs difficult to solve.

By observing the CPLEX output when solving the LP relaxation, we noticed frequent occurrences of unscaled infeasibility. CPLEX's preprocessor scales the rows of the mixed-integer programming formulation before solving it, and unscaled infeasibility occurs if the optimal solution found for the scaled problem is not feasible for the original problem. This seems to imply that our problem is ill-conditioned. Consider the constraints in the formulation (14). The reservation prices in the problem instances are generally in the range of 500 to 1500. That is, the coefficients of some of the variables are more than 1500 times the coefficients of other variables, making the problem quite ill-conditioned. We can attempt to solve this problem by scaling the reservation prices before using them in the model since the optimal solution is the same regardless of the unit of the reservation prices. In a future work, we explore the impact of this type of specialized scaling on the total computation time.

Table 10 shows the results with the real data. Similar to the Uniform and Weighted Uniform Model, the real data is more difficult to solve, in general, than the randomly generated data. This is most evident for $n = 10$ and 20, where the optimality gap and the heuristic gap is significantly lower with the real data than with the randomized data.

11.4 Price Sensitive Model

Table 11 shows some preliminary computational results of running small problem instances with the Price Sensitive Model formulation (18).

The first ten cases (t*) each has 3 products and 3 segments. The next six cases (rand*) each has 5 products and 5 segments and the reservation prices are random numbers that range from 500 to 1200. The rest of the cases are subsets of real data and the file name $(n \times m)$ indicates the number of segments and the number of products, respectively, in the inputs. The model was run with default parameter settings of CPLEX 9.1 and a time limit of two hours (7200 CPU seconds).

Since the formulation has a second-order cone constraint, only small problems can be solved quickly. The smaller cases can be solved to optimality fairly quickly, but the solutions for the last two cases ("10×10" and "10×20") found by CPLEX after 2 hours have large optimality gaps.

n	m	Provably Optimal(%)	# of SimplxItns	# of Nodes	Final Gap(%)	Time (CPUsec)	Heuristic Gap(%)
2	2	100	9	0	0.00	0.01	0.00
2	5	100	19	0	0.00	0.01	0.00
2	10	100	31	0	0.00	0.01	0.00
2	20	100	58	0	0.00	0.01	0.00
2	60	100	125	0	0.00	0.03	0.00
2	100	100	154	0	0.00	0.04	0.00
5	2	100	38	0	0.00	0.01	0.00
5	5	100	106	0	0.00	0.03	0.00
5	10	100	142	0	0.00	0.04	0.00
5	20	100	217	0	0.00	0.06	0.00
5	60	100	261	0	0.00	0.08	0.00
5	100	100	573	0	0.00	0.26	0.00
10	2	100	109	0	0.00	0.02	0.00
10	5	100	307	11	0.00	0.16	0.00
10	10	100	2252	0	0.00	0.90	0.00
10	20	100	5134	488	0.00	3.83	0.00
10	60	100	1885	0	0.00	1.26	0.00
10	100	100	3259	0	0.00	3.98	0.00
20	2	100	214	0	0.00	0.04	0.00
20	5	100	4401	454	0.00	1.12	0.00
20	10	100	356098	24550	0.01	64.34	0.56
20	20	60	6150940	232358	0.05	1659.42	0.43
20	60	60	1193300	38903	0.04	1102.32	0.06
20	100	80	73764	0	0.00	141.27	0.00
60	2	100	702	7	0.00	0.41	20.34
60	5	100	265222	11806	0.01	81.94	15.02
60	10	0	12324100	444651	3.01	7200.00	2.32
60	20	0	5982610	184930	2.81	7200.00	0.51
60	60	0	977550	27395	0.98	7200.00	0.00
60	100	0	333850	9169	0.35	7200.00	0.00
100	2	100	1025	9	0.00	0.95	50.37
100	5	80	8067590	273307	0.03	4100.81	19.49
100	10	0	7034250	189925	5.58	7200.00	4.80
100	20	0	1562800	50215	4.75	7200.00	0.00
100	60	0	241156	6956	1.20	7200.00	0.00
100	100	0	86500	836	0.65	7200.00	0.00

Table 5: Uniform Model on Randomly Generated Data.

n	m	CPLEX Status	# of SimplxItns	# of Nodes	Final Gap(%)	Time (CPUsec)	Heuristic Gap(%)
2	2	Optimal	5	0	0.00	0.06	0.00
2	5	Optimal	37	0	0.00	0.06	0.00
2	10	Optimal	28	0	0.00	0.06	0.00
2	20	Optimal	28	0	0.00	0.06	0.00
2	60	Optimal	96	0	0.00	0.08	0.00
2	100	Optimal	125	0	0.00	0.10	0.00
5	2	Optimal	40	1	0.00	0.07	0.00
5	5	Optimal	501	99	0.00	0.15	0.18
5	10	Optimal	14520	3066	0.01	1.61	0.00
5	20	Optimal	124	0	0.00	0.10	0.00
5	60	Optimal	146	0	0.00	0.17	0.00
5	100	Optimal	537	27	0.01	1.17	0.12
10	2	Optimal	123	7	0.00	0.08	2.94
10	5	Optimal	14702	4473	0.01	1.80	0.46
10	10	Optimal	3099905	976193	0.01	478.43	1.47
10	20	Feasible	36883946	4521011	7.01	7200.00	1.19
10	60	Feasible	16801245	3206741	2.27	7200.00	1.24
10	100	Feasible	14817894	2201874	0.69	7200.00	0.00
20	2	Optimal	152	0	0.00	0.13	0.34
20	5	Optimal	148607	33304	0.01	22.45	6.08
20	10	Feasible	33346360	5977052	1.18	7200.00	9.66
20	20	Feasible	20586030	1288903	19.08	7200.00	5.53
20	60	Feasible	9038508	435610	11.20	7200.00	2.93
20	100	Feasible	6982060	250213	5.95	7200.00	13.42
60	2	Optimal	5757	420	0.00	2.10	9.57
60	5	Feasible	21893848	3472747	1.77	7200.00	20.11
60	10	Feasible	15824025	1460216	18.64	7200.00	8.83
60	20	Feasible	8570162	290388	21.44	7200.00	13.74
60	60	Feasible	2180690	43495	25.11	7200.00	14.93
60	100	Feasible	996247	6782	21.66	7200.00	8.28
100	2	Optimal	20970	1360	0.00	6.78	9.71
100	5	Feasible	14311720	2001758	7.43	7200.00	15.85
100	10	Feasible	9416746	644540	18.70	7200.00	16.89
100	20	Feasible	3513959	142581	30.80	7200.00	19.42
100	60	Feasible	785803	5144	26.30	7200.00	15.80
100	100	Feasible	375860	659	30.80	7200.00	0.00

Table 6: Uniform Model on Real Data with Heuristic

n	m	Provably Optimal(%)	# of SimplxItns	# of Nodes	Final Gap(%)	Time (CPUsec)	Heuristic Gap(%)
2	2	100	9	0	0.00	0.01	0.00
2	5	100	8	0	0.00	0.01	0.00
2	10	100	11	0	0.00	0.01	0.00
2	20	100	13	0	0.00	0.01	0.00
2	60	100	24	0	0.00	0.02	0.00
2	100	100	126	0	0.00	0.04	0.00
5	2	100	43	0	0.00	0.01	0.00
5	5	100	99	0	0.00	0.02	0.00
5	10	100	123	0	0.00	0.03	0.00
5	20	100	121	0	0.00	0.05	0.00
5	60	100	199	0	0.00	0.07	0.00
5	100	100	466	0	0.00	0.22	0.00
10	2	100	91	0	0.00	0.03	0.00
10	5	100	272	17	0.00	0.11	0.00
10	10	100	1937	0	0.00	0.60	0.00
10	20	100	3054	306	0.00	2.30	0.00
10	60	100	1647	0	0.00	1.15	0.00
10	100	100	2565	0	0.00	3.46	0.00
20	2	100	241	0	0.00	0.06	0.00
20	5	100	3114	390	0.00	0.84	0.00
20	10	100	218660	18240	0.01	39.31	0.55
20	20	80	2910690	138381	0.02	726.41	0.42
20	60	60	994118	29001	0.03	861.67	0.06
20	100	80	45913	0	0.00	98.81	0.00
60	2	100	1938	106	0.00	0.89	19.75
60	5	100	804911	37932	0.01	202.56	14.83
60	10	0	14506800	420194	2.68	7200.00	2.22
60	20	0	6361410	236692	2.42	7200.00	0.49
60	60	0	1229010	31682	0.92	7200.00	0.00
60	100	0	428255	13577	0.30	7200.00	0.00
100	2	100	2720	122	0.00	2.29	56.39
100	5	20	16598900	594915	1.10	7177.44	19.09
100	10	0	7778290	202018	6.45	7200.00	3.88
100	20	0	1845340	56619	4.39	7200.00	0.00
100	60	0	309144	8180	1.16	7200.00	0.00
100	100	0	93974	1500	0.64	7200.00	0.00

Table 7: Weighted Uniform Model on Randomly Generated Data.

n	m	CPLEX Status	# of SimplxItns	# of Nodes	Final Gap(%)	Time (CPUsec)	Heuristic Gap(%)
2	2	Optimal	3	0	0.00	0.01	0.00
2	5	Optimal	45	0	0.00	0.01	0.00
2	10	Optimal	20	0	0.00	0.01	0.00
2	20	Optimal	18	0	0.00	0.01	0.00
2	60	Optimal	4	0	0.00	0.02	0.00
2	100	Optimal	3	0	0.00	0.03	0.00
5	2	Optimal	42	0	0.00	0.01	0.00
5	5	Optimal	391	96	0.00	0.06	0.18
5	10	Optimal	23329	4752	0.01	2.03	1.09
5	20	Optimal	50	0	0.00	0.02	0.00
5	60	Optimal	90	0	0.00	0.03	0.00
5	100	Optimal	421	70	0.01	0.42	0.10
10	2	Optimal	101	9	0.00	0.03	3.08
10	5	Optimal	12343	3907	0.01	1.41	0.42
10	10	Optimal	3348430	1045730	0.01	507.41	1.04
10	20	Feasible	39369300	6194330	6.74	7200.00	0.99
10	60	Feasible	18731400	3423080	1.45	7200.00	1.30
10	100	Feasible	15682000	2204020	0.58	7200.00	0.00
20	2	Optimal	147	2	0.00	0.06	0.36
20	5	Optimal	123083	28175	0.01	18.33	6.00
20	10	Feasible	36021100	5402250	3.93	7200.00	9.05
20	20	Feasible	24832400	1800900	18.30	7200.00	6.04
20	60	Feasible	10603800	862087	8.79	7200.00	3.17
20	100	Feasible	8833870	694551	4.20	7200.00	11.46
60	2	Optimal	4894	684	0.00	1.46	9.66
60	5	Feasible	22890000	3716240	1.99	7200.00	20.18
60	10	Feasible	15735000	1173870	15.68	7200.00	10.66
60	20	Feasible	8696510	373766	20.35	7200.00	14.73
60	60	Feasible	2530950	116001	18.26	7200.00	16.78
60	100	Feasible	1312000	42641	14.40	7200.00	9.19
100	2	Optimal	17588	2030	0.00	5.44	9.68
100	5	Feasible	14179400	2392980	7.23	7200.00	16.28
100	10	Feasible	9981990	790005	18.38	7200.00	16.71
100	20	Feasible	5245070	175370	26.13	7200.00	20.11
100	60	Feasible	1121010	33001	17.46	7200.00	16.05
100	100	Feasible	625020	21315	16.72	7200.00	7.80

Table 8: Weighted Uniform Model on Real Data.

n	m	Provably Optimal(%)	# of SimplxItns	# of Nodes	Final Gap(%)	Time (CPUsec)	Heuristic Gap(%)
2	2	100	14	0	0.00	0.01	0.00
2	5	100	15	0	0.00	0.01	0.00
2	10	100	33	0	0.00	0.01	0.00
2	20	100	48	0	0.00	0.01	0.00
2	60	100	134	0	0.00	0.03	0.00
2	100	100	203	0	0.00	0.06	0.00
5	2	100	171	0	0.00	0.04	0.00
5	5	100	261	0	0.00	0.10	0.00
5	10	100	320	0	0.00	0.18	0.00
5	20	100	1056	0	0.00	0.48	0.00
5	60	100	600	0	0.00	0.28	0.00
5	100	100	1134	0	0.00	0.75	0.00
10	2	100	1392	41	0.00	0.33	0.00
10	5	100	6969	247	0.00	2.55	1.73
10	10	100	91468	3385	0.00	40.18	0.00
10	20	80	307804	8433	0.03	275.30	0.00
10	60	80	8988	0	0.00	10.54	0.00
10	100	100	6067	0	0.00	19.02	0.00
20	2	100	10312	117	0.00	4.59	0.00
20	5	100	406673	7447	0.00	279.50	1.42
20	10	0	5066530	62721	6.82	7200.00	1.42
20	20	0	1937260	13536	3.41	7200.00	0.00
20	60	0	191894	15	0.40	7200.00	0.00
20	100	20	71510	0	0.00*	1528.05	0.00
60	2	100	519426	2345	0.00	1095.90	21.39
60	5	0	466665	161	29.10	7200.00	0.00
60	10	0	206470	0	14.86**	7200.00	0.00
60	20	0	159213	0	noLP	7200.00	0.00
60	60	0	128080	0	noLP	7200.00	0.00
60	100	0	159868	0	noLP	7200.00	0.00
100	2	0	636346	403	35.18	7200.00	51.89
100	5	0	153009	0	43.76*	7200.00	0.00
100	10	0	139430	0	noLP	7200.00	0.00
100	20	0	135284	0	noLP	7200.00	0.00
100	60	0	173941	0	noLP	7200.00	0.00
100	100	0	171525	0	noLP	7200.00	0.00

Table 9: Share-of-Surplus Model with Restricted Prices on Randomly Generated Data. In column "OptGap", ' *', ' **' , and 'noLP' indicate that the LP relaxation of two out of the five, three out of the five, and all five of the instances, respectively, could not be solved within the two hour time limit.

n	m	CPLEX Status	# of SimplxItns	# of Nodes	Final Gap(%)	Time (CPUsec)	Heuristic Gap(%)
2	2	Optimal	9	0	0.00	0.01	0.00
2	5	Optimal	45	0	0.00	0.01	0.00
2	10	Optimal	29	0	0.00	0.01	0.00
2	20	Optimal	52	0	0.00	0.01	0.00
2	60	Optimal	177	0	0.00	0.03	0.00
2	100	Optimal	216	0	0.00	0.04	0.00
5	2	Optimal	174	14	0.00	0.03	7.40
5	5	Optimal	1366	172	0.00	0.23	1.90
5	10	Optimal	88171	13920	0.01	11.90	3.39
5	20	Optimal	184	0	0.00	0.05	0.00
5	60	Optimal	369	0	0.00	0.21	0.00
5	100	Optimal	30968	17285	0.01	112.99	0.02
10	2	Optimal	1097	44	0.00	0.24	0.98
10	5	Optimal	41285	3391	0.01	7.67	10.39
10	10	Optimal	20140700	1218800	0.01	5328.99	18.76
10	20	Feasible	11489200	678326	13.59	7200.00	1.70
10	60	Feasible	2528930	27068	13.72	7200.00	4.97
10	100	Feasible	1398250	7331	12.28	7200.00	0.00
20	2	Optimal	7378	113	0.00	2.75	0.52
20	5	Optimal	3348750	137019	0.01	1680.05	10.60
20	10	Feasible	5765580	67567	24.20	7200.00	20.22
20	20	Feasible	1944430	16779	34.80	7200.00	11.01
20	60	Feasible	159487	250	40.17	7200.00	0.00
20	100	Feasible	122578	0	42.21	7200.00	0.00
60	2	Optimal	619097	2249	0.00	1089.83	13.97
60	5	Feasible	514955	95	37.08	7200.00	13.39
60	10	Feasible	184065	0	43.32	7200.00	0.00
60	20	Feasible	120800	0	noLP	7200.00	0.00
60	60	Feasible	60800	0	noLP	7200.00	0.00
60	100	Feasible	39400	0	noLP	7200.00	0.00
100	2	Feasible	684915	681	15.93	7200.00	14.29
100	5	Feasible	139800	0	34.75	7200.00	17.14
100	10	Feasible	134000	0	noLP	7200.00	0.00
100	20	Feasible	103800	0	noLP	7200.00	0.00
100	60	Feasible	46900	0	noLP	7200.00	0.00
100	100	Feasible	32700	0	noLP	7200.00	0.00

Table 10: Share-of-Surplus Model with Restricted Prices on Real Data.

Problem Name	CPLEX Status	# of SimplxItns	# of Nodes	Final Gap(%)	Time (CPUsec)
t1	Optimal	861	106	0.00	0.61
t2	Optimal	616	54	0.00	0.74
t3	Optimal	453	48	0.00	0.53
t4	Optimal	980	86	0.00	1.00
t5	Optimal	716	84	0.00	0.69
t6	Optimal	1023	98	0.00	0.76
t7	Optimal	579	46	0.00	0.67
t8	Optimal	980	86	0.00	1.01
t9	Optimal	940	86	0.00	0.78
t10	Optimal	507	52	0.00	0.61
rand1	Optimal	20625	1682	0.00	21.30
rand2	Optimal	12025	735	0.00	14.47
rand3	Optimal	7254	498	0.00	9.67
rand4	Optimal	4899	316	0.00	6.34
rand5	Optimal	7443	474	0.00	10.08
rand6	Optimal	11705	873	0.00	12.85
2x2	Optimal	8	0	0.00	0.01
2x5	Optimal	768	56	0.00	0.59
5x2	Optimal	785	81	0.00	0.54
5x5	Optimal	25662	2151	0.00	23.62
5x10	Optimal	2685098	182506	0.01	4232.98
10x5	Optimal	970831	79289	0.01	1331.47
10x10	Feasible	1655571	105230	71.96	7200.00
10x20	Feasible	168276	7984	85.77	7200.00

Table 11: Price Sensitive Model on Small-Scale Randomly Generated Data.

12 Conclusion

We presented ways to formulate and solve product pricing models using mathematical programming approaches. We have discussed four different probabilistic choice models, all of which are based on reservation prices and are formulated as convex mixed-integer programming problems. The Uniform Distribution Model assumes that Pr_{ij}, the probability that segment i buys product j, is uniform among all products with nonnegative surplus. The Weighted Uniform Model assumes that Pr_{ij} is proportional to the reservation price R_{ij}. In the Share-of-Surplus Model, the probability Pr_{ij} depends on the surplus of the products. Using the assumption that demand increases as price decreases, the Price Sensitive Model uses Pr_{ij} that is inversely proportional to the price of the products with nonnegative surplus. A few special properties of the models have been shown and comparisons of the models' optimal solutions provide some indication of how the models behave. We have proposed and tested a few simple heuristics for finding feasible solutions, which results in strong, often optimal, integer solutions from empirical experiments. Computational results of the various models are also presented and they show that the proposed models are difficult to solve for larger problems.

Further research is required to explore ways to improve the solution time of all the models. For example, more investigations on different cuts is needed, especially on the valid inequalities discussed in Section 9. We also need to examine the structural properties of the problem instances that are especially difficult to solve.

For the Share-of-Surplus Model, we may want to investigate other monotonically increasing functions to describe the probability which would perhaps lead to formulations that are easier to solve. We may examine the effect of the value of the constant c on the problem (14) and determine the ideal value for the constant. In addition, we currently do not fully understand the effect of scaling the reservation prices and this area will be explored further.

All the models discussed in this paper assume that the company has no competitors. We should explore ways to consider competitor products in our models in order to correctly model the loss of revenue when the customers buy from other companies. We can easily incorporate competitor products in our formulations by considering the surplus of every segment for every competitor product. However, this may unrealistically increase the denominator of Pr_{ij} and collecting such detailed competitor information is very difficult. The challenge is to determine how to include competitor information without explicitly considering each competitor product individually.

The motive of this paper is to show how some marketing models of customer choice behavior can be modelled exactly using mixed-integer programming. This work illustrates the modeling power of integer and convex nonlinear programming techniques and we hope to extend our work to other product pricing and customer choice models in the future.

A Supplemental Proofs

Proof of Theorem 2.1. Suppose we are given the optimal β_{ij}'s for Problem (3). Then, the problem simplifies to:

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} N_i t_i, \\
\text{s.t.} \quad & t_i \leq \frac{\sum_{j:\beta_{ij}=1} \pi_j}{\sum_j \beta_{ij}}, && \forall i : \sum_j \beta_{ij} \geq 1, \\
& t_i = 0, && \forall i : \sum_j \beta_{ij} = 0, \\
& \pi_j \leq R_{ij}, && \forall j : \sum_j \beta_{ij} \geq 1, \forall i : \beta_{ij} = 1, \\
& \pi_j \geq \overline{R}_j + 1, && \forall j : \sum_j \beta_{ij} = 0, \\
& \pi_j \leq \overline{R}_j + 1, && \forall j : \sum_j \beta_{ij} = 0.
\end{aligned}
$$

The first constraint is from $\sum_{j=1}^m a_{ij} \leq \sum_{j=1}^m p_{ij}$, where $\sum_{j=1}^m a_{ij} = \sum_{j:\beta_{ij}=1} a_{ij} = t_i \sum_j \beta_{ij}$ and $\sum_{j=1}^m p_{ij} = \sum_{j:\beta_{ij}=1} \pi_j$. The last three sets of constraints are from the inequalities defining P (1). Thus, in the optimal solution, $t_i = \frac{\sum_{j:\beta_{ij}=1} \pi_j}{\sum_j \beta_{ij}}$ if $\sum_j \beta_{ij} \geq 1$ and $t_i = 0$ otherwise. Then, $\pi_j = \min_{i:\beta_{ij}=1} R_{ij}$ if $\sum_i \beta_{ij} \geq 1$ and $\pi_j = \overline{R}_j + 1$ otherwise.

$\square$

Proof of Lemma 2.1. To show the inclusion, suppose $(\hat{t}_0, \hat{\boldsymbol{t}}, \hat{\boldsymbol{x}}, \hat{\boldsymbol{a}}) \in F_2$. Let $\bar{\beta}_{ij} = \sum_{l:R_{lj} \leq R_{ij}} \hat{x}_{lj}$, $\bar{\pi}_j = \sum_{i=1}^n R_{ij} \hat{x}_{ij}$, $\bar{p}_{ij} = \sum_{l:R_{lj} \leq R_{ij}} R_{lj} \hat{x}_{lj}$, $\bar{a}_{ij} = \hat{a}_{ij}$, $i = 1, \ldots, n$, $j = 1, \ldots, m$. It is easy to see that $(\hat{\boldsymbol{t}}, \bar{\boldsymbol{\beta}}, \bar{\boldsymbol{\pi}}, \bar{\boldsymbol{p}}, \bar{\boldsymbol{a}}) \in F_1$. Thus, $\Pi_t(F_2) \subseteq \Pi_t(F_1)$.

To show that the inclusion is strict, let $n = 2$, $m = 2$, $N_1 = N_2 = 1$, and $\boldsymbol{R} = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}$ be the matrix of reservation prices where rows are the segments and products are the columns. F_1 contains the following point:

$$
\boldsymbol{t} = \begin{bmatrix} 2 \\ 0.5 \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} 0 & 1 \\ 0 & 0.5 \end{bmatrix}, \quad \boldsymbol{\pi} = \begin{bmatrix} 2 & 2 \end{bmatrix}, \quad \boldsymbol{p} = \begin{bmatrix} 0 & 2 \\ 0 & 0.5 \end{bmatrix}, \quad \boldsymbol{a} = \begin{bmatrix} 0 & 2 \\ 0 & 0 \end{bmatrix}
$$

where again, the rows correspond to segments and the columns correspond to products. The dummy segment in (4) has reservation prices $R_{01} = 2$ and $R_{02} = 3$. We will show that $\boldsymbol{t} = \begin{bmatrix} 2 \\ 0.5 \end{bmatrix} \notin \Pi_t(F_2)$.

Given $t_1 = 2$ and $t_2 = 0.5$, the fifth set of constraints in (4) are $a_{11} \geq 2 - 2x_{01}$ and $a_{12} \geq 2 - 2x_{02}$. The second set of constraints yield $a_{11} + a_{12} \leq x_{11} + 2x_{12} + x_{22}$. Combining it with the above two inequalities and the first set of constraints gives us $x_{01} + x_{12} + x_{02} - x_{21} \geq 2$, implying $x_{01} + x_{12} + x_{02} \geq 2$. With the first set of constraints, this yields $x_{01} \geq 1$ and $x_{12} + x_{02} \geq 1$, implying $x_{11} = x_{21} = 0$ and $x_{22} = 0$. The sixth set of constraints $t_2 \leq x_{21} + x_{22} = 0$ contradicts $t_2 = 0.5$. Thus, $\boldsymbol{t} = \begin{bmatrix} 2 \\ 0.5 \end{bmatrix} \notin \Pi_t(F_2)$.

□

Proof of Lemma 6.1. The maximum revenue we can get from segment i is $N_i(\max_j R_{ij}) = N_i R_{ip(i)}$. This happens when segment i only buys product $p(i)$. Hence, the objective value of any feasible solution is at most $\sum_{i=1}^{n} N_i R_{ip(i)}$.

Consider the solution with the x variables assigned as in the lemma and

$$\pi_j := \begin{cases} \max_{k=1,\ldots,n} R_{kj}, & \text{if } j \in J, \\ R_{0j}, & \text{otherwise.} \end{cases}$$

Because of the assumptions in the Lemma, the solution is feasible with exactly one segment with nonnegative surplus for each product $j \in J$ and no segment buying any products $j \notin J$. That implies every segment only buys the product with the maximum reservation price. The corresponding objective value is $\sum_{i=1}^{n} N_i R_{ip(i)}$, and thus the solution is optimal. □

Proof of Lemma 6.2. If the x variables are known, then $\pi_j = R_{ij}$ where i is the segment such that $x_{ij} = 1$. If the β variables are known, then $\pi_j = \min_{i:\beta_{ij}=1} R_{ij}$.

If the optimal prices are known, we know that each π_j equals the reservation price of some segment. Then in the optimal solution,

$$x_{ij} := \begin{cases} 1, & \text{if } R_{ij} = \pi_j, \\ 0, & \text{otherwise.} \end{cases} \quad \text{and} \quad \beta_{ij} := \begin{cases} 1, & \text{if } R_{ij} \geq \pi_j, \\ 0, & \text{otherwise.} \end{cases}$$

These are the only values that would make the solution feasible.

□

Proof of Lemma 6.3. Suppose in an optimal solution, $\beta_{st} = 0$. We know that $\beta_{it} = 0$ for all segments i in all three models. Let v be the optimal value. Consider the objective value v' if β_{st} is set to 1. We will have $\pi_t = R_{st}$.

In the Uniform Model, if $\sum_j \beta_{sj} = 0$, then clearly the objective value increases by $N_s R_{st}$. If $\sum_j \beta_{sj} \geq 1$, then

$$v' - v = N_s \left(\frac{R_{st} + \sum_j p_{sj}}{1 + \sum_j \beta_{sj}} - \frac{\sum_j p_{sj}}{\sum_j \beta_{sj}} \right)$$

$$= N_s \left(\frac{R_{st} \sum_j \beta_{sj} - \sum_j p_{sj}}{(\sum_j \beta_{sj})(1 + \sum_j \beta_{sj})} \right)$$

$$= N_s \left(\frac{R_{st} \sum_j \beta_{sj} - \sum_j \pi_j \beta_{sj}}{(\sum_j \beta_{sj})(1 + \sum_j \beta_{sj})} \right)$$

$$= N_s \left(\frac{\sum_j (R_{st} - \pi_j) \beta_{sj}}{(\sum_j \beta_{sj})(1 + \sum_j \beta_{sj})} \right). \tag{28}$$

R_{st} is the maximum reservation price and each of the π_j's equals to a reservation price, so $R_{st} \geq \pi_j \quad \forall j$. The condition $\sum_j \beta_{sj} \geq 1$ implies that $\beta_{sk} = 1$ for some product $k \neq t$, and we know that $R_{st} > \pi_k$. So the expression (28) is strictly positive. This contradicts the fact that it is an optimal solution. Therefore, $\beta_{st} \geq 1$ in every optimal solution.

Similarly, in the Weighted Uniform Model, if $\sum_j \beta_{sj} = 0$, then clearly the objective value increases by $N_s R_{st}$. If $\sum_j \beta_{sj} = 1$, then

$$\begin{aligned} v' - v &= N_s \left(\frac{R_{st}\pi_t + \sum_j R_{sj}\pi_j\beta_{sj}}{R_{st} + \sum_j R_{sj}\beta_{sj}} - \frac{\sum_j R_{sj}\pi_j\beta_{sj}}{\sum_j R_{sj}\beta_{sj}} \right) \\ &= N_s R_{st} \left(\frac{\pi_t \sum_j R_{sj}\beta_{sj} - \sum_j R_{sj}\pi_j\beta_{sj}}{(\sum_j R_{sj}\beta_{sj})(R_{st} + \sum_j R_{sj}\beta_{sj})} \right) \\ &= N_s R_{st} \left(\frac{\sum_j (\pi_t - \pi_j) R_{sj}\beta_{sj}}{(\sum_j R_{sj}\beta_{sj})(R_{st} + \sum_j R_{sj}\beta_{sj})} \right) \\ &> 0, \end{aligned}$$

where $\pi_t = R_{st} > \pi_j, \quad \forall j$.

In the Share-of-Surplus Model with restricted prices, if $\sum_j \beta_{sj} = 0$, then the objective value increases by $N_s R_{st}$. Otherwise,

$$\begin{aligned} v' - v &= N_s \left(\frac{\pi_t(R_{st} - \pi_t + c) + \sum_j \pi_j (R_{sj} - \pi_j + c)\beta_{sj}}{(R_{st} - \pi_t + c) + \sum_j (R_{sj} - \pi_j + c)\beta_{sj}} - \frac{\sum_j \pi_j (R_{sj} - \pi_j + c)\beta_{sj}}{\sum_j (R_{sj} - \pi_j + c)\beta_{sj}} \right) \\ &= N_s (R_{st} - \pi_t + c) \left(\frac{\sum_j (\pi_t - \pi_j)(R_{sj} - \pi_j + c)\beta_{sj}}{(\sum_j (R_{sj} - \pi_j + c)\beta_{sj})((R_{st} - \pi_t + c) + \sum_j (R_{sj} - \pi_j + c)\beta_{sj})} \right) \\ &> 0 \end{aligned}$$

where we let $c > 0$ to avoid singularity.

In all three models, we showed that the solution is not optimal if $\beta_{st} = 0$. So in any optimal solution, segment s buys product t. $\square$

Proof of Lemma 8.1. The proof directly follows from Lemma 6.1. $\square$

Proof of Lemma 8.2. The time it takes to examine a product k is $O(mf(n,m))$ since we consider up to m products that product k can swap with. A product is examined multiple times only if its price increases after a swap. Since a product's price always equals to a segment's reservation price, it can only increase at most n times. So there are at most $O(nm)$ iterations to examine a product, and each iteration has a runtime of $O(mf(n,m))$.

Therefore, the runtime of Heuristic 2 is $O(nm^2 f(n,m))$. $\square$

Proof of Lemma 9.1. The Hessian of f is

$$\nabla^2 f = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0}^T & 0 \end{bmatrix}$$

where $\boldsymbol{A} := \begin{bmatrix} 2a & \boldsymbol{e}^T \\ \boldsymbol{e} & 2\boldsymbol{B} \end{bmatrix}$, $\boldsymbol{B}$ is an $m \times m$ diagonal matrix with $b, b, \ldots, b$ on the diagonal, and $\boldsymbol{e}$ is a vector of ones. Since f is twice continuously differentiable, f is convex iff $\boldsymbol{A}$ is a positive semi-definite matrix.

If $b \leq 0$, then $\boldsymbol{A}$ is not positive semi-definite and f is not convex. So we can assume $b > 0$.

The Schur-complement of $\boldsymbol{B}$ in $\boldsymbol{A}$ is $2a - \frac{1}{2b}(\boldsymbol{e}^T\boldsymbol{e})$, thus

$$\boldsymbol{A} \succeq 0 \quad \Leftrightarrow \quad a - \frac{m}{4b} \geq 0 \quad \Leftrightarrow \quad ab \geq \frac{m}{4}.$$

$\square$

Proof of Lemma 9.2. As in the proof of the previous lemma, F is twice continuously differentiable. The Hessian of F is

$$\nabla^2 F = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0}^T & \boldsymbol{0} \end{bmatrix},$$

where $\boldsymbol{A} := \begin{bmatrix} 2a & \boldsymbol{e}^T \\ \boldsymbol{e} & 2\boldsymbol{B} \end{bmatrix}$. Therefore, F is convex iff $\boldsymbol{A}$ is positive semidefinite. If for some j, $b_j \leq 0$, then $\boldsymbol{A}$ is not positive semidefinite. Therefore, $\boldsymbol{b} > \boldsymbol{0}$. Let $\bar{\boldsymbol{b}} := (\frac{1}{\sqrt{b_1}}, \frac{1}{\sqrt{b_2}}, \ldots, \frac{1}{\sqrt{b_m}})^T$. Also, if $a \leq 0$ then $\boldsymbol{A}$ is not positive semidefinite; thus, $a > 0$. The Schur complement of a in $\boldsymbol{A}$ is $2\boldsymbol{B} - \frac{1}{2a}\boldsymbol{e}\boldsymbol{e}^T$. Thus,

$$\boldsymbol{A} \succeq 0 \quad \Leftrightarrow \quad 2\boldsymbol{B} - \frac{1}{2a}\boldsymbol{e}\boldsymbol{e}^T \succeq 0$$

$$\Leftrightarrow \quad 4I - \frac{1}{a}\bar{\boldsymbol{b}}\bar{\boldsymbol{b}}^T \succeq 0$$

$$\Leftrightarrow \quad 4\bar{\boldsymbol{b}}^T\bar{\boldsymbol{b}} - \frac{1}{a}(\bar{\boldsymbol{b}}^T\bar{\boldsymbol{b}})^2 \geq 0$$

$$\Leftrightarrow \quad a \geq \frac{\bar{\boldsymbol{b}}^T\bar{\boldsymbol{b}}}{4} = \sum_{j=1}^{m} \frac{1}{4b_j}.$$

□

Proof of Lemma 9.3. For given β_{ij}'s, finding the most violated subset P_{ik}^* for (22) is equivalent to solving

$$\begin{array}{rll} \max & \sum_{j=1}^m \beta_{ij} z_j, & \\ \text{s.t.} & \sum_{j=1}^m z_j = k+1, & \\ & 0 \leq z_j \leq 1, & j = 1, \ldots, m. \end{array}$$

Since the feasible region of the above LP is an integral polyhedron, and since the LP is clearly feasible and bounded, it has an optimal 0-1 solution corresponding to the characteristic vector of P_{ik}^*. The Dual of this LP is:

$$\begin{array}{rll} \min & (k+1)q + \sum_{j=1}^m p_j, & \\ \text{s.t.} & q + p_j \geq \beta_{ij}, & j = 1, \ldots, m, \\ & p_j, & j = 1, \ldots, m. \end{array}$$

If there exist β_{ij}'s and y_{il} which satisfy (22) for all covers P_{ik}, then it must satisfy (22) for P_{ik}^*. Thus, from strong duality, there exists q and p_j satisfying the constraints for the Dual LP and $\sum_{j \in P_{ik}^*} \beta_{ij} = (k+1)q + \sum_{j=1}^m p_j$.

Conversely, if there exist q and p_j which satisfy the constraints of the Dual LP and there is a y_{il} such that $(k+1)q + \sum_{j=1}^m p_j + \sum_{l=0}^k y_{il} \leq k+1$, then from weak duality, $\sum_{j \in P_{ik}} \beta_{ij} \leq (k+1)q + \sum_{j=1}^m p_j$ for all P_{ik}'s and thus, $\sum_{j \in P_{ik}} \beta_{ij} + \sum_{l=0}^k y_{il} \leq k+1$ for all P_{ik}'s. □

Proof of Lemma 9.4. The system (24) are valid inequalities for the pure 0-1 formulation (5). Thus, $\hat{x}_{ij}$ is a feasible integer solution to (4) if and only if $\hat{x}_{ij}$ and $\hat{y}_{ik} = 1$ where $k = \sum_{l: R_{lj} \leq R_{ij}} x_{lj}$ is a feasible integer solution to (5).

From Farkas' Lemma, (24) is infeasible if and only if there exist u, v, and w_k, $k = 0, \ldots, m$, where

$$\begin{array}{rl} u + kv + \sum_{l=0}^k w_k \geq 0, & k = 0, \ldots, m, \\ u + \sum_{j=1}^m \bar{\beta}_{ij} v + \left(k + 1 - \sum_{j \in P_{ik}^*} \bar{\beta}_{ij}\right) w_k < 0, & \\ w_k \geq 0, & k = 0, \ldots, m. \end{array}$$

Therefore, $\sum_{j=1}^{m} v\beta_{ij} + \sum_{j\in P_{ik}^*} w_k\beta_{ij} \leq \sum_{k=0}^{m}(k+1)w_k$ is a valid inequality for (4) and it is violated by $\bar{\beta}_{ij}$.

$\square$

B Pseudo-code of Heuristic Algorithms

Algorithm 1 Heuristic 0

heuristic0(numSegments, numProducts, N, R, beta, pi)

for each segment i and product j, set $\beta_{ij} = 0$
for segment $i = 1$ to numSegments **do**
    set $\beta_{ij^*} = 1$ where $j^* = argmax_j R_{ij}$
    for all l such that $R_{lj^*} \geq R_{ij^*}$ **do**
        set $\beta_{lj^*} = 1$
return β

Algorithm 2 Heuristic 1

heuristic1(numSegments, numProducts, N, R, β, π)

β := heuristic0(numSegments, numProducts, N, R, β, π)
k := -1;
repeat
    increase := 0;
    make a heap H where the elements are products and the comparator compares the product prices
    while *increase* ≤ 0 and H is not empty **do**
        j := H.pop()
        increase := swap(j, k, β, π)
until *increase* ≤ 0

Algorithm 3 Make one swap if possible

swap(j, k, β, π)

for seg $i = argmin_i\{R_{ij} : \beta_{ij} = 1\}$, find the greatest increase in objective value if segment i buys another product k (or does not buy any product) instead of j.
if increase in obj val ≤ 0 **then**
    return 0
$\beta_{ij} := 0$
if product k is found **then**
    $\beta_{ik} := 1$
make the solution feasible and set the prices π to the appropriate values
return the increase in objective value after ONE swap

Algorithm 4 Heuristic 2

heuristic2(numSegments, numProducts, N, R, β, π)

β := heuristic0(numSegments, numProducts, N, R, β, π)
increase := 0;
make a heap H where the elements are products and the comparator compares the product prices
while H is not empty **do**
  k := -1;
  $\pi_{temp} := \pi_j$
  j := H.pop()
  increase := swap(j, k, β, π)
  if $increase > 0$ **then**
    if product j is still bought by some segment **then**
      H.push(j)
    if $k \geq 0$ and $\pi_k \geq \pi_{temp}$ **then**
      H.push(k)

Algorithm 5 Heuristic 3

heuristic3(numSegments, numProducts, N, R, β, π)

β := heuristic0(numSegments, numProducts, N, R, β, π)
increase := 0;
make a heap H where the elements are products and the comparator compares the product prices
while H is not empty **do**
  k := -1;
  $\pi_{temp} := \pi_j$
  j := H.pop()
  increase := swap(j, k, β, π)
  if $increase > 0$ **then**
    if product j is still bought by some segment **then**
      H.push(j)
    if $k \geq 0$ **then**
      if $\pi_k \geq \pi_{temp}$ **then**
        H.push(k)
      else
        for products l where $\pi_k \leq \pi_l \leq \pi_{temp}$ **do**
          H.push(l)

C Supplementary Tables

			Heuristic 1			Heuristic 2			Heuristic 3		
n	m	Initial Obj Val	# Swaps	Time	Final Obj Val	# Swaps	Time	Final Obj Val	# Swaps	Time	Final Obj Val
2	2	2656.00	0	0.003	2656	0	0.003	2656	0	0.003	2656
2	5	121520	0	0.003	121520	0	0.002	121520	0	0.003	121520
2	10	165960	0	0.006	165960	0	0.006	165960	0	0.006	165960
2	20	207680	0	0.010	207680	0	0.010	207680	0	0.010	207680
2	60	66801	0	0.025	66801	0	0.024	66801	0	0.024	66801
2	100	66801	0	0.039	66801	0	0.040	66801	0	0.039	66801
5	2	176584	1	0.004	212238	1	0.004	212238	1	0.004	212238
5	5	131346	5	0.020	164038.67	5	0.014	164038.67	5	0.015	164039
5	10	177403	10	0.079	217832	4	0.020	212348	10	0.048	217832
5	20	124311	0	0.022	124311	0	0.022	124311	0	0.022	124311
5	60	377480	0	0.049	377480	0	0.049	377480	0	0.049	377480
5	100	316906	8	0.473	318770	8	0.276	318770	8	0.277	318770
10	2	543760	0	0.003	543760	0	0.003	543760	0	0.004	543760
10	5	307678	3	0.018	323976	3	0.012	323976	3	0.016	323976
10	10	320574	11	0.099	379851	9	0.033	375924	11	0.044	379851
10	20	448921	21	0.365	555829	21	0.114	555829	21	0.134	555829
10	60	489794	11	0.324	624070	10	0.235	624070	10	0.236	624070
10	100	528288	7	0.807	605906	7	0.305	605906	7	0.371	605906

Table 12: Comparison of Heuristics (1)

			Heuristic 1			Heuristic 2			Heuristic 3		
n	m	Initial Obj Val	# Swaps	Time	Final Obj Val	# Swaps	Time	Final Obj Val	# Swaps	Time	Final Obj Val
20	2	503054	8	0.019	544933	8	0.013	544933	8	0.016	544933
20	5	552958	9	0.042	597752	9	0.022	597752	9	0.023	597752
20	10	624238	16	0.128	746544	15	0.048	743491	15	0.055	743491
20	20	698462.78	35	0.951	823206	31	0.148	822773	35	0.209	823206
20	60	934624	48	5.134	1174843	37	0.647	1094627	48	0.958	1174844
20	100	806249	25	4.256	868827	10	0.445	851503	17	0.699	863241
60	2	1136086	6	0.019	1228719	6	0.013	1228719	6	0.014	1228719
60	5	1407568	4	0.028	1526584	4	0.020	1526584	4	0.025	1526584
60	10	1480566	39	0.378	1770829	39	0.115	1770829	39	0.123	1770829
60	20	1697868	49	1.241	1957967	56	0.279	1978848	49	0.286	1957967
60	60	1976530	92	21.621	2476842	88	1.376	2437774	91	1.558	2476277
60	100	2824349	83	52.689	3218884	55	2.385	3171053	76	5.013	3215687
100	2	2201501	9	0.028	2368924	9	0.020	2368924	9	0.020	2368924
100	5	2177005	19	0.103	2493507	19	0.049	2493507	19	0.053	2493507
100	10	2307042	43	0.415	2703764	43	0.146	2703764	43	0.153	2703764
100	20	2384416	45	1.854	2736795	45	0.280	2736795	45	0.285	2736795
100	60	3548047	89	21.130	4128381	88	1.725	4127951	88	1.789	4127951
100	100	3568616	132	108.240	4380587	120	3.938	4338345	126	6.679	4366574

Table 13: Comparison of Heuristics (2)

		Heur. 1 – Heur. 2		Heur. 1 – Heur. 3		Heur. 2 – Heur. 3	
n	m	Time	Obj Val	Time	Obj Val	Time	Obj Val
2	2	0	0	0	0	0	0
2	5	0.001	0	0	0	-0.001	0
2	10	0	0	0	0	0	0
2	20	0	0	0	0	0	0
2	60	0.001	0	0.001	0	0	0
2	100	-0.001	0	0	0	0.001	0
5	2	0	0	0	0	0	0
5	5	0.007	0	0.006	0	-0.001	0
5	10	0.060	5484	0.031	0	-0.028	-5484
5	20	0	0	0	0	0	0
5	60	0	0	0	0	0	0
5	100	0.197	0	0.196	0	-0.001	0
10	2	0	0	-0.001	0	-0.001	0
10	5	0.006	0	0.002	0	-0.004	0
10	10	0.065	3927	0.055	0	-0.011	-3927
10	20	0.251	0	0.231	0	-0.020	0
10	60	0.089	0	0.088	0	-0.001	0
10	100	0.503	0	0.436	0	-0.066	0
20	2	0.006	0	0.003	0	-0.003	0
20	5	0.020	0	0.019	0	-0.001	0
20	10	0.080	3053	0.073	3053	-0.007	0
20	20	0.802	433	0.742	0	-0.061	-433
20	60	4.487	80216	4.175	0	-0.311	-80216
20	100	3.811	17324	3.558	5586	-0.254	-11738

Table 14: Comparison of Heuristics (3)

		Heur. 1 – Heur. 2		Heur. 1 – Heur. 3		Heur. 2 – Heur. 3	
n	m	Time	Obj Val	Time	Obj Val	Time	Obj Val
60	2	0.006	0	0.005	0	-0.001	0
60	5	0.008	0	0.003	0	-0.005	0
60	10	0.263	0	0.255	0	-0.008	0
60	20	0.962	-20881	0.956	0	-0.007	20881
60	60	20.245	39068	20.064	565	-0.182	-38503
60	100	50.304	47831	47.677	3197	-2.627	-44634
100	2	0.009	0	0.008	0	-0.001	0
100	5	0.055	0	0.051	0	-0.004	0
100	10	0.268	0	0.262	0	-0.007	0
100	20	1.574	0	1.569	0	-0.005	0
100	60	19.406	430	19.341	430	-0.064	0
100	100	104.302	42242	101.562	14013	-2.741	-28229

Table 15: Comparison of Heuristics (4)

		Difference in Prices			
Range	Test #	U - W	U - S	U - P	S - P
1000-1100	1	0	0	0	0
	2	0	0	0	0
	3	0	0	0	0
	4	0	0	0	0
	5	0	1	0	1
	6	0	0	0	0
	7	0	4	1	3
	8	0	0	0	0
	9	0	0	0	0
	10	0	0	0	0
1000-1500	1	0	0	0	0
	2	0	126	0	126
	3	0	0	0	0
	4	0	35	0	35
	5	0	73	0	73
	6	0	356	0	356
	7	0	0	0	0
	8	0	45	0	45
	9	0	0	0	0
	10	0	0	0	0
1000-1700	1	0	0	0	0
	2	0	146	0	146
	3	0	11	0	11
	4	0	0	0	0
	5	0	389	389	0
	6	0	0	0	0
	7	0	0	0	0
	8	0	0	0	0
	9	0	0	0	0
	10	0	105	0	105

Table 16: Comparison of the Prices in the Models' Solutions for Random Tests (1)

		Difference in Prices			
Range	Test #	U - W	U - S	U - P	S - P
1000-2000	1	0	199	0	199
	2	0	257	0	257
	3	0	0	0	0
	4	0	0	0	0
	5	0	0	0	0
	6	0	0	0	0
	7	0	0	0	0
	8	0	768	0	768
	9	0	156	0	156
	10	0	0	0	0
1000-3000	1	0	10	0	10
	2	0	0	0	0
	3	0	1	1	0
	4	0	716	0	716
	5	0	0	0	0
	6	0	0	0	0
	7	0	137	0	137
	8	0	0	0	0
	9	0	0	0	0
	10	0	0	0	0

Table 17: Comparison of the Prices in the Models' Solutions for Random Tests (2)

		Difference in Objective Value					
Range	Test #	U - W	U - S	U - P	W - S	W - P	P - S
1000-1100	1	-0.01	1.00	0.01	1.01	0.02	0.99
	2	0	0	0	0	0	0
	3	0.07	45.09	0.49	45.02	0.42	44.60
	4	-0.04	4.66	0.02	4.70	0.06	4.64
	5	-0.05	12.18	0.24	12.23	0.29	11.93
	6	-0.38	21.90	0.28	22.28	0.66	21.62
	7	-0.10	32.63	0.66	32.73	0.76	31.97
	8	0	0	0	0	0	0
	9	0.07	9.39	0.14	9.32	0.07	9.25
	10	-0.10	24.17	0.18	24.27	0.28	23.99
1000-1500	1	-1.36	-20.63	0.62	-19.27	1.98	-21.26
	2	-5.17	132.60	10.96	137.77	16.13	121.65
	3	-0.31	44.17	1.65	44.48	1.96	42.53
	4	-0.22	9.00	1.42	9.22	1.64	7.58
	5	0.22	14.00	0.23	13.78	0.01	13.77
	6	-1.30	117.19	5.10	118.49	6.40	112.09
	7	0	0	0	0	0	0
	8	-7.00	112.90	8.77	119.90	15.77	104.13
	9	0	0	0	0	0	0
	10	-0.01	8.44	0.08	8.45	0.09	8.36
1000-1700	1	-2.05	73.47	3.61	75.52	5.66	69.86
	2	0.27	55.86	1.40	55.59	1.13	54.46
	3	0.22	15.50	0.44	15.28	0.22	15.06
	4	-0.12	-5.75	0.03	-5.63	0.15	-5.78
	5	-8.93	83.78	2.76	92.71	11.69	81.03
	6	0	0	0	0	0	0
	7	0	0	0	0	0	0
	8	-3.33	136.89	3.78	140.22	7.11	133.11
	9	0.50	30.60	0.65	30.10	0.15	29.94
	10	-4.74	113.11	7.19	117.85	11.93	105.92

Table 18: Comparison of the Objective Values for Random Tests (1)

		Difference in Objective Value					
Range	Test #	U - W	U - S	U - P	W - S	W - P	P - S
1000-2000	1	-9.94	201.44	19.48	211.38	29.42	181.96
	2	-19.04	487.80	83.00	506.84	102.04	404.80
	3	0	0	0.01	0	0.01	-0.01
	4	0	0	0	0	0	0
	5	0.52	18.24	0.20	17.72	-0.32	18.05
	6	0	0	0	0	0	0
	7	-4.44	113.49	6.21	117.93	10.65	107.28
	8	-2.11	185.00	23.07	187.11	25.18	161.93
	9	-1.93	18.43	1.12	20.36	3.05	17.31
	10	0	0	0.01	0	0.01	-0.01
1000-3000	1	-0.90	81.00	3.43	81.90	4.33	77.57
	2	-0.20	33.77	0.37	33.97	0.57	3.39
	3	-0.20	208.21	8.75	208.41	8.95	199.47
	4	-35.80	203.74	21.62	239.54	57.42	182.13
	5	-2.50	177.71	11.55	180.21	14.05	166.16
	6	0	0	0.01	0	0.01	-0.01
	7	12.20	326.37	30.57	314.17	18.37	295.79
	8	3.80	54.62	11.83	50.82	8.03	42.80
	9	0	0	0.01	0	0.01	-0.01
	10	-17.60	180.03	10.80	197.63	28.40	169.23

Table 19: Comparison of the Objective Values for Random Tests (2)

D Estimating the Reservation Price

The reservation price data used in the computational experiments of Section 11 are estimated from actual purchase orders of a Canadian travel company. The customers are partitioned into segments according to their demographic information, purchase lead time and other characteristics. Suppose after the segmentation, there are n customers, with N_i customers in segment i, $i = 1, \ldots, n$. The company offers m products.

From the historical data, we know what fraction of customers of each segment purchased each product and how much they paid for it. Let

$$\begin{aligned} fr_{ij} &:= \text{ the fraction of segment } i \text{ customers who purchased product } j, \\ B_i &:= \{j : fr_{ij} > 0\} \text{ , i.e., set of products purchased by segment } i, \\ p_{ij} &:= \text{ the price that customers of segment } i \text{ paid for product } j. \end{aligned}$$

The price paid for a particular product may be slightly different from customer to customer depending on the time of sales and other anomalies. Thus, the above p_{ij} value is the average price paid by segment i for product j.

To estimate the reservation price R_{ij} of segment i for product j, we assumed that customers behaved according to the share-of-surplus model of Section 4. Thus, fr_{ij} should be approximately equal to

$$\frac{R_{ij} - p_{ij}}{\sum_{k \in B_i} R_{ik} - p_{ik}},$$

where R_{ij}'s are now variables and p_{ij}'s are data.

We fit R_{ij}'s and the share-of-surplus model to the data using least squares regression, i.e., for each segment i, we solved for R_{ij}'s, $j = 1, \ldots, m$, that minimizes

$$\sum_{j \in B_i} \left(f_{ij} - \frac{R_{ij} - p_{ij}}{\sum_{k \in B_i} R_{ik} - p_{ik}} \right)^2$$

or

$$\sum_{j \in B_i} \left(f_{ij} (\sum_{k \in B_i} R_{ik} - p_{ik}) - R_{ij} - p_{ij} \right)^2$$

subject to

$$\begin{aligned} R_{ij} - p_{ij} &\geq 0, \quad j \in B_i, \\ \textstyle\sum_{k \in B_i} R_{ik} - p_{ik} &\geq \delta \end{aligned}$$

where $\delta > 0$.

There are some further details that need to be addressed. One of the key issues is estimating R_{ij} for $j \notin B_i$. Currently, we have these R_{ij}'s set to 0, which is clearly an underestimate.

Although we do not have any direct information about segment i's preference level of product j, we may be able to infer this from other segments that do purchase product j. As a future work, we can consider using data mining techniques such as clustering and collaborative filtering to determine these R_{ij}'s.

References

[1] E. Balas. Disjunctive programming and a hierarchy of relaxations for discrete optimization problems. *1997,SIAM J. Alg. Disc. Meth.*, 6:466–486, 1985.

[2] G. Dobson and S. Kalish. Positioning and pricing a product line. *Marketing Science*, 7:107–125, 1988.

[3] G. Dobson and S. Kalish. Heuristics for pricing and positioning a product-line using conjoint and cost data. *Management Science*, 39:160–175, 1993.

[4] W. Hanson and R.K. Martin. Optimal bundle pricing. *Management Science*, 36:155–174, 1990.

[5] W.A. Kamakura and G.J. Russell. A probabilistic choice model for market segmentation and elasticity structure. *Journal of Marketing Research*, 26:379–390, 1989.

[6] U.G. Kraus and C.A. Yano. Product line selection and pricing under a share-of-surplus choice model. *European Journal of Operational Research*, 150:653–671, 2003.

[7] J. I. McGill and G. J. Van Ryzin. Revenue management: Research overview and prospects. *Transportation Science*, 33:233–256, 1999.

[8] G. L. Nemhauser and L.A. Wolsey. *Integer Programming and Combinatorial Optimization*. John Wiley and Sons, 1999.

[9] G. Van Ryzin and K.T. Talluri. *The Theory and Practice of Revenue Management*. Kluwer Academic Publishers, 2004.