A Framework for Automated Bundling and Pricing Using Purchase Data

Michael Benisch and Tuomas Sandholm

School of Computer Science, Carnegie Mellon University

Abstract. We present a framework for automatically suggesting high-profit bundle discounts based on historical customer purchase data. We develop several search algorithms that identify profit-maximizing prices and bundle discounts. We introduce a richer probabilistic valuation model than prior work by capturing complementarity, substitutability, and covariance, and we provide a hybrid search technique for fitting such a model to historical shopping cart data. As new purchase data is collected, it is integrated into the valuation model, leading to an online technique that continually refines prices and bundle discounts. To our knowledge, this is the first paper to study bundle discounting using shopping cart data. We conduct computational experiments using our fitting and pricing algorithms that demonstrate several conditions under which offering discounts on bundles can benefit the seller, the buyer, and the economy as a whole. One of our main findings is that, in contrast to products typically suggested by recommender systems, the most profitable products to offer bundle discounts on appear to be those that are occasionally purchased together and often separately.

1 Introduction

Business-to-customer retail sales account for nearly four trillion dollars in the United States annually, and the percentage of this shopping done online increased three-fold between 2002 and 2007 [25]. Yet, despite the increased computational power, connectivity, and data available today, most online and brick-and-mortar retail mechanisms remain nearly identical to their centuries-old original form: item-only catalog pricing (i.e., take-it-or-leave-it offers). These are the default of B2C trade and are used by massive online retailers like Amazon, Best Buy, and Dell. However, they are fundamentally inexpressive because they do not allow sellers to offer discounts on different combinations, or *bundles*, of items.

Recently, some electronic retailers have started offering large numbers of bundle discounts (e.g., motherboards and memory at the popular computer hardware site, New Egg, and songs or albums on music sites), and brick-and-mortar retailers often offer bundle discounts on select items, such as food and drinks. Such discounts make the item-only catalog more expressive, and can be viewed as part of the general trend toward increased expressiveness in economic mechanisms. Increases in expressiveness have been shown to yield better outcomes in the design of general economic mechanisms [6], and in a number of specific domains such as sourcing auctions [23] and advertisement markets [7, 28].

Researchers in economics, operations research, and computer science have studied issues surrounding choosing prices and bundles in various types of catalog settings for decades. However, this work has either been i) largely theoretical in nature rather than operational (e.g., [1, 3, 13, 20, 24]), ii) focused on specific types of customer survey data which are not available in many settings (e.g., [15, 17, 22]), or iii) focused on specific sub-problems (e.g., pricing information goods [4, 9, 16, 19, 29], item-only pricing [5], or unit-demand and single-minded customers [14]). Despite the ability to collect substantial amounts of data about actual customer responses to different pricing schemes, retailers in most domains are still lacking practical techniques to identify promising bundle discounts.

In this paper, we introduce an automated framework that suggests profit-maximizing prices, bundles, and discounts, the first, to our knowledge, to attempt bundle discounting using shopping cart data. Our framework uses a pricing algorithm to compute high-profit prices and a fitting algorithm to estimate a customer valuation model. As new purchase data is collected, it can be integrated into the model fitting process, leading to an online technique that continually refines prices and discounts.

In Section 4, we conduct computational experiments that test each component of our framework individually and one set that tests the framework as a whole. Our results reveal that, in contrast to the products typically suggested by recommender systems, the most profitable products to offer bundle discounts on appear to be those that are only occasionally purchased together and often separately. We also use data from a classic shopping cart generator [2] to estimate the gains in profit and surplus that can be expected by using our framework in a realistic setting. We conservatively estimate that a seller with shopping cart data like that of the generator, who already has optimally priced items, can increase profits by almost 3% and surplus by over 8% using only bundles of size two (even if he has a thousand items for sale). All of our results taken together suggest that this line of work could have material practical implications.

The setting we consider involves a seller with m different kinds of items who wishes to choose a set of prices to offer on different combinations of those items to one customer at a time. However, we generalize our framework to consider settings with more than one customer by measuring expectations for profit and revenue, which implies that item prices cannot depend on the identity of the customer. We also consider the special case where a seller can only offer discounts on bundles and must hold the item prices fixed for some exogenous reason (e.g., due to existing policies or competition). We assume the seller has a cost function that can be approximated by assigning each item a fixed cost per unit sold (in the case of digital goods, which have no marginal cost to produce, we assume the seller can estimate some form of amortized cost), and his goal is to maximize expected *profit* (revenue minus cost). The seller chooses a price catalog, $\pi(b)$, which specifies a take-it-or-leave-it price for each bundle, b, of items. In an item-priced catalog, the price of a bundle is the sum of its parts. (We will be studying richer price catalogs than that, but we still will not be pricing each bundle separately in order to keep the process tractable.) The customer has a valuation,

$v(b)$, for each bundle b and chooses to purchase the bundle that maximizes her *surplus* (valuation minus price). We make the usual assumption of free disposal (i.e., the value of a bundle is at least as much as the value of any sub-bundle). We measure expected values of revenue, seller's profit, surplus, and *efficiency* (buyer's surplus plus seller's profit).

Using this model, we can easily prove that an item-price-only catalog is arbitrarily inefficient for some valuation distributions. This follows from our recent application-independent theory that proves an upper bound on efficiency based on the expressiveness of the mechanism [6].

2 Searching for profit maximizing prices

To study the impact of the item-only price catalog's inexpressiveness in practice, we first develop pricing algorithms that can determine the seller's profit-maximizing prices for a given type of catalog, cost function, and distribution over customer valuations. Each of our algorithms takes as input an estimate of the buyer's probability function, $\hat{P}$, the seller's cost function, $c(b)$, a set of priceable bundles, $\hat{B}$ (determined by the type of catalog), lower and upper bounds on the price of each bundle, $L(b)$ and $U(b)$ (also determined by the type of catalog, and can be used to ensure certain prices are fixed), and a seed price catalog, $\pi^{(0)}$ (which need not be intelligently generated). We assume that the algorithm can choose any arbitrary prices for the different bundles as long as the price of a bundle is no greater than the sum of prices for any collection of sub-bundles that contain all of its items.[1] The algorithms each call $\hat{P}$ repeatedly with candidate catalogs in order to identify the one with the highest expected profit: $\max_\pi \sum_{b \in B} \hat{P}(b|\pi) \times (\pi(b) - c(b))$.

Exhaustive pricing (EX): For each priceable bundle, $\hat{b} \in \hat{B}$, this algorithm discretizes the space between $L(\hat{b})$ and $U(\hat{b})$ into k evenly-spaced prices and checks the expected profit of every possible mapping of prices to priceable bundles. It finds an optimal solution (subject to discretization), but is intractable with more than two items and even with two items if k is too large. For a fully expressive catalog (i.e., one where each bundle is priced separately) with m items, this algorithm calls $\hat{P}$ with k^{2^m-1} different catalogs, and $\hat{P}$ can, itself, be costly to compute. Thus, we propose this algorithm be used primarily as a tool to compare results with the other algorithms on small instances.

Hill-climbing pricing (HC): Starting with the seed catalog, this algorithm computes the improvement in expected profit achieved by adding or subtracting a fixed Δ from each priceable bundle, which involves $2|\hat{B}|$ calls to $\hat{P}$, in each step. It updates the catalog with the change that leads to the greatest improvement, and repeats this process until there are no more improving changes. The resulting catalog is returned, and, since the catalog is only updated when an improvement is possible, it is guaranteed to have the highest observed expected revenue.

Gradient-ascent pricing (GA): Starting with the seed catalog, this algorithm computes the gradient, or partial derivative, of the expected profit function,

[1] This ensures the catalog is consistent so that a customer cannot get a better price on a bundle by purchasing its components in some other combinations.

which involves $|\hat{B}|$ calls to $\hat{P}$ in each step. The partial derivative, $\mathbf{d}(b)$, of the expected profit function with respect to a bundle, b, is estimated by measuring the change in expected profit when a fixed Δ is added to $\pi(b)$. The resulting vector of derivatives, $\mathbf{d}$, is normalized to sum to one, and the algorithm updates its best candidate catalog by adding $\mathbf{d}(b) \times \Delta$ to the price of each priceable bundle. The algorithm continues this process until no more improvements in expected profit are possible. The resulting catalog is returned, and, as with the hill-climbing algorithm, it is guaranteed to be the one with the highest expected revenue that was explored throughout the search. In our experiments, this algorithm achieved near-optimal expected revenue on most instances, while performing poorly on a few, with a relatively few number of calls to $\hat{P}$.

Pivot-based pricing (PVT): This algorithm generalizes hill-climbing by searching for the best adjustment to the current prices of up to k bundles at a time. For each k-or-less-sized combination of priceable bundles, β, this algorithm measures the change in expected profit from simultaneously adjusting all the prices in β. Each price can be incremented by Δ, decremented by Δ, or not changed. At each step, the algorithm tests all of those possibilities and selects the one that increases expected profit the most. The hill-climbing algorithm above is a special case of this where $k = 1$. However, for larger values of k it generalizes that algorithm to consider more complex types of price adjustments. This process involves $\binom{|\hat{B}|}{k} \times (3^k - 1)$ calls to $\hat{P}$ at each step. Even with $k = 2$, our early tests show this is the only one of the algorithms (other than the exhaustive one), that achieves optimal expected revenue on nearly every instance.

3 Estimating a rich customer valuation model

The problem of estimating a customer valuation model from historical purchase data is an essential part of our bundling framework because it allows us to use the pricing algorithms presented in the previous section in a practical setting. It is also a problem of interest in its own right, as it extends the classic *market basket analysis* problem first introduced by Agrawal *et al.* [2]. Market basket analysis is a commonly studied data mining problem that involves counting the frequencies of different bundles in a collection of customer purchase histories. Simply counting these occurrences can be challenging when there is a large set of items and each customer buys several of them at once. Almost all of the work on this problem has focused on building recommender systems that suggest products frequently purchased together. Many algorithms have been developed for finding bundles with such statistical properties, including one that was developed and patented by Google co-founder Sergey Brin and others [8]. However, as our experiments in Section 4 show, our framework predicts that the most profitable items to bundle are those with the opposite profile.

The valuation modeling problem that we consider extends the market basket analysis problem to involve predictions about what would happen to the purchase frequencies under different price catalogs. (There has been significant recent progress on inferring valuation distributions from bids or other indications of demand in a variety of applications [17, 18, 21, 26], but that work typically used bids in auctions or survey information.)

The inputs to the two problems are essentially the same, although in the case of our valuation problem we include the price catalogs that were on offer at the time of purchase, which can provide additional information about sensitivities to price changes. The close relationship between these two problems allows us to use a classic data generator for the market basket problem in our experiments.

3.1 Deriving the maximum likelihood estimate

For the valuation modeling problem, we are given a set of historical purchase observations, $D = \{\langle b_1, \pi_1\rangle, \langle b_2, \pi_2\rangle, \ldots, \langle b_n, \pi_n\rangle\}$, where each observation, i, includes a bundle that was purchased, b_i, by a distinct customer, i, and the prices of all bundles at the time, π_i. We assume that these purchases are made based on each customer's surplus-maximizing behavior with valuations drawn from an underlying valuation model. We also assume that each purchase is independent of all others since we consider each observation to be from a distinct customer. Under these assumptions, it is relatively straightforward to show that the maximum likelihood estimate (i.e., model that maximizes the likelihood of the data) for the customer valuations yields a $\hat{P}$ that matches the observed purchase frequencies as closely as possible. (Details omitted due to space constraints.)

3.2 Fitting the valuation model to purchase data

The valuation model we will fit allows for normally distributed valuations on each item, pair-wise covariance between valuations for items, as well as normally distributed terms for complementarity (or substitutability in case such a term is negative). This model significantly generalizes prior ones [10, 11, 24, 27] by allowing for heterogeneous complementarity and substitutability between products.

Specifically, our model parameters include a mean and variance for each priceable bundle in $\hat{B}$ and covariances between individual items' valuations. While the draw, $x_{\{i\}}$, from the distribution of an item i represents that item's valuation, $v(\{i\})$, to the customer, a draw from the distribution for a bundle b of two or more items represents a complementarity bonus (or substitutability penalty if negative). The valuation for a bundle is then the sum of the draws of all the bundles (including individual items) it contains: $v(b) = \sum_{b' \subseteq b} x_{b'}$. Under this model, a customer's valuation can be thought of as a hyper-graph where each (hyper-)edge is associated with a real-valued random variable representing the valuation bonus or penalty for receiving a bundle containing the items connected by the (hyper-)edge. This allows us to model any possible distribution over valuations (without loss of generality), and can be viewed as a probabilistic generalization of the classic $k-$wise valuation model introduced by Conitzer and Sandholm for combinatorial auctions [12].

To go from a valuation model to the probability function, $\hat{P}$, we use a Monte-Carlo method to sample customers (10,000 in our experiments) according to the valuation distribution, and, for a given catalog, we simulate their surplus-maximizing purchasing behavior (taking into account that disposal is free). This simulation is relatively straightforward since items that are not connected by a complementarity or substitutability edge can be considered independently.

In order to identify the model parameters that maximize the likelihood of the observed data, we use a hybrid search technique. It begins by performing a tree search over the variance and covariance parameters. A range for each of these parameters is given as input that is discretized into a specified number of values (in our experiments we use six values per parameter). At each leaf node, a local search is performed to find the means that maximize the data likelihood given the values of the variance parameters at that leaf (see Figure 1).[2] In our experiments, we use a pivot-based search, as described in Section 2, for this step. The parameter settings resulting in the highest overall likelihood are returned, and in the case of a tie an even mixture of all the tied models is used (i.e., simulated customers are sampled from each with equal probability).

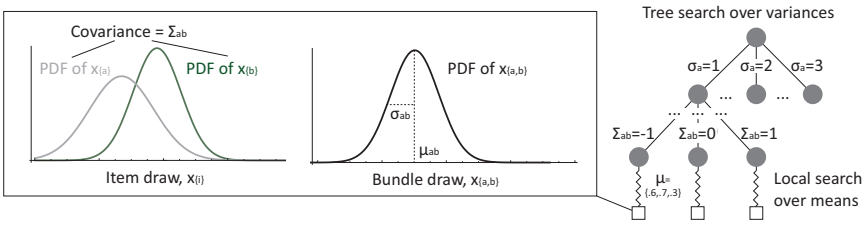

Fig. 1. Illustration of the search technique we use for estimating a customer valuation model from historical purchase data. We use a tree search over variances and a pivot-based search over means. Leaves are evaluated based on how closely the corresponding model predicts the observed data and (optionally) how closely the model's optimal profit matches the profit achieved by existing prices.

Most existing shopping cart data involve only a single catalog and do not include information about customers' surplus-maximizing behavior under alternative prices. Thus, this data tends to be under-specified for the purposes of inferring a valuation model. To address this, on such instances we utilize the existing item prices as an additional piece of information to fit our model. Specifically, among models that fit the observed purchase data (approximately), we prefer models whose profit under the optimal item-pricing for that model is close to the profit of the existing item prices under the model. Our algorithm does this test once at every leaf of the search tree (after the best model for the leaf has been computed as described above). If there are still several leaves that are (approximately) as good at explaining the purchase data and the existing prices, we use an even mixture over those models (we use at most the top five models).

4 Empirical results

We will now discuss the results from several sets of compuational experiments that test our pricing and fitting algorithms and reveal some interesting economic insights that emerge as a consequence of our customer valuation model. The next

[2] We found empirically that choosing standard deviations with tree search and means with local search yielded better results due to the tight relationship between the appropriate means and standard deviations.

two subsections focus on pricing and fitting two-item instances. The third set of results provides an estimate of the potential achievable by offering bundle discounts on pairs of items from a seller with a thousand items and realistic shopping cart data.

4.1 Results with pricing algorithms

The first set of experiments involves using the search techniques described in Section 2 to find high-profit prices on a generic class of instances similar to the models used in prior work [10, 11, 24, 27]. We compare the results and performance of the pricing algorithms on symmetric two-item instances where the customer's valuation for each item is drawn from a normal distribution with mean 0.5 and standard deviation 0.5. We vary the pairwise covariance from $-.25$ to .25 and we vary the mean of the pair-wise complementarity (or substitutability when negative) term from -1.5 to 0.5 (the standard deviation for this term is held constant at 0.5). Each algorithm (other than the exhaustive one) uses an item-only catalog with all prices set to 0.5 as a seed and a step size $\Delta = 0.05$ to price fully expressive catalogs. The EX algorithm considers $k = 15$ different prices for each bundle and finds the optimal prices subject to this discretization. The PVT algorithm considers all possible gradients for two item instances.

The following table reports each algorithm's average fraction of the highest expected profit, efficiency, and surplus, as well as the average number of calls to $\hat{P}$ over five instances with a fully expressive catalog for each parameter setting. The best value in each column is in bold.

Algorithm	E[prof.]	E[eff.]	E[surp.]	$\hat{P}$ calls
PVT	**99.99%**	97.88%	86.92%	267.25
EX ($k = 15$)	99.24%	97.70%	88.28%	3375.00
GA	97.57%	**99.14%**	90.78%	49.61
HC	89.76%	93.73%	**96.32%**	**4.08**

Other than the unscalable exhaustive algorithm, the pivot-based algorithm is the only one to achieve optimal profit on every instance. Therefore, it is the algorithm we use in the rest of the paper for pricing. (Gradient ascent also performed well and may scale better for larger instances.)

Figure 2 shows the increase in expected profit and surplus from allowing sellers to offer profit-maximizing bundle discounts, while varying the levels of covariance, complementarity, and substitutability. The values represent averages over five runs but deviate very little.

For this set of results, we assume the seller holds the item prices fixed at the optimal item-only catalog values to isolate the impact of offering bundle discounts from the potential confound of our system improving the item prices as well. We believe this also represents a practical constraint in many markets and is a policy that sellers are likely to take when first adopting the bundle discounts suggested by our framework. This has the effect of depressing the seller's expected profit gain, but it ensures that the customer surplus cannot decrease.

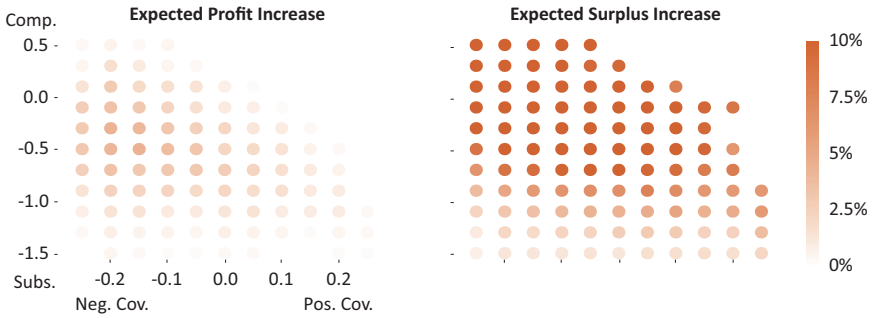

Fig. 2. The intensity of each dot is the increase in expected profit or surplus achieved by profit-maximizing bundle discounts for different levels of covariance (x-axis) and complementarity (or substitutability) (y-axis), ranging from 0% to 10%. Here, we assume the seller holds the item prices fixed at the optimal item-only catalog values to isolate the impact of bundle discounts.

For the scenarios we consider, the seller's greatest predicted increase in expected profit (about 4.6%) occurs when valuations are highly negatively correlated and the items are slightly substitutable. However, too much substitutability diminishes the predicted profit benefits. Others have also identified negative correlation and substitutability as motivators for offering bundle discounts [10, 11, 27], but they did not use a rich enough valuation model to fully explore the impact of heterogeneous complementarity or substitutability. (That work also did not address the model fitting problem that must be solved to operationalize this insight.)

Unsurprisingly, due to the discount-only pricing we imposed, our results also show a large predicted increase in surplus (averaging around 9%) throughout the parameter space. Together with the seller's predicted increase in profit, this leads to substantial efficiency increases.

Another set of experiments (not shown due to space constraints) demonstrates that when our system is also free to adjust the prices of the items, additional increases in profit are possible but usually at the expense of the customer surplus. This may be desirable for the seller in the short term, but maintaining surplus can be an important long-term goal if there are competing sellers.

4.2 Results with the fitting algorithm

We now present experiments that use the fitting algorithm from Section 3 to find models that predict an observed set of purchase data. We allow the search algorithm to consider standard deviations between 0.5 to 3.5 at intervals of 0.5, and we focus on symmetric two-item instances where both items occur with the same frequency in the shopping cart data. (Results on asymmetric instances were similar.) These experiments test our algorithm in the ubiquitous scenario where shopping cart data is accompanied by a single item-only price catalog.

Figure 3 shows the predicted increases in expected profit and surplus achievable by a bundle discount, assuming that the individual items are optimally priced and that at those prices they have the same profit margin (the value of

the profit margin does not matter). As in Figure 2, we assume the seller can only offer a discount on the existing item prices and cannot change them. (When we relax this assumption, we find additional opportunities to increase profit at the expense of customer surplus.) We consider instances where the item frequencies range from 2.5% to 40% and the *co-occurrence* percentages from 2.5% to 87.5%. We define co-occurrence as the fraction of baskets containing the less frequent item (for symmetric items either can be used) that also contain the other. We also increased our sampling frequency in an interesting area of the parameter space where item frequency is less than 15% and co-occurence is less than 20%. This is illustrated on each chart by a higher concentration of small points in the bottom left corner. (Again, the values are averaged over five runs.)

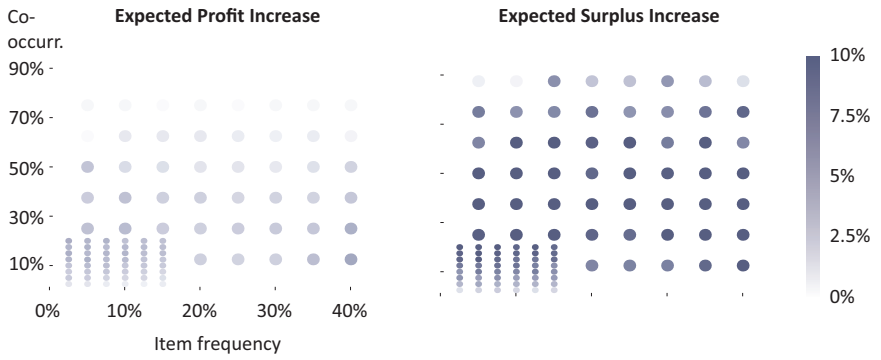

Fig. 3. The intensity of each dot represents the predicted increase in expected profit or surplus achieved by profit-maximizing bundle discounts on single-catalog instances with varying item frequencies (x-axis) and co-occurrence percentages (y-axis), ranging from 0% to 10%. As in Figure 2, we assume the seller holds the item prices fixed at the optimal item-only catalog values to isolate the impact of offering bundle discounts from the potential confound of our system improving the item pricing as well.

These results are consistent with those in Figure 2, since the seller's greatest predicted increase in expected profit (about 4.6%) occurs when products are occasionally bought together (co-occurrence probability less than 20%) and frequently bought separately. This set of results also predicts large increases in surplus throughout the parameter space (averaging about 9%), as in Figure 2.

Taken together, our results illustrate why new techniques are needed beyond those used for building recommender systems, which typically identify items that are commonly purchased or consumed together. When it comes to items that can be profitably bundled together at a discount, our framework suggests those with the opposite profile. Our results also explain why recommender systems are highly popular among users: a recommendation can be viewed as a small discount (in the form of time saved), and we see that even a small discount on highly co-occurring products leads to a substantial increase in surplus.

4.3 Results with a shopping cart generator

Our final set of experiments estimates the potential increase in expected profit and surplus achievable by bundling products from a seller with shopping cart data like that generated by Agrawal and Srikant's classic generator [2]. We use the standard parameters in the generator: for each instance, we generate 10,000 shopping carts with 100-1,000 items (N), 100-2,000 potentially popular bundles (L) of size 2-4 (I), and an average of 2-20 purchases per customer (B). We assume the seller had optimally priced the individual items, and that those prices involved a uniform profit across all items.

Pricing all—or a huge number of—bundles is undesirable for several reasons: i) presenting complex catalogs to customers may be infeasible and/or it may confuse/burden them, ii) it is intractable in terms of computation and information, and iii) even non-overlapping bundles can interact: as one bundle is discounted, some customers might shift from buying other things to that bundle. Therefore, we only consider discounting bundles of two items, and further narrow them down as follows. We only consider item pairs priceable if the items are not directly or indirectly *related* to any other items. We consider two items related if their joint purchase frequency is more than a fixed threshold different than the product of their individual purchase frequencies (we use a threshold of 1% for these experiments). We construct a graph where the items are nodes and edges connect items that are related. Then, only connected components of size two and pairs of isolated items are considered priceable.

The profit and surplus increases for each priceable pair are then estimated using the results behind Figure 3 and a set of similar results on asymmetric instances. The increase for a given pair is estimated as the average value for the five most similar instances (based on the frequencies of the two items and the bundle). Priceable pairs are then greedily selected to actually be discounted based on their predicted profit increase. Once a pair is selected, all other pairs containing either of the selected items are removed from consideration. The following table shows the total predicted profit and surplus increases for various parameter settings of the generator (values are averaged over five instances).

N	B	L	I	E[prof.] inc.	E[surp.] inc.
1000	20	2000	4	2.80%	8.34%
1000	20	1000	4	1.10%	3.01%
100	2	200	2	0.89%	2.65%
100	2	100	2	0.15%	0.86%

For the standard parameter settings, the first row shows almost 3% profit increase using our algorithms to select pairs of items to bundle and discount. This increase in profit is accompanied by more than an 8% increase in customer surplus, and, thus, a significant efficiency increase. The table also shows that increasing the number of items and potentially popular bundles increases the benefits from our approach. This is because it leads to a sparser relatedness graph and, thus, increases the number of safely priceable items for our algorithms.

These figures are conservative because they assume that the seller had already priced the individual items optimally, only considers pairs of items where neither item is related to any other, and offers at most one bundle per item.

5 Conclusions and future research

We introduced a framework for automatically mining purchase data and suggesting profit-maximizing prices, bundles, and discounts. It uses a pricing algorithm to compute high-profit prices on items and some bundles, and a fitting algorithm to estimate a customer valuation model. New data can be integrated into model fitting in an online manner leading to continually refined prices and discounts.

Some obvious directions for future research include less conservative methods for selecting pricable bundles, discounting bundles of more than two items, and live experiments where the catalogs that we offer serve as demand queries about the customers' valuations that are then incorporated back into our model. These experiments could be carried out similarly to the ones described by Jedidi *et al.* [17], but would involve actual purchases by subjects rather than survey data.

There are also several assumptions made here that could be relaxed in future work. For example, we assumed that each purchase in the shopping cart data was independent, but it may be possible to develop a model that captures repeat purchases by the same customers. We also assumed that the cost of selling an item could be described by a marginal unit cost. It would be interesting to extend our work here to include considerations of non-linear cost functions (e.g., with large start-up costs) or limited-inventory items. Finally, we assumed that the true customer valuations were drawn from distributions that could be accurately fit by our valuation model. However, it would be interesting to consider the effects of mis-representing these valuations because, for example, they are drawn from a different kind of distribution than ours (e.g., lognormal rather than normal).

References

1. William James Adams and Janet L. Yellen. Commodity bundling and the burden of monopoly. *Quarterly Journal of Economics*, 90(3):475–498, Aug 1976.
2. Rakesh Agrawal and Ranakrishnan Srikant. Fast algorithms for mining association rules. In *VLDB*, 1994.
3. Mark Armstrong. Multiproduct nonlinear pricing. *Econometrica*, 64(1):51–75, 1996.
4. Yannis Bakos and Erik Brynjolfsson. Bundling information goods: Pricing, profits, and efficiency. *Management Science*, 45(12):1613–1630, Dec 1999.
5. Maria-Florina Balcan, Avrim Blum, and Yishay Mansour. Item pricing for revenue maximization. In *ACM EC*, 2008.
6. Michael Benisch, Norman Sadeh, and Tuomas Sandholm. A theory of expressiveness in mechanisms. In *AAAI*, 2008.
7. Michael Benisch, Norman Sadeh, and Tuomas Sandholm. Methodology for designing reasonably expressive mechanisms with application to ad auctions. In *IJCAI*, 2009.
8. Sergey Brin, Rajeev Motwani, Jeffrey D. Ullman, and Shalom Tsur. Dynamic itemset counting and implication rules for market basket data. In *SIGMOD*, 1997.
9. Chirstopher H. Brooks and Edmund H. Durfee. Toward automated pricing and bundling of information goods. In *Knowledge-based Electronic Markets*, 2000.

10. Zmbl Bulut, lk Grler, and Alper Sen. Bundle pricing of inventories with stochastic demand. *European Journal of Operational Research*, 197(3):897–911, 2009.
11. Chenghuan Sean Chu, Phillip Leslie, and Alan Sorensen. Nearly optimal pricing for multiproduct firms. NBER Working Paper 13916, 2008.
12. Vincent Conitzer, Tuomas Sandholm, and Paolo Santi. Combinatorial auctions with k-wise dependent valuations. In *AAAI*, 2005.
13. Robert E. Dansby and Cecilia Conrad. Commodity bundling. *The American Economic Review*, 74(2):377–381, May 1984.
14. Venkatesan Guruswami, Jason D. Hartline, Anna R. Karlin, David Kempe, Claire Kenyon, and Frank McSherry. On profit-maximizing envy-free pricing. In *SODA*, 2005.
15. Ward Hanson and R. Kipp Martin. Optimal bundle pricing. *Management Science*, 36(2):155–174, 1990.
16. Lorin M. Hitt and Pei-yu Chen. Bundling with customer self-selection: A simple approach to bundling low-marginal-cost goods. *Management Science*, 51(10):1481–1493, 2005.
17. Kamel Jedidi, Sharan Jagpal, and Puneet Manchanda. Measuring heterogeneous reservation prices for product bundles. *Marketing Science*, 22(1):107–130, 2003.
18. Przemyslaw Jeziorski and Ilya Segal. What makes them click: Empirical analysis of consumer demand for internet search advertising. Mimeo, June 2009.
19. Jeffrey O. Kephart, Christopher H. Brooks, Rajarshi Das, Jeffrey K. MacKie-Mason, Robert Gazzale, and Edmund H. Durfee. Pricing information bundles in a dynamic environment. In *ACM EC*, 2001.
20. R. Preston McAfee, John McMillan, and Michael D. Whinston. Multiproduct monopoly, commodity bundling, and correlation of values. *Quarterly Journal of Economics*, 104(2):371–383, May 1989.
21. Michael Ostrovsky and Michael Schwarz. Reserve prices in internet advertising auctions: A field experiment. In *ACM EC*, 2011.
22. Paat Rusmevichientong, Benjamin Van Roy, and Peter W. Glynn. A nonparametric approach to multiproduct pricing. *Operations Research*, 54(1):82–98, 2006.
23. Tuomas Sandholm. Expressive commerce and its application to sourcing: How we conducted $35 billion of generalized combinatorial auctions. *AI Magazine*, 28(3):45–58, 2007.
24. Richard Schmalensee. Gaussian demand and commodity bundling. *The Journal of Business*, 57(1):S211–S230, Jan 1984.
25. US Census Bureau. Statistical abstract of the United States, wholesale & retail trade: Online retail sales, 2010. `http://www.census.gov/`.
26. H. R. Varian. The non-parametric approach to demand analysis. *Econometrica*, 50:945 – 974, 1982.
27. R. Venkatesh and W. Kamakura. Optimal bundling and pricing under a monopoly: Contrasting complements and substitutes from independently valued products. *Journal of Business*, 76(2):211–231, 2003.
28. W. Walsh, C. Boutilier, T. Sandholm, R. Shields, G. Nemhauser, and D. Parkes. Automated channel abstraction for advertising auctions. In *AAAI*, 2010.
29. Shin-yi Wu, Lorin M. Hitt, Pei-yu Chen, and G. Anandalingam. Customized bundle pricing for information goods: A nonlinear mixed-integer programming approach. *Management Science*, 54(3):608–622, 2008.