Assortment Optimization Over Time

James M. Davis [*]   Huseyin Topaloglu   David P. Williamson[†]

Abstract

In this note, we introduce the problem of assortment optimization over time. In this problem, we have a sequence of time periods and can only introduce one new product per time step, and we are not allowed to remove products from our assortment that have already been introduced. The goal is to determine which products to introduce so as to maximize the total revenue realized over all the time steps under some choice model. We show how to give a 1/2-approximation algorithm for this problem under a general choice model, for which the multinomial logit choice model is a special case. We further show a $(1 - 1/e)$-approximation algorithm if the revenue function is monotone and submodular. Finally, we show that the problem is NP-hard to compute for a natural special case of the general choice model whose revenue function is monotone and submodular.

1 Introduction

A fundamental problem in revenue management involves finding a profitable assortment of products to offer to customers given that customers choose within the offered assortment according to a certain choice model. The main tradeoff in this problem setting is that if we offer a limited assortment, then customers may often decide to leave without making a purchase, but if we offer a wide assortment, then we may end up adding products with low revenue contributions to the assortment, which may dilute the revenue potential. This fundamental tradeoff is often complicated by the fact that it may take time for a firm to build up the product assortment it wants to offer to its customers. In this paper, we consider assortment optimization problems over time, where a firm has to gradually build up its product portfolio. We consider a firm that sequentially decides which product, if any, to add to its offered assortment at each time period. Customers arriving at a particular time period choose among the current assortment according to a particular choice model. The goal is to decide how to build the product portfolio over time to maximize the total expected revenue over a finite planning horizon.

There is growing literature on assortment optimization problems, as these problems are faced on strategic and tactical levels in numerous retail and revenue management settings. In brick-and-mortar retail, firms are interested in finding the right mix of products to offer and such decisions are made infrequently on a strategic level. In revenue management, airlines and hotels continuously update their product offerings in response to the remaining time in the selling horizon and the

remaining capacity, and so the assortment problems in those setting have more of an operational nature. Several authors studied the so-called static assortment problem, where the goal is to find an assortment of products that maximizes the expected revenue obtained from each customer. Talluri and van Ryzin [18] show that if the customers choose according to the multinomial logit model, then the static assortment problem can be solved efficiently by checking the expected revenue from each assortment that includes a certain number of products with the largest revenues. Rusmevichientong et al. [17] solve the same problem when there is a limit on the number of products that can be offered. Bront et al. [3], Desir and Goyal [8] and Mendez-Diaz et al. [15] study the static assortment problem when customers choose according to a mixture of multinomial logit models, show that the problem is NP-complete, strengthen an integer programming formulation for the problem with valid inequalities and propose various approximation schemes. Davis et al. [7] consider the same problem under the nested logit model and show that the corresponding static assortment problem can be solved efficiently. Alptekinoglu and Semple [1] focus on static assortment problems when customers choose under the exponomial choice model.

In dynamic assortment problems, the offered assortment is adjusted over time, possibly due to depleted product inventories, better understanding of customer choice processes or changes in customer tastes. Honhon et al. [13] and Mahajan and van Ryzin [14] study the problem of finding an assortment to offer and the corresponding stocking quantities with the understanding that customers choose only among the products that are still in stock. Bernstein et al. [2] and Golrezaei et al. [12] consider the problem of dynamically customizing the assortment offerings based on the preferences of each customer and remaining product inventories. Caro et al. [5] and Cinar and Martinez-de-Albeniz [6] study assortment problems where the attractiveness of the products diminishes over time and they seek optimal policies to replace such products. Caro and Gallien [4] and Ulu et al. [19] develop models where the assortment offering needs to be adjusted over time in response to a better understanding of the customer choice process.

The rest of the note is organized as follows. §2 describes our assortment problem, where the firm needs to gradually build its product portfolio. §3 analyzes approximation algorithms for the problem. §4 discusses the complexity of the problem.

2   Preliminaries

Let I be the set of items that can be offered for sale, and let $n = |I|$. We let r_j be the revenue earned when item j is sold. Let $P_j(S)$ be the probability that item j is purchased if $S \subseteq I$ is offered for sale; then $P_j(S) = 0$ if $j \notin S$. We consider choice models where the following two properties hold. First,
$$P_j(S) \geq P_j(T) \qquad \forall T \; \forall j \in S \subset T;$$
that is, the probability of purchasing item j cannot increase if we offer a larger set of items. This holds for utility maximization models, for example. Second, $\sum_{j \in S} P_j(S) \leq 1$ for any non-empty set of items S. If an assortment S is offered for sale, then with probability $1 - \sum_{j \in S} P_j(S)$, no item is purchased. The expected revenue for a set S of items is $R(S) = \sum_{j \in S} r_j P_j(S)$. For lack of a better term, let us call such choice models *monotone* choice models. The multinomial logit choice model is one example of a monotone choice model.

Additionally, we will also consider the case in which we only know that the revenue function $R(S)$ is *monotone* (that is, $R(S) \leq R(T)$ for any $S \subseteq T \subseteq I$) and *submodular* (that is, for any

2

$j \notin S \subseteq T \subseteq I$, then $R(S \cup \{j\}) - R(S) \geq R(T \cup \{j\}) - R(T))$, without knowing anything about the underlying choice model.

We will consider variants of the assortment optimization problem in which there is a capacity c on the number of items that can be offered for sale; that is, we wish to find a set S_c^* with $|S_c^*| \leq c$ that maximizes $R(S_c^*)$. We call this the *capacitated assortment optimization problem.* For an optimal solution S_c^*, let OPT_c be the expected revenue obtained; that is, $\mathrm{OPT}_c = R(S_c^*)$. For our first result, we will suppose that we have a polynomial-time algorithm for the capacitated assortment optimization problem under the given monotone choice model. For example, there is an optimal algorithm for the problem for the multinomial logit and nested logit choice models (see Feldman and Topaloglu [11] and Rusmevichientong et al. [17]). For our second result, we will use a well-known greedy $(1 - 1/e)$-approximation algorithm for finding a maximum valued set S with $|S| \leq c$ for any monotone, submodular set function due to Nemhauser et al. [16]. The greedy algorithm repeatedly chooses an element to add to the set S until $|S| = c$, and each time adds the element in $I - S$ that maximizes the marginal gain; that is, it selects $j \in I - S$ that maximizes $R(S \cup \{j\}) - R(S)$ and adds it to S.

We wish to study the *assortment optimization problem over time.* Intuitively, we would like to find a sequence of items in which we can offer at most one new item for sale at each of T time steps that maximizes the overall expected revenue achieved over the given time horizon. Once an item is offered for sale, it remains available to purchase for the remainder of the time horizon. More precisely, we would like to find sets $S_1, S_2, \ldots, S_T$ such that $|S_t| \leq t$ for all $1 \leq t \leq T$ and $S_1 \subseteq S_2 \subseteq \cdots \subseteq S_T$ that maximizes $\sum_{t=1}^{T} R(S_t)$.

3   Algorithm

In this section, we first give an $1/2$-approximation algorithm for the assortment optimization problem over time for monotone choice models, given a polynomial-time algorithm for the capacitated assortment optimization problem under the choice model. We then give a $(1 - 1/e)$-approximation algorithm for the assortment optimization problem over time for monotone submodular revenue functions.

The first algorithm works as follows. We initially use the polynomial-time algorithm to find an optimal assortment of at most capacity t for all t, $1 \leq t \leq T$. Let τ be the capacity for which we get the largest revenue assortment and let S_τ be the assortment. We then order the items of S_τ by nonincreasing value of $r_j P_j(S_\tau)$; we will offer items for sale in this order. Without loss of generality, assume that items of S_τ are indexed by $1, 2, \ldots, \tau$ so that $r_1 P_1(S_\tau) \geq r_2 P_2(S_\tau) \geq \cdots \geq r_\tau P_\tau(S_\tau)$. Then we set $S_1 = \{1\}$, $S_2 = \{1, 2\}$, $\ldots$, $S_{\tau-1} = \{1, \ldots, \tau - 1\}$, and $S_t = S_\tau$ for all $\tau \leq t \leq T$.

We now analyze the algorithm. Let OPT be the overall expected revenue of an optimal assortment over time. We observe the following.

Observation 3.1
$$\mathrm{OPT} \leq \sum_{t=1}^{T} \mathrm{OPT}_t \leq T \cdot \mathrm{OPT}_\tau = T \cdot R(S_\tau^*),$$

where $\mathrm{OPT}_t = R(S_t^*)$ *is the optimal expected revenue for the capacitated assortment optimization problem with capacity t, S_t^* is the optimal solution to this capacitated problem, and τ is such that* $\mathrm{OPT}_\tau \geq \mathrm{OPT}_t$ *for* $1 \leq t \leq T$.

We then get the following easy lemma bounding the value of the algorithm's solution in terms of the overall optimum.

Lemma 3.2 *Given a monotone choice model, for any of the sets S_t constructed above, $1 \le t \le \tau$, we have that*

$$R(S_t) \ge \frac{t}{\tau} \, \mathrm{OPT}_\tau \, .$$

Proof: Consider the following inequalities:

$$
\begin{aligned}
OPT_\tau &= \sum_{j \in S_\tau} r_j P_j(S_\tau) \\
&= \sum_{j \in S_t} r_j P_j(S_\tau) + \sum_{j \in S_\tau / S_t} r_j P_j(S_\tau) \\
&\le \sum_{j \in S_t} r_j P_j(S_t) + \sum_{j \in S_\tau / S_t} r_j P_j(S_\tau) \\
&\le \sum_{j \in S_t} r_j P_j(S_t) + \frac{\tau - t}{\tau} \sum_{j \in S_\tau} r_j P_j(S_\tau) \\
&= R(S_t) + \frac{\tau - t}{\tau} \, OPT_\tau,
\end{aligned}
$$

where the final inequality follows by our choice of the highest revenue items in S_τ. Rearranging terms gives the claimed inequality. ∎

Theorem 3.3 *The algorithm is a 1/2-approximation algorithm for the assortment optimization problem over time for any monotone choice model such that there is a polynomial-time algorithm for the capacitated assortment optimization problem.*

Proof: The expected value of our solution is $\sum_{1 \le t \le \tau} R(S_t) + \sum_{\tau < t \le T} R(S_\tau^*)$. Then

$$
\begin{aligned}
\sum_{1 \le t \le \tau} R(S_t) + \sum_{\tau < t \le T} R(S_\tau^*) &\ge \sum_{1 \le t \le \tau} \frac{t}{\tau} R(S_\tau^*) + \sum_{\tau < t \le T} R(S_\tau^*) \\
&= R(S_\tau^*) \left(T - \tau + 1 + \sum_{1 \le t \le \tau} \frac{t}{\tau} \right) \\
&= R(S_\tau^*) \left(T - \tau + \frac{\tau(\tau + 1)}{2\tau} \right) \\
&= R(S_\tau^*) \left(T - \tau + \frac{\tau + 1}{2} \right) \\
&= R(S_\tau^*) \left(T - \frac{\tau}{2} + \frac{1}{2} \right) \\
&\ge \frac{T}{2} \cdot R(S_\tau^*) \ge \frac{1}{2} \, \mathrm{OPT},
\end{aligned}
$$

where the final inequality uses Observation 3.1. ∎

We now show that we can obtain a $(1 - 1/e)$-approximation algorithm for assortment optimization over time given that the revenue function R is monotone and submodular. We simply run the greedy algorithm, and let S_1 be the first item selected by the greedy algorithm, S_2 be the first two elements selected by the greedy algorithm, and so on. If $T \geq |I|$, then for time steps $t \geq T$, we let $S_t = I$. We can now show the following.

Theorem 3.4 *This algorithm gives a $(1 - 1/e)$-approximation algorithm for assortment optimization over time when the revenue function is monotone and submodular.*

Proof: Let S_t^* be the optimal assortment of t items for revenue function, where $S_t^* = I$ for $t \geq T$. Then because the greedy algorithm is a $(1 - 1/e)$-approximation algorithm for the capacitated assortment optimization problem when the revenue function is monotone and submodular, we know that for any $t \geq 1$,

$$R(S_t) \geq \left(1 - \frac{1}{e}\right) R(S_t^*).$$

Therefore, we have a revenue of

$$\sum_{t=1}^{T} R(S_t) \geq \left(1 - \frac{1}{e}\right) \sum_{t=1}^{T} R(S_t^*) \geq \left(1 - \frac{1}{e}\right) \text{OPT},$$

since by Observation 3.1 $\sum_{t=1}^{T} \text{OPT}_t = \sum_{t=1}^{T} R(S_t^*)$ must be an upper bound on the optimal revenue. $\blacksquare$

4 Hardness

We now show that assortment optimization over time is NP-hard under a particular monotone choice model whose revenue function is monotone and submodular. Our reduction is from the min-sum set cover problem. In this problem, we are given a hypergraph $H = (V, E)$ as input, and the output is a sequence of the elements of V; we can think of the output as a bijective function $f : V \to \{1, \ldots, n\}$, where $n = |V|$. We extend the function f to the hyperedges so that $f(e) = \min_{v \in e} f(v)$. Then the goal of the min-sum set cover problem is to find a bijection f so as to minimize $\sum_{e \in E} f(e)$. Feige et al. [10] show that this is an NP-hard problem; in particular, they show that there is no $(2 - \epsilon)$-approximation algorithm for the problem, even for r-uniform d-regular hypergraphs, unless $P = NP$. A hypergraph is r-uniform if $|e| = r$ for all $e \in E$, and is d-regular if each vertex $v \in V$ is in exactly d of the hyperedges. We will need the d-regularity for our reduction.

Theorem 4.1 *It is NP-complete to decide whether the expected revenue of an assortment optimization over time instance is at least C for a choice model that is monotone, or a revenue function that is monotone and submodular.*

Proof: Given an instance of the min-sum set cover problem in which we have an d-regular hypergraph, we create an instance of assortment optimization over time as follows. We create an item j of revenue $r_j = |E|$ for each vertex $j \in V$. We set $P_j(S) = 0$ if $j \notin S$, and otherwise

$$P_j(S) = \frac{1}{|E|} \sum_{e \in E} \frac{|e \cap \{j\}|}{|e \cap S|},$$

5

where we assume $0/0 = 0$ in the case that $e \cap S = \emptyset$.

We now show that this choice model is monotone. Since there are exactly d hyperedges that contain j and $|E| \geq d$, then $0 \leq P_j(S) \leq 1$ for all j and all S, and furthermore $P_j(S) \geq P_j(T)$ when $j \in S \subset T$, since for any edge $e \in E$, if $j \notin e$, then $|e \cap \{j\}|/|e \cap S| = |e \cap \{j\}|/|e \cap T| = 0$, while if $j \in e$, $1/|e \cap S| \geq 1/|e \cap T|$. Furthermore, for any nonempty set S,

$$\sum_{j \in S} P_j(S) = \frac{1}{|E|} \sum_{j \in S} \sum_{e \in E} \frac{|e \cap \{j\}|}{|e \cap S|}$$

$$= \frac{1}{|E|} \sum_{e \in E} \sum_{j \in S} \frac{|e \cap \{j\}|}{|e \cap S|}$$

$$= \frac{1}{|E|} \sum_{e \in E} \frac{|e \cap S|}{|e \cap S|}$$

$$= \frac{1}{|E|} \sum_{e \in E} \mathbb{I}(e \cap S),$$

where $\mathbb{I}(e \cap S)$ is the indicator function that is 1 if $e \cap S \neq \emptyset$ and is 0 otherwise. Thus $\sum_{j \in S} P_j(S) \leq 1$.

We now show that the revenue function $R(S)$ is monotone and submodular. From the above, we have that

$$R(S) = |E| \sum_{j \in S} P_j(S) = \sum_{e \in E} \mathbb{I}(e \cap S)$$

so the revenue for a set S is simply the number of hyperedges with which S has a nonempty intersection. Clearly this function is monotone. It is also submodular since the number of additional hyperedges intersected by adding a new element j to S is at least as large as the number of additional hyperedges intersected by adding j to $T \supseteq S$; that is, for $T \supseteq S$,

$$R(S \cup \{j\}) - R(S) = \sum_{e \in E} \mathbb{I}(e \cap (S \cup \{j\})) - \sum_{e \in E} \mathbb{I}(e \cap S)$$

$$\geq \sum_{e \in E} \mathbb{I}(e \cap (T \cup \{j\})) - \sum_{e \in E} \mathbb{I}(e \cap T) = R(T \cup \{j\}) - R(T).$$

Now suppose we are given some ordering of the vertices given by a function f. Consider the sets $S_1 = \{f^{-1}(1)\}$, $S_2 = \{f^{-1}(1), f^{-1}(2)\}$, ..., $S_n = \{f^{-1}(1), \ldots, f^{-1}(n)\}$. Then the same ordering of the items gives a revenue of

$$|E| \sum_{i=1}^{n} \sum_{j \in S_i} P_j(S_i) = \sum_{i=1}^{n} \sum_{j \in S_i} \sum_{e \in E} \frac{|e \cap \{j\}|}{|e \cap S_i|}$$

$$= \sum_{e \in E} \sum_{i=1}^{n} \sum_{j \in S_i} \frac{|e \cap \{j\}|}{|e \cap S_i|}$$

$$= \sum_{e \in E} \sum_{i=1}^{n} \frac{|e \cap S_i|}{|e \cap S_i|}$$

$$= \sum_{e \in E} \sum_{i=1}^{n} \mathbb{I}(e \cap S_i).$$

We now observe that $\sum_{i=1}^{n} \mathbb{I}(e \cap S_i) + f(e) = n + 1$, since $f(e)$ is the smallest j for which $e \cap S_j \neq \emptyset$, while $\mathbb{I}(e \cap S_i)$ is 1 for $j \leq i \leq n$ and 0 for $i < j$.

Thus for any ordering of the vertices, the sum of the expected revenue plus the sum $\sum_{e \in E} f(e)$ is

$$|E| \sum_{j \in V} \sum_{i=1}^{n} P_j(S_i) + \sum_{e \in E} f(e) = \sum_{e \in E} \sum_{i=1}^{n} \left(\mathbb{I}(e \cap S_i) + f(e) \right) = (n+1)|E|.$$

Hence maximizing revenue for this instance is equivalent to minimizing the min-sum set cover objective function. Therefore, given an instance of the min-sum set cover problem in which we must check if the objective is at most B, we can reduce it to this instance of assortment optimization over time and check if the expected revenue is at least $(n+1)|E| - B$. Therefore, the decision version of our incremental assortment optimization problem is also NP-complete. ∎

We now say a few words about the particular choice model implied by the probabilities $P_j(S)$ given above. We can view each edge $e \in E$ as representing a particular customer type who is solely interested in the items in e but is indifferent between them. Given a set S of products, a uniformly random customer type e arrives, and selects uniformly at random amongst any of the $e \cap S$ items offered (if there are any). Consider now the capacitated version of this problem: suppose we want to choose $S \subseteq V$, $|S| \leq k$, to maximize $|E| \sum_{j \in S} P_j(S)$. From the above, we have that

$$R(S) = |E| \sum_{j \in S} P_j(S) = \sum_{e \in E} \mathbb{I}(e \cap S),$$

so we need to pick S, $|S| \leq k$, so as to maximize the number of hyperedges e with which S has some intersection (we say that S *covers* a hyperedge e if $S \cap e \neq \emptyset$). This problem is known as the *maximum coverage problem*, and it is well-known to be an NP-hard problem (see Feige [9]).

References

[1] A. Alptekinoglu and J. H. Semple. The exponomial choice model: A new alternative for assortment and price optimization. Technical report, Penn State, Smeal College of Business, 2013.

[2] F. Bernstein, A. G. Kok, and L. Xie. Dynamic assortment customization with limited inventories. Technical report, Duke University, Fuqua School of Business, 2011.

[3] J. J. M. Bront, I. Mendez Diaz, and G. Vulcano. A column generation algorithm for choice-based network revenue management. *Operations Research*, 57(3):769–784, 2009.

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

[5] F. Caro, V. Martinez de Albeniz, and P. Rusmevichientong. The assortment packing problem: Multiperiod assortment planning for short-lived products. *Management Science*, (to appear), 2014.

[6] E. Cinar and V. Martinez-de-Albeniz. A closed-loop approach to dynamic assortment planning. Technical report, University of Navara, IESE Business School, 2014.

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

[8] A. Desir and V. Goyal. An FPTAS for capacity constrained assortment optimization. Technical report, Columbia University, School of Industrial Engineering and Operations Research, 2013.

[9] Uriel Feige. A threshold of ln n for approximating set cover. *Journal of the ACM*, 45:634–652, 1998.

[10] Uriel Feige, László Lovász, and Prasad Tetali. Approximating min sum set cover. *Algorithmica*, 40:219–234, 2004.

[11] J. B. Feldman and H. Topaloglu. Capacity constraints across nests in assortment optimization under the nested logit model. Technical report, Cornell University, School of Operations Research and Information Engineering, 2014.

[12] N. Golrezaei, H. Nazerzadeh, and P. Rusmevichientong. Real-time optimization of personalized assortments. *Management Science*, (to appear), 2014.

[13] Dorothee Honhon, Vishal Gaur, and Sridhar Seshadri. Assortment planning and inventory decisions under stock-out based substitution. *Operations Research*, 58(5):1364–1379, 2010.

[14] S. Mahajan and G. van Ryzin. Stocking retail assortments under dynamic customer substitution. *Operations Research*, 49(3):334–351, 2001.

[15] I. Mendez-Diaz, J. J. M. Bront, G. Vulcano, and P. Zabala. A branch-and-cut algorithm for the latent-class logit assortment problem. *Discrete Applied Mathematics*, 36:383–390, 2010.

[16] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions — I. 14:265–294, 1978.

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

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

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