

Sequential Assortment Optimization in Random Consideration Set Model

Wenxiao Yang*

*Department of Mathematics, University of Illinois at Urbana-Champaign

2022

1 Introduction

Consider a consumer who visits a platform's website and continues to browse the platform's webpages. The platform would like to recommend products to the consumer; if the consumer notices several recommended products, he would buy the one he likes best if it can satisfy him. Should the platform modify the recommended products list when the consumer continues to browse webpages and recommendation lists will be offered again and again? The answer is, of course, yes; if any products were recommended to the consumer but were not purchased by the consumer, some information must exist and could be used to improve the next recommendation list provided to the consumer. However, most studies on choice models just look at how to deliver the best one-time list; they don't look at how platforms should use past recommendation list information to provide second, third, or beyond lists.

One assumption in Manzini and Mariotti (2014) and Gallego and Li (2017) is the consumer chooses non-purchase option if and only if the consideration set is empty. However, in this model, I set the consumer has his own reserve value, if all products' values in consideration set don't exceed his reserve value, he will still choose non-purchase option even if the consideration set is nonempty. This setting is closer to reality than previous setting, and we can utilize the previous offered list information based on this new setting.

Consistent ordering setting is inherited in this model. We assume the ordering of products $1, 2, \dots, n$ is $1 \prec 2 \prec \dots \prec n$. If a consumer's own reserve value is less than 1 we say his position is 0, if the consumer's own reserve value is less than 1 we sat his position is 0, if the value is higher than n , we say his position is n , if the value is higher than i and less than $i + 1$, $i = 1, 2, \dots, n - 1$, we say his position is i .

Process:

1. A consumer with reserve price's position v enters the first webpage.
2. The platform provides an offered set S (recommendation list).
3. The consumer forms a consideration set $C \subseteq S$. If the product with the highest order is higher than consumer's reserve price's position, i.e., $v < \max C$, the consumer will purchase the product with the highest order. Otherwise, the consumer chooses non-purchase option.
4. If consumer choose non-purchase in this webpage, he will continue to browse next webpage (repeat 2,3) unless he has viewed T webpages.

Inference:

Set the prior probability (at the beginning) of v at position $i, i = 0, \dots, n$ is $R^1 = \{\rho_0^1, \dots, \rho_n^1\}$ and $\sum_{i=0}^n \rho_i^1 = 1$.

The probability that a product i is noticed by the consumer is $\lambda_i, i = 1, \dots, n$. Set $\lambda_{n+1} = 0$.

The platform's profit of selling product i is $p_i, i = 1, \dots, n$. Set $p_0 = 0$.

1.1 Prior Probability Update Algorithm

2^{nd} offered set

Given 1^{st} offered set $S^1 = \{s_1^1, s_2^1, \dots, s_{m_1}^1\}$, the consumer selected non-purchase option. ($s_i^1, i = 1, \dots, m_1$ increases as i increases). Set $s_0^1 = 0, s_{m_1+1}^1 = n + 1$.

The non-purchase probability given S^1 and S^1 is

$$\begin{aligned} P(\text{non-purchase on } S^1) &= \sum_{i=0}^n P(\text{non-purchase on } S^1 | v = i) P(v = i) \\ &= \sum_{j=0}^{m_1} P(\text{non-purchase on } S^1 | s_j^1 \leq v < s_{j+1}^1) P(s_j^1 \leq v < s_{j+1}^1) \\ &= \sum_{j=0}^{m_1} \prod_{z=j+1}^{m_1+1} (1 - \lambda_{s_z^1}) \sum_{i=s_j^1}^{s_{j+1}^1-1} \rho_i^1 \end{aligned}$$

Since the consumer selected non-purchase option, there are two possibilities of the consumer's reserve price's location:

- (1) there is no item in S that can satisfy the consumer, i.e., $v \geq s_{m_1}^1$;

The probability of this situation that $v = k, k \geq s_{m_1}^1$:

$$P(v = k | \text{non-purchase on } S^1) = \frac{P(v = k, \text{non-purchase on } S^1)}{P(\text{non-purchase on } S^1)} = \frac{P(v = k)}{P(\text{non-purchase on } S^1)}$$

- (2) all items that can satisfy the consumer are overlooked, i.e., $\{x \in S^1 : x > v\} \cap C^1 = \emptyset$ (C^1 is the consideration set corresponding to S^1)

The probability of this situation that $v = k, s_j^1 \leq k < s_{j+1}^1, j = 0, 1, 2, \dots, m_1 - 1$:

$$\begin{aligned} P(v = k | \text{non-purchase on } S^1) &= P(\text{non-purchase on } S^1 | v = k) \frac{P(v = k)}{P(\text{non-purchase on } S^1)} \\ &= P(\text{non-purchase on } S^1 | s_j^1 \leq v < s_{j+1}^1) \frac{P(v = k)}{P(\text{non-purchase on } S^1)} \end{aligned}$$

In general, for a k s.t. $s_j^1 \leq k < s_{j+1}^1, j = 0, 1, 2, \dots, m_1$

$$\begin{aligned} P(v = k | \text{non-purchase on } S^1) &= P(v = k) \frac{\overbrace{P(\text{non-purchase on } S^1 | s_j^1 \leq v < s_{j+1}^1)}^{\text{only need compute } m \text{ times}}}{\underbrace{P(\text{non-purchase on } S^1)}_{\text{only need compute one time}}} \\ &= \frac{\rho_k^1 \prod_{z=j+1}^{m_1+1} (1 - \lambda_{s_z^1})}{\sum_{j=0}^{m_1} \prod_{z=j+1}^{m_1+1} (1 - \lambda_{s_z^1}) \sum_{i=s_j^1}^{s_{j+1}^1-1} \rho_i^1} \end{aligned}$$

$(t+1)^{th}$ **offered set**

Hence, in a more general situation, at providing t^{th} **offered set**, we have

- The **prior probability** of the consumer's position at i , ρ_i^t , $i = 0, 1, \dots, n$. $\sum_{i=0}^n \rho_i^t = 1$

The set of the ρ_i^t 's is $R^t = \{\rho_0^t, \rho_1^t, \dots, \rho_n^t\}$

- The t^{th} **offered set** in increasing preference degree order, $S^t = \{s_1^t, s_2^t, \dots, s_{m_t}^t\}$.

To assist computation, we also set $s_0^t = 0, s_{m_t+1}^t = n+1$

If the consumer choose non-purchase option in t^{th} offered set and stay at the website, the platform need to provide $(t+1)^{th}$ offered set. The platform can renew the prior probability set R^t to R^{t+1}

$$(1) \gamma^t = P(\text{non-purchase on } S^t) = \sum_{j=0}^{m_t} \prod_{z=j+1}^{m_t+1} (1 - \lambda_{s_z^t}) \sum_{i=s_j^t}^{s_{j+1}^t-1} \rho_i^t$$

$$(2) \eta_j^t = P(\text{non-purchase on } S^t | s_j^t \leq v < s_{j+1}^t) = \prod_{z=j+1}^{m_t+1} (1 - \lambda_{s_z^t}), \text{ for } j = 0, 1, \dots, m_t.$$

$$(3) \text{ Then, we can update } \rho_i^t \text{ to } \rho_i^{t+1}.$$

For $j = 0, \dots, m_t$:

For $k = s_j^t, \dots, s_{j+1}^t - 1$:

$$\rho_k^{t+1} = \frac{\eta_j^t}{\gamma^t} \rho_k^t$$

1.2 Optimization

Decide the t^{th} offered set $S^t = \{s_1^t, \dots, s_{m_t}^t\}$ with prior probability set R^t . If the consumer choose non-purchase option the prior probability set updated in next offered set's decision is $R^{t+1}(S^t)$ which is related to the S^t chosen in t^{th} .

What we want is to find the consumer's reserve price v 's position and try to match the consumer with the product that has the highest price among all products that are acceptable to the consumer.

1.3 Transformation

Set decision variables $x_i^t \in \{0, 1\}$, $X^t = \{x_1^t, \dots, x_n^t\} \in \{0, 1\}^n$.

$$\begin{aligned} P(\text{non-purchase on } S^t) &= \sum_{j=0}^n P(\text{non-purchase on } S^t | v = j) P(v = j) \\ &= \sum_{j=0}^n \rho_j^t \prod_{z=j+1}^{n+1} (1 - x_z^t \lambda_z) \end{aligned}$$

$$\begin{aligned} P(v = k | \text{non-purchase on } S^t) &= P(v = k) \frac{P(\text{non-purchase on } S^t | v = k)}{P(\text{non-purchase on } S^t)} \\ &= \rho_k^t \frac{\prod_{z=k+1}^n (1 - x_z^t \lambda_z)}{\sum_{j=0}^n \rho_j^t \prod_{z=j+1}^{n+1} (1 - x_z^t \lambda_z)} \\ &= \frac{\rho_k^t \prod_{z=k+1}^n (1 - x_z^t \lambda_z)}{\sum_{j=0}^n \rho_j^t \prod_{z=j+1}^{n+1} (1 - x_z^t \lambda_z)} \end{aligned}$$

Given X^t , the probability that the consumer buys $x_j^t, j = 1, \dots, n$ is $\sum_{i=0}^{j-1} \rho_i^t \prod_{z=j+1}^{n+1} (1 - x_z^t \lambda_z) x_j^t \lambda_j p_j$. However, the real profit of the platform is only related to v (the real reserve price of the consumer, unknown to the platform). The optimization problem is

$$\begin{aligned}
\Pi_v^t(X^t) &= \prod_{z=v+1}^{n+1} (1 - x_z^t \lambda_z) \Pi_v^{t+1}(X^{t+1}) + \sum_{j=v+1}^n \prod_{z=j+1}^{n+1} (1 - x_z^t \lambda_z) x_j^t \lambda_j p_j \\
\Pi_v^{T+1}(X) &= 0, \quad \forall X \in \{0, 1\}^n, \forall v = 0, 1, \dots, n \\
\Pi_n^t(X) &= 0, \quad \forall t = 1, \dots, T \\
X^t(R^t) &= \operatorname{argmax}_{X^t} \sum_{i=0}^n \rho_i^t \prod_{z=i+1}^{n+1} (1 - x_z^t \lambda_z) \Pi_i^{t+1}(X^{t+1}) + \sum_{i=0}^{n-1} \rho_i^t \sum_{j=i+1}^n \prod_{z=j+1}^{n+1} (1 - x_z^t \lambda_z) x_j^t \lambda_j p_j \\
&= \operatorname{argmax}_{X^t} \sum_{i=0}^{n-1} \rho_i^t \left[\prod_{z=i+1}^{n+1} (1 - x_z^t \lambda_z) \Pi_i^{t+1}(X^{t+1}) + \sum_{j=i+1}^n \prod_{z=j+1}^{n+1} (1 - x_z^t \lambda_z) x_j^t \lambda_j p_j \right] \\
&= \operatorname{argmax}_{X^t} \sum_{i=0}^{n-1} \rho_i^t \Pi_i^t(X^t)
\end{aligned}$$

1.4 Algorithm

$$X^T(R^T) = \operatorname{argmax}_X \sum_{i=0}^n \rho_i^T \sum_{j=i+1}^{n+1} \prod_{z=j+1}^{n+1} (1 - x_z \lambda_z) x_j \lambda_j p_j$$

Set $L(S) = \sum_{i=0}^n \rho_i^T \Pi_i^T(X^T)$, where $x_j = \begin{cases} 1 & \text{if } j \in S \\ 0 & \text{if } j \notin S \end{cases}$

Let k s.t. $i \prec k, \forall i \in S$

$$L(S \cup \{k\}) = \lambda_k p_k \sum_{i=0}^{k-1} \rho_i^T + (1 - \lambda_k) L(S)$$

Accept k if and only if $p_k \sum_{i=0}^{k-1} \rho_i^T > L(S)$. We can use $O(n)$ algorithm in book to compute the optimal $(S^T)^*$, i.e., $(X^T)^*$ when given R^T .

$$\Pi_v^T = \sum_{j=v+1}^n \prod_{z=j+1}^{n+1} (1 - x_z^T \lambda_z) x_j^T \lambda_j p_j$$

Compute the optimal offered set at T is quite easy. However, when we decide the offered set before T , we are facing a trade-off between purchase at that time and more accurate estimation in the next time. To help further analysis, we need to measure how the accuracy of R^T influences final Π_v^T .

Let's compare the algorithm between Given R^T and given v :

1. Given R^T : Let k s.t. $i \prec k, \forall i \in S$

$$L_R(S \cup \{k\}) = \lambda_k p_k \sum_{i=0}^{k-1} \rho_i^T + (1 - \lambda_k) L_R(S)$$

Accept k if and only if $p_k > \frac{L_R(S)}{\sum_{i=0}^{k-1} \rho_i^T}$. Underestimation (distribute some on $i < v$) increases $L_R(S)$, overestimation (distribute some on $i > v$) decreases $\sum_{i=0}^{k-1} \rho_i^T$. Both types of inaccuracy increase the standard of accepting $k > v$.

2. Given v : Let $k > v$ s.t. $i \prec k, \forall i \in S$

$$L_v(S \cup \{k\}) = \lambda_k p_k + (1 - \lambda_k) L_v(S)$$

Accept k if and only if $p_k > L_v(S)$, any $i \leq v$ won't be accepted in S .

(1) If R^T is inaccurate, some $i \leq v$ with high p_i will be accepted in S (no influences on Π_v^T)

(2) If R^T is inaccurate, some $i > v$ with low p_i will be rejected in S (decrease Π_v^T)

Hence, inaccuracy actually won't increase the purchase probability of low price products, it increases the purchase probability of high price product but decreases the total purchase probability.

Proposition 1. *As R^t convergence to $\{v\}$ (may prove converge firstly), the total purchase probability increases, but the average profit of selling products decreases.*

.....