The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)

Dynamic Learning of Sequential Choice Bandit Problem under Marketing Fatigue

Junyu Cao
University of California, Berkeley
Berkeley, California 94720
jycao@berkeley.edu

Wei Sun
IBM Research
Yorktown Height, New York 10591
sunw@us.ibm.com

Abstract

Motivated by the observation that overexposure to unwanted marketing activities leads to customer dissatisfaction, we consider a setting where a platform offers a sequence of messages to its users and is penalized when users abandon the platform due to marketing fatigue. We propose a novel sequential choice model to capture multiple interactions taking place between the platform and its user: Upon receiving a message, a user decides on one of the three actions: accept the message, skip and receive the next message, or abandon the platform. Based on user feedback, the platform dynamically learns users' abandonment distribution and their valuations of messages to determine the length of the sequence and the order of the messages, while maximizing the cumulative payoff over a horizon of length T. We refer to this online learning task as the *sequential choice bandit* problem. For the offline combinatorial optimization problem, we show a polynomial-time algorithm. For the online problem, we propose an algorithm that balances exploration and exploitation, and characterize its regret bound. Lastly, we demonstrate how to extend the model with user contexts to incorporate personalization.

1 Introduction

Service providers and retailers routinely rely on emails and app notifications to interact with their users. When it is done well, these messages act as digital reminders that increase customer engagement, raise brand awareness and conversion. However, frequent messaging can easily backfire. Marketing fatigue, which refers to an overexposure to unwanted marketing messages, could aggravate users and prompt them to forgo receipt of future messages by unsubscribing or deleting the app.

Motivated by this dilemma, we consider a setting where a platform needs to learn a policy which consists of a sequence of messages for its users. It has to decide *the order of the messages* as well as *the length of the sequence* from a pool of available messages. The messages are presented to a user sequentially. Upon reviewing a message i, a user takes one of the three actions: 1) accept the message and exit. In this case, the platform earns a reward r_i. If the user does not select the current message, she can either 2) receive the next message unless the sequence runs out, or 3) abandon the platform. When a user abandons, the platform incurs a penalty cost c from losing that user. Based on users' feedback, the platform learns two pieces of information in order to determine the optimal sequence, namely, users' valuations of individual messages and users' abandonment distribution. The objective of the platform is to maximize its expected payoff which is the revenue after subtracting the penalty cost due to abandonment. We refer to the online learning task which the platform faces as the *sequential choice bandit* (*SC-Bandit*) problem.

To draw a connection between this problem and the earlier motivating example, messages can represent digital marketing content such as an email or app notification regarding a product or service that a marketer wishes to promote. He[1] earns revenue whenever a user interacts with the content (e.g., click or purchase). The interaction is an indication that the content is of interest to that user. When the user ignores the content, there is a possibility that she will unsubscribe or delete the app. We can think of the abandonment cost c as the cost of user acquisition as the marketer replenishes his customer base. Based on a survey[2], the cost of customer acquisition is estimated to be 5 to 25 times higher than keeping an existing customer. Therefore, fatigue control is a critical component of digital marketing content dissemination.

There are several challenges associated with analyzing the SC-Bandit problem. Firstly, even in the offline setting where users' valuations and abandonment distribution are known, the sequence optimization problem is combinatorial in nature without an obvious efficient algorithm. Secondly, the sequential behavior of users complicates the learning task: while one can observe the response to the first offered message, the feedback to subsequent messages is not guaranteed due to abandonment. Thirdly, one needs to simultaneously learn valuations and abandonment distribution from users' feedback which depends on these two pieces of information jointly. The contribution of our work is fourfold:

1. We propose a novel sequential choice model which captures multiple interactions including abandonment between users and a platform.
2. We prove that the offline combinatorial optimization

[1]We refer to a marketer as *he*, and a user as *she*.

[2]https://hbr.org/2014/10/the-value-of-keeping-the-right-customers

problem allows an efficient polynomial-time algorithm.

3. For the online problem where valuations and abandonment distribution are unknown to the platform, we propose a learning algorithm and show that the regret is bounded above by $O(N\sqrt{T \log T})$ where N is the number of available messages and T is the time duration.
4. We incorporate personalization by solving a contextual SC-Bandit problem where valuations and abandonment distribution can vary with user features.

2 Literature Review

Multi-armed bandit problem Our work is closely related to the multi-armed bandit (MAB) problem, which has been well studied in the literature (e.g., Robbins 1985; Sutton, Barto, and others 1998). Several popular extensions include MAB with linear payoffs (Auer 2002; Agrawal and Goyal 2013), ranked bandits (Radlinski, Kleinberg, and Joachims 2008; Slivkins, Radlinski, and Gollapudi 2013), and the combinatorial MAB problem (Chen, Wang, and Yuan 2013). Our problem can be viewed as a combinatorial bandit problem where a platform chooses a set of messages to be displayed in a certain order. A naive approach is to treat each possible combination as an arm. However, the number of arms increases exponentially with the number of messages under this approach. Other combinatorial bandit work assuming linear reward (Auer 2002; Rusmevichientong and Tsitsiklis 2010) or independent rewards (Chen, Wang, and Yuan 2013) cannot be directly applied to our model. Our setup also shares some similarities with cascading bandits (Kveton et al. 2015). The task there is to select m messages with the highest click probabilities, where m is exogenous and the rewards are the same for all messages. In contrast, our task is to determine both m (the length of the sequence) and the order of the messages which have different revenues.

Some recent work such as Schmit and Johari 2018 has studied users' abandonment. In their setting, a user has a threshold drawn from an unknown distribution and she abandons if the platform's action x exceeds that threshold. The platform needs to learn the distribution while optimizing x to maximize its discounted reward. One of the key differentiators and novelty of our work is how we model abandonment in the presence of sequential behavior. The decision to abandon is an interplay of user's valuations which determine whether a user will select the message, and the abandonment distribution. The platform needs to learn both quantities and solve an integer programming problem to obtain the optimal policy.

Dynamic learning of assortment optimization problems Assortment optimization refers to the problem of selecting a set of products to offer to a group of customers so as to maximize the revenue that is realized when customers make purchases according to their preferences. It is a central topic in the operations management research literature. We refer the reader to Kök, Fisher, and Vaidyanathan 2008 for a comprehensive review. Talluri and Van Ryzin 2004 formulate the assortment planning problem by using a discrete choice model which is a multinomial logit model (Train 2009; Luce 2012) to describe user behavior.

More recent literature (Caro and Gallien 2007; Rusmevichientong, Shen, and Shmoys 2010; Agrawal et al. 2017a; 2017b; Cheung and Simchi-Levi 2017) focus on the dynamic assortment problem where the customer preferences are unknown a priori and need to be learnt. Our work can be viewed as a dynamic assortment problem to determine a set of messages and a specific display order. Existing dynamic assortment problems model a single interaction between the platform and a user, who can either choose an item from the assortment or leave without a purchase. In contrast, our model captures multiple interactions between the two - the sequential nature of the decision-making process is a key novelty of our work. The order of messages plays a crucial role in the analysis as message rewards vary and users could abandon the platform when unsatisfying messages are received.

3 Model

In this section, we formally introduce our setting. Assume there are N different messages for the platform to choose from. Let X be the set of these N messages. Each message i generates revenue r_i when it is selected by a user. Customers arrive at time $t = 1, \cdots, T$. For a customer arriving at time t, the platform determines a sequence of messages $\mathbf{S}^t = S_1^t \oplus S_2^t \oplus \cdots \oplus S_m^t$, where S_i^t consists of a single message for any $1 \le i \le m$ and "$\oplus$" denotes the operator of union which also preserves the order. The platform's decision includes both the order of the messages as well as the total length of the sequence m.

Messages are displayed sequentially to a user according to the pre-specified order. Thus, messages at the front of the sequence will be displayed first and are considered to have higher priorities. If a user selects a message, she exits the platform and no further messages will be shown to her. The platform earns r_i. On the other hand, when a message is not selected, we consider its content unsatisfying, since they are not of sufficient interest to the user. When that happens, the user can either choose to abandon the platform, or see the next message if the sequence runs out. Abandonment will cause a penalty cost c to the platform.

Abandonment distribution under marketing fatigue We assume the probability that a user abandons the platform upon receiving each unsatisfying message is p. Each user arriving at time t can be characterized by a random variable W^t, drawn from a distribution F_W. W^t is a proxy for user patience, which measures the maximum number of unsatisfying messages that a user can tolerate before abandoning the platform. Under this setup, it implies that F_W is a geometric distribution with parameter p. Let $q = 1 - p$.

The probability of upon receiving the k^{th} unsatisfying message is $P(W = k) = q^{k-1}(1 - q)$. The probability that a user has not abandoned after k unsatisfying messages is $P(W > k) = q^k$, which is also the probability that a user's patience is larger than k.

Sequential choice model For every message i, its probability of being selected is u_i, where $0 \le u_i < 1$. This quantity can be directly derived from users' valuation of message i which reflects users' preferences. For the rest of the paper, we will refer to u_i as valuation to avoid confusion with $p_i(\mathbf{S})$ which we will define next. When message i is part of a sequence $\mathbf{S} = S_1 \oplus S_2 \oplus \cdots \oplus S_m$, the probability of being selected which is denoted as $p_i(\mathbf{S})$, depends on its position in the sequence as well as the content of other messages shown earlier. Formally,

$$p_i(\mathbf{S}) = \begin{cases} u_i, & \text{if } i \in S_1 \\ P(W \ge l) \prod_{k=1}^{l-1} (1 - u_{I(k)}) u_i, & \text{if } i \in S_l, l \ge 2 \\ 0, & \text{if } i \notin \mathbf{S}, \end{cases}$$

where $I(\cdot)$ denote the index function, i.e., $I(k) = i$ if and only if $S_k = \{i\}$. With the exception being at S_1 where it is the first message in the sequence, the probability of selecting message $I(l)$ at the subsequent levels is the joint probability that 1) the user has not yet abandoned at $l-1$ level, $P(W > l-1) = P(W \ge l)$; 2) she has not selected any earlier messages, $\prod_{k=1}^{l-1}(1 - u_{I(k)})$; 3) she selects message $I(l)$ when it is displayed, $u_{I(l)}$.

Given a sequence of messages $\mathbf{S} = S_1 \oplus S_2 \oplus \cdots \oplus S_m$, define $p_a(\mathbf{S})$ as the total abandonment probability over its entire length, which can be expressed as

$$p_a(\mathbf{S}) = \sum_{k=1}^{m} P(W = k) \prod_{j=1}^{k} (1 - u_{I(j)}).$$

It sums over the joint probabilities of not selecting the first k messages and abandoning at the k^{th} level upon receiving the k^{th} unsatisfying message.

Payoff optimization problem Let $U(\mathbf{S}, \mathbf{u}, q)$ denote the total payoff that the platform receives from a given sequence of messages $\mathbf{S}$ when the valuation is $\mathbf{u}$ and abandonment follows the geometric distribution with parameter $1-q$. For simplicity of notation, we use $U(\mathbf{S})$ to denote $U(\mathbf{S}, \mathbf{u}, q)$. The expected payoff which the platform is trying to optimize is defined as

$$E[U(\mathbf{S})] = \sum_{i \in X} p_i(\mathbf{S}) r_i - c p_a(\mathbf{S}),$$

where c is the cost of losing a customer due to abandonment. In contrast to the traditional assortment problems which only focus on revenue maximization, the objective in our model also includes a penalty of losing customer.

The platform's optimization problem is defined as follows,

$$\begin{aligned} \max_{\mathbf{S}} \quad & E[U(\mathbf{S})] \\ s.t. \quad & S_i \cap S_j = \emptyset, \forall i \ne j. \end{aligned} \tag{3.1}$$

The constraint specifies that the sequence cannot contain duplicated messages. It is included to avoid unrealistic solutions where the optimal sequence consisting of identical messages due to the memoryless property of geometric distribution. We denote the optimal sequence of messages as $\mathbf{S}^* = \mathrm{argmax}_{\mathbf{S}} E[U(\mathbf{S})]$.

4 Characterization of the Optimal Sequence

In this section, we describe an algorithm to solve the optimal payoff optimization problem when the valuation $\mathbf{u}$ and abandonment distribution F_W are both known to the platform. It is an integer programming problem as the platform needs to choose a subset from all available messages and also specify the order. In addition, the choice probability of a particular message $p_i(\mathbf{S})$ depends on its valuation, as well as the valuation of previous messages shown to the user. This dependence makes the problem much more complicated. We will show in the following result that under the assumption of geometric abandonment distribution, there exists an efficient algorithm for our problem.

Theorem 1. *For message $i \in \{1, \cdots, N\}$, define its score as follows,*

$$\theta_i := \frac{r_i u_i - cp(1-u_i)}{1 - q(1-u_i)}.$$

Without loss of generality, assume messages are sorted in decreasing order of their scores, i.e., $\theta_1 \ge \theta_2 \ge \cdots \ge \theta_N$. Then the optimal sequence of messages is $\mathbf{S}^* = \{1\} \oplus \{2\} \oplus \cdots \oplus \{m\}$, where $m = \max\{i : r_i u_i - cp(1-u_i) > 0\}$.

Due to the space constraint, we only include proof sketches for the key results in the paper. All detailed proofs can be found in the supplementary material.

Proof sketch: We prove Theorem 1 by contradiction. If the optimal sequence $\mathbf{S}^*$ is not ordered by the decreasing order of θ, then there exists $S^*_k = \{i\}$, $S^*_{k+1} = \{j\}$ such that $\theta_i < \theta_j$. We compare the payoff generated under this sequence with an alternative sequence whose order of i and j is switched. We show that the alternative sequence generates a higher payoff, which is a contradiction to the fact that $\mathbf{S}^*$ is the optimal. ■

The score θ_i can be interpreted as follows: $r_i u_i$ is the expected revenue when displaying message i, while $cp(1-u_i)$ is its expected abandonment cost. Thus, the numerator denotes the expected payoff of message i. The denominator is the probability of two events: 1) choose message i; 2) abandon the platform after viewing message i. Therefore, the score θ_i is a *normalized* expected payoff, conditioned on the probability conditional on the event that message i is making an impact to the payoff.

Theorem 1 states that all messages with a positive expected payoff should be included in the optimal sequence whose order is determined by their scores. Theorem 1 provides an efficient algorithm with complexity $O(N \log N)$ (where the complexity comes from sorting N messages) and shows that this problem is polynomial-time solvable.

A special case of Theorem 1 is when $p = 0$, i.e., users never abandon the platform. The following result states that under this scenario, the optimal sequence only depends on the revenue of the messages.

Proposition 2. *With the abandonment probability $p = 0$, the optimal sequence is ordered by its revenue. That is, $r_{I(1)} \ge$*

$r_{I(2)} \geq \cdots \geq r_{I(N)}$*, where* I *is the index function of the optimal sequence* $\mathbf{S}^*$.

With $p = 0$, a user will either select one of the messages and generate revenue r_i, or leave without any selection after the entire sequence has been shown. Without the risk of user abandonment, the platform can show *all* available messages to a user. In addition, as messages are viewed sequentially, those with higher revenue should have higher priorities and be shown first.

In the next result, we show that we can compare the expected payoff generated under different abandonment distributions if they follow a stochastic order which is stated below for completeness. We want to emphasize that Proposition 4 holds under any general distribution for user abandonment, and is not restricted to the geometric distribution.

Definition 3 (Stochastic order)**.** *Real random variable* W_1 *is stochastically larger than or equal to* W_2*, denoted as* $W_1 \gtrsim_{s.t.} W_2$*, if*

$$P(W_1 > x) \geq P(W_2 > x) \text{ for all } x \in \mathbb{R}.$$

Proposition 4. *Assume* $\mathbf{S}'$ *and* $\mathbf{S}''$ *are the optimal sequences generated under abandonment distribution* W_1 *and* W_2 *respectively. If* $W_1 \gtrsim_{s.t.} W_2$*, we have*

$$E[U(\mathbf{S}', \mathbf{u}, F_{W_1})] \geq E[U(\mathbf{S}'', \mathbf{u}, F_{W_2})],$$

where $U(\mathbf{S}, \mathbf{u}, F_W)$ *denotes the payoff under strategy* $\mathbf{S}$ *when the valuation and abandonment distribution are* $\mathbf{u}$ *and* F_W *respectively.*

The definition of $W_1 \gtrsim_{s.t.} W_2$ implies that users under F_{W_1} are more patient, as they are less likely to abandon the platform upon receiving the same number of unsatisfying messages than their counterparts under F_{W_2}. Thus, intuitively, Proposition 4 states that the expected payoff is higher when users are more patient.

5 Online Learning

In the previous section, we assumed that both valuations and the user abandonment distribution are known to the platform. It is natural to ask what the platform should do in the absence of such knowledge. In this section we will present an exploration-exploitation algorithm for the SC-Bandit problem and characterize its regret bound. We would like to contrast our model with traditional bandit settings (e.g., Auer 2002): 1) Due to the sequential user behavior and the presence of abandonment, only partial feedback is obtained for learning; 2) The algorithm has to tease out two unknown quantities which jointly influence user feedback. The aforementioned features of the SC-Bandit makes the analysis of its regret bound much more challenging and involved.

5.1 Algorithm

We will present a UCB-based algorithm for the SC-Bandit problem to learn the users' valuation u_i for message i and as well as the abandonment distribution parameter q. To characterize the upper confidence bounds, we first identify the unbiased estimators $\hat{u}_i(t)$ and $\hat{q}(t)$ respectively.

Define $T_i(t)$ as the total number of users who observe message i by time t and $c_i(t)$ as the total number of users selecting message i. Note that a user does not necessarily observe message i even if i is included in the offered sequence $\mathbf{S}$ if she abandons the platform before this message is shown.

Let $n_a(t)$ denote the number of users who abandon the platform by time t. We use $n_e(t)$ to denote the number of times that users refuse a message without abandonment by time t. For example, suppose a user at $t = 1$ refuses the first two messages and abandons upon receiving the third message, then $n_e(1) = 2$ and $n_a(1) = 1$. Let $N_q(t) = n_e(t) + n_a(t)$, which denotes the total number of times users turn down unsatisfying messages by time t.

Lemma 5 (Unbiased estimator)**.** $\hat{u}_i(t) = c_i(t)/T_i(t)$ *is an unbiased estimator for* u_i*. Moreover,* $\hat{q}(t) = n_e(t)/N_q(t)$ *is an unbiased estimator for* q*.*

With Lemma 5 which gives the unbiased estimators, define the upper confidence bound for valuation $\mathbf{u}$ and abandonment distribution parameter q as follows,

$$u_{i,t}^{UCB} = \hat{u}_i(t) + \sqrt{2 \log t / T_i(t)} \tag{5.1}$$

and

$$q_t^{UCB} = \hat{q}(t) + \sqrt{2 \log t / N_q(t)}. \tag{5.2}$$

Algorithm 1 proposed below is an exploration-exploitation algorithm for the SC-Bandit problem which simultaneously learns valuations and abandonment distribution. For a user arriving at time t, we use $u_{i,t-1}^{UCB}$ and q_{t-1}^{UCB} to calculate the current optimal sequence of messages and offer them sequentially to the user. Let k_t denote the last message seen by user t, which occurs when one of the following feedback is observed: 1) the user chooses a message; 2) the user abandons the platform; 3) the sequence runs out. We update the upper confidence bound to $u_{i,t}^{UCB}$ and q_t^{UCB} respectively when the last message k_t is shown.

5.2 Regret Bound

The regret for a policy π is defined as follows,

$$Reg_\pi(T; \mathbf{u}, q) = E_\pi \left[\sum_{t=1}^{T} U(\mathbf{S}^*, \mathbf{u}, q) - U(\mathbf{S}^t, \mathbf{u}, q) \right],$$

where $\mathbf{S}^*$ is the optimal sequence when $\mathbf{u}$ and q are known to the platform, while $\mathbf{S}^t$ is the sequence offered to the user arriving at time t. E_π denotes the expectation under the policy π.

To analyze the regret, we first establish the following results. In Lemma 6, we provide the concentration analysis of $u_{i,t}^{UCB}$ and q_t^{UCB} using Hoeffding's inequality.

Lemma 6 (Concentration bound)**.** *For any* $T_i(t)$ *and* $N_q(t)$*, we have*

$$P\left(u_{i,t}^{UCB} - \sqrt{8 \frac{\log t}{T_i(t)}} < u_i < u_{i,t}^{UCB} \right) \geq 1 - \frac{2}{t^4}$$

and

$$P\left(q_t^{UCB} - \sqrt{8 \frac{\log t}{N_q(t)}} < q < q_t^{UCB} \right) \geq 1 - \frac{2}{t^4}.$$

Algorithm 1: An exploration-exploitation algorithm for SC-Bandit under marketing fatigue

Initialization: Available messages X with known revenues $\mathbf{r}$; set $u_{i,0}^{UCB} = 1$ for all $i \in X$ and $q_0^{UCB} = 1$; $n_e(0) = 0$; $n_a(0) = 0$; $t = 1$;
while $t < T$ **do**
  Compute $\mathbf{S}^t = \text{argmax}_{\mathbf{S}} \quad E[U(\mathbf{S}, \mathbf{u}_{t-1}^{UCB}, q_{t-1}^{UCB})]$ according to Theorem 1;
  Offer sequence $\mathbf{S}^t$, observe the user's feedback upon receiving the k_t messages;
  for $i = 1 : k_t$ **do**
    update $u_{I(i),t}^{UCB}$ according to Equation (5.1);
  end
  update $n_e(t), n_a(t)$;
  update q_t^{UCB} according to Equation (5.2);
  $t = t + 1$;
end

Next, Lemma 7 shows that with the optimal sequence $\mathbf{S}^*$ determined under $\mathbf{u}$ and q, its expected payoff is smaller than or equal to the payoff under the same strategy $\mathbf{S}^*$ when valuation $\mathbf{u}$ and the abandonment distribution parameter q are higher. Note that this result only holds for $\mathbf{S}^*$, and does not generally hold for other sequence $\mathbf{S}$.

Lemma 7. *Assume $\mathbf{S}^*$ is the optimal sequence of messages. Under the condition that $0 \leq \mathbf{u} \leq \mathbf{u}^{UCB}$ and $0 \leq q \leq q^{UCB}$, we have*

$$E[U(\mathbf{S}^*, \mathbf{u}^{UCB}, q^{UCB})] \geq E[U(\mathbf{S}^*, \mathbf{u}, q)].$$

Proof sketch: Define $E[U(\mathbf{S}_j)]$ as the expected payoff conditioned on a user is receiving the j^{th} message in a sequence. We show that this term can be expressed as $E[U(\mathbf{S}_j)] = r_{I(j)}u_{I(j)} + (1 - u_{I(j)}) (qE[U(\mathbf{S}_{j+1})] - pc)$, which is a sum of the expected payoff generated if message $I(j)$ is selected and the future payoff if $I(j)$ is not selected. Note that the inequality $r_{I(j)} \geq E[U(\mathbf{S}_j^*)] \geq qE[U(\mathbf{S}_{j+1}^*)] - pc$ must hold. Otherwise, removing message $I(j)$ will improve the expected payoff. Using this condition, we prove by induction that $E[U(\mathbf{S}_j^*, \mathbf{u}^{UCB}, q^{UCB})] \geq E[U(\mathbf{S}_j^*, \mathbf{u}, q)]$ for all j. By definition, $E[U(\mathbf{S}_1^*)] = E[U(\mathbf{S}^*)]$, and this completes the proof. ■

Putting everything together, we characterize a regret bound of our online learning algorithm in Theorem 8.

Theorem 8 (Performance bounds for Algorithm 1)**.** *Given valuation u_i of message i, $i \in X$ and parameter q, the regret of policy π during time T is bounded by*

$$Regret_\pi(T; \mathbf{u}, q) = O\left(N\sqrt{T \log T}\right)$$

where N is the total number of messages.

We want to highlight the difficulty of this regret analysis due to the incomplete feedback we observe after a sequence is offered. We are unable to estimate the parameter q if users keep on selecting messages. Meanwhile, we are also unable to estimate the valuation $\mathbf{u}$ for those messages which are offered but are not seen by a user. The complete proof can be found in the supplementary material.

Proof sketch: Define the "large" probability event $D_t := \bigcap_{i=1}^N \left(u_{i,t}^{UCB} - \sqrt{8 \log t / T_i(t)} < u_i < u_{i,t}^{UCB}\right) \cap \left(q_t^{UCB} - \sqrt{8 \log t / N_q(t)} < q < q_t^{UCB}\right)$. To bound the regret, we consider the quantity $E[U(\mathbf{S}^*, \mathbf{u}, q) - U(\tilde{\mathbf{S}}^t, \mathbf{u}, q)]$ given D_t and given its complement D_t^c, respectively.

Define $\tilde{\mathbf{S}}^t$ as the optimal sequence when the valuation is $\mathbf{u}_t^{UCB}$ and the geometric parameter for the abandonment distribution is $1 - q_t^{UCB}$. By the definition of $\tilde{\mathbf{S}}^t$ and $\mathbf{S}^*$, and with Lemma 7, we have $E_\pi[U(\tilde{\mathbf{S}}^t, \mathbf{u}, q)] \leq E_\pi[U(\mathbf{S}^*, \mathbf{u}, q)] \leq E_\pi[U(\mathbf{S}^*, \mathbf{u}_t^{UCB}, q_t^{UCB})] \leq E_\pi[U(\tilde{\mathbf{S}}^t, \mathbf{u}_t^{UCB}, q_t^{UCB})]$ on D_t. Thus, the difference $E_\pi\left[\sum_{t=1}^T U(\mathbf{S}^*, \mathbf{u}, q) - U(\tilde{\mathbf{S}}^t, \mathbf{u}, q)\right]$ can be bounded above by the expected difference between $U(\tilde{\mathbf{S}}^t, \mathbf{u}_t^{UCB}, q_t^{UCB})$ and $U(\tilde{\mathbf{S}}^t, \mathbf{u}, q)$. This quantity can be further expressed as the sum of two terms which can be analyzed separately, namely, one term is related to the estimated error $q_t^{UCB} - q$, while another is related to the error $(u_{i,t}^{UCB} - u_i)1(i \in \tilde{\mathbf{S}}^t)$.

Next, using the coupling method, we bound the error term on q. To analyze the regret term of $\mathbf{u}$, we derive the relation between the probability of exploring message i and the expected regret caused by the error of $u_{i,t}^{UCB} - u_i$. With Lemma 6, the regret on D_t^c can be bounded. Combining the regret on D_t and on D_t^c, we show that the total regret can be bounded above by $O(N\sqrt{T \log T})$. ■

6 Personalization with Contextual SC-Bandit

Thus far, we have considered a setting where the platform determines an optimal sequence $\mathbf{S}^*$ for all its users who share the identical abandonment distribution and valuations. In this section, we consider a more realistic setting where the abandonment distribution and valuations could differ across users based on some user context $\mathbf{x}$. In other words, instead of learning the homogeneous parameter q and u_i, the platform needs to learn $q(\mathbf{x})$ and $u_i(\mathbf{x})$ which will be used to determine personalized messaging sequences.

Contextual bandit is an active research area that has received lots of attention in recent years (Chu et al. 2011; Li et al. 2010; 2012; Cheung and Simchi-Levi 2017). A common assumption is a linear relationship between the reward and the context. In our setting, since both q and $\mathbf{u}$ denote probabilities, and the observed actions are either 0 or 1 (i.e., whether a user selects a message or abandons the platform), we use the logit model to model $q(\mathbf{x})$ and $u_i(\mathbf{x})$ respectively. That is,

$$q(\mathbf{x}) = e^{\alpha^T \mathbf{x}} / (1 + e^{\alpha^T \mathbf{x}}),$$

and

$$u_i(\mathbf{x}) = e^{\beta_i^T \mathbf{x}} / (1 + e^{\beta_i^T \mathbf{x}}),$$

where $\alpha \in \tilde{\Theta}$ and $\beta_i \in \Theta_i$ are the unkown parameters to be learnt. Then $q(\mathbf{x})$ and $u_i(\mathbf{x})$ are formulated as generalized linear models (GLM), i.e., $q(\mathbf{x}) = \mu(\alpha^T \mathbf{x})$ and $u_i(\mathbf{y}) = \mu(\beta_i^T \mathbf{x})$, where $\mu(x) = \exp(x)/(1 + \exp(x))$.

Next, we will propose an exploration-exploitation algorithms for the contextual SC-Bandit problem. We adapt the GLM-UCB algorithm proposed by (Filippi et al. 2010) for our contextual SC-Bandit problem. The key difference from the non-contextual version is that, during each update, we calculate the maximum quasi-likelihood estimator of the parameter $\hat{\beta}_i$, and then update $u_i^{UCB}(\mathbf{x})$ with $\mu(\hat{\beta}_i^T \mathbf{x})$ plus an "exploration bonus" term defined by $\rho(t)\|\mathbf{x}\|_{M_{i,t}^{-1}}$, where $\rho(t)$ is a slowly varying function which can be set as $\rho(t) = \sqrt{2\log(t)}$, $\|v\|_M = \sqrt{v' M v}$ denotes the matrix norm induced by the positive semidefinite matrix M with $M_{i,t} = \lambda \mathbf{I} + \sum_{k=1}^{t-1} \mathbf{x}_k \mathbf{x}_k' 1(\text{user } \mathbf{x}_k \text{ observed message } i)$ where $1(\cdot)$ is the indicator function, and λ is a constant. The update is similar for $q_t^{UCB}(\mathbf{x})$, i.e., $q_t^{UCB}(\mathbf{x}) = \mu(\hat{\alpha}_t^T \mathbf{x}) + \rho(t)\|\mathbf{x}\|_{\tilde{M}_t^{-1}}$ where $\tilde{M}_t = \lambda' \mathbf{I} + \sum_{k=1}^{t-1} \mathbf{x}_k \mathbf{x}_k' n_k$, n_k denotes the number of messages that user k observes, and λ' is a constant.

To initialize the algorithm, for the first N users, we offer the i^{th} user message i. In each iteration, we first update $\hat{\alpha}_{t-1}$ and $\hat{\beta}_{i,t-1}$ based on prior user feedback. Next, we update $q_{t-1}^{UCB}(\mathbf{x})$ and $u_{i,t-1}^{UCB}(\mathbf{x})$ for the user t with feature $\mathbf{x}$. The optimal messaging sequence is obtained by solving the optimization problem $\max_{\mathbf{S}} E[U(\mathbf{S}, \mathbf{u}_{t-1}^{UCB}(\mathbf{x}), q_{t-1}^{UCB}(\mathbf{x}))]$. For completeness, the GLM-UCB algorithm is given below.

Algorithm 2: GLM-UCB algorithm I for contextual SC-Bandit under marketing fatigue

Initialization: Available messages X with known revenues $\mathbf{r}$. Offer each message $1, 2, \cdots, N$ to user $1, 2, \cdots, N$, observe decision;
Update $M_{i,t}, \tilde{M}_t$; $t = N$;
while $t < T$ **do**
  Update $\hat{\alpha}_t$ and $\hat{\beta}_{i,t}$ by quasi-MLE; $t = t + 1$;
  Observe user's contextual information $\mathbf{x}_t$ Compute $\mathbf{S}^t = \operatorname{argmax}_{\mathbf{S}} \quad E[U(\mathbf{S}, \mathbf{u}_{t-1}^{UCB}(\mathbf{x}_t), q_{t-1}^{UCB}(\mathbf{x}_t))]$ according to Theorem 1 where $q_{t-1}^{UCB}(\mathbf{x}_t)$ and $\mathbf{u}_{t-1}^{UCB}(\mathbf{x}_t)$ are computed by

$$u_{i,t-1}^{UCB}(\mathbf{x}_t) = \mu(\hat{\beta}_{i,t-1}^T \mathbf{x}_t) + \rho(t)\|\mathbf{x}_t\|_{M_{i,t-1}^{-1}}, \forall i$$

  and

$$q_{t-1}^{UCB}(\mathbf{x}_t) = \mu(\hat{\alpha}_{t-1}^T \mathbf{x}_t) + \rho(t)\|\mathbf{x}_t\|_{\tilde{M}_{t-1}^{-1}}.$$

  Offer personalized messaging sequence $\mathbf{S}^t$, observe the user's decision;
  Update $M_{i,t}, \tilde{M}_t$;
end

7 Numerical Experiments

In this section, we first investigate the robustness of Algorithm 1 which is our proposed UCB-algorithm for the SC-Bandit problem by comparing how the regret changes with respect to different values of $\mathbf{u}$. Next, we compare our Algorithm 1 and 2 with two benchmarks in the non-contextual and contextual settings respectively.

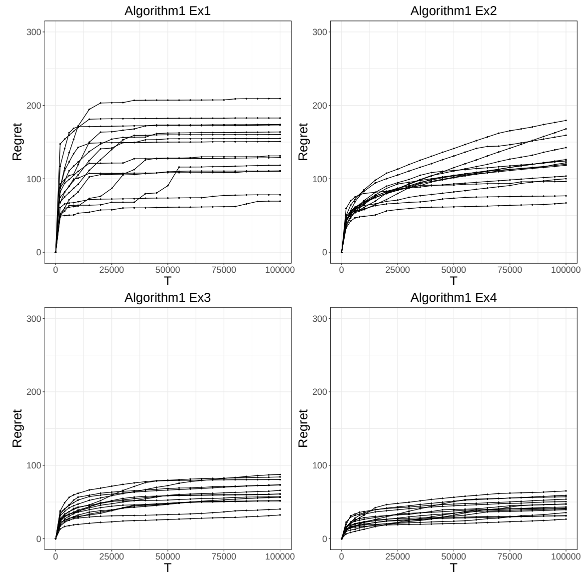

Figure 1: Comparison of Algorithm 1 when $\mathbf{u}$ is uniformly generated from [0,0.1], [0,0.2], [0.0.3], and [0.0.5], respectively.

7.1 Robustness of the SC-Bandit Algorithm

Experiment setup We consider a setting with $N = 30$, revenue r_i is uniformly distributed between [0,1], abandonment distribution probability $p = 0.1$ and the cost of abandonment $c = 0.5$. We present four scenarios, when the valuation $\mathbf{u}$ is uniformly generated from [0,0.1], [0,0.2], [0.0.3], and [0.0.5], respectively.

Result Figure 1 shows the results based on 15 independent simulations for different scenarios of $\mathbf{u}$. The average regrets are 141.13, 121.91, 59.69, and 44.64, respectively. Figure 1 suggests that when u_is are more spread out, it is easier for the algorithm to learn them to a large degree. Meanwhile, Figure 1 also reveals something more subtle. When $\mathbf{u}$ is generated uniformly from [0,0.1], Algorithm 1 is able to find the optimal sequence before $T = 25000$ for a large fraction of the simulations. On the other hand, when $\mathbf{u}$ is generated uniformly from [0,0.3] or [0,0.5], the regret continues to increase after the initial 100,000 iterations, indicating that the algorithm has not found the optimal sequence yet. The intuition is that with higher valuations, the length of the optimal sequence could become longer. As a result, it is slower to learn the values of $\mathbf{u}$ precisely (especially for those messages which are placed later in the sequence), despite learning their approximate values quickly.

7.2 Comparison with Benchmark Algorithms

We analyze two benchmarks and compare their results with our algorithm. The first benchmark is an explore-then-exploit algorithm, while the second "enhances" the first benchmark by exploiting the knowledge it has already learned during its exploration phase.

Benchmark-1 With the explore-then-exploit approach, there is an exploration phase where every message is learnt for at least $\gamma \log(t)$ times during the time period $[0, t]$, where γ is a tuning parameter. After this phase, the algorithm uses the estimated parameters to determine an optimal sequence which is offered to all subsequent users. We want to highlight that our setting differs from the traditional multi-armed bandit problem where an arm will be pulled if it is selected. In our setting, messages which are to appear later in the sequence may not be viewed by a user. Thus, in order to guarantee that message i is explored, we only offer a single message in a sequence during the exploration phase, i.e., $\mathbf{S}^t = \{i\}$.

Benchmark-2 This algorithm is a variant of Benchmark-1. During its exploration phase, suppose this benchmark aims to learn the valuation of message i. It first solves the optimal sequence problem based on the valuations of the messages which it has already learned, and then appends message i to the beginning of the sequence. Thus, Benchmark-2 learns faster than Benchmark-1 as it offers more messages each time. In addition, it can optimize the sub-sequence to earn higher revenue than its counterpart, making it a competitive baseline. The optimization problem one needs to solve here is nearly identical to (3.1) with an additional constraint that $S_1 = \{i\}$. It can be proven that the optimal solution with $S_1 = \{i\}$ as the first message orders the messages of the remaining sequence according to θ_i as defined in Theorem 1.

Experiment setup for SC-Bandit without contexts We consider a setting that $N = 30$, r_i is uniformly distributed between [0,1], $p = 0.1$, $c = 0.5$ and $\mathbf{u}$ is uniformly generated from [0,0.1].

Experiment setup for SC-Bandit with contexts We consider a setting with $N = 30$, r_i is uniformly distributed between $[0, 1]$. The user feature $\mathbf{x}$ is uniformly generated from $[0, 1]^3$. The coefficient related to the abandonment distribution is $\alpha = (0.25, 0.5, 1, 0.8)$ where α_1 is the intercept. The coefficient related to the valuation of message i, β_i, is uniformly generated from $[-2.5, 0]^2 \times [0, 0.5]^2$ where $\beta_{i,1}$ is the intercept.

Result Figure 2 and 3 shows the average regret of our algorithm and the two benchmarks under the non-contextual and contextual settings respectively. It is clear that our algorithm outperforms both benchmarks. In particular, Benchmark-2 does better than Benchmark-1 as it incorporates learning during its exploration phase.

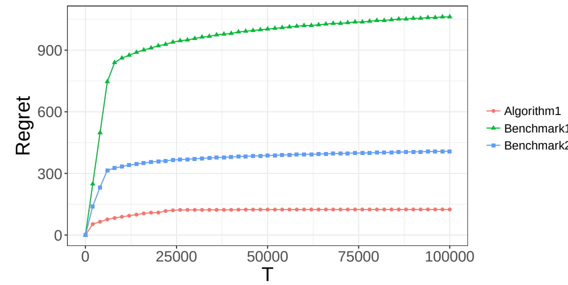

Figure 2: Comparison between Algorithm 1 and two benchmark algorithms in the non-contextual bandit setting.

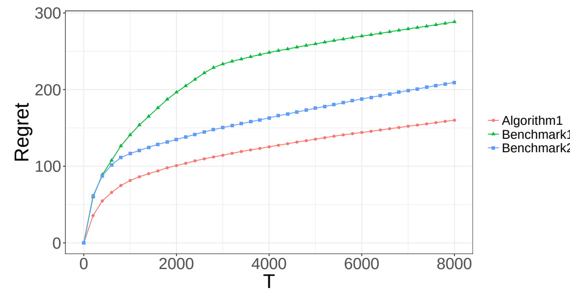

Figure 3: Comparison between Algorithm 2 and two benchmark algorithms in the contextual bandit setting.

8 Conclusion

In this work, we studied dynamic learning of a sequential choice bandit problem when users could abandon the platform due to marketing fatigue. We showed that there exists an efficient algorithm to solve the offline optimization problem that determines an optimal sequence of messages to offer to users. For the online learning problem, we proposed an exploration-exploitation algorithm and showed that the resulting regret is bounded by $O\left(N\sqrt{T \log T}\right)$. Lastly, we proposed a GLM-UCB algorithm to incorporate personalization with user contexts.

There are several future directions of this work. Firstly, as users' preferences may vary over time, it is interesting to incorporate the temporal dimension into the setting. Secondly, different user actions could reveal different levels of interest (e.g., the amount of time a user spent on a message, a user clicked on a message but did not complete a purchase etc.). One question is how to construct and analyze a more accurate user behavior model by utilizing such data. Thirdly, Thompson Sampling would be another natural algorithm to solve the problem we proposed, especially for the personalized version. However, analyzing this setting and providing theoretical results remain a challenging problem.

References

Agrawal, S., and Goyal, N. 2013. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, 127–135.

Agrawal, S.; Avadhanula, V.; Goyal, V.; and Zeevi, A.

2017a. Mnl-bandit: A dynamic learning approach to assortment selection. *arXiv preprint arXiv:1706.03880*.

Agrawal, S.; Avadhanula, V.; Goyal, V.; and Zeevi, A. 2017b. Thompson sampling for the mnl-bandit. *arXiv preprint arXiv:1706.00977*.

Auer, P. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3(Nov):397–422.

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

Chen, W.; Wang, Y.; and Yuan, Y. 2013. Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, 151–159.

Cheung, W. C., and Simchi-Levi, D. 2017. Thompson sampling for online personalized assortment optimization problems with multinomial logit choice models.

Chu, W.; Li, L.; Reyzin, L.; and Schapire, R. 2011. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 208–214.

Filippi, S.; Cappe, O.; Garivier, A.; and Szepesvári, C. 2010. Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*, 586–594.

Kök, A. G.; Fisher, M. L.; and Vaidyanathan, R. 2008. Assortment planning: Review of literature and industry practice. In *Retail supply chain management*. Springer. 99–153.

Kveton, B.; Szepesvari, C.; Wen, Z.; and Ashkan, A. 2015. Cascading bandits: Learning to rank in the cascade model. In *International Conference on Machine Learning*, 767–776.

Li, L.; Chu, W.; Langford, J.; and Schapire, R. E. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, 661–670. ACM.

Li, L.; Chu, W.; Langford, J.; Moon, T.; and Wang, X. 2012. An unbiased offline evaluation of contextual bandit algorithms with generalized linear models. In *Proceedings of the Workshop on On-line Trading of Exploration and Exploitation 2*, 19–36.

Luce, R. D. 2012. *Individual choice behavior: A theoretical analysis*. Courier Corporation.

Radlinski, F.; Kleinberg, R.; and Joachims, T. 2008. Learning diverse rankings with multi-armed bandits. In *Proceedings of the 25th international conference on Machine learning*, 784–791. ACM.

Robbins, H. 1985. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*. Springer. 169–177.

Rusmevichientong, P., and Tsitsiklis, J. N. 2010. Linearly parameterized bandits. *Mathematics of Operations Research* 35(2):395–411.

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

Schmit, S., and Johari, R. 2018. Learning with abandonment. In *International Conference on Machine Learning*, 4516–4524.

Slivkins, A.; Radlinski, F.; and Gollapudi, S. 2013. Ranked bandits in metric spaces: learning diverse rankings over large document collections. *Journal of Machine Learning Research* 14(Feb):399–436.

Sutton, R. S.; Barto, A. G.; et al. 1998. *Reinforcement learning: An introduction*. MIT press.

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

Train, K. E. 2009. *Discrete choice methods with simulation*. Cambridge university press.