

When to Target Customers? Retention Management using Constrained Dynamic Off-Policy Policy Learning

Ryuya Ko

The University of Texas at Austin

Kosuke Uetake

kosuke.uetake@yale.edu

Yale University

Kohei Yata

University of Wisconsin–Madison

Ryosuke Okada

Research Article

Keywords: Customer retention, dynamic treatment regimes, policy learning, personalized targeting, budget constraints, off-policy learning

Posted Date: October 7th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10807536/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

Additional Declarations: No competing interests reported.

When to Target Customers? Retention Management using Constrained Dynamic Off-Policy Policy Learning

Anonymous Authors

Abstract

We propose a method to learn personalized customer retention management strategies when customers' intentions to purchase evolve over time. Working with a Japanese online platform, we first implement a large-scale randomized experiment, in which coupons are sent to first-time buyers at different times. The experimental data allow us to estimate personalized dynamic retention policies using off-policy policy learning methods. We extend the existing methods by allowing intertemporal budget constraints and feasibility constraints. Our offline evaluation results based on the test sample show that the optimal dynamic policy is more cost-effective than baseline policies. Finally, we test the optimal policy online to confirm its performance.

JEL: C41, C44, C61, C93

Keywords: Customer retention; dynamic treatment regimes; policy learning; personalized targeting; budget constraints; off-policy learning

1 Introduction

Managing customer retention is a central part of customer relationship management (CRM). In particular, it is well known in both academia and practice that the attrition rate at the early stage of the customer life cycle is quite high [e.g., Fader and Hardie, 2007, 2010, Kim, 2022], but attrition tends to decrease as customers repeat purchases over time. Hence, it is essential to increase the retention of first-time buyers to increase the overall customer lifetime value.¹ Common strategies to retain first-time buyers include sending special thank-you messages and providing coupons to encourage another purchase.²

¹Throughout the paper, we use the term “retention” instead of “attrition” even though our application is not a contractual setting such as subscription businesses. Our method can be applied to both contractual and non-contractual settings.

²There are a lot of blog posts and articles online on how to send appreciation emails to first-time buyers. For example, see <https://www.drip.com/blog/customer-appreciation-emails>.

The recent advancement in the availability of Big Data and machine learning techniques has enabled companies to develop sophisticated data-driven CRM strategies. In particular, companies can now design personalized targeting policies such as sending messages or coupons. At the same time, improving those policies entails several challenges. The first challenge is that retention management inherently involves dynamics in customers’ behavior and interests. For example, it is well known that the retention probability tends to decline as the length of time since the customer’s first-time purchase increases, as known as the “recency trap” [Neslin et al., 2013]. It may be too early to provide retention incentives right after the first purchase, or it can be too late to do so a month after the purchase. Thus, timing matters. Hence, it is crucial to incorporate dynamics in policy learning for retention management. Existing retention management strategies, however, mainly focus on static settings where a company sees customers at one point in time and decides how to treat them right away.

The second challenge is that many marketing campaigns have a budget ceiling as they do not want to waste their resource. A common rule of thumb is that the marketing budget is 2–5% of their revenue for business-to-business companies and 5–10% for business-to-customer companies.³ Hence, marketing managers need to efficiently select marketing strategies within the budget. When the resource is limited such that only a subset of customers can receive retention incentives, identifying a cost-effective personalized targeting strategy is even more important. Even absent explicit budget constraints, companies may have other types of constraints or internal rules, which restrict what kind of policies they can implement in practice. We refer to these constraints as feasibility constraints.

In this paper, we aim to develop a framework that maximizes the retention rate of first-time buyers in a cost-efficient way. Our framework is to learn the optimal dynamic retention strategy, which decides actions (e.g. coupons) over time in response to the customer’s evolving states, given a budget constraint and feasibility constraints. After discussing the related literature in Section 2 and the background of our project in Section 3, the paper is organized as follows. In Section 4, we first propose a general framework of constrained dynamic personalization and discuss the necessary data variations to learn the optimal policy. Our method follows the recent literature on the estimation of optimal dynamic treatment regimes (DTRs) in statistics. DTRs are sequential decision rules that adapt over time to the changing status of each customer. A DTR, for example, determines whether or not to offer a coupon as a function of state variables such as the customer’s past purchase history, past browsing history, and past responses to emails.

³See, e.g., <https://www.bdc.ca/en/articles-tools/marketing-sales-export/marketing/what-average-marketing-budget-for-small-business>.

Learning optimal DTRs is challenging since the treatment assignment today affects not only the current customer behavior but also future treatment assignments and outcomes. We extend the existing method to learn optimal DTRs by explicitly incorporating feasibility and budget constraints. This further complicates the dynamic optimization problem, because the current treatment assignment affects the future assignments and outcomes not only through customer state dynamics but also through inter-temporal constraints. Incorporating both dynamics and constraints, however, is crucial in developing practical and cost-effective targeting policies.

Following the general framework of DTRs, in Section 5, we discuss the experimental design to generate the data necessary for estimating dynamic policies. To identify the optimal policy, the experiment needs to satisfy a condition that the data-generating policy chooses every *feasible* action with a positive probability, where the set of feasible actions may depend on the state variables and past actions. In our case, we use experimental data from a large e-commerce platform in Japan. The company imposes a feasibility constraint that each user receives at most one coupon during the campaign. Hence, our experiments randomize actions at each stage given this feasibility constraint. Our experimental design allows us to learn dynamic policies under the feasibility constraint as well as an additional inter-temporal budget constraint.

We use two experimental data sets from the platform. In the first experiment, there are two actions over three periods, i.e., a retention email and a retention email with a \$10 coupon. Two actions are randomized at each period (2 days, 10 days, and 30 days after the first transaction) under the company’s feasibility constraint that each consumer receives at most one coupon during the period. In the second experiment, there are three actions over three periods, i.e., no email, a retention email, and a retention email with a \$10 coupon. Similar to the first experiment, three actions are randomized at each period under the constraint that each user receives at most one coupon. The purpose of the second experiment is to allow us to separately estimate the effects of a retention email and a coupon. Moreover, it is possible to learn richer dynamic policies with the second experiment data as there is no feasibility constraint for sending retention emails.

In Section 6, we provide details of learning optimal dynamic personalized policies using the data discussed in Section 5. We first provide the characterization of optimal DTRs under budget and feasibility constraints, which transforms the constrained problem into a sequence of unconstrained problems. We then propose a learning algorithm that allows us to use existing approaches to learning unconstrained optimal DTRs as subroutines. There are mainly two approaches to unconstrained problems. The first approach is the indirect approach, which first estimates the value function and then maxi-

mizes it to derive the optimal dynamic policy. In particular, we use Q -learning [e.g., Murphy, 2005b]. Q -learning is a data-driven dynamic programming procedure that estimates an optimal DTR by maximizing the conditional expectation of the cumulative sum of the current and future payoffs given the current state and action, known as the Q -function. We estimate the Q -function by various machine learning methods such as LASSO to avoid fully parameterizing underlying data-generating processes and to handle a large number of state variables. The second approach is the direct approach, which does not require estimating the Q -function. In particular, we use Backward Outcome Weighted Learning (BOWL) [e.g., Zhao et al., 2015]. The BOWL approach reframes the estimation of an optimal DTR as a sequential weighted classification problem, starting from the very end period. This reformulation is helpful as one can use existing off-the-shelf classification algorithms.

In Section 7, we show the offline evaluation results. To avoid overfitting, we randomly split our experimental data into the training set (70%) and the test set (30%), apply our method to the training set to learn optimal DTRs, and then use the test set to evaluate their performance. We first find that, when there is no budget constraint, the optimal DTR based on BOWL achieves a slightly higher retention rate than the static optimal policies. More importantly, our *budget-constrained* optimal DTRs are much more cost-effective than alternative policies. We find that our budget-constrained dynamic policy based on BOWL substantially improves cost efficiency. Measured by the increase in retention probability per yen of incentive spending, the policy outperforms all benchmark policies while maintaining large gains in customer retention. Moreover, our results reveal customers’ heterogeneous responses to the timing of incentives. Specifically, it is not always optimal to send incentives right after the first purchase; for some customers, it may be more beneficial to send incentives later.

Lastly, in Section 8, we present the online evaluation results. Following the promising offline evaluation results, the company implemented our optimal DTRs in a large-scale online experiment. The online results confirm the offline findings: the budget-constrained policy based on BOWL achieves the highest retention uplift per unit of incentive cost among the competing policies. Based on these results, the company adopted our algorithm for its retention management.

Our contributions are the following. First, we propose an empirical framework to create dynamic personalized targeting strategies for retention management when there exist budget constraints (or other constraints that limit what fraction of the customers can be treated). As Ascarza et al. [2021] point out, many companies do not effectively use their customer data to achieve their goals within their budget. Our framework can address the concern. Second, we apply the methodology to experimental data from a leading Japanese e-commerce company and demonstrate that our approach generates more

cost-effective retention policies than existing benchmark strategies. In both offline and online evaluations, our budget-constrained dynamic policy achieves the largest retention uplift per unit of incentive cost. The company estimated that our algorithm generated more than \$10 million in additional value over one year relative to its status quo strategy, demonstrating the practical relevance of our approach.

2 Related Literature

Our paper is related to the marketing literature on proactive churn/attrition management.⁴ Churn management is one of the key priorities for most businesses as customer retention is a major component of customer lifetime value (CLV) and hence a cornerstone of successful CRM [Ascarza et al., 2018, 2021]. As summarized by Neslin et al. [2013], a popular industry practice for data-driven retention management is to flag risky customers who are likely to churn using behavioral and demographic variables. By proactively predicting customer churn, firms can try to convince them to stay [e.g., Neslin et al., 2006]. Recently, a few papers have gone beyond this practice. Ascarza [2018] points out that it is not effective to target customers with higher predicted retention probabilities as they may not be responsive to marketing interventions and propose to determine targeting based on uplift. Also, Lemmens and Gupta [2020] note that it is crucial to take the financial impact of a retention intervention based on CLV into account. Gopalakrishnan and Park [2024] show that it is not always optimal to send coupons for retention right after purchases by running a randomized experiment. Our approach adds to the literature by developing a method that estimates a (counterfactual) dynamic targeting policy that maximizes retention given budget constraints.⁵

In marketing, a growing number of papers propose methods for learning optimal personalized policies.⁶ Hitsch et al. [2024] propose a policy learning method based on the estimation of the conditional average treatment effect (CATE) using k-nearest neighbors. Simester et al. [2020] consider an efficient policy evaluation method when existing policies and new policies are compared. Yoganarasimhan et al.

⁴Although “churn” is used in the context of contractual settings such as subscription business and “attrition” is used in non-contractual settings, we use those words almost interchangeably as our method can be used in both cases.

⁵For a review of the literature, see, e.g., Ascarza et al. [2018].

⁶In economics, there is a strand of papers on policy learning. Athey and Wager [2021], for example, develop methods for policy learning with observational data. Their method can be used to optimize various types of treatment allocation such as binary treatments and infinitesimal changes in continuous treatments. Kitagawa and Tetenov [2018] study policy learning in a nonparametric setting and obtain regret bounds for the Empirical Welfare Maximization (EWM) method. Bhattacharya and Dupas [2012], Luedtke and van der Laan [2016], Qiu et al. [2022], and Sun [2026] study how to incorporate a certain policy constraint in a static setting of policy learning, not in a dynamic setting as ours. Sakaguchi [2025] extends the EWM method to a dynamic setting with inter-temporal constraints. Our method is different from Sakaguchi [2025] in that our approach transforms the constrained problem into a sequence of unconstrained problems, which allows us to use the existing approaches such as Q -learning and BOWL. Moreover, our method is computationally light and can accommodate a large number of state variables.

[2023] estimate the CATE with different machine learning models and compare the performance of targeting policies constructed based on these models. Yang et al. [2023] consider how to derive targeting policies when an outcome of interest is observed only in the long term by applying the idea of statistical surrogacy [Athey et al., 2026] and optimal policy learning. Daljord et al. [2023] study the optimal design of compliance promotions and develop an experimental framework for optimizing promotional policies when customer participation is endogenous.

Recently, a few papers in marketing explicitly examine dynamic personalized policies. The most closely related paper is Liu [2023]. Liu [2023] develops a dynamic reinforcement learning algorithm for dynamic pricing in e-commerce and finds that the dynamic reference price effect plays an important role. Also, Wang et al. [2023] propose a sequential targeting strategy using deep reinforcement learning and apply it to the experimental data from a mobile app promotion campaign. In a different context, Kar et al. [2015] and Raffeian [2023] examine dynamic ad targeting and show the importance of the inter-temporal externalities. We add to this brand-new literature by proposing a method to estimate dynamic cost-effective policies, which explicitly take constraints into account. Moreover, we differ from those papers as we consider a non-Markov dynamic setup.⁷ Finally, our paper investigates a substantively different topic on retention management.

Methodologically, our paper relates to an evolving literature of reinforcement learning under constraints. Achiam et al. [2017] is a seminal paper in the literature and develop the constrained policy optimization approach (CPO). The paper proposes an algorithm to solve the optimization problem based on trust region optimization. The method by Achiam et al. [2017] considers the online policy learning where the new policy is learned under the new data governed by the current policy. We consider the offline learning situation. Another relevant paper in the literature is Le et al. [2019], which examines the offline constrained policy learning. Although they propose a conceptually similar algorithm, their method mainly focuses on stationary Markov cases and hence it is not directly applicable in our case. Also, their algorithm learns policies by minimizing the estimated duality gap of the constrained optimization problem, while our algorithm learns policies by directly maximizing the cross-validation performance.

⁷In the literature of personalized medicine, DTRs are developed in non-Markov settings to adaptively select clinical treatments in response to the factors emerging over time [e.g., Murphy, 2003, Murphy et al., 2007, Zhao et al., 2009, Zhang et al., 2013, to name a few]. Another related paper is Nie et al. [2021], which study when to start treatment and learn the optimal policy.

3 Background

The company we work with is one of the largest e-commerce platforms in Japan (about \$2,300 million transaction volume in 2021) that mainly sells apparel products for young female customers. There are more than 1,500 retailers on the platform and more than 8 million active users. The users buy products from the retailers through the platform’s website or the mobile app. Those apparel brands and retailers delegate marketing activities to the platform company that manages inventories at the company’s warehouse and handles shipping directly to consumers.

The company’s retention strategy primarily focuses on enhancing the retention of first-time purchasers instead of their overall lifetime value (LTV). There are a few reasons for doing so. First, the company’s analysis indicates that the impact of a second purchase on LTV is significantly higher than subsequent purchases. As measuring LTV directly is time-consuming, using a second purchase as a proxy can yield an effective and time-efficient policy [Yang et al., 2023]. Second, the retention rates for first-time buyers are substantially lower than those for repeat customers. Hence, incentivizing a second purchase among these customers could improve both retention rates and overall LTVs.

The company’s original retention strategy for first-time buyers is, as suggested by many popular marketing strategy books, to send “thank you” messages to first-time buyers to show appreciation and to provide them with some information such as various rankings of items and the company’s app. They are neither personalized nor dynamically optimized. The company sends an email three times, 2 days, 10 days, and 30 days after the first purchase.⁸ These emails aim to encourage second purchases.

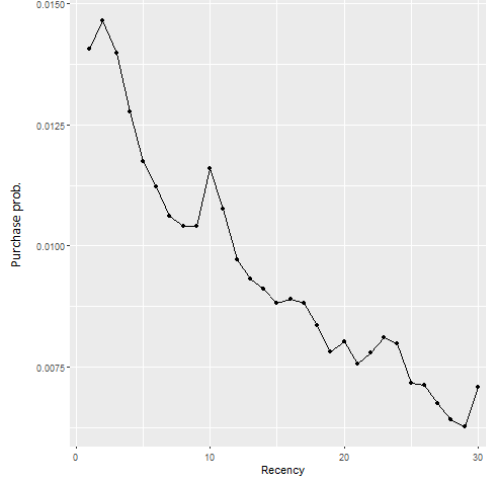
Although the company has seen positive impacts of the current appreciation emails on retention, the company would like to increase the retention rate by providing financial incentives and personalization. As to providing incentives, the timing is crucial. Neslin et al. [2013] highlight a well-known concept of the recency trap: the retention probability diminishes as the time since the last purchase increases. Figure 1 shows the relationship between the days since the last purchase and the average purchase probability in our application, suggesting a similar declining pattern.⁹

This observed pattern underscores the importance of timing for targeting customers. Delaying incentives might reduce their effectiveness due to diminished purchasing intent. Conversely, sending incentives immediately after the first purchase might be unnecessary, as they may purchase for the

⁸Observe that the customer typically receives their product a few days after checking out. Hence, the company sends the second email about a week after the package delivery, and the third one about four weeks after the delivery.

⁹Previous papers consider the recency trap in longer terms than ours. Although we are not allowed to disclose the average purchase intervals, customers in the platform we study buy more frequently than other prior studies.

Figure 1: Recency and Retention Probability



Notes: The graph shows the time (days) since the first purchase on the horizontal axis and purchase probabilities, i.e., retention probabilities on the vertical axis. The purchase probabilities are calculated from the experimental data we use for the estimation of optimal dynamic policies. The company sends incentives 2 days, 10 days, and 30 days after the customer's first purchase, giving hikes around those days.

second time even without such incentives. This is also of practical importance for the company, who is concerned with its budget constraint. While the company's general objective is to maximize the retention rate, the company hesitates to distribute too generous incentives. Those considerations led us to design a cost-effective, personalized retention strategy.

4 Framework for Dynamic Policy Personalization with Constraints

In this section, we introduce a general model and problem setup for dynamic policy personalization with constraints. We also discuss the requirements for data collection to learn optimal dynamic policies. Our proposed learning method is presented later in Section 6.

4.1 Non-Markov Non-Stationary Dynamic Environment

We consider the following dynamic environment. Time is discrete and finite, denoted by $t = 1, 2, \dots, T$. All the variables defined below are user-level. The company observes the state variables $X_t \in \mathcal{X}_t$ for each user in period t . The state variables include the user demographic information, past purchase information, browsing information, responses to past marketing activities, etc. Our methodology can easily accommodate a large number of state variables. In our application, we use more than 100 variables for X_t , which evolve over time, and can be affected by the company's action. The company can

choose an action $A_t \in \mathcal{A}_t$ for each user every period, where $\mathcal{A}_t$ is the action set. Our application considers two settings. In the first setting, the company either sends an incentive or not in addition to the appreciation message, so $\mathcal{A}_t = \{0, 1\}$, where option 1 is sending the incentive. In the second setting, the company has an additional option of not sending the appreciation message. The outcome and cost of interest are $Y \in \mathbb{R}$ and $C \in \mathbb{R}$, which will be realized after period T . Note that it is straightforward to apply our method to the case where the outcome and cost are the discounted cumulative values, e.g., $\sum_t \beta^t Y_t$ and $\sum_t \beta^t C_t$, where Y_t and C_t are the period- t outcome and cost and β is the discount factor. Also, intermediate outcomes (such as Y_t) can be included in the state vector X_t , even if the primary outcome of interest is their terminal value. In our application, we consider customer retention and the total coupon amount used within 2 months after the first purchase as the final outcome and cost, respectively, while including retention and coupon usage at intermediate periods as state variables.

We introduce the history $H_t \in \mathcal{H}_t$ as a summary of the user-level events up to period t . More precisely, we define $H_1 = X_1$ and $H_t = (H_{t-1}, A_{t-1}, X_t)$ for all $t > 1$. Note that the history includes not only the state variables but also the actions taken up to that period. The initial state distribution is denoted by $P_{X_1}(x_1)$, and the state transition distribution from periods $t-1$ to t is denoted by $P_{X_t}(x_t|h_{t-1}, a_{t-1})$. The final outcome Y and cost C are then determined by the entire history up to period T , h_T , and the action taken in period T , a_T , following the distribution $P_{Y,C}(y, c|h_T, a_T)$.¹⁰ Note that the model is non-Markov, i.e., the state transition depends not only on the previous state but also on the entire history up to that period. Also, the environment is non-stationary, i.e., the state space and its transition distribution change over time. Although this model can be viewed as Markov if we treat history as the state, the model is still a *non-stationary* Markov environment. Hence, we may not be able to use off-the-shelf batch (offline) reinforcement learning algorithms that typically assume a *stationary* Markov environment.¹¹

With this dynamic environment, a *dynamic treatment regime* (DTR) is a sequence of decision rules $\mathbf{d} = (d_1, \dots, d_T)$, where $d_t : \mathcal{H}_t \rightarrow \Delta(\mathcal{A}_t)$ is a function that maps the history up to time t into a probability distribution over actions. We use $d_t(a_t|h_t)$ to denote the probability of choosing action a_t given history h_t . If the decision rule is deterministic and chooses an action with probability one, then we use $d_t(h_t)$

¹⁰We consider estimating optimal policies without explicitly modeling the consumer's dynamic decision problem. This is a standard approach used in the literature of DTR and off-policy policy learning. Our approach is agnostic about how consumers respond to actions or how consumers form their beliefs, as those are embedded into the state transition probabilities and reward probabilities, i.e., $P_{X_t}(x_t|h_{t-1}, a_{t-1})$ and $P_{Y,C}(y, c|h_T, a_T)$, which we can directly estimate from the data.

¹¹In principle, it is possible to reformulate a non-stationary Markov decision problem into a stationary Markov decision problem by concatenating the state spaces in all periods ($t = 1, \dots, T$) and treating it as the time-invariant state space. We discuss some of the issues of this approach in Appendix D.

to denote the action. In our application, a simple example of a rule d_t is to send an incentive if the user visits the platform's website after the previous period $t - 1$.

When a DTR $\mathbf{d}$ is applied to the above dynamic environment, the trajectory $H = (X_1, A_1, \dots, X_T, A_T, Y, C)$ is generated by the following process.

1. $X_1 \sim P_{X_1}, A_1 \sim d_1(\cdot | H_1)$.
2. $X_t \sim P_{X_t}(\cdot | H_{t-1}, A_{t-1})$ and $A_t \sim d_t(\cdot | H_t)$ for $t = 2, \dots, T$.
3. $(Y, C) \sim P_{Y,C}(\cdot | H_T, A_T)$.

That is, a DTR generates the action for each period, which has direct impacts on the final outcome Y and cost C and indirect impacts through the state variables. We denote the distribution of H under DTR $\mathbf{d}$ by $P_{\mathbf{d}}$ and the expectation with respect to $P_{\mathbf{d}}$ by $E_{\mathbf{d}}$.

4.2 Objective and Constraints

Suppose that the company aims to choose a DTR that maximizes the expected value of the outcome Y under constraints:

$$\mathbf{d}^* \in \arg \max_{\mathbf{d}} E_{\mathbf{d}}[Y] \text{ s.t. (BC) and (FC),}$$

where we consider two constraints (BC) and (FC), defined below. $E_{\mathbf{d}}[Y]$ can be interpreted as the average outcome in the target population when DTR $\mathbf{d}$ is used to determine actions for each user, assuming that the target population consists of an infinite number of users following the dynamic process described above.

1. Inter-temporal Budget Constraint (BC):

$$E_{\mathbf{d}}[C] \leq B, \tag{BC}$$

where $B > 0$ is a per-user budget. In our empirical context, C is the coupon amount that the customer uses within 2 months after the first purchase, $E_{\mathbf{d}}[C]$ is the average coupon usage per customer under DTR $\mathbf{d}$, and B is the budget ceiling over time specified by the company on how much can be spent per customer. The above formulation also allows for a constraint on the fraction or number of times the company takes a particular action. For example, if the company's action is either sending a coupon or not, denoted by $A_t \in \{0, 1\}$, and the company has a limited

number of coupons, we can write the constraint as $E_{\mathbf{d}}[\sum_{t=1}^T A_t] \leq B$, where B is the capacity on the average number of coupons per customer.

2. Feasibility Constraint (FC):

$$\{a_t \in \mathcal{A}_t : d_t(a_t | h_t) > 0\} \subset \mathcal{A}_t(h_t) \text{ for all } h_t \in \mathcal{H}_t \text{ and } t = 1, \dots, T, \quad (\text{FC})$$

where $\mathcal{A}_t(h_t) \subset \mathcal{A}_t$ is the set of feasible actions when the history up to period t is h_t . If the DTR $\mathbf{d}$ is deterministic, the constraint is equivalent to $d_t(h_t) \in \mathcal{A}_t(h_t)$ for all $h_t \in \mathcal{H}_t$ and $t = 1, \dots, T$. This constraint allows us to prohibit the DTR from choosing particular actions in particular histories by appropriately specifying the feasible set $\mathcal{A}_t(h_t)$. In our application, we consider the following feasibility constraints:

- (a) The DTR sends a coupon to each user at most once over time.
- (b) The DTR does not send a coupon if the user has already made a second purchase.

These constraints can be imposed by restricting the feasible action set depending on the past actions and user's behaviors.¹²

Dynamic and Static Policies In our application, a dynamic policy can update its decisions as new information about the user becomes available over time. Although DTRs satisfying feasibility constraint (a) above do not send a coupon to the same user multiple times, they are still dynamic because decisions in periods ($t > 1$) can depend on updated state variables. Specifically, in the first setting with two actions, a DTR decides whether to send a coupon in period 1 based on the initial state variables (X_1). If it does not send a coupon to some users in period 1, it decides whether to send one in period 2 based on the updated history ($H_2 = (X_1, A_1, X_2)$), including user behaviors such as search activities between periods 1 and 2. In the final period, it makes the same decision for users who have not yet received a coupon based on all information available up to that point, ($H_T = (X_1, A_1, \dots, X_{T-1}, A_{T-1}, X_T)$). A static

¹²Specifically, we can incorporate constraints (a) and (b) by letting $a_t \in \{0, 1\}$ denote whether to send a coupon and setting

$$\mathcal{A}_1(h_1) = \begin{cases} \{0\} & \text{if } \text{second}_1 = 1 \\ \{0, 1\} & \text{otherwise} \end{cases}$$

and for $t > 1$

$$\mathcal{A}_t(h_t) = \begin{cases} \{0\} & \text{if } a_1 + \dots + a_{t-1} = 1 \text{ or } \text{second}_t = 1 \\ \{0, 1\} & \text{otherwise,} \end{cases}$$

where second_t indicates whether the user has already made a second purchase before the beginning of period t and is assumed to be included in the state vector x_t .

policy, which we use as a benchmark, instead determines the entire action profile $((A_1, \dots, A_T))$ using only the initial state (X_1) .

The key feature of the dynamic policy is therefore that later decisions can incorporate information that becomes available after the initial purchase. For example, the firm can wait to observe additional signals about a customer's purchase behavior before deciding whether and when to send an incentive. Such updating may allow the firm to avoid sending incentives to customers who are likely to purchase without them and to allocate incentives later when they are expected to have a greater incremental effect. This flexibility may be particularly useful when the marketing budget is limited. Appendix A illustrates this mechanism in a simple model and provides conditions under which the use of information revealed over time improves policy performance, with potentially larger gains under a budget constraint. We return to the comparison with static policies in Section 6.3.

4.3 Data-generating Process

To learn optimal DTRs, suppose that we observe data $\{H^{(i)}\}_{i=1}^n = \{(X_1^{(i)}, A_1^{(i)}, \dots, X_T^{(i)}, A_T^{(i)}, Y^{(i)}, C^{(i)})\}_{i=1}^n$ of n individuals, where trajectories $H^{(1)}, \dots, H^{(n)}$ are independently and identically generated by the process described in Section 4.1 with some common DTR $\mathbf{d}^0$. If the data are generated from an experiment, as in our empirical setting, $\mathbf{d}^0$ is the known random assignment policy used by the experiment.¹³ We denote the distribution of the observed trajectory of user i , $H^{(i)}$, by P and the expectation with respect to P by E , both of which are unknown.

We assume the following *overlap condition*. Recall that $\mathcal{A}_t(h_t)$ is the set of feasible actions when the history up to period t is h_t .

Assumption 1. For any $t = 1, 2, \dots, T$ and $(a_t, h_t) \in \mathcal{A}_t(h_t) \times \mathcal{H}_t$, $d_t^0(a_t|h_t) > 0$.

This assumption states that the data-generating DTR $\mathbf{d}^0$ chooses every *feasible* action $a_t \in \mathcal{A}_t(h_t)$ with a positive probability in every history $h_t \in \mathcal{H}_t$. As a result, if we hypothetically observe an infinite amount of data generated by DTR $\mathbf{d}^0$, we observe every trajectory H that can be realized under any feasible DTR. This assumption is standard in the literature of DTR and is satisfied when the data are collected in a sequential multiple assignment randomized trial (SMART) [Murphy, 2005a]. In our application, we design an experiment that randomizes feasible actions so that Assumption 1 holds, as detailed in Section 5.

Under Assumption 1, the average outcome $E_{\mathbf{d}}[Y]$ and average cost $E_{\mathbf{d}}[C]$ are identified (i.e.

¹³If we use observational data, our approach requires the estimation of the data-generating policy.

uniquely determined) for every feasible DTR $\mathbf{d}$ satisfying (FC), from the distribution of the observed trajectory. For example, we can write

$$E_{\mathbf{d}}[Y] = E \left[Y \prod_{t=1}^T \frac{d_t(A_t|H_t)}{d_t^0(A_t|H_t)} \right] \text{ and } E_{\mathbf{d}}[C] = E \left[C \prod_{t=1}^T \frac{d_t(A_t|H_t)}{d_t^0(A_t|H_t)} \right],$$

where the expectations on the right-hand sides are taken with respect to the distribution of $H = (X_1, A_1, \dots, X_T, A_T, Y, C)$ under $\mathbf{d}^0$.¹⁴ Note that the denominator $d_t^0(A_t|H_t)$ can never be 0 by construction, since A_t is the observed action assigned by the rule d_t^0 . This way of adjusting distributions is called the inverse probability weighting or importance sampling technique [Precup et al., 2000].

Our learning objective is to use the data $\{H^{(i)}\}_{i=1}^n$ to choose a DTR that maximizes $E_{\mathbf{d}}[Y]$ subject to the budget and feasibility constraints. In principle, we can solve the constrained optimization problem for an optimal DTR by estimating $E_{\mathbf{d}}[Y]$ and $E_{\mathbf{d}}[C]$ for all feasible DTRs. However, such a brute-force approach is computationally infeasible, especially if there are a large number of state variables. In Section 6, we show that we only need to search through a certain class of feasible DTRs for an optimal one and develop a computationally feasible learning algorithm.

5 Experimental Designs and Data

5.1 Experimental Designs

This section describes the designs of the randomized experiments that we will use for estimating the optimal policies. The company conducted two experiments: one with two actions and another with an additional action. The data from each experiment are used to estimate the optimal policy for each setting.

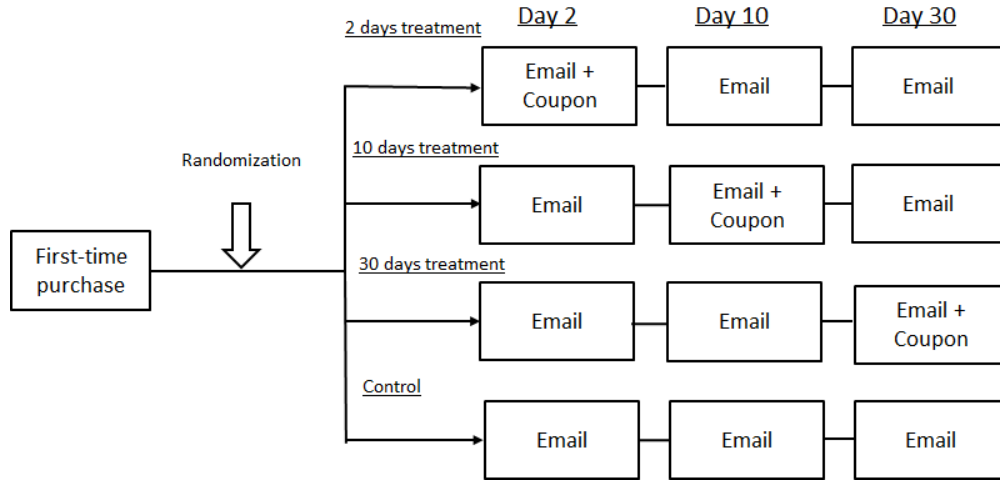
First Experiment. The company conducted the first experiment from September to December 2020. As explained in Section 3, the company focuses on customers who have just made their first purchase.

¹⁴To build intuition for this result, suppose, for simplicity, that H is discrete under $\mathbf{d}^0$. Then, the first equation can be derived as follows:

$$\begin{aligned} E \left[Y \prod_{t=1}^T \frac{d_t(A_t|H_t)}{d_t^0(A_t|H_t)} \right] &= \sum_{(x_1, a_1, \dots, x_T, a_T, y)} y \prod_{t=1}^T \frac{d_t(a_t|h_t)}{d_t^0(a_t|h_t)} P_{X_1}(x_1) d_1^0(a_1|x_1) P_{X_2}(x_2|x_1, a_1) \cdots d_T^0(a_T|h_T) P_Y(y|h_T, a_T) \\ &= \sum_{(x_1, a_1, \dots, x_T, a_T, y)} y P_{X_1}(x_1) d_1(a_1|x_1) P_{X_2}(x_2|x_1, a_1) \cdots d_T(a_T|h_T) P_Y(y|h_T, a_T) = E_{\mathbf{d}}[Y], \end{aligned}$$

where the sum is taken over all trajectories $(x_1, a_1, \dots, x_T, a_T, y)$ that occur with positive probability under $\mathbf{d}^0$. By the overlap condition, this set of trajectories includes every trajectory that occurs with positive probability under $\mathbf{d}$, and hence the final equality holds.

Figure 2: Experimental Design (First Experiment)



There are about 150,000–200,000 first-time buyers per month during the experiment period.¹⁵ We randomly choose about 70% of those first-time buyers for the experiment.

In the first experiment, customers in the control group receive only appreciation emails, each of which contains information about how to use the platform or a generic ranking of the items sold on the platform. They receive appreciation emails at three different times: 2 days, 10 days, and 30 days after the first purchase. The customers in the treatment group receive the financial incentive of 1,000 points (about \$10) along with the appreciation emails.¹⁶ Since the platform had never provided coupons for first-time buyers, the customers did not know if they could receive coupons until they made a purchase.

In the treatment group, there are three subgroups depending on when the customers receive the incentive, as shown in Figure 2. The customers in the 2-day treatment group receive the incentive 2 days after the first purchase, and they receive only the appreciation emails 10 days and 30 days after the first purchase. The 10-day treatment and the 30-day treatment groups are similarly defined. Hence, each first-time buyer in the treatment groups receives the incentive at most once during the experimental period. The company imposes this constraint for two reasons. First, the company does not want to provide too many coupons. Second, the company's main interest is to examine the timing of sending a coupon rather than the frequency of sending coupons.

The randomization is done at the user level. We randomly assign each user right after their first

¹⁵The non-disclosure agreement (NDA) does not allow us to report the exact number of customers involved in the experiment.

¹⁶The company was not willing to offer %-off coupons because such a coupon can be very costly for the platform if users purchase expensive items. The design of coupons is beyond our research question. Also, there is no expiration date for those points.

purchase to one of the four groups: the control (39.76%), the 2-day treatment (19.15%), the 10-day treatment (20.5%), or the 30-day treatment (20.59%).

Note that the experiment randomly divides the customers into four groups who receive different action profiles $(A_1, A_2, A_3) \in \{0, 1\}^3$ such that $A_1 + A_2 + A_3 \leq 1$ prior to the first treatment, where A_t indicates whether to send a coupon at time t (i.e., day 2, day 10, or day 30). Implementing this randomization design is equivalent to implementing a sequential randomization design that randomly determines each customer's treatment prior to each period based on the past treatment profile. More specifically, our design is equivalent to the following sequential design:

- At $t = 1$ (day 2), the action A_1 (whether to send a coupon) is randomized.
- At $t = 2$ (day 10), A_2 is randomly assigned conditional on A_1 , where $\Pr(A_2 = 1|A_1 = 1) = 0$, while $\Pr(A_2 = 1|A_1 = 0) \in (0, 1)$.
- At $t = 3$ (day 30), A_3 is randomly assigned conditional on (A_1, A_2) , where $\Pr(A_3 = 1|A_1 + A_2 = 1) = 0$, while $\Pr(A_3 = 1|A_1 + A_2 = 0) \in (0, 1)$.¹⁷

Our experiment can be used to estimate optimal *dynamic* policies within the class of policies that send a coupon to each user at most once over time ($A_1 + A_2 + A_3 \leq 1$). Specifically, the experiment involves randomization in A_t at each period conditional on the history H_t , including the past treatment profile. While there is no randomization for the users who have already received a coupon, we do have randomization for the users who have not done so yet, which allows us to optimize the actions for the latter group.¹⁸

When learning optimal policies, we also impose an additional feasibility constraint that they do not send a coupon if the user has already made a second purchase, as mentioned in Section 4.2.¹⁹ It is also straightforward to see that the data generated by the experimental policy satisfies Assumption 1. Thus, we can estimate a constrained optimal dynamic policy with the data generated by the experiment.

¹⁷The reason for this equivalence is as follows. Clearly, randomizing (A_1, A_2, A_3) among the four profiles $(0, 0, 0)$, $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$ at $t = 1$ is equivalent to first randomizing A_1 only, then randomizing A_2 for the users with $A_1 = 0$, and lastly randomizing A_3 for those with $A_1 = A_2 = 0$, all done at $t = 1$. An important point here is that the assignment of A_2 for those with $A_1 = 0$ does not influence the evolution of their state from $t = 1$ to $t = 2$, since the action A_2 does not take effect until $t = 2$. This means that randomizing A_2 for those with $A_1 = 0$ at $t = 1$ is statistically equivalent to doing so at $t = 2$. For the same reason, randomizing A_3 for those with $A_1 = A_2 = 0$ at $t = 1$ is statistically equivalent to doing so at $t = 3$.

¹⁸For example, we can estimate an optimal decision rule for period 3 by using the subsample such that $A_1 = A_2 = 0$, to which A_3 is randomly assigned. Note that when estimating an optimal decision rule for period 3 under the feasibility constraint, we do not need to observe treatment profiles like $(1, 0, 1)$, $(1, 0, 0)$, $(0, 1, 1)$, $(0, 1, 0)$, $(1, 1, 1)$, and $(1, 1, 0)$, since we do not need to optimize the actions for the users who have already received a coupon ($A_1 + A_2 = 1$); the only option for them is to send an email without a coupon.

¹⁹This constraint is imposed to save the cost of coupons.

Table 1: Summary Statistics (First Experiment)

Variable	2 day		10 day		30 day		Control	
	mean	sd	mean	sd	mean	sd	mean	sd
Female	0.631	0.483	0.631	0.483	0.629	0.483	0.634	0.482
Age	30.24	12.69	30.31	12.76	30.23	12.68	30.25	12.67
Quantity: first buy	1.699	1.520	1.697	1.482	1.691	1.588	1.697	1.493
Sales: first buy (JPY)	8065.3	8213.6	8127.3	8405.5	8102.7	8361.3	8089.4	8157.9
Points used: first buy	557.5	756.9	556.5	760.2	556.5	762.0	560.3	860.9
# of sessions/day (pre 1st buy)	0.697	2.016	0.698	2.032	0.703	2.049	0.704	2.047
# of PVs/day (pre delivery)	15.08	31.78	15.38	33.02	15.24	32.73	15.23	32.64
# of favorites/day (days 2–10)	0.207	1.036	0.178	1.119	0.179	1.047	0.180	1.018
# of messages sent (days 10–30)	32.21	37.81	32.31	37.16	31.60	37.28	31.61	37.29

Note: The first six columns report the mean and standard deviation of each variable for each of the three treatment groups. The last two columns are for the control group. The number of observations in each treatment is not reported due to the non-disclosure agreement (NDA).

Table 1 reports the summary statistics of a subset of the variables we use.²⁰ In the first two columns, we show the mean and standard deviation of each variable for the customers in the 2-day treatment. Similarly, the third and fourth columns are for the customers in the 10-day treatment, the fifth and sixth columns for the customers in the 30-day treatment, and the seventh and eighth columns for the customers in the control group.

For user demographics, about 63% of users are female and the average age of users is 30. The number of items purchased (quantity) and total spending (sales) on the user’s first purchase occasion are on average 1.7 items and 8,100 JPY, respectively, across conditions. Also, users apply about 550 points for the first purchase across all conditions.²¹

The last four rows contain a subset of the user behavior variables. Since we use more than 100 behavioral variables based on the user access and behavior data (sessions, page views (PVs), favorites, messages sent, etc.) before and after the first purchase, it is not possible to report the summary statistics of all of those variables. Hence, we *randomly* choose four variables and show their summary statistics in the table.²²

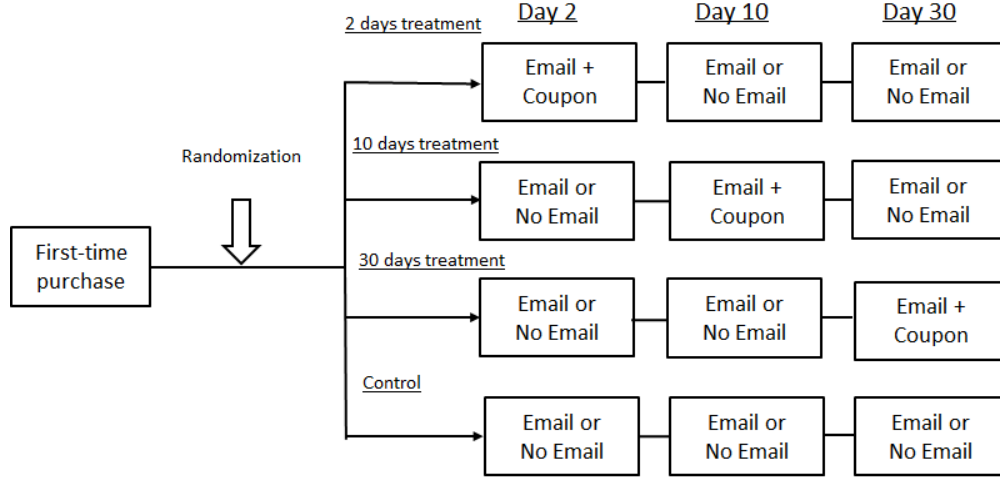
Second Experiment. The company conducted another experiment between June and August 2021 aiming to optimize whether to send an appreciation email as well as when to send a coupon. In the experiment, first-time buyers are randomly assigned to one of the four groups: the control group, who receive no coupon, and the three treatment groups, who receive a coupon 2 days, 10 days, or 30 days

²⁰We show the results of the balance check in Appendix B.

²¹Most users receive 500 to 2000 points when they sign up for the platform. They can use them immediately, especially for their first purchase.

²²Note that we do not mean those reported variables are more important than other variables. We simply choose those variables at random. This is required by the company for confidentiality issues.

Figure 3: Experimental Design (Second Experiment)



after the first purchase. Moreover, when users do not receive a coupon, an email is randomly sent. Hence, there are 20 subgroups (combinations of the assigned treatments at the three timings) in total, where there are about 5% of users for each subgroup. Figure 3 visualizes the design.

Similarly to the first experiment, this experimental design is equivalent to randomly assigning three actions (email with a coupon, email only, and no email) in each of the three time periods, under the feasibility constraint that the “email with a coupon” action can be taken at most once. It is straightforward to see that the data generated by this experiment satisfy Assumption 1 to estimate the optimal DTR of sending emails and coupons given the feasibility constraint. Moreover, note that one can estimate richer dynamic policies with the data from the second experiment as there is no feasibility constraint for sending emails (without coupons). That is, the optimal policy may not always send an email to a customer, depending on the evolution of the customer’s state variables.

Table 2 reports the summary statistics of a subset of the variables we use, computed from the data from the second experiment. To save space, we do not report the summary statistics for all possible treatment conditions, and we report the summary statistics only for the three treatment groups based on the timing the coupons are sent. For example, the day-2 group includes the users who receive a coupon on day 2 regardless of the treatments on days 10 and 30. We also report the summary statistics for the control group, which consists of the users who do not receive an incentive or an email on any of the three days. For both demographic and behavioral variables, we do not find any significant differences across conditions.

Table 2: Summary Statistics (Second Experiment)

Variable	2 day		10 day		30 day		Control (no email)	
	mean	sd	mean	sd	mean	sd	mean	sd
Female	0.632	0.482	0.631	0.483	0.628	0.483	0.632	0.482
Age	32.18	18.23	32.21	18.25	32.05	18.21	31.97	17.98
Quantity: first buy	1.013	0.179	1.013	0.172	1.014	0.221	1.015	0.252
Sales: first buy (JPY)	4057	4187	4090	4298	4063	4082	4070	4090
# of sessions/day (pre 1st buy)	0.777	2.108	0.769	2.067	0.779	2.129	0.759	2.032
# of PVs/day (pre delivery)	12.81	28.34	12.75	29.30	12.96	29.38	12.60	27.27
# of favorites/day (2-10 day)	0.155	0.805	0.128	0.911	0.135	0.824	0.119	0.640
# of messages sent (10-30 day)	5.145	17.389	5.167	17.941	5.192	18.177	5.260	19.651

Note: The first six columns report the mean and standard deviation of each variable for each of the three treatment groups who receive incentives 2, 10, and 30 days after their first purchase. The last two columns are for the control group with no emails.

5.2 Average Treatment Effects

Before we explain the estimation of DTR with the experiment data, we report the average treatment effects in this section.

First Experiment. First, we estimate the average treatment effects with the first experiment data. We compare the average outcomes between each of the three treatment groups and the control group. Specifically, we estimate the following linear regression:

$$Y_i = \beta_0 + \sum_{t=1}^3 \beta_{1t} A_{it} + \varepsilon_{it},$$

where Y_i is the outcome of interest and $A_{it} \in \{0, 1\}$ is the dummy indicating receiving a coupon at time t (day 2, 10, or 30). β_{1t} indicates the average treatment effect for time t . As the outcome variables, we consider the following three: whether or not the user makes the second purchase (retention), the total purchase (sales) in JPY, and the number of items purchased (quantity).²³ The outcomes are measured 8 weeks or 12 weeks after the first purchase, not after each treatment.

As Panel (A) of Table 3 shows, we find that the average treatment effect of a coupon on the retention rate is 2.3% for the 2-day treatment and is monotonically decreasing as the coupon is sent at later timings. For total spending, the treatment effect is 267 JPY for the 2-day treatment, while not statistically significant for the other treatments. The treatment effects on quantity show a similar pattern to the effects on retention. Although we cannot disclose the baseline due to the NDA, the treatment effects are economically significant.

²³The estimation sample includes the users who make the second purchase before they receive incentives. When we remove all users who purchase before they receive coupons, the results are virtually the same as Table 3.

Table 3: Average Treatment Effects (First Experiment)

	Retention	Sales (JPY)	Quantity	Retention	Sales (JPY)	Quantity
	(A): 8 Week Outcomes			(B): 12 Week Outcomes		
2 day	0.023 (0.001)	266.7 (109.4)	0.091 (0.023)	0.023 (0.002)	421.9 (147.4)	0.118 (0.030)
10 day	0.015 (0.002)	23.5 (107.0)	0.041 (0.022)	0.011 (0.002)	-35.0 (144.1)	0.023 (0.029)
30 day	-0.002 (0.002)	102.4 (106.8)	0.024 (0.022)	0.010 (0.002)	157.1 (143.9)	0.029 (0.029)

Note: The first column reports the treatment effects on whether a customer makes any purchases within 8 weeks (Panel (A)) or 12 weeks (Panel (B)) since their first purchase. The second column reports the treatment effects on total sales and the third column reports the treatment effects on the number of items purchased. The standard errors are in parentheses. The table does not report the constants as the constants reveal the baseline retention rates, sales, and quantities, which is prohibited due to the NDA.

Since customers who are assigned to the 30-day treatment have less time to make purchases until the outcomes are measured 8 weeks after the first purchase, the treatment effect may mechanically decrease. To mitigate such a concern, we extend the length of the period we measure outcomes to 12 weeks so that those in the 30-day treatment have sufficient time to make a second purchase. Panel (B) reports the treatment effects for the outcomes within 12 weeks. We find that the treatment effects are qualitatively similar to Panel (A). Thus, we focus on the 8-week outcome as the main outcome when learning optimal DTRs. The optimal uniform strategy is to send incentives 2 days after the first purchase.

Second Experiment. Next, we estimate the average treatment effects with the second experiment data. Notice that there are two treatments, an email with a coupon and an email without a coupon. This design allows us to separately estimate the effect of coupons and emails. At the same time, since we have two sorts of treatments at different timings, we estimate the ATE of one treatment (incentive/email on day 2, 10, or 30) by controlling the treatment conditions of the other timings. Specifically, to estimate the ATE of financial incentives on one timing, we use the subsample of the users who receive the email on at least one of the other timings: for example, to estimate the ATE of the 2-day treatment of financial incentives, we select the users who receive the financial incentive on day 2 and the email on day 10 or 30 or both, and then compare their outcomes with the outcomes of the users who receive nothing on day 2 but receive the email on day 10 or 30 or both. Similarly, to estimate the ATE of emails, we condition on the subsample that receives financial incentives on one of the other timings. In Appendix C.1, we report the results of the ATE under different conditions.

The estimation results in Table 4 show that the appreciation email without coupons might increase

Table 4: Average Treatment Effects on 8-Week Outcomes (Second Experiment)

	Retention	Sales (JPY)	Quantity	Retention	Sales (JPY)	Quantity
	(A): Financial incentive			(B): Only appreciation email		
2 day	0.043 (0.003)	341.4 (164.6)	0.132 (0.032)	0.001 (0.003)	-39.7 (151.9)	-0.007 (0.032)
10 day	0.026 (0.003)	88.7 (111.8)	0.037 (0.029)	0.002 (0.003)	116.9 (161.7)	0.024 (0.037)
30 day	0.014 (0.003)	17.3 (182.7)	0.041 (0.019)	0.001 (0.003)	-121.7 (110.0)	-0.032 (0.036)

Note: In each panel, the first column reports the treatment effects on whether a customer makes any purchases within 8 weeks since their first purchase. The second column reports the treatment effects on total sales and the third column reports the treatment effects on the number of items purchased. For the first panel, we condition on the subsample of customers who receive an email on at least one of the other timings. For the second panel, we condition on the subsample of customers who receives financial incentive on one of the other timings. The standard errors are in parentheses. The table does not report the constants as the constants reveal the baseline retention rates, sales, and quantities, which is prohibited due to the NDA.

retention, although estimates are mostly statistically insignificant and economically negligible.²⁴ The 2-day and 30-day treatment effects of emails on retention are 0.1% and the 10-day treatment effect is 0.2%. Sales also increase by 117 JPY by emails for the 10-day treatment, but not for the 2-day or 30-day treatment (40 and 122 JPY decrease, both statistically insignificant). The results also reveal that the coupons in addition to the appreciation emails increase retention, sales, and the number of items purchased. The 2-day coupons increase the retention rate by 4.3% and sales by 341 JPY.

5.3 Heterogeneity in Treatment Effects

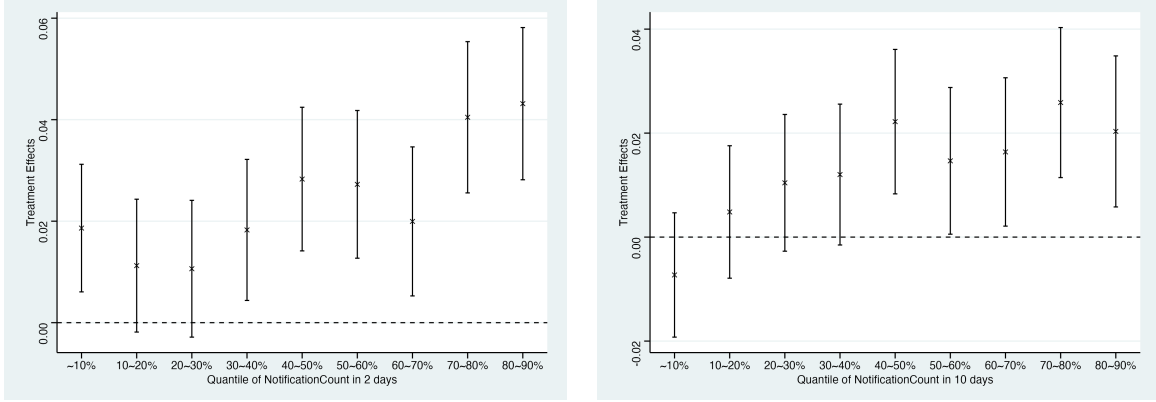
Although the average treatment effects are informative to see which treatment is more effective than others on average, it may not necessarily be the case that the 2-day treatment is optimal for all users. To see how much heterogeneity exists in the treatment effect, we now estimate the conditional average treatment effects (CATEs) for the first experiment by considering the following regression:

$$Y_i = \beta_0(X_i) + \sum_{t=1}^3 \beta_{1t}(X_i)A_{it} + \varepsilon_i,$$

where Y_i is the indicator of retention, $A_{it} \in \{0, 1\}$ indicates receiving a coupon at time t (day 2, 10, or 30), $X_i \in \mathbb{R}$ is a variable for which we examine heterogeneity in the treatment effect, $\beta_0(X_i)$ is the conditional mean of Y_i given X_i for the control group, and $\beta_{1t}(X_i)$ indicates the average treatment effect for period t conditional on X_i . We parametrize $\beta_0(X_i)$ and $\beta_{1t}(X_i)$ so that they may vary depending on which decile of X_i the user belongs to.

²⁴To save space, we report the results for the outcomes measured 8 weeks after the first purchase. Results for 12-week outcomes are presented in Appendix C.1.

Figure 4: (b) 10 day, the number of messages sent



Note: The figures show the average treatment effects (2-day treatment for Panel (a) and 10-day treatment for Panel (b)) on the retention rate within 8 weeks against the number of messages sent, which is split into deciles. Error bars indicate the 95% confidence intervals.

We estimate the model by the ordinary least squares (OLS) with the first experiment data. Figure ?? shows the estimates of the conditional average treatment effects of the 2-day treatment (Panel (a)) and 10-day treatment (Panel (b)) on the 8-week retention rate for each decile of the number of messages sent to the user until the delivery of their first purchased item.²⁵ The treatment effects vary significantly across users: for both 2-day and 10-day treatments, the average treatment effect tends to increase as the user receives more messages. In Appendix C.2, we present the results for the 30-day treatment and for heterogeneity with respect to different conditional variables.²⁶ Substantial heterogeneity leads us to consider personalized policies.

6 Learning Optimal DTRs

In this section, we propose our strategy for learning the optimal dynamic policy under the budget and feasibility constraints described in Section 4. This section focuses on the baseline setting where the action is binary. In Section 6.3 and Appendix F, we discuss our learning algorithm for the setting with multiple actions.

In Section 6.1, we first show that a threshold rule is an optimal dynamic policy for the binary-action case. We then present an algorithm for estimating the optimal policy using the experimental data in Section 6.2. Our algorithm requires two types of base estimation methods, one for estimating threshold

²⁵These messages do not include the appreciation email designed for first-time buyers; rather, they include several different types of messages sent to all users who satisfy certain criteria.

²⁶As a robustness check, we also estimate the CATE with respect to low-dimensional variables with high-dimensional controls, using double/debiased machine learning [Chernozhukov et al., 2018]. The results, available upon request, also exhibit clear heterogeneity.

rules and the other for evaluating the performance of DTRs. Sections 6.2.1 and 6.2.2 discuss possible approaches to these two types of estimation.

6.1 Optimal DTRs

Recall that our optimization problem is given by

$$\begin{aligned} & \max_{\mathbf{d}} E_{\mathbf{d}}[Y] \\ & \text{s.t. } E_{\mathbf{d}}[C] \leq B \\ & \{a_t \in \mathcal{A}_t : d_t(a_t|h_t) > 0\} \subset \mathcal{A}_t(h_t) \text{ for all } h_t \in \mathcal{H}_t \text{ and } t = 1, \dots, T. \end{aligned} \quad (6.1)$$

We show that a *threshold DTR* is a solution to this optimization problem. To define threshold DTRs, we introduce the *Q-function*, sequentially defined as follows. Given a DTR $\mathbf{d} = (d_1, \dots, d_T)$, let $\underline{\mathbf{d}}_t = (d_1, \dots, d_t)$ and $\bar{\mathbf{d}}_t = (d_t, \dots, d_T)$. For the final period T , the Q -function is the conditional mean outcome given the history and final action:

$$Q_T^Y(h_T, a_T) = E[Y|H_T = h_T, A_T = a_T].$$

For $t = T-1, \dots, 1$, the Q -function is the conditional mean outcome given the history and current action, assuming the future actions are determined by $\bar{\mathbf{d}}_{t+1} = (d_{t+1}, \dots, d_T)$:

$$\begin{aligned} Q_t^Y(h_t, a_t; \bar{\mathbf{d}}_{t+1}) &= E_{\bar{\mathbf{d}}_{t+1}}[Y|H_t = h_t, A_t = a_t] \\ &= E \left[\sum_{a_{t+1} \in \mathcal{A}_{t+1}} d_{t+1}(a_{t+1}|H_{t+1}) Q_{t+1}^Y(H_{t+1}, a_{t+1}; \bar{\mathbf{d}}_{t+2}) \middle| H_t = h_t, A_t = a_t \right], \end{aligned}$$

where we set $Q_T^Y(h_T, a_T; \bar{\mathbf{d}}_{T+1}) = Q_T^Y(h_T, a_T)$. We define the Q -function for the cost C analogously by $Q_T^C(h_T, a_T) = E[C|H_T = h_T, A_T = a_T]$ and $Q_t^C(h_t, a_t; \bar{\mathbf{d}}_{t+1}) = E_{\bar{\mathbf{d}}_{t+1}}[C|H_t = h_t, A_t = a_t]$.

Threshold DTRs make decisions based on these Q -functions.

Definition 1 (Threshold DTRs). The threshold DTR with parameter $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_T) \in \mathbb{R}_+^T$, denoted by $\mathbf{d}(\boldsymbol{\lambda}) = (d_t(\cdot; \bar{\boldsymbol{\lambda}}_t))_{t=1}^T$ with $\bar{\boldsymbol{\lambda}}_t = (\lambda_t, \dots, \lambda_T)$, is defined recursively as follows. For the final period T ,

$$d_T(h_T; \lambda_T) \in \arg \max_{a_T \in \mathcal{A}_T(h_T)} \{Q_T^Y(h_T, a_T) - \lambda_T Q_T^C(h_T, a_T)\}.$$

For $t = T - 1, \dots, 1$,

$$d_t(h_t; \bar{\lambda}_t) \in \arg \max_{a_t \in \mathcal{A}_t(h_t)} \{Q_t^Y(h_t, a_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})) - \lambda_t Q_t^C(h_t, a_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}))\},$$

where $\bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}) = (d_{t+1}(\cdot; \bar{\lambda}_{t+1}), \dots, d_T(\cdot; \lambda_T))$.

By construction, threshold DTRs only choose among feasible actions, satisfying the feasibility constraint. Here, λ is a hyperparameter that controls the cost under the threshold DTR. When the action is binary and both actions are feasible, the period- t rule can be written as

$$d_t(h_t; \bar{\lambda}_t) = \mathbf{1} \left[\frac{Q_t^Y(h_t, 1; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})) - Q_t^Y(h_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}))}{Q_t^C(h_t, 1; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})) - Q_t^C(h_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}))} \geq \lambda_t \right], \quad (6.2)$$

provided that the denominator is positive. In other words, the rule assigns the treatment (action 1) to individuals for which the ratio between the conditional average treatment effects on the outcome and cost exceeds a certain threshold. Given the future rules $\bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})$, increasing the period- t threshold λ_t leads to a smaller fraction of individuals who receive the treatment in period t . Furthermore, we can show from the definitions of the Q -functions and the threshold rule that $d_t(\cdot; \bar{\lambda}_t)$ maximizes

$$E_{\underline{\mathbf{d}}'_{t-1}, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})}[Y - \lambda_t C]$$

among period- t rules d_t (given any arbitrary $\underline{\mathbf{d}}'_{t-1}$).²⁷ Therefore, λ_t can be interpreted as the “shadow price” of the budget constraint. If there is no budget constraint, the choice of $\lambda = (0, \dots, 0)$ leads to an optimal DTR.

In the next proposition, we show that when the action is binary, the threshold DTR achieves the largest average outcome under the budget constraint with an appropriate choice of λ .

Proposition 1. *Let $\mathcal{A}_t = \{0, 1\}$ for all $t = 1, \dots, T$. Suppose that a solution to the maximization problem (6.1) exists, and let $\mathbf{d}^*$ denote a solution. Suppose also that there exists $\lambda^* \in \mathbb{R}_+^T$ such that $E_{\mathbf{d}(\lambda^*)}[C] = B$ and that $E_{\underline{\mathbf{d}}_t^*, \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*)}[C] = B$ for all $t = 1, \dots, T - 1$.²⁸ Then $\mathbf{d}(\lambda^*)$ solves (6.1).*

Proof. See Appendix G.1. ■

²⁷See Appendix G.2 for the details.

²⁸The existence of a threshold DTR with binding budget constraints excludes the case where the average cost discontinuously changes as λ changes so that a randomized DTR is optimal. We can show the existence of such a threshold DTR under primitive conditions such as the absolute continuity of the probability measures of the Q -values.

6.2 Proposed Algorithm

The result in the previous section suggests solving the problem (6.1) over the class of threshold DTRs. That is, the DTR $\mathbf{d}(\lambda^*)$ is optimal, where

$$\lambda^* \in \arg \max_{\lambda \in \mathbb{R}_+^T} V^Y(\mathbf{d}(\lambda)) \text{ s.t. } V^C(\mathbf{d}(\lambda)) \leq B, \quad (6.3)$$

where $V^Y(\mathbf{d}) = E_{\mathbf{d}}[Y]$ and $V^C(\mathbf{d}) = E_{\mathbf{d}}[C]$.

Based on the above observation, we propose an algorithm for using the data $\{H^{(i)}\}_{i=1}^n$ to estimate $\mathbf{d}(\lambda^*)$, presented in Definition 2. Our algorithm learns the optimal parameter value λ^* by applying a hyperparameter optimization method to the problem (6.3) where the true average outcome and cost of $\mathbf{d}(\lambda)$, $V^Y(\mathbf{d}(\lambda))$ and $V^C(\mathbf{d}(\lambda))$, are replaced with their cross-validation estimates, $\hat{V}_{\text{CV}}^Y(\lambda)$ and $\hat{V}_{\text{CV}}^C(\lambda)$. In other words, this optimization aims to maximize the estimated outcome $\hat{V}_{\text{CV}}^Y(\lambda)$ among the values of $\lambda \in \mathbb{R}_+^T$ for which the estimated cost $\hat{V}_{\text{CV}}^C(\lambda)$ is below B .

To estimate $V^Y(\mathbf{d}(\lambda))$ and $V^C(\mathbf{d}(\lambda))$ for each candidate value of λ , we first estimate the threshold DTR $\mathbf{d}(\lambda)$ and then estimate the average outcome and cost of the estimated rule. If we use the full data $\{H^{(i)}\}_{i=1}^n$ both for estimating $\mathbf{d}(\lambda)$ and estimating its average outcome and cost, substantial bias might arise due to overfitting. To remove the potential bias, we use K -fold cross-validation, a sample splitting procedure that uses independent, separate samples for the estimation of the rule and for the evaluation of the performance of the estimated rule.

In Section 6.2.1, we develop two methods to estimate the threshold DTR $\mathbf{d}(\lambda)$ used in Step 1(b)i and Step 2 of the algorithm, by modifying existing methods. The two methods allow us to use either an off-the-shelf regression method or a classification method in machine learning. In our application, we try various regression or classification methods and choose the best one in terms of the cross-validation performance of the resulting DTR.

In Section 6.2.2, we introduce the Inverse Probability Weighting estimator, a standard estimator for $V^Y(\mathbf{d})$ and $V^C(\mathbf{d})$ for a given feasible DTR $\mathbf{d}$, which can be used in Step 1(b)ii of the algorithm.

6.2.1 Estimation of Threshold DTRs

We develop two methods to estimate the threshold DTR for given $\lambda \in \mathbb{R}_+^T$. The two methods build on two existing methods for learning an optimal DTR without a budget constraint, Q -learning [e.g., Watkins, 1989, Murphy, 2005b] and Backward Outcome Weighted Learning [BOWL; Zhao et al., 2015], respectively.

Definition 2 (Algorithm for Learning an Optimal DTR).

1. Use a hyperparameter optimization method^a to optimize λ with the optimization problem set to

$$\max_{\lambda \in \mathbb{R}_+^T} \hat{V}_{CV}^Y(\lambda) \text{ s.t. } \hat{V}_{CV}^C(\lambda) \leq B,$$

where for each $\lambda \in \mathbb{R}_+^T$, $\hat{V}_{CV}^Y(\lambda)$ and $\hat{V}_{CV}^C(\lambda)$ are obtained as follows:

- (a) Take a K -fold random partition $(I_k)_{k=1}^K$ of trajectory indices $\{1, \dots, n\}$ such that each fold is of equal size. For each $k = 1, \dots, K$, let $I_k^c = \{1, \dots, n\} \setminus I_k$.
- (b) For each $k = 1, \dots, K$:
 - i. Use the data excluding the k -th fold, $\{H^{(i)}\}_{i \in I_k^c}$, to estimate $\mathbf{d}(\lambda)$. Let $\hat{\mathbf{d}}_k(\lambda)$ denote the estimated DTR.
 - ii. Use the k -th fold of the data, $\{H^{(i)}\}_{i \in I_k}$, to estimate $V^Y(\hat{\mathbf{d}}_k(\lambda))$ and $V^C(\hat{\mathbf{d}}_k(\lambda))$. Let $\hat{V}_k^Y(\lambda)$ and $\hat{V}_k^C(\lambda)$ denote the estimates.
- (c) Take the average of the estimates over K folds to calculate

$$\hat{V}_{CV}^Y(\lambda) = \frac{1}{K} \sum_{k=1}^K \hat{V}_k^Y(\lambda) \text{ and } \hat{V}_{CV}^C(\lambda) = \frac{1}{K} \sum_{k=1}^K \hat{V}_k^C(\lambda).$$

2. Let $\hat{\lambda}$ denote the optimizer from Step 1. Set $\lambda = \hat{\lambda}$ and apply the estimation method used in Step 1(b)i to the full data $\{H^{(i)}\}_{i=1}^n$ to estimate $\mathbf{d}(\hat{\lambda})$. The resulting DTR is our estimate for the optimal DTR $\mathbf{d}(\lambda^*)$.

^aPossible hyperparameter optimization methods include a grid search and Bayesian optimization [see, e.g., Nogueira, 2014]. In our empirical application, we tried both approaches but we use a grid search.

Q-Learning. Q-learning is a sequential method that estimates an unconstrained optimal DTR. Moving backward from the final period, it estimates the Q -function by a regression method and derives the rule by maximizing the estimated Q -function over actions.

We propose a modified version of Q-learning to estimate threshold DTRs, presented in Definition 3. This approach estimates Q -functions and resulting rules backward from $t = T$ to $t = 1$. For the final period T , we first use a regression method to estimate the Q -functions $Q_T^Y(h_T, a_T) = E[Y|H_T = h_T, A_T = a_T]$ and $Q_T^C(h_T, a_T) = E[C|H_T = h_T, A_T = a_T]$. We then plug the estimates into the expression for the rule $d_T(h_T; \lambda_T)$ in Definition 1. For $t < T$, to estimate the period- t Q -function given the future

Definition 3 (Q-Learning for Threshold DTRs). Construct an estimator $\hat{\mathbf{d}}(\lambda) = (\hat{d}_t(\cdot; \bar{\lambda}_t))_{t=1}^T$ for the threshold DTR $\mathbf{d}(\lambda) = (d_t(\cdot; \bar{\lambda}_t))_{t=1}^T$ by iterating the following three steps backward from $t = T$ to $t = 1$:

1. Run a regression of $Y^{(i)}$ (when $t = T$) or $\hat{Y}_{t+1}^{(i)}$ (when $t < T$) on $H_t^{(i)}$ and $A_t^{(i)}$ and a regression of $C^{(i)}$ (when $t = T$) or $\hat{C}_{t+1}^{(i)}$ (when $t < T$) on $H_t^{(i)}$ and $A_t^{(i)}$, where $\hat{Y}_{t+1}^{(i)}$ and $\hat{C}_{t+1}^{(i)}$ are defined in Step 3 of the previous iteration. Let $\hat{Q}_t^Y(h_t, a_t)$ and $\hat{Q}_t^C(h_t, a_t)$ denote the estimated regression functions from the two regressions.

2. Set

$$\hat{d}_t(h_t; \bar{\lambda}_t) \in \arg \max_{a_t \in \mathcal{A}_t(h_t)} \{ \hat{Q}_t^Y(h_t, a_t) - \lambda_t \hat{Q}_t^C(h_t, a_t) \}.$$

3. Let $\hat{Y}_t^{(i)} = \hat{Q}_t^Y(H_t^{(i)}, \hat{d}_t(H_t^{(i)}; \bar{\lambda}_t))$ and $\hat{C}_t^{(i)} = \hat{Q}_t^C(H_t^{(i)}, \hat{d}_t(H_t^{(i)}; \bar{\lambda}_t))$ for $i = 1, \dots, n$.

rules $\bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})$, note that we can write it as

$$\begin{aligned} Q_t^Y(h_t, a_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})) &= E_{\bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})}[Y | H_t = h_t, A_t = a_t] \\ &= E[Q_{t+1}^Y(H_{t+1}, d_{t+1}(H_{t+1}; \bar{\lambda}_{t+1}); \bar{\mathbf{d}}_{t+2}(\bar{\lambda}_{t+2})) | H_t = h_t, A_t = a_t]. \end{aligned}$$

This suggests estimating the Q -function by regression of $\hat{Y}_{t+1}$ on H_t and A_t , where $\hat{Y}_{t+1}$ is an estimate for $Q_{t+1}^Y(H_{t+1}, d_{t+1}(H_{t+1}; \bar{\lambda}_{t+1}); \bar{\mathbf{d}}_{t+2}(\bar{\lambda}_{t+2}))$ from the previous iteration. Once we estimate the Q -functions both for the outcome and cost, we plug them into the expression in Definition 1 to estimate $d_t(h_t; \bar{\lambda}_t)$.

When $\lambda_t = 0$ for all t , our algorithm is standard Q -learning for maximizing $V^Y(\mathbf{d})$. In our application, we try several machine-learning regression methods such as LASSO, Random Forest, and LightGBM (Light Gradient Boosting Machine) to estimate Q -functions.²⁹

Backward Outcome Weighted Learning. BOWL is another sequential method that estimates an unconstrained optimal DTR, which can be applied to the binary-action case. This approach aims to directly maximize a nonparametric estimate of the average outcome over DTRs. This is in contrast to

²⁹In principle, one can apply deep reinforcement learning (DRL) methods to estimate the Q -functions. There are a few challenges in our application. First, typically DRL is applied to the case with a longer time horizon. Since our application has only three periods, the benefits of using DRL may be limited. Second, typical applications of DRL consider a Markov situation, while our application considers a non-Markov situation. Hence, widely used DRL algorithms may not work.

regression-based methods such as Q -learning, which indirectly attempts such maximization by plugging the estimated Q -functions into the optimal DTR.

Specifically, BOWL casts the problem of maximizing the average outcome over DTRs as a sequence of binary classification problems, where the objective of each problem is to minimize a weighted misclassification error. Going backward from the final period, BOWL directly estimates the optimal rule by applying a classification method to the binary classification problem.

We propose a modified version of BOWL to estimate threshold DTRs. A key observation is that the threshold DTR solves a sequence of minimization problems.

Proposition 2. *Let $\mathcal{A}_t = \{0, 1\}$ for all $t = 1, \dots, T$. Under Assumption 1, the threshold DTR $\mathbf{d}(\lambda) = (d_t(\cdot; \bar{\lambda}_t))_{t=1}^T$, defined in Definition 1, satisfies the following: $d_T(\cdot; \lambda_T)$ solves*

$$\min_{d_T: \mathcal{H}_T \rightarrow \{0, 1\}} E \left[\frac{Y - \lambda_T C}{d_T^0(A_T | H_T)} \mathbf{1}[A_T \neq d_T(H_T)] \right] \text{ s.t. } d_T(h_T) \in \mathcal{A}_T(h_T) \text{ for all } h_T \in \mathcal{H}_T,$$

and for $t = T - 1, \dots, 1$, $d_t(\cdot; \bar{\lambda}_t)$ solves

$$\min_{d_t: \mathcal{H}_t \rightarrow \{0, 1\}} E \left[\frac{(Y - \lambda_t C) \prod_{j=t+1}^T \mathbf{1}[A_j = d_j(H_j; \bar{\lambda}_j)]}{\prod_{j=t}^T d_j^0(A_j | H_j)} \mathbf{1}[A_t \neq d_t(H_t)] \right] \\ \text{ s.t. } d_t(h_t) \in \mathcal{A}_t(h_t) \text{ for all } h_t \in \mathcal{H}_t.$$

Proof. See Appendix G.2. ■

The intuition for this result is as follows. From the definitions of the Q -functions and the threshold rule, we can show that the period- T rule $d_T(\cdot; \lambda_T)$ maximizes

$$E_{\underline{\mathbf{d}}_{T-1}, d_T} [Y - \lambda_T C] = E \left[\frac{(Y - \lambda_T C) \mathbf{1}[A_T = d_T(H_T)]}{d_T^0(A_T | H_T)} \right]$$

among deterministic rules d_T that satisfy the feasibility constraint. Here, the expectation on the left-hand side is taken with respect to the distribution if we follow the data-generating rule $\mathbf{d}^0$ prior to the final period and then follow a period- T candidate rule d_T , while that on the right-hand side is under the data-generating rule $\mathbf{d}^0$. The equality of the two expected values follows from the inverse probability weighting technique. When the action is binary and hence $\mathbf{1}[A_T = d_T(H_T)] = 1 - \mathbf{1}[A_T \neq d_T(H_T)]$, maximizing the right-hand side over d_T is equivalent to minimizing $E \left[\frac{Y - \lambda_T C}{d_T^0(A_T | H_T)} \mathbf{1}[A_T \neq d_T(H_T)] \right]$, leading to the result in Proposition 2. A similar argument applies to periods $t < T$.

Definition 4 (BOWL for Threshold DTRs). Construct an estimator $\hat{\mathbf{d}}(\boldsymbol{\lambda}) = (\hat{d}_t(\cdot; \bar{\boldsymbol{\lambda}}_t))_{t=1}^T$ for the threshold DTR $\mathbf{d}(\boldsymbol{\lambda}) = (d_t(\cdot; \bar{\boldsymbol{\lambda}}_t))_{t=1}^T$ by iterating the following three steps backward from $t = T$ to $t = 1$:

1. Apply a weighted classification method to the subsample for which both actions are feasible in period t (i.e. $\mathcal{A}_t(H_t^{(i)}) = \{0, 1\}$), using $A_t^{(i)}$ as the true class, $H_t^{(i)}$ as the feature vector, and $\frac{Y^{(i)} - \lambda_T C^{(i)}}{d_T^0(A_T^{(i)} | H_T^{(i)})}$ (when $t = T$) or $\frac{(Y^{(i)} - \lambda_t C^{(i)}) \prod_{j=t+1}^T \mathbf{1}[A_j^{(i)} = \hat{A}_j^{(i)}]}{\prod_{j=t}^T d_j^0(A_j^{(i)} | H_j^{(i)})}$ (when $t < T$) as the weight for observation i .^a Here, $\hat{A}_{t+1}^{(i)}, \dots, \hat{A}_T^{(i)}$ are defined in Step 3 of the iterations prior to t . Let $\tilde{d}_t(h_t)$ denote the constructed classification rule.

2. Set

$$\hat{d}_t(h_t; \bar{\boldsymbol{\lambda}}_t) = \begin{cases} 1 & \text{if } \mathcal{A}_t(h_t) = \{1\} \\ 0 & \text{if } \mathcal{A}_t(h_t) = \{0\} \\ \tilde{d}_t(h_t) & \text{if } \mathcal{A}_t(h_t) = \{0, 1\}. \end{cases}$$

3. Let $\hat{A}_t^{(i)} = \hat{d}_t(H_t^{(i)}; \bar{\boldsymbol{\lambda}}_t)$ for $i = 1, \dots, n$.

^aFor the final period, to prevent negative weights, we can instead use $L_T = \mathbf{1}[W_T \geq 0]$ as the true class and $|W_T|$ as the weight, where $W_T = \frac{Y - \lambda_T C}{d_T^0(A_T | H_T)}(2A_T - 1)$. This is based on the fact that minimizing $E \left[\frac{Y - \lambda_T C}{d_T^0(A_T | H_T)} \mathbf{1}[A_T \neq d_T(H_T)] \right]$ is equivalent to minimizing $E[|W_T| \mathbf{1}[L_T \neq d_T(H_T)]]$, which can be shown by some algebra. For $t < T$, we can use $L_t = \mathbf{1}[W_t \geq 0]$ as the true class and $|W_t|$ as the weight, where $W_t = \frac{(Y - \lambda_t C) \prod_{j=t+1}^T \mathbf{1}[A_j = \hat{A}_j]}{\prod_{j=t}^T d_j^0(A_j | H_j)}(2A_t - 1)$. We implement this approach in our application.

The objective functions in Proposition 2 can be viewed as a weighted misclassification error of the classification problem where we aim to classify the observed action A_t based on H_t . The weight for each misclassification event is, for example, given by $\frac{Y - \lambda_T C}{d_T^0(A_T | H_T)}$ for the final period $t = T$. Our proposed approach, presented in Definition 4, estimates a threshold DTR by sequentially solving the weighted classification problems.

When $\lambda_t = 0$ for all t and there is no feasibility constraint, our algorithm is standard BOWL for maximizing $V^Y(\mathbf{d})$. In our application, we try several classification methods such as SVM (Support Vector Machine), Random Forest, and logistic regression in Step 1.³⁰

³⁰Since the objective function of the weighted classification problem is non-convex and discontinuous, Zhao et al. [2015] propose to replace the 0-1 loss $\mathbf{1}[A_t \neq d_t(H_t)]$ with a hinge loss $\phi(A_t f_t(H_t)) = \max(1 - A_t f_t(H_t), 0)$, where $A_t \in \{-1, 1\}$ and $f_t : \mathcal{H}_t \rightarrow \mathbb{R}$ is the decision function so that $d_t(h_t) = \text{sign}(f_t(h_t))$.

6.2.2 Evaluation of DTRs

Our proposed algorithm requires a method for estimating the average outcome $V^Y(\mathbf{d}) = E_{\mathbf{d}}[Y]$ and cost $V^C(\mathbf{d}) = E_{\mathbf{d}}[C]$ of a given feasible DTR $\mathbf{d} = (d_1, \dots, d_T)$. Standard methods include the Inverse Probability Weighting (IPW) estimator [Precup et al., 2000, Strehl et al., 2010]. For a deterministic DTR $\mathbf{d}$, the IPW estimator using data $\{H^{(i)}\}_{i=1}^n$ is given by

$$\hat{V}_{\text{IPW}}^Y(\mathbf{d}) = \frac{1}{n} \sum_{i=1}^n Y^{(i)} \prod_{t=1}^T \frac{\mathbf{1}[A_t^{(i)} = d_t(H_t^{(i)})]}{d_t^0(A_t^{(i)} | H_t^{(i)})}, \quad \hat{V}_{\text{IPW}}^C(\mathbf{d}) = \frac{1}{n} \sum_{i=1}^n C^{(i)} \prod_{t=1}^T \frac{\mathbf{1}[A_t^{(i)} = d_t(H_t^{(i)})]}{d_t^0(A_t^{(i)} | H_t^{(i)})}.$$

Under Assumption 1, the IPW estimator is unbiased and consistent for any DTR that satisfies the feasibility constraint. The IPW-based evaluation has been used in existing papers such as Hitsch et al. [2024] and Yoganarasimhan et al. [2023].

In our application, we use the IPW estimator for several reasons.³¹ First, IPW does not involve any first-stage estimation, while alternative methods, such as the Doubly Robust (DR) estimator [Jiang and Li, 2016], require the estimation of high-dimensional functions and can be sensitive to the choice of the first-stage estimation method. The absence of a first-stage estimation in IPW also significantly increases the computational speed of our proposed algorithm. Second, since our application has only three periods, the scope of variance reduction by other methods may be limited. Third, alternative approaches often focus on Markov settings and may not be directly applicable to our non-Markov setting. However, we note that alternative estimators such as DR may perform better in other empirical scenarios, including the case with a long time horizon and the case where the data-generating policy $\mathbf{d}^0$ is unknown and needs to be estimated.

6.3 Discussion

Here, we provide a comparison between Q -Learning and BOWL and additional details on the implementation. We also compare our dynamic approach with static approaches and discuss a possible approach to the extension to multi-action settings.

³¹While the IPW estimator is unbiased, its variance tends to be large due to potentially extreme weights, especially if the number of periods T is large. To reduce the variance, a variety of methods have been proposed in the literature of “off-policy policy evaluation” (OPE) for reinforcement learning [Jiang and Li, 2016, Thomas and Brunskill, 2016, Farajtabar et al., 2018, Kallus and Uehara, 2020]. These methods typically combine IPW with regressions for estimating Q -functions, reducing the variance at the cost of introducing some bias. We try the DR estimator in addition to the IPW estimator in our application. The results are qualitatively similar.

Comparison between Q-Learning and BOWL. Although both Q-learning and BOWL can theoretically find the optimal DTR given the availability of infinite data and no misspecification, it may be useful to compare the two approaches.

Q-learning fits a model to best predict Q -functions rather than to maximize the expected outcome of the resulting DTR [Murphy, 2005b]. Consequently, even in an ideal setting with infinite data, if the model for the Q -function is misspecified, the DTR induced from maximizing the fitted Q -function may be far from optimal. Moreover, even if the model is correctly specified, the performance of Q-learning may be poor with finite data, since minimizing the prediction error for the Q -function does not necessarily lead to a larger value of the resulting DTR.

By contrast, BOWL is designed to directly maximize the value of the DTR. Even if the true optimal DTR does not belong to the space of classifiers used by a classification algorithm, the derived classifier maximizes the expected outcome over the restricted classifier space, at least with infinite data. BOWL may also perform relatively well with finite data since minimizing the weighted classification error translates into maximizing the estimated outcome. Existing simulation studies show that BOWL tends to outperform Q-learning in most situations except the case where the model for the Q -function is correctly specified and the sample size is large [Zhao et al., 2012, 2015].

That said, in our proposed algorithm (Definition 2), we can tune the hyperparameters of the regression or classification algorithm to optimize the cross-validation performance of the resulting DTR, not the prediction or classification error. Hence, the aforementioned drawbacks of Q-learning may be alleviated. We therefore recommend trying both Q-learning and BOWL with different regression or classification algorithms and choosing the best one in terms of cross-validation performance.

Comparison with Static Approaches. As we discussed in Section 4.2, our approach differs from two natural static alternatives. The first determines the entire action profile $(A_1, \dots, A_T)$ using only the initial state X_1 . Because it cannot condition on information observed later, it forgoes the value of adapting decisions to customers' evolving states. Appendix A formalizes this benefit of dynamic policies.

Another static approach would be to estimate each period's policy d_t separately using (H_t, A_t, Y) . This approach ignores that the value of the current action depends on the policies implemented in future periods. It also cannot readily accommodate an inter-temporal budget constraint because decisions are optimized separately across periods. By contrast, our dynamic programming approach jointly optimizes the sequence of policies using updated histories and future continuation values.

Extension to Multiple Actions. In the case with three or more actions, the class of threshold DTRs, given by Definition 1, may not contain an optimal DTR. While there is no global optimality guarantee, as a heuristic we consider finding an optimal DTR among threshold DTRs, i.e., solving the problem (6.3). Specifically, we propose using our algorithm (Definition 2) in Section 6.2; it can be directly applied to estimate an optimal threshold DTR for the case with multiple actions without any modifications. As subroutines of the algorithm, one can directly use Q -learning (Definition 3) to estimate threshold DTRs and use the IPW estimator to evaluate DTRs even with multiple actions. By contrast, BOWL is originally designed for the binary-action case and is not directly applicable to the multiple-action case. In Appendix F, we propose a modification of BOWL based on the transformation of the problem of maximizing the average outcome into a sequence of multiple binary classification problems.

Bias Reduction Techniques. In the procedure of Q -learning in Section 6.2.1, after estimating the Q -functions at each iteration, we use them both for choosing the action and estimating the Q -values of the chosen action. This may result in biased estimates of the Q -values due to overfitting [Lan et al., 2020], which may in turn produce estimation errors in the next iteration. To reduce the bias, we propose using sample splitting. That is, at each iteration, we split the sample into two subsamples and construct two sets of estimated Q -functions using the two subsamples separately. We then use one set to choose the action and use the other to estimate the Q -values for the chosen action. We implement this sample-splitting procedure in our empirical application. The procedure of BOWL also has a problem of potential overfitting bias. We implement an analogous sample-splitting procedure for BOWL to reduce the bias in our empirical application.³²

7 Offline Evaluation Results

In this section, we evaluate the performance of our policies using offline evaluation methods. To do so, we randomly split our experimental data into the training set (70%) and the test set (30%), use the training set to learn optimal policies, and then use the test set to estimate their average outcome and cost by the IPW estimator (see Section 6.2.2).

We first consider the baseline setting with two actions and examine the benefits of our constrained dynamic personalized policies based on the first experimental data. We also show the results for the

³²Specifically, at each iteration, when calculating $\hat{A}_t^{(i)}$ in Step 3 of the procedure, we split the sample into two subsamples and construct two rules using the two subsamples separately. We then calculate the decision $\hat{A}_t^{(i)} = \hat{a}_t(H_t^{(i)}; \hat{\lambda}_t)$ for each i by using the rule constructed from the subsample that does not include trajectory i .

extended setting with three actions from the second experimental data.

7.1 Baseline Model

7.1.1 Policies

Remember that the baseline model has two actions: sending a coupon with an appreciation email and only an email. We consider a class of policies that satisfy the inter-temporal budget constraint and feasibility constraints.³³ The budget constraint limits the per-user cost measured by the coupon amount used within 2 months after the first purchase. The feasibility constraints are: (1) a coupon is sent to each user at most once over the three possible timings, and (2) a coupon is not allowed to be sent to users who have already made a second purchase.

We evaluate the performance of our proposed DTRs with and without the budget constraint and the following two benchmark static policies without the budget constraint:

1. Static uniform policy: This policy sends a coupon 2 days after the first purchase to all users who have not yet made a second purchase.
2. Static personalized policy: This policy decides whether to send a coupon and at which timing solely based on the variables available 2 days after the first purchase. In other words, it chooses the action profile (A_1, A_2, A_3) among four possible profiles based on the initial state X_1 . We use the data on $(X_1, A_1, \dots, A_3, Y)$ to estimate the static optimal policy by single-period Q -learning or BOWL treating the action profile as a single action, without imposing the budget constraint.³⁴ As discussed in Section 6.3, this policy does not use updated state variables, unlike dynamic policies.

Note that the two benchmark policies may not satisfy the inter-temporal budget constraint. Also, the static personalized policy is the optimal static policy that allocates coupons based on the conditional average treatment effect of sending a coupon at each of the three possible timings relative to sending only an email.

³³We also tried the policy learning approach proposed by Zhou et al. [2023]. However, as the authors note, a key disadvantage of this approach is that it is hard to handle a problem with many variables. Unfortunately, this approach is computationally infeasible for our problem.

³⁴If this policy is applied to the dynamic environment, it may send a coupon 10 or 30 days after the first purchase even if the user has already made a second purchase since such information is not available in the initial state. After learning the policy, we adjust it so that it does not send a coupon to users who have already made a second purchase, so that the feasibility constraints are satisfied.

7.1.2 Choice of Algorithms, Outcomes, and State Variables

Here we summarize our choice of the algorithms and variables for policy learning.

Algorithms. For Q -learning, one can estimate Q -functions with various machine learning algorithms such as LASSO, Random Forest, LightGBM (Light Gradient Boosting Machine), and deep learning.³⁵ For BOWL, one can use any classification algorithm to solve the weighted classification error minimization problem such as SVM, logistic regression, Random Forest, and SGDC (Stochastic Gradient Descent Classifier).³⁶

To see which algorithm performs better in our empirical setting, we compare the performance of different algorithms to estimate dynamic optimal policies without budget constraints. Based on the results, we choose to use SGDC with L2 regularization and hinge loss (SGDCClassifier in Python) for BOWL and LASSO (Lasso in Python) for Q -learning, since these algorithms provide sufficiently high retention rates with smaller costs than other algorithms.³⁷ These algorithms are also attractive in their feasibility and computational time.

For both SGDC with L2 regularization and hinge loss and LASSO, we tune the regularization parameters and set the other hyperparameters to the default values. For each model, we choose the regularization parameters and $(\lambda_1, \lambda_2, \lambda_3)$ (if we consider budget constraints) jointly through grid search to optimize the 5-fold cross-validation estimates of the expected outcome and cost in the training sample.³⁸ See Appendix E.1 for training details, including model specification and hyperparameter tuning.

Target Outcome and State Variables. We learn policies to maximize the probability of making a second purchase within 2 months of the first purchase, to reflect the company’s primary objective of retention management for first-time buyers. As discussed in Section 4, our framework can encompass a large number of state variables. When learning the optimal policies, we use more than 100 state variables such as the user’s demographics, past purchase behavior, browsing behavior, and history of responses to the company’s marketing activities. Note that a dynamic policy assigns an action based

³⁵We do not use the deep reinforcement learning approach here for two reasons. First, as long as we are aware, there is no explicit reinforcement learning algorithm that can accommodate inter-temporal constraints easily. Second, there seem to be no established deep reinforcement learning algorithms that can be used to learn the optimal DTR under a non-Markov non-stationary decision process as in our case. Since our setup of retention management for first-time buyers necessitates a non-Markov strategy, it is important to consider a non-stationary dynamic programming problem.

³⁶SGDC is a linear classifier including SVM, optimized by the stochastic gradient descent. We try using the L1 norm (as LASSO), the L2 norm (as Ridge regression), and combinations of them (the elastic net) for regularization.

³⁷The results are available upon request.

³⁸We also tried Bayesian optimization [see, e.g., Nogueira, 2014] for hyperparameter tuning, but grid search produced policies with better cross-validation performance.

on all the information available at each t , including the past and current state variables and the actions taken up to that period. Hence, the number of the input variables increases over time.

7.1.3 Offline Evaluation Results

Tables 5 and 6 summarize the offline results on the test set for the static uniform policy, the static personalized policies, and the dynamic personalized policies. We learn optimal personalized policies using either BOWL with SGDC or Q-learning with LASSO. When learning dynamic optimal policies, we consider two different levels of the per-user budget, 20 and 30 JPY. Those values are selected by the company based on their past campaign data. For comparison purposes, we also evaluate dynamic optimal policies without a budget constraint.

As a main performance measure, Table 5 reports the uplift in the second purchase probability relative to the no-incentive policy that always sends an appreciation email only, where the uplift is defined as the difference in the purchase probability rather than the percentage change.^{39,40} We also report the uplift per unit of marketing cost, defined as the increase in the second purchase probability per yen of incentive spending. This metric measures the cost efficiency of each policy and is particularly useful for comparing policies under different budget constraints. In addition, Table 6 reports pairwise differences in the second purchase probability and average marketing cost across selected policies, together with their standard errors.⁴¹

A few observations arise from Tables 5 and 6. First, without a budget constraint, both static and dynamic personalized policies based on Q-learning achieve exactly the same performance as the uniform policy of always sending coupons on day 2. This suggests that these personalized policies do not personalize allocation (which is confirmed by their treatment allocations; see Table 7). We suspect that, in this empirical setting, the treatment effects (the differences in the Q-values between actions) depend on high-dimensional state variables in a complex manner relative to the sample size, resulting in the failure of Q-learning to capture the potential heterogeneity sufficiently.

By contrast, the static personalized policy based on BOWL outperforms the uniform policy. It increases the second purchase probability by 2.09%, compared with 1.86% under the uniform policy,

³⁹Due to the NDA, we are not allowed to disclose the baseline retention rates but are allowed to report only the rate uplifts.

⁴⁰We note that all of the policies have positive uplifts not only in the second purchase probability but also in the third purchase probability, relative to the no-incentive policy. This result mitigates the potential concern that providing incentives merely moves forward the timing of the second purchase without increasing the total number of purchases.

⁴¹We calculate the standard errors as follows. Let $\rho^{(i)}(\mathbf{d}) = \prod_{t=1}^T \frac{\mathbf{1}[A_t^{(i)}=d_t(H_t^{(i)})]}{d_t^0(A_t^{(i)}|H_t^{(i)})}$, so that $\hat{V}_{\text{IPW}}^Y(\mathbf{d}) = \frac{1}{n} \sum_{i=1}^n Y^{(i)} \rho^{(i)}(\mathbf{d})$. Then, for two DTRs $\mathbf{d}$ and $\mathbf{d}'$, we calculate the standard error of $\hat{V}_{\text{IPW}}^Y(\mathbf{d}) - \hat{V}_{\text{IPW}}^Y(\mathbf{d}')$ by $\frac{1}{\sqrt{n}} \sqrt{\frac{1}{n} \sum_{i=1}^n (Y^{(i)}(\rho^{(i)}(\mathbf{d}) - \rho^{(i)}(\mathbf{d}')) - (\hat{V}_{\text{IPW}}^Y(\mathbf{d}) - \hat{V}_{\text{IPW}}^Y(\mathbf{d}'))^2}$.

Table 5: Offline Uplift Performance on Test Set (Baseline Model)

Uplift (Relative to No-Incentive Policy)	2nd	Cost (JPY)	Uplift/Cost
Static Uniform Policy without Budget Constraint	1.86%	48.95	0.0380%
Static Personalized Policy without Budget Constraint			
BOWL	2.09%	47.08	0.0443%
Q-Learning	1.86%	48.95	0.0380%
Dynamic Personalized Policy without Budget Constraint			
BOWL	2.14%	46.92	0.0456%
Q-Learning	1.86%	48.95	0.0380%
Dynamic Personalized Policy with Budget Constraint			
BOWL: Budget = 30	1.66%	33.39	0.0496%
BOWL: Budget = 20	1.23%	21.22	0.0580%
Q-Learning: Budget = 30	0.76%	28.24	0.0268%
Q-Learning: Budget = 20	0.59%	17.97	0.0328%

Note: The uplift is the difference from the no-incentive policy in the second purchase probability within 2 months after the first purchase, and the cost is the average incentive cost incurred within 2 months after the first purchase.

Table 6: Offline Uplift Performance Comparison on Test Set (Baseline Model)

Uplift Difference	2nd	Cost (JPY)
Static Uniform vs. No Incentive	1.86% (0.46)	48.95 (1.60)
Static BOWL vs. Static Uniform	0.23% (0.20)	−1.87 (0.71)
Dynamic BOWL w/o BC vs. Static Uniform	0.28% (0.18)	−2.04 (0.61)
Dynamic BOWL w/o BC vs. Static BOWL	0.05% (0.21)	−0.16 (0.74)
Dynamic BOWL w/o BC vs. Dynamic BOWL: Budget = 30	0.48% (0.30)	13.53 (1.21)

Note: This table reports the differences between policies in the second purchase probability and average cost within 2 months after the first purchase. The standard errors are in parentheses. BC stands for Budget Constraint.

while also using slightly less marketing expenditure (47.08 versus 48.95 JPY per user). Consequently, the personalized policy achieves a higher uplift per unit of marketing cost (0.0443% versus 0.0380%). These results demonstrate the value of personalizing incentives based on users' initial state variables.

The dynamic personalized policy based on BOWL without a budget constraint further improves performance. Relative to the static personalized policy, it achieves a slightly higher second purchase uplift (2.14% versus 2.09%) with a slightly lower average cost (46.92 versus 47.08 JPY), resulting in a modest improvement in cost efficiency. Thus, updating decisions based on users' evolving states yields additional, albeit modest, gains beyond static personalization.

More importantly, imposing a budget constraint substantially improves the cost efficiency of the dynamic BOWL policy. With a budget of 30 JPY, the second purchase uplift declines only moderately to 1.66%, while the average marketing cost falls from 46.92 to 33.39 JPY (see Table 5). As a result, the uplift per unit of marketing cost increases from 0.0456% to 0.0496%. Tightening the budget further to 20

Table 7: Offline Treatment Allocation on Test Set (Baseline Model)

	2 day	10 day	30 day	Total
Static Uniform Policy without Budget Constraint	88.99%	0.00%	0.00%	88.99%
Static Personalized Policy without Budget Constraint				
BOWL	76.25%	10.20%	1.45%	87.91%
Q-Learning	88.99%	0.00%	0.00%	88.99%
Dynamic Personalized Policy without Budget Constraint				
BOWL	78.93%	4.30%	4.39%	87.63%
Q-Learning	88.99%	0.00%	0.00%	88.99%
Dynamic Personalized Policy with Budget Constraint				
BOWL: Budget = 30	52.18%	7.65%	13.84%	73.67%
BOWL: Budget = 20	0.56%	30.13%	34.58%	65.28%
Q-Learning: Budget = 30	36.44%	6.81%	20.63%	63.87%
Q-Learning: Budget = 20	29.46%	17.05%	14.07%	60.57%

JPY increases this measure to 0.0580%. The budget-constrained Q-learning policies also substantially reduce marketing expenditure, although the associated retention uplift declines more sharply, so their gains in cost efficiency are more limited. Overall, the results show that the budget constraint allows the dynamic BOWL policy to substantially reduce incentive spending while preserving a large share of the retention gains.

The cost efficiency of our dynamic policies stems from withholding unnecessary coupons. In Table 7, we show each policy’s allocation of coupons to the three timings. The uniform policy sends coupons to 89% of the first-time buyers on day 2; 11% of users have already made a second purchase at that point. The static personalized policy based on BOWL allocates coupons to a nonnegligible fraction of users (10%) on day 10. Our dynamic policies, especially under a budget constraint, further allocates incentives to later timings. With the budget constraint at 30 JPY, the BOWL-based policy sends a coupon to 8% and 14% of users on days 10 and 30, respectively. With the budget constraint at 20 JPY, almost no user receives a coupon on day 2, and 30% and 35% receive one on days 10 and 30, respectively.

By taking the dynamics of consumer purchase behavior into consideration, our dynamic optimal policies save coupons that would otherwise be sent to users likely to make a purchase without incentives. The reserved coupons are sent later to other customers based on their changing responsiveness to the incentive, enhancing the cost performance. This advantage is more evident under budget constraints, where the strategy of mass-distributing coupons on day 2 is no longer feasible. This is in line with the intuition of why dynamic policies improve the performance, which is outlined in Section 4.2.

7.2 Extension

Next, we consider the extended model with three actions: sending an email with a coupon, sending only a non-incentive appreciation email, and sending nothing. We impose the same feasibility constraints on the action of sending a coupon as in the baseline model, while no feasibility constraints are imposed on the action of sending a non-incentive email. Thus, unlike in the baseline model, we are allowed to make personalized actions (either sending a no-incentive email or not) to the users who already made a second purchase or already received a coupon. We expect a potentially larger benefit of dynamic personalization in this extended setting.

We use the training set from the second experiment to learn two optimal dynamic policies with three actions: one without a budget constraint and one with a per-user budget of 90 JPY. This budget is set much higher than in the baseline setting because, after the first experiment, the company introduced a campaign encouraging users to spend more accumulated points, which mechanically increased coupon redemption and therefore the realized marketing cost. To learn these policies, we focus on BOWL using SGDC with L2 regularization and hinge loss, since it outperforms alternative algorithms in the baseline model.⁴² To examine the benefit of introducing the additional action of sending nothing, we also learn an optimal dynamic policy restricted to two actions: sending a coupon and sending a non-incentive email.

Tables 8 and 9 report the offline evaluation results on the test set. Unlike in the baseline analysis, the uplifts are measured relative to the no-email policy that never sends either an appreciation email or a coupon. Hence, the estimates capture the combined effects of both emails and financial incentives. The marketing cost now measures both coupon expenditures and the operational cost of sending emails, although the latter is economically negligible (less than 1 JPY per email). In addition to the three dynamic policies, we also evaluate a static uniform policy as a benchmark. On day 2, this policy sends a coupon to users who have not yet made a second purchase and sends a non-incentive email to all others. On days 10 and 30, it sends a non-incentive email to everyone.

Compared to the static uniform policy, both dynamic policies without a budget constraint achieve a higher second purchase probability with a lower average cost, resulting in a substantial improvement in cost efficiency. Between the two dynamic policies, the three-action policy slightly dominates the two-action policy: adding the option to send nothing increases the uplift from 6.41% to 6.55% while reducing average cost from 118.41 JPY to 113.78 JPY, improving the uplift per unit of marketing cost

⁴²See Appendices F and E.2 for the algorithm and training details in the extended setting with three actions.

Table 8: Offline Uplift Performance on Test Set (Extension)

Uplift (Relative to No-email Policy)	2nd	Cost (JPY)	Uplift/Cost
Static Uniform Policy without Budget Constraint			
2-Day-Later Incentive + No-Incentive email	6.00%	123.61	0.0486%
Dynamic Personalized Policy without Budget Constraint			
BOWL (Two Actions)	6.41%	118.41	0.0541%
BOWL (Three Actions)	6.55%	113.78	0.0576%
Dynamic Personalized Policy with Budget Constraint			
BOWL (Three Actions): Budget = 90	5.34%	81.95	0.0652%

Note: The uplift is the difference from the no-email policy in the second purchase probability within 2 months after the first purchase. The cost includes both coupon spending and the operational cost of sending emails incurred within 2 months after the first purchase.

Table 9: Offline Uplift Performance Comparison on Test Set (Extension)

Uplift Difference	2nd	Cost (JPY)
Static Uniform (2-Day-Later Incentive) vs. No email	6.00% (0.82)	123.61 (5.20)
BOWL (Two Actions) w/o BC vs. Static Uniform	0.40% (0.30)	−5.20 (2.15)
BOWL (Three Actions) w/o BC vs. Static Uniform	0.55% (0.55)	−9.83 (5.92)
BOWL (Three Actions) w/o BC vs. BOWL (Two Actions) w/o BC	0.15% (0.56)	−4.630 (5.894)
BOWL (Three Actions) w/o BC vs. BOWL (Three Actions): Budget = 90	1.21% (0.82)	31.83 (6.32)

Note: This table reports the differences between policies in the second purchase probability and average cost within 2 months after the first purchase. The standard errors are in parentheses. BC stands for Budget Constraint.

from 0.0541% to 0.0576%. Importantly, as in the baseline setting, imposing a budget constraint further improves the cost efficiency of the dynamic policy. With a budget of 90 JPY, although the second purchase uplift drops to 5.34%, the average cost falls to 81.95 JPY, yielding a significantly higher uplift per unit of marketing cost of 0.0652%.

Table 10 reports how the learned policies allocate financial incentives and appreciation emails over time. Without a budget constraint, the three-action dynamic policy sends coupons aggressively soon after the first purchase, assigning incentives to 74% of users on day 2, while allocating relatively few coupons at later timings. The benchmark two-action dynamic policy exhibits an even stronger concentration on early incentives, assigning coupons to 80% of users on day 2.

By contrast, the budget-constrained policy spreads incentives more gradually over time. The policy sends coupons to 44% of users on day 2, 3% on day 10, and 20% on day 30. At the same time, it actively uses non-incentive appreciation emails, assigning them to 33%, 39%, and 58% of users on days 2, 10, and 30, respectively. These patterns suggest that the constrained policy strategically substitutes lower-cost appreciation emails for coupons, especially for users who are less likely to require immediate financial incentives.

Table 10: Offline Treatment Allocation on Test Set (Extension)

	Incentive Delivery Share				Non-Incentive Delivery Share		
	2 day	10 day	30 day	Total	2 day	10 day	30 day
Static Uniform (2-Day-Later Incentive)	89.75%	0.00%	0.00%	89.75%	10.25%	100.00%	100.00%
BOWL (Two Actions) w/o BC	80.25%	4.70%	3.95%	88.90%	20.75%	96.29%	97.05%
BOWL (Three Actions) w/o BC	73.68%	0.00%	5.15%	78.82%	21.48%	89.55%	51.40%
BOWL (Three Actions): Budget = 90	43.80%	2.89%	20.48%	67.16%	32.65%	38.70%	58.05%

Note: BC stands for Budget Constraint.

These allocation patterns provide intuition for why the budget-constrained policy substantially reduces marketing expenditures while maintaining positive retention effects. The policy can save costs by avoiding immediate incentives for users who are likely to make repeat purchases even without coupons. At the same time, delaying incentives for a subset of users allows the policy to target customers whose responsiveness to financial incentives is stronger at later stages of the customer lifecycle. Our results therefore suggest that both the timing of incentives and the distinction between financial and non-financial interventions are important dimensions of effective retention management.

8 Online Evaluation Results

Finally, the company evaluated the performance of the learned policies through two online A/B tests. The first experiment evaluated the policies developed for the baseline model. First-time buyers were randomly assigned to one of five candidate policies: a static personalized policy based on BOWL, a dynamic personalized policy based on Q-learning without a budget constraint, dynamic personalized policies based on BOWL with and without a budget constraint, and the benchmark no-email policy. The budget-constrained policy used a per-user budget of 30 JPY. The second experiment evaluated the policy developed for the extension model. In this experiment, first-time buyers were randomly assigned to either the budget-constrained dynamic BOWL policy, with a budget of 127.47 JPY, or the benchmark no-email policy. The first experiment was conducted in the fall of 2021, and the second in the summer of 2022. Each experiment ran for several months.

Table 11 reports the uplift in the second purchase probability and the corresponding uplift per unit of marketing cost within 2 months after the first purchase, relative to the no-email policy.⁴³ For the baseline model, the online results closely mirror the offline evaluation. Without a budget constraint,

⁴³For the baseline model, the realized marketing cost in the online experiment is substantially higher than in the offline evaluation because, after the first experiment, the company introduced a campaign encouraging customers to redeem accumulated coupon points.

Table 11: Online Uplift Performance

Uplift (Relative to No-email Policy)	2nd	Cost (JPY)	Uplift/Cost
Online Experiment 1: Baseline Model			
Static Personalized Policy without Budget Constraint			
BOWL	5.97%	122.8	0.0486%
Dynamic Personalized Policy without Budget Constraint			
BOWL	5.87%	120.9	0.0486%
Q-Learning	5.16%	107.2	0.0481%
Dynamic Personalized Policy with Budget Constraint			
BOWL: Budget = 30	4.55%	94.4	0.0482%
Online Experiment 2: Extension Model			
Dynamic Personalized Policy with Budget Constraint			
BOWL: Budget = 127.47	3.26%	93.9	0.0347%

Note: The first panel reports the results from the online A/B test for the baseline model, which compares the listed policies against the company’s no-email policy. The second panel reports a separate online A/B test conducted for the extension model. Because the two experiments were conducted at different times and on different user populations, the performance measures should not be compared across panels. Uplift is the increase in the second purchase probability relative to the no-email policy. Uplift/Cost reports the increase in the second purchase probability per yen of marketing cost.

the static and dynamic policies based on BOWL achieve similar improvements in the second purchase probability and both outperform the dynamic policy based on Q-learning. The budget-constrained BOWL policy generates a somewhat smaller retention uplift but achieves comparable uplift per unit of marketing cost while substantially reducing marketing expenditures. These findings confirm that the offline policy evaluation accurately predicts out-of-sample performance in a live environment.

The second online experiment evaluates the extension model. The budget-constrained dynamic policy increases the second purchase probability by 3.26%, with an uplift per unit of marketing cost of 0.0347%. Although both the retention uplift and cost efficiency are lower than in the offline evaluation, the policy continues to generate economically meaningful improvements over the no-email policy. We suspect that this difference reflects changes in the economic environment: by the time of the second experiment, Japan had largely emerged from the pandemic, reducing consumers’ demand for online shopping and their responsiveness to promotional incentives.

9 Conclusion

This paper proposes a method to learn an optimal dynamic targeting policy for customer retention management when there is an inter-temporal budget constraint. Dynamic policies are crucial for customer retention management as the states of consumers such as an intention for future purchases inherently evolve over time. Moreover, since most marketing campaigns have certain budget ceilings,

it is practically important to consider a cost-efficient way to target consumers.

To do so, we extend the existing methods of estimating optimal DTRs to account for an intertemporal budget constraint. In particular, we characterize an optimal DTR under a budget constraint and provide an algorithm to learn it using experimental data. Our algorithm incorporates two methods, Q-learning and BOWL, into a constrained optimization problem.

Our empirical application is a large online e-commerce platform in Japan. We personalize whether to send a “thank you” message and a coupon to those who make their first purchase to urge second purchases. The company runs a series of large-scale randomized experiments with more than 100,000 monthly new buyers, which allow us to estimate dynamic personalized policies in the offline setting before the company actually implements them for the entire population.

Our offline evaluation shows that the optimal dynamic policies, particularly those with a budget constraint, are highly cost-effective. Moreover, the company tests our proposed policies on their platform, and the online evaluation results confirm that our policies are highly effective.

Our paper contributes to the literature on retention management by providing a method to achieve cost-effective dynamic policies. There are, however, some limitations. First, in our application, the company fixes three days when emails and coupons can be sent to customers. It is important and interesting to extend the method to the case where the company can choose the timing more flexibly beyond what’s observed in the data. Second, while we focus on encouraging the second purchase because it is the firm’s primary business objective and a strong predictor of future engagement, future work could extend the framework to optimize long-run customer lifetime value. Such an extension would require balancing short-run retention gains against the long-run effects of promotional interventions on future purchasing behavior and profitability.

References

- Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning*, pages 22–31, 2017.
- Eva Ascarza. Retention futility: Targeting High-Risk customers might be ineffective. *Journal of Marketing Research*, 55(1):80–98, February 2018.
- Eva Ascarza, Scott Neslin, Oded Netzer, et al. In pursuit of enhanced customer retention management: Review, key issues, and future directions. *Customer Needs and Solution*, 5:65–81, 2018.

- Eva Ascarza, Michael Ross, and Bruce Hardie. Why you aren't getting more from your marketing ai. *Harvard Business Review*, 99(4):48–54, 2021.
- Susan Athey and Stefan Wager. Policy learning with observational data. *Econometrica*, 89(1):133–161, 2021.
- Susan Athey, Raj Chetty, Guido W Imbens, and Hyunseung Kang. The surrogate index: Combining short-term proxies to estimate long-term treatment effects more rapidly and precisely. *The Review of Economic Studies*, 93(4):2284–2312, 07 2026.
- Debopam Bhattacharya and Pascaline Dupas. Inferring welfare maximizing treatment assignment under budget constraints. *Journal of Econometrics*, 167(1):168–196, March 2012.
- Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/Debiased Machine Learning for Treatment and Structural Parameters. *The Econometrics Journal*, 21(1):C1–C68, 01 2018. doi: 10.1111/ectj.12097.
- Øystein Daljord, Carl F Mela, Jason M. T. Roos, Jim Sprigg, and Song Yao. The design and targeting of compliance promotions. *Marketing Science*, 42(5):866–891, 2023.
- Peter S. Fader and Bruce G. S. Hardie. Customer-base valuation in a contractual setting: The perils of ignoring heterogeneity. *Marketing Science*, 29(1):85–93, 2010. doi: 10.1287/mksc.1080.0482.
- Peter S. Fader and Bruce G.S. Hardie. How to project customer retention. *Journal of Interactive Marketing*, 21(1):76–90, 2007. ISSN 1094-9968. doi: <https://doi.org/10.1002/dir.20074>.
- Mehrdad Farajtabar, Yinlam Chow, and Mohammad Ghavamzadeh. More robust doubly robust off-policy evaluation. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1447–1456, 2018.
- Arun Gopalakrishnan and Young-Hoon Park. The right timing and the right offer for purchase conversion? evidence from a field experiment on shopping cart interventions. *Available at SSRN: <https://ssrn.com/abstract=3896585>*, 2024.
- Günter J Hitsch, Sanjog Misra, and Walter W. Zhang. Heterogeneous treatment effects and optimal targeting policy evaluation. *Quantitative Marketing and Economics*, 22:115–168, 2024.

- Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 652–661, 2016.
- Nathan Kallus and Masatoshi Uehara. Double Reinforcement Learning for Efficient Off-Policy Evaluation in Markov Decision Processes. In *Proceedings of the 37th International Conference on Machine Learning*, pages 5078–5088, 2020.
- Wreetabrata Kar, Viswanathan Swaminathan, and Paulo Albuquerque. Selection and ordering of linear online video ads. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 203–210, September 2015.
- Yewom Kim. Customer retention under imperfect information. Available at SSRN: <https://ssrn.com/abstract=3709043>, 2022.
- Toru Kitagawa and Aleksey Tetenov. Who should be treated? empirical welfare maximization methods for treatment choice. *Econometrica*, 86(2):591–616, 2018.
- Qingfeng Lan, Yangchen Pan, Alona Fyshe, and Martha White. Maxmin q-learning: Controlling the estimation bias of q-learning. In *International Conference on Learning Representations (ICLR 2020)*, 2020. URL <https://openreview.net/forum?id=BkgOu3Etwr>.
- Hoang M. Le, Cameron Voloshin, and Yisong Yue. Batch policy learning under constraints. In *Proceedings of the 36th International Conference on Machine Learning*, pages 3703–3712, 2019.
- Aurélien Lemmens and Sunil Gupta. Managing churn to maximize profits. *Marketing Science*, 39(5): 956–973, 2020.
- Xiao Liu. Dynamic coupon targeting using batch deep reinforcement learning: An application to livestream shopping. *Marketing Science*, 42(4):637–658, 2023.
- Alexander R Luedtke and Mark J van der Laan. Statistical inference for the mean outcome under a possibly non-unique optimal treatment strategy. *Annals of Statistics*, 44(2):713–742, 2016.
- S A Murphy. Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 65(2):331–355, May 2003.
- S. A. Murphy. An experimental design for the development of adaptive treatment strategies. *Statistics in Medicine*, 24(10):1455–1481, 2005a.

- Susan A. Murphy. A generalization error for q-learning. *Journal of Machine Learning Research*, 6(37):1073–1097, 2005b.
- Susan A Murphy, Kevin G Lynch, David Oslin, James R McKay, and Tom TenHave. Developing adaptive treatment strategies in substance abuse research. *Drug Alcohol Dependence*, 88(Suppl 2):S24–30, May 2007.
- Scott A. Neslin, Sunil Gupta, Wagner Kamakura, Junxiang Lu, and Charlotte H. Mason. Defection detection: Measuring and understanding the predictive accuracy of customer churn models. *Journal of Marketing Research*, 43(2):204–211, 2006. doi: 10.1509/jmkr.43.2.204. URL <https://doi.org/10.1509/jmkr.43.2.204>.
- Scott A Neslin, Gail Ayala Taylor, Kimberly D Grantham, and Kimberly R McNeil. Overcoming the “re-cency trap” in customer relationship management. *Journal of the Academy of Marketing Science*, 41(3):320–337, May 2013.
- Xinkun Nie, Emma Brunskill, and Stefan Wager. Learning When-to-Treat policies. *Journal American Statistical Association*, 116(533):392–409, January 2021.
- Fernando Nogueira. Bayesian Optimization: Open Source Constrained Global Optimization Tool for Python. <https://github.com/fmfn/BayesianOptimization>, 2014.
- Doina Precup, Richard S Sutton, and Satinder P Singh. Eligibility traces for Off-Policy policy evaluation. In *Proceedings of the 17th International Conference on Machine Learning*, pages 759–766, June 2000.
- Hongxiang Qiu, Marco Carone, and Alex Luedtke. Individualized treatment rules under stochastic treatment cost constraints. *Journal of Causal Inference*, 10(1):480–493, December 2022.
- Omid Rafieian. Optimizing user engagement through adaptive ad sequencing. *Marketing Science*, 42(5):910–933, 2023.
- Shosei Sakaguchi. Estimation of optimal dynamic treatment assignment rules under policy constraints. *Quantitative Economics*, 16(3):981–1022, 2025.
- Duncan Simester, Artem Timoshenko, and Spyros I. Zoumpoulis. Efficiently evaluating targeting policies: Improving on champion vs. challenger experiments. *Management Science*, 66(8):3412–3424, 2020. doi: 10.1287/mnsc.2019.3379.

- Alex Strehl, John Langford, Lihong Li, and Sham M Kakade. Learning from Logged Implicit Exploration Data. In *Advances in Neural Information Processing Systems 23*, pages 2217–2225, 2010.
- Liyang Sun. Empirical welfare maximization with constraints. *Journal of Econometrics*, 253:106169, 2026.
- Philip Thomas and Emma Brunskill. Data-efficient off-policy policy evaluation for reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 2139–2148, 2016.
- Wen Wang, Beibei Li, Xueming Luo, and Xiaoyi Wang. Deep reinforcement learning for sequential targeting. *Management Science*, 69(9):5439–5460, December 2023.
- C. J. C. H. Watkins. *Learning from Delayed Rewards*. PhD thesis, King’s College, Cambridge, 1989.
- Jeremy Yang, Dean Eckles, Paramveer S Dhillon, and Sinan Aral. Targeting for long-term outcomes. *Management Science*, 2023.
- Hema Yoganarasimhan, Ebrahim Barzegary, and Abhishek Pani. Design and evaluation of optimal free trials. *Management Science*, 69(6):3220–3240, 2023.
- Baqun Zhang, Anastasios A Tsiatis, Eric B Laber, and Marie Davidian. Robust estimation of optimal dynamic treatment regimes for sequential treatment decisions. *Biometrika*, 100(3):681–694, May 2013.
- Ying-Qi Zhao, Donglin Zeng, Eric B Laber, and Michael R Kosorok. New statistical learning methods for estimating optimal dynamic treatment regimes. *Journal of the American Statistical Association*, 110(510):583–598, 2015.
- Yingqi Zhao, Donglin Zeng, A John Rush, and Michael R Kosorok. Estimating individualized treatment rules using outcome weighted learning. *Journal of the American Statistical Association*, 107(449): 1106–1118, September 2012.
- Yufan Zhao, Michael R Kosorok, and Donglin Zeng. Reinforcement learning design for cancer clinical trials. *Statistics in Medicine*, 28(26):3294–3315, November 2009.
- Zhengyuan Zhou, Susan Athey, and Stefan Wager. Offline multi-action policy learning: Generalization and optimization. *Operations Research*, 71(1):148–183, 2023.

Online Appendix

A Illustrating Benefits of Dynamic Policies through a Simple Model

To understand the benefit of dynamic policies relative to static policies in our setup, this section provides a simple model of a dynamic environment with three periods and three types of customers. We show that an appropriate parameter profile yields a positive dynamic gain from adopting a dynamic policy (DTR in the general environment) that benefits from observing updated states.

A.1 Setup

Firm actions. The firm chooses whether to send a coupon to each customer in periods $t = 1, 2, 3$. Let $A_t \in \{0, 1\}$ indicate whether the firm sends a coupon at time t . We impose the feasibility constraint that the firm can send a coupon to each customer at most once over time, so that

$$A_1 + A_2 + A_3 \leq 1.$$

Latent customer types. There are three types of customers, $T \in \{S, E, L\}$, with the following type probabilities:

$$\Pr(T = S) = \frac{1}{2}, \quad \Pr(T = E) = \Pr(T = L) = \frac{1}{4}.$$

Different types have different purchase probabilities conditional on the firm's coupon decisions. Let $Y_t \in \{0, 1\}$ indicate whether the customer purchases in period t and $Y \in \{0, 1\}$ indicate whether the customer purchases by the end of period 3. For simplicity, assume that each customer purchases at most once over time, so that $Y = Y_1 + Y_2 + Y_3$. Suppose each type's demand is specified as follows:

- If $T = S$ (safe), the customer purchases in period 1 regardless of coupons:

$$\Pr(Y = 1 \mid A_1 = a_1, A_2 = a_2, A_3 = a_3, T = S) = \Pr(Y_1 = 1 \mid A_1 = a_1, A_2 = a_2, A_3 = a_3, T = S) = 1$$

for all (a_1, a_2, a_3) . Hence, if the firm sends a coupon to a safe-type customer, it is wasted.

- If $T = E$ (early-responsive), a purchase can occur only in the period a coupon is sent, and the effec-

tiveness of the coupon declines with time:

$$\Pr(Y = 1 \mid A_1 = 1, T = E) = \Pr(Y_1 = 1 \mid A_1 = 1, T = E) = 1,$$

$$\Pr(Y = 1 \mid A_2 = 1, T = E) = \Pr(Y_2 = 1 \mid A_2 = 1, T = E) = m,$$

$$\Pr(Y = 1 \mid A_3 = 1, T = E) = \Pr(Y_3 = 1 \mid A_3 = 1, T = E) = r,$$

$$\Pr(Y = 1 \mid A_1 = A_2 = A_3 = 0, T = E) = 0.$$

Here $1 > m > r > 0$.

- If $T = L$ (late-responsive), a purchase can occur only in the period a coupon is sent, and the effectiveness of the coupon increases with time:

$$\Pr(Y = 1 \mid A_1 = 1, T = L) = \Pr(Y_1 = 1 \mid A_1 = 1, T = L) = r,$$

$$\Pr(Y = 1 \mid A_2 = 1, T = L) = \Pr(Y_2 = 1 \mid A_2 = 1, T = L) = m,$$

$$\Pr(Y = 1 \mid A_3 = 1, T = L) = \Pr(Y_3 = 1 \mid A_3 = 1, T = L) = 1,$$

$$\Pr(Y = 1 \mid A_1 = A_2 = A_3 = 0, T = L) = 0.$$

Information structure. The firm has only imperfect information about the customer type. Information regarding customer type is revealed through purchase behavior or other observed actions over time. This corresponds to observing H_t in the general model presented in Section 4. Specifically:

- At the beginning of period $t = 1$, the firm has no information about the customer type.
- At the beginning of period $t = 2$, the firm observes Y_1 , which indicates whether the customer purchased in period $t = 1$. Also, if the customer did not receive a coupon at $t = 1$ and has not yet purchased, the firm observes a noisy signal $X \in \{E, L\}$ satisfying

$$\Pr(X = E \mid T = E) = \Pr(X = L \mid T = L) = q,$$

$$\Pr(X = L \mid T = E) = \Pr(X = E \mid T = L) = 1 - q,$$

with $q \in [1/2, 1]$. Recall that any safe customer buys in period $t = 1$ with or without a coupon. If the customer has not yet received a coupon or purchased, their type must be $T \in \{E, L\}$. Note

that, since $\Pr(T = E) = \Pr(T = L)$, by Bayes' rule,

$$\begin{aligned}\Pr(T = E \mid X = E) &= \Pr(T = L \mid X = L) = q, \\ \Pr(T = L \mid X = E) &= \Pr(T = E \mid X = L) = 1 - q.\end{aligned}$$

Hence, X is informative about the customer type when $q > 1/2$.

- At the beginning of period $t = 3$, the firm observes Y_2 , which indicates whether the customer purchased in period $t = 2$. The firm receives no additional information.

Under this information structure, the information available to the firm at the beginning of each period is summarized as follows:

$$H_1 = \emptyset \text{ (no information),} \quad H_2 = \begin{cases} (A_1, Y_1) & \text{if } A_1 = 1 \text{ or } Y_1 = 1, \\ (A_1, Y_1, X) & \text{if } A_1 = Y_1 = 0, \end{cases} \quad H_3 = (H_2, A_2, Y_2).$$

Objective and policies. The firm attempts to maximize the purchase probability, $\mathbb{E}[Y]$, by optimizing its policy (DTR), which is either static or dynamic.

- **Static policy:** A static policy must commit at $t = 1$ to a full action profile (A_1, A_2, A_3) (or a probability distribution over action profiles) subject to the feasibility and budget constraints. It cannot condition on X , since X is observed only at the beginning of period $t = 2$.
- **Dynamic policy:** A dynamic policy chooses action A_t (or a probability distribution over actions) given history H_t for each period $t = 1, 2, 3$, subject to the feasibility and budget constraints.

We assume that the firm knows the optimal static and dynamic policies. For example, if the firm conducted an experiment that randomized actions in the past, it could use the experimental data to estimate the purchase probability for any candidate policy, and then choose the best-performing one. Here, we do not model how the firm obtains optimal static and dynamic policies; instead, we focus on the structures of these policies and their comparison.

Budget constraint. We impose the constraint that the expected number of coupons per customer does not exceed B :

$$\mathbb{E}[A_1 + A_2 + A_3] \leq B.$$

Since $A_1 + A_2 + A_3 \leq 1$ under the feasibility constraint, $B = 1$ corresponds to the case with no binding budget constraint. We compare cases with and without a budget constraint.

A.2 Gain from Adopting a Dynamic Policy

The following proposition shows the positive gain from adopting a dynamic policy.

Proposition A.1. *Assume that the parameter profile (r, m, q) satisfies $qr + (1 - q) < m < (1 - q)r + q$. Then, the optimal dynamic policy achieves a higher purchase probability than the optimal static policy, regardless of whether the budget is constrained or unconstrained. The gain from the dynamic policy is higher when the budget is constrained ($B \in [1/4, 1/2]$) than when it is unconstrained ($B = 1$).*

As shown later, $qr + (1 - q)$ and $(1 - q)r + q$ are the probabilities of purchase when the firm does not send a coupon in period $t = 2$ to customers with signal $X = E$ and $X = L$, respectively. The parameter m is the probability of purchase when the firm sends a coupon in period $t = 2$ to customers with either signal value. The condition $qr + (1 - q) < m < (1 - q)r + q$ means that it is optimal for the firm to change the timing of the coupon based on the signal, leading to a dynamic gain. Given the coupon effectiveness parameters (r, m) , this condition tends to hold if q is larger. This implies that if the signal is sufficiently informative, a dynamic gain exists.

The budget constraint amplifies this dynamic advantage through two mechanisms. First, the dynamic policy enables the firm to avoid spending coupons on safe customers who buy in period 1 anyway. Second, after observing the signal, the dynamic policy allows the firm to prioritize late-responsive customers, who yield a higher return per coupon, over early-responsive customers when distributing the budget.

Below, we show Proposition A.1 and present the optimal static and dynamic policies with and without a budget constraint.

A.2.1 Unconstrained Budget ($B = 1$)

Optimal static policy. An optimal static policy chooses the time to send a coupon that yields the highest purchase probability $\mathbb{E}[Y]$ evaluated at the beginning of period 1. Under the type structure described above, the purchase probability for each static policy is calculated as follows:

- Coupon at $t = 1$:

$$\mathbb{E}[Y \mid A_1 = 1] = \sum_{t \in \{S, E, L\}} \Pr(T = t) \Pr(Y = 1 \mid A_1 = 1, T = t) = \frac{1}{2} + \frac{1}{4} + \frac{1}{4}r = \frac{1}{2} + \frac{1+r}{4}.$$

- Coupon at $t = 2$:

$$\mathbb{E}[Y | A_2 = 1] = \sum_{t \in \{S, E, L\}} \Pr(T = t) \Pr(Y = 1 | A_2 = 1, T = t) = \frac{1}{2} + \frac{1}{4}m + \frac{1}{4}m = \frac{1}{2} + \frac{m}{2}.$$

- Coupon at $t = 3$:

$$\mathbb{E}[Y | A_3 = 1] = \sum_{t \in \{S, E, L\}} \Pr(T = t) \Pr(Y = 1 | A_3 = 1, T = t) = \frac{1}{2} + \frac{1}{4}r + \frac{1}{4} = \frac{1}{2} + \frac{1+r}{4}.$$

Hence, an optimal static policy sends a coupon at $t = 1$ or 3 if $\frac{1+r}{2} \geq m$, and at $t = 2$ if $\frac{1+r}{2} < m$. The purchase probability under this policy is given by

$$\mathbb{E}_{\text{stat}}[Y] = \begin{cases} \frac{1}{2} + \frac{1+r}{4} & \text{if } \frac{1+r}{2} \geq m, \\ \frac{1}{2} + \frac{m}{2} & \text{if } \frac{1+r}{2} < m. \end{cases}$$

Optimal dynamic policy. An optimal dynamic policy can be derived by optimizing the action given history via backward induction from $t = 3$. In each period, we only need to consider customers who have not yet received a coupon or purchased by the beginning of the period, as sending a coupon to other customers is either infeasible or does not increase the outcome Y .

- **Period $t = 3$.** Since a coupon is effective for both types E and L , it is optimal to send a coupon if the customer has not yet received a coupon or purchased by the beginning of period 3 (i.e., $A_1 = A_2 = Y_1 = Y_2 = 0$), regardless of the signal X . The purchase probability under this rule depends on the signal X as follows:

$$\mathbb{E}[Y | X = E, A_3 = 1] = \sum_{t \in \{E, L\}} \Pr(T = t | X = E) \Pr(Y = 1 | A_3 = 1, T = t) = qr + (1-q),$$

$$\mathbb{E}[Y | X = L, A_3 = 1] = \sum_{t \in \{E, L\}} \Pr(T = t | X = L) \Pr(Y = 1 | A_3 = 1, T = t) = (1-q)r + q.$$

Since $q \in [1/2, 1]$, $\mathbb{E}[Y | X = E, A_3 = 1] \leq \mathbb{E}[Y | X = L, A_3 = 1]$; the coupon is more effective when the signal indicates that the customer is more likely to be late-responsive.

- **Period $t = 2$.** Consider a customer who has not yet received a coupon or purchased by the beginning of period 2 (i.e., $A_1 = Y_1 = 0$). Since the effectiveness of sending a coupon at $t = 2$ is the

same for types E and L , if the firm sends a coupon at $t = 2$, the purchase probability is given by

$$\mathbb{E}[Y \mid X, A_2 = 1] = m,$$

regardless of $X \in \{E, L\}$. On the other hand, if the firm waits at $t = 2$, the customer does not purchase at $t = 2$, and the optimal period-3 rule sends a coupon at $t = 3$. Hence, the purchase probability is given by

$$\mathbb{E}[Y \mid X = E, A_2 = 0, A_3 = 1] = qr + (1 - q),$$

$$\mathbb{E}[Y \mid X = L, A_2 = 0, A_3 = 1] = (1 - q)r + q.$$

Consequently, an optimal period-2 rule is as follows: if $m \geq (1 - q)r + q$, it sends a coupon regardless of X ; if $qr + (1 - q) < m < (1 - q)r + q$, it sends a coupon if $X = E$ and does not if $X = L$; and if $m \leq qr + (1 - q)$, it does not send a coupon regardless of X .

- **Period $t = 1$.** If the firm sends a coupon at $t = 1$, the purchase probability is identical to that of the static policy of sending a coupon at $t = 1$:

$$\mathbb{E}[Y \mid A_1 = 1] = \frac{1}{2} + \frac{1 + r}{4}.$$

On the other hand, if the firm waits at $t = 1$, the customer purchases at $t = 1$ if $T = S$ and does not otherwise. In the latter case, suppose that the firm waits at $t = 2$ and sends a coupon at $t = 3$ regardless of signal X . While this period-2 rule may be suboptimal, it provides a useful lower bound on the purchase probability. Under this rule, the purchase probability is

$$\begin{aligned} & \Pr(T = S)\mathbb{E}[Y \mid A_1 = 0, T = S] + \sum_{t \in \{E, L\}} \Pr(T = t)\mathbb{E}[Y \mid A_1 = A_2 = 0, A_3 = 1, T = t] \\ &= \frac{1}{2} + \frac{1}{4}r + \frac{1}{4} = \frac{1}{2} + \frac{1 + r}{4} = \mathbb{E}[Y \mid A_1 = 1]. \end{aligned}$$

If the firm waits at $t = 1$ and uses the optimal period-2 rule instead at $t = 2$, the purchase probability is no smaller than that under the above suboptimal period-2 rule, and thus no smaller than $\mathbb{E}[Y \mid A_1 = 1]$. Therefore, it is optimal to wait at $t = 1$.

An optimal dynamic policy is summarized as follows:

- In period $t = 1$, it does not send a coupon.

- In period $t = 2$, it may send a coupon to customers who have not yet received a coupon or purchased by the beginning of period 2, depending on the signal. Specifically, if $m \geq (1-q)r + q$, it sends a coupon regardless of X ; if $qr + (1-q) < m < (1-q)r + q$, it sends a coupon if $X = E$ and does not if $X = L$; and if $m \leq qr + (1-q)$, it does not send a coupon regardless of X .
- In period $t = 3$, it sends a coupon to any customer who has not yet received a coupon or purchased by the beginning of period 3.

Optimal dynamic policy vs. static policy. Note that this optimal dynamic policy is equivalent to the static policy of sending a coupon at $t = 2$ if $m \geq (1-q)r + q$, and equivalent to the static policy of sending a coupon at $t = 3$ if $m \leq qr + (1-q)$. In these cases, there is no value in accounting for dynamics, as simply allocating a coupon at $t = 2$ or $t = 3$ remains optimal even after the firm acquires the signal. However, the optimal dynamic policy differs from any static policy if

$$qr + (1-q) < m < (1-q)r + q. \quad (\text{A.1})$$

Equation (A.1) implies that the signal is sufficiently informative (q is sufficiently large) for the firm to personalize coupon timing based on the signal. Under this condition, the purchase probability under the optimal dynamic policy is given by

$$\begin{aligned} \mathbb{E}_{\text{dyn}}[Y] &= \Pr(T = S) + \Pr(T \in \{E, L\}, X = E)\mathbb{E}[Y \mid X = E, A_2 = 1] + \Pr(T \in \{E, L\}, X = L)\mathbb{E}[Y \mid X = L, A_3 = 1] \\ &= \frac{1}{2} + \frac{1}{4}m + \frac{1}{4}((1-q)r + q) = \frac{1}{2} + \frac{m + (1-q)r + q}{4}. \end{aligned}$$

The dynamic-static gap is then given by

$$\Delta = \mathbb{E}_{\text{dyn}}[Y] - \mathbb{E}_{\text{stat}}[Y] = \begin{cases} \frac{m - (qr + (1-q))}{4} & \text{if } \frac{1+r}{2} \geq m, \\ \frac{-m + (1-q)r + q}{4} & \text{if } \frac{1+r}{2} < m. \end{cases}$$

Thus, $\Delta > 0$ for any (r, m, q) satisfying (A.1). Given r and q , the gap is maximized when $m = \frac{1+r}{2}$.

Additionally, the optimal dynamic policy sends a coupon if the customer type is either E or L , and refrains if the type is S . Hence, the expected number of coupons per customer is

$$\mathbb{E}_{\text{dyn}}[A_1 + A_2 + A_3] = \Pr(T \in \{E, L\}) = \frac{1}{2}.$$

This stands in sharp contrast to any static policy that commits to sending a coupon at the start of $t = 1$. When $B \geq 1/2$, the budget constraint is non-binding, and the dynamic policy above is also the constrained optimal policy.

A.2.2 Constrained Budget ($B \in [1/4, 1/2]$)

Now, consider the case where $B \leq 1/2$, in which the unconstrained optimal dynamic policy violates the budget constraint. To keep the exposition simple, we focus on $B \in [1/4, 1/2]$.

Optimal static policy. Under a budget constraint, a static policy must restrict the fraction of customers receiving a coupon to B . The optimal timing remains identical to that in the unconstrained case. Hence, an optimal static policy sends a coupon to $100B\%$ of customers (or sends a coupon to any given customer with probability B) at $t = 1$ or 3 if $\frac{1+r}{2} \geq m$, and at $t = 2$ if $\frac{1+r}{2} < m$. The purchase probability under this policy is given by

$$\mathbb{E}_{\text{stat,constrained}}[Y] = \begin{cases} \frac{1}{2} + \frac{(1+r)B}{4} & \text{if } \frac{1+r}{2} \geq m, \\ \frac{1}{2} + \frac{mB}{2} & \text{if } \frac{1+r}{2} < m. \end{cases}$$

Optimal dynamic policy. Under a budget constraint, an optimal dynamic policy cannot be derived using simple backward induction due to the inter-temporal nature of the constraint. To derive the optimal policy, we focus on the region where (A.1) holds. First, note that it is suboptimal to send a coupon to a positive fraction of customers at $t = 1$. To see this, suppose a fraction $p \in (0, B]$ of customers receives a coupon at $t = 1$. These customers yield a purchase probability of $\mathbb{E}[Y \mid A_1 = 1]$ per person. Alternatively, if a fraction p of customers is assigned coupons according to the unconstrained optimal dynamic policy, they yield a purchase probability of $\mathbb{E}_{\text{dyn}}[Y]$ per person. Because $\mathbb{E}_{\text{dyn}}[Y] > \mathbb{E}[Y \mid A_1 = 1]$, sending coupons to any positive fraction of customers at $t = 1$ is suboptimal. Thus, a constrained optimal dynamic policy waits at $t = 1$.

Next, consider the optimal rule for periods 2 and 3 for customers who have neither received a coupon nor purchased by the start of period 2. As shown in the analysis for the unconstrained case, if the signal is $X = E$, sending a coupon at $t = 2$ yields a higher purchase probability than waiting until $t = 3$, while consuming the exact same budget. Thus, waiting at $t = 2$ when $X = E$ is suboptimal even under a budget constraint. Similarly, sending a coupon at $t = 2$ when $X = L$ is suboptimal. Therefore, we only need to consider rules that send coupons to a fraction p_E of customers at $t = 2$ when $X = E$, and to a fraction p_L of customers at $t = 3$ when $X = L$. We optimize p_E and p_L to maximize the pur-

chase probability subject to the budget constraint. The policy parameterized by p_E and p_L yields the purchase probability:

$$\begin{aligned}\mathbb{E}_{p_E, p_L}[Y] &= \Pr(T = S) + \Pr(T \in \{E, L\}, X = E)\mathbb{E}[Y | X = E, A_2 = 1]p_E \\ &\quad + \Pr(T \in \{E, L\}, X = L)\mathbb{E}[Y | X = L, A_3 = 1]p_L \\ &= \frac{1}{2} + \frac{1}{4}mp_E + \frac{1}{4}((1-q)r + q)p_L = \frac{1}{2} + \frac{mp_E + ((1-q)r + q)p_L}{4}.\end{aligned}$$

The expected number of coupons per customer is

$$\mathbb{E}_{p_E, p_L}[A_1 + A_2 + A_3] = \Pr(T \in \{E, L\}, X = E)p_E + \Pr(T \in \{E, L\}, X = L)p_L = \frac{p_E + p_L}{4}.$$

Since $m < (1-q)r + q$, spending budget on customers with $X = L$ yields a higher marginal purchase probability than on customers with $X = E$. Therefore, the optimal policy sets $p_L^* = 1$ and allocates the remaining budget $4B - 1$ to customers with $X = E$, so $p_E^* = 4B - 1$. Note that the condition $B \geq 1/4$ ensures that $p_E^* \geq 0$, so that all customers with $X = L$ receive coupons before remaining coupons are allocated to customers with $X = E$.

Thus, if $qr + (1-q) < m < (1-q)r + q$ and $B \in [1/4, 1/2]$, the constrained optimal dynamic policy is structured as follows:

- In period $t = 1$, it does not send a coupon.
- In period $t = 2$, it sends a coupon to $100(4B - 1)\%$ of customers with $X = E$ and sends no coupons to customers with $X = L$.
- In period $t = 3$, it sends a coupon to all customers with $X = L$ and sends no coupons to customers with $X = E$.

The resulting purchase probability under this policy is

$$\mathbb{E}_{\text{dyn, constrained}}[Y] = \frac{1}{2} + \frac{m(4B - 1) + (1-q)r + q}{4}.$$

Optimal dynamic policy vs. static policy. The dynamic-static gap is

$$\begin{aligned}\Delta_{\text{constrained}} &= \mathbb{E}_{\text{dyn,constrained}}[Y] - \mathbb{E}_{\text{stat,constrained}}[Y] \\ &= \begin{cases} \frac{(1+r)B}{4} + \left(\frac{1+r}{2} - m\right)\left(\frac{1}{2} - B\right) + \frac{m - (qr + (1-q))}{4} & \text{if } \frac{1+r}{2} \geq m, \\ \frac{mB}{2} + \frac{-m + (1-q)r + q}{4} & \text{if } \frac{1+r}{2} < m. \end{cases}\end{aligned}$$

Thus, $\Delta_{\text{constrained}} > 0$ for any (r, m, q) satisfying (A.1). Moreover, the dynamic-static gap is strictly larger under a constrained budget than under an unconstrained budget. This captures two additional benefits of the dynamic policy: it avoids wasting scarce coupons on safe customers, and it prioritizes late-responsive customers over early-responsive ones after observing the signal due to their higher purchase probability per coupon.

A.2.3 Summary

Table A.1 summarizes the purchase probabilities and dynamic-static gaps with and without a budget constraint when $m = \frac{1+r}{2}$.

Table A.1: Purchase probabilities under optimal static vs. dynamic policies when $m = (1+r)/2$.

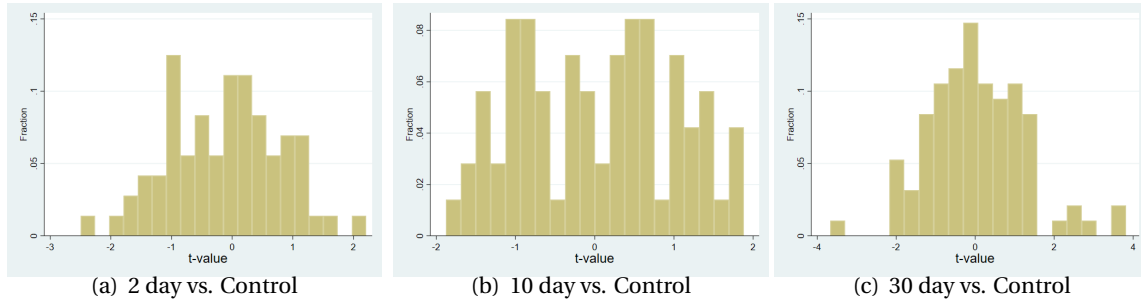
Budget	$\mathbb{E}_{\text{stat}}[Y]$	$\mathbb{E}_{\text{dyn}}[Y]$	Δ
$B = 1$	$\frac{1}{2} + \frac{1+r}{4}$	$\frac{1}{2} + \frac{1+r}{8} + \frac{(1-q)r + q}{4}$	$\frac{(2q-1)(1-r)}{8}$
$B \in [1/4, 1/2]$	$\frac{1}{2} + \frac{(1+r)B}{4}$	$\frac{1}{2} + \frac{(1+r)(4B-1)}{8} + \frac{(1-q)r + q}{4}$	$\frac{(1+r)B}{4} + \frac{(2q-1)(1-r)}{8}$

In summary, a dynamic policy takes into account the tradeoff between waiting to observe a signal and delaying the first coupon (which may be less effective later): the firm gives up the chance to target early-responsive customers at their optimal timing. Even so, when the period-2 signal is sufficiently informative, the ability to choose between $t = 2$ and $t = 3$ after learning is valuable enough to generate a positive dynamic gain even under an unconstrained setting ($B = 1$). When the budget binds ($B \in [1/4, 1/2]$), the gain is larger because the same information also enables the firm to avoid wasting scarce coupons on safe-type customers and prioritize late-responsive customers who yield a higher purchase probability per coupon.

B Balance Check

We check the balance between the treatment groups and the control group to see if the randomization is done accurately. Since we use more than 100 variables, we do not report the mean comparison of each variable in a table. Rather, in Figure A.1, we show the histograms of the t-values for the mean comparison test between the treatment and control groups for the variables used in the estimation of the optimal DTR. As the graphs show, t-values are located between -2 and 2 for most of the variables. Hence, there is no significant difference between the control and treatment groups.

Figure A.1: Balance Check (First Experiment)



Note: Each figure plots the histogram of t-values for a mean-comparison test between each treatment group and the control group.

C Details on Treatment Effects

C.1 Average Treatment Effects

Table A.2 shows the estimation results of ATEs on 12-week outcomes in the second experiment. Notice that in this experiment, we have two sorts of treatments, an email with a coupon incentive and an email without a coupon, for each treatment timing (2, 10, or 30 days after the first purchase). As in Table 4, to estimate the ATE of financial incentives on one timing, we use the subsample of the users who receive the email on at least one of the other timings. To estimate the ATE of emails, we condition the subsample that receives financial incentives on one of the other timings.

Table A.3 shows the estimation results of ATEs on 12-week outcomes of each treatment net of the other treatments. Specifically, for treatment of financial incentives, we condition on the subsample that receives no email on the other timings. For emails, we condition on the subsample that receives no financial incentives on the other timings. The results are robust to the choice of conditions, while the treatment effects are generally smaller than the ATEs in Table A.2.

Table A.2: Average Treatment Effects on 12-Week Outcomes (Second Experiment)

	Retention	Sales (JPY)	Quantity	Retention	Sales (JPY)	Quantity
	(A): Financial incentive			(B): Only appreciation email		
2 day	0.037 (0.003)	416.2 (202.4)	0.162 (0.041)	0.000 (0.003)	-114.1 (193.4)	-0.020 (0.043)
10 day	0.024 (0.003)	99.1 (151.9)	0.052 (0.039)	0.002 (0.003)	111.5 (202.7)	0.027 (0.047)
30 day	0.013 (0.003)	-10.7 (229.1)	0.023 (0.045)	0.000 (0.003)	-214.1 (140.7)	-0.058 (0.048)

Note: In each panel, the first column reports the treatment effects on whether a customer makes any purchases within 12 weeks since their first purchase. The second column reports the treatment effects on total sales and the third column reports the treatment effects on the number of items purchased. For the first panel, we condition on the subsample of customers who receive an email on at least one of the other timings. For the second panel, we condition on the subsample of customers who receives financial incentive on one of the other timings. The standard errors are in parentheses. The table does not report the constants as the constants reveal the baseline retention rates, sales, and quantities, which is prohibited due to the NDA.

Table A.3: Robustness Check of ATE Estimates (Second Experiment)

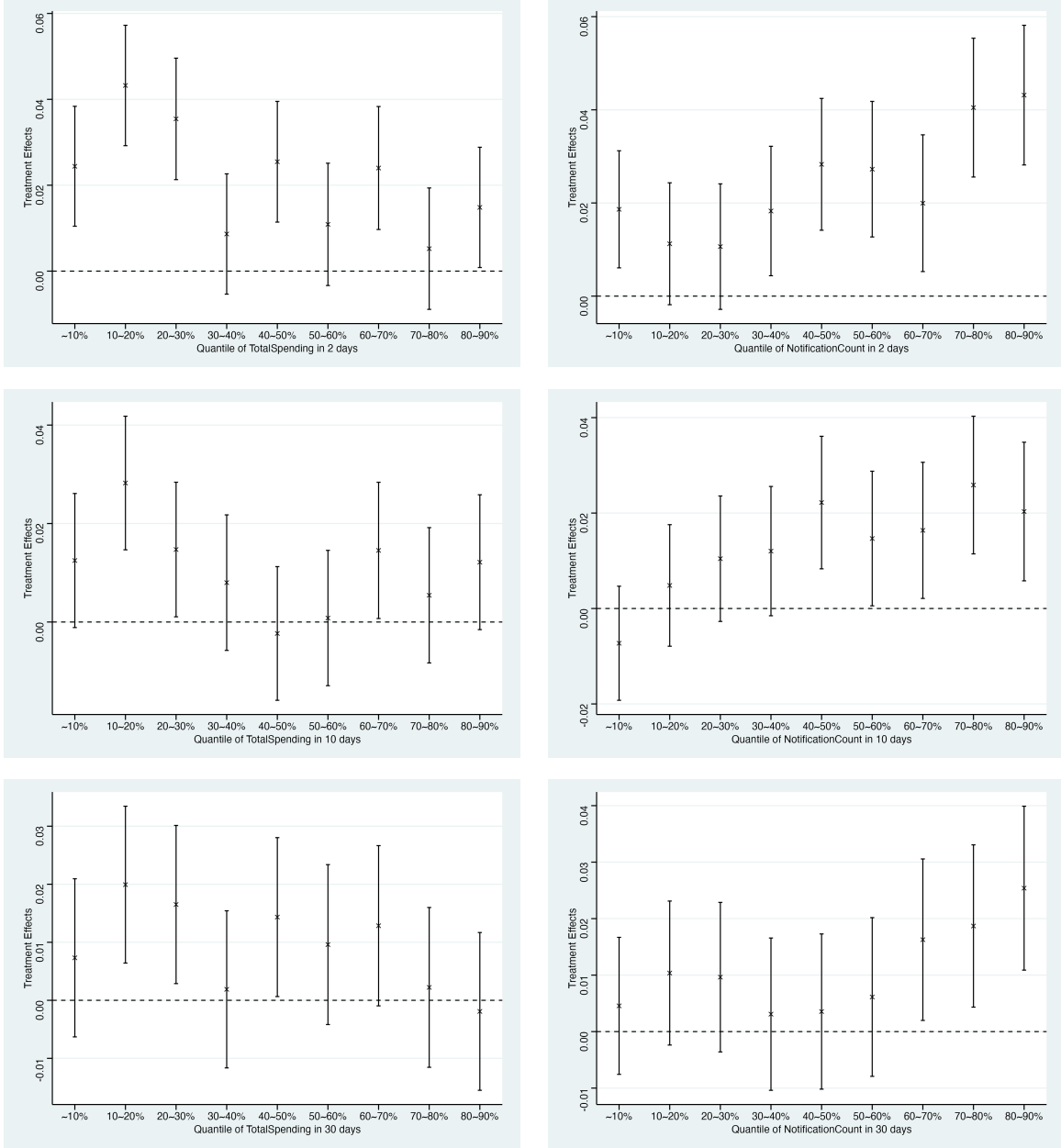
	Retention	Sales (JPY)	Quantity	Retention	Sales (JPY)	Quantity
	(A): Financial incentive			(B): Only appreciation email		
2 day	0.026 (0.005)	139.7 (236.8)	0.152 (0.087)	-0.002 (0.003)	-114.1 (192.4)	0.098 (0.038)
10 day	0.012 (0.005)	-148.8 (237.4)	0.023 (0.087)	-0.003 (0.003)	26.9 (184.1)	-0.025 (0.038)
30 day	0.004 (0.005)	416.0 (237.5)	0.075 (0.087)	-0.001 (0.003)	-94.8 (184.1)	0.015 (0.038)

Note: In each panel, the first column reports the treatment effects on whether a customer makes any purchases within 12 weeks since their first purchase. The second column reports the treatment effects on total sales and the third column reports the treatment effects on the number of items purchased. For the first panel, we condition on the subsample of customers who did not receive any email on the other dates. For the second panel, we condition on the subsample of customers who did not receive any financial incentive on the other dates. The standard errors are in parentheses. The table does not report the constants as the constants reveal the baseline retention rates, sales, and quantity, which is prohibited due to the NDA.

C.2 Conditional Average Treatment Effects

Next, we provide the results of the conditional average treatment effects (CATEs) omitted from Section 5.3. Figure ?? shows the CATEs of the 2, 10, and 30-day treatments on retention within 8 weeks of the first purchase for each decile of the total spending for the first purchase (left panels) and the number of messages sent until the delivery of the first item (right panels). The CATEs are estimated by the linear regression presented in Section 5.3. Both conditioning variables exhibit clear heterogeneity in the treatment effects.

Figure A.2: Average treatment effects across deciles of the number of messages sent (30 day)



Note: The figures show the average treatment effects of the 2, 10, and 30-day treatments on the retention rate within 8 weeks against the total spending for the first purchase (left panels) and the number of messages sent since the delivery of the first item (right panels). The total spending and the number of messages are split into deciles. Error bars indicate the 95% confidence intervals.

D On the Reformulation of Non-stationary Markov Decision Problem

Our method allows us to study non-stationary dynamic problems. One may wonder if it is possible to reformulate a nonstationary problem into a stationary Markov decision problem. Using a different notation for clarity, we can define the state as (x, y_1, y_2, y_3) , where x represents static features and y_t rep-

resents time-dependent features like purchase history. One could still define the state as (x, y_1, y_2, y_3) , concatenating all y states and the problem is a stationary Markov.

While it is theoretically possible that a non-Markov or non-stationary environment can be transformed into a stationary Markov decision process by augmenting the state space, this transformation does not imply that existing methods for stationary Markov problems are practically applicable in our setting.

First, the required state augmentation leads to a rapidly expanding and high-dimensional state space that includes the entire history of actions and states. In our application with rich behavioral data, this makes estimation of value functions or transition dynamics infeasible due to the curse of dimensionality.

Second, our problem involves inter-temporal budget constraints and feasibility constraints that are path-dependent. Incorporating these constraints into a Markov representation requires tracking additional state variables such as past treatments and remaining budget, further increasing the complexity of the state space and rendering standard RL approaches difficult to apply, especially in an offline setting.

Third, our setting is fundamentally an offline policy learning problem based on experimental data. Many existing reinforcement learning methods for stationary Markov problems rely on online exploration or repeated interactions, which are not available in our context.

Fourth, the reformulation approach ignores the fact that the optimal Q -function and DTR do not depend on future states, potentially leading to statistical inefficiency. For example, if we use Q -learning, the reformulation approach learns a single, joint Q -function as a function of (x, y_1, y_2, y_3) and a . In our dynamic setting, the Q -function has the following structure:

$$Q(x, y_1, y_2, y_3, a) = \begin{cases} Q_1(x, y_1, a) & \text{if } (y_2, y_3) \text{ is missing (i.e., in period 1)} \\ Q_2(x, y_1, y_2, a) & \text{if } y_3 \text{ is missing (i.e., in period 2)} \\ Q_3(x, y_1, y_2, y_3, a) & \text{otherwise (i.e., in period 3),} \end{cases}$$

where Q_t is the period- t Q -function. Without specifying the joint Q -function in this separable form, it is still possible to use off-the-shelf RL methods to learn it after padding missing values of future states. However, recovering the above separable structure from the data is statistically challenging and requires a large number of observations. Our approach directly exploits this known structure, enabling efficient use of the data.

Our contribution is therefore not to challenge the theoretical equivalence, but to develop a computationally tractable and empirically implementable method for learning optimal dynamic policies under constraints without requiring full state augmentation or transition modeling.

E Training Details

We present training details for the baseline model with two actions and the extension with three actions.

E.1 Baseline Model

E.1.1 Model Specification

Q-Learning. We split users into types based on whether they have already received a coupon and whether they have already made a second purchase. We learn the parameter of the Q -function separately for these types.

Specifically, for periods $t = 2, 3$, there are four types:

1. those who have already received a coupon and made a second purchase;
2. those who have already received a coupon but have yet to make a second purchase;
3. those who have yet to receive a coupon but have already made a second purchase; and
4. those who have yet to receive a coupon or make a second purchase.

Types 1–3 would not receive a coupon at t under our constraints that we send a coupon at most once and do not send a coupon to those have already made a second purchase. Therefore, for these types, we need to set or learn the Q -functions $Q_t^Y(h_t, a_t)$ and $Q_t^C(h_t, a_t)$ for $a_t = 0$ only. For types 1 and 3, we set $Q_t^Y(h_t, a_t) = 1$, since they have already made a second purchase. For type 3, we set $Q_t^C(h_t, a_t) = 0$, since they would never receive a coupon. We learn the other Q -functions for types 1–3 from the data. Also, for type 4, we learn the Q -functions for $a_t = 0$ and $a_t = 1$ separately, using the corresponding subsamples.

For period $t = 1$, no user has already recieved a coupon, so there are two types:

1. those who have already made a second purchase; and
2. those who have yet to make a second purchase.

We do not need to set or learn the Q -functions for type 1, since we would never send a coupon to them and $t = 1$ is the last iteration of the backward procedure. For type 2, we learn the Q -functions for $a_t = 0$ and $a_t = 1$ separately, using the corresponding subsamples.

BOWL. There is no need to learn the rules for types 1–3 in periods $t = 2, 3$ and for type 1 in period $t = 1$, since they would not receive a coupon under our constraints. We learn the rules for type 4 in periods $t = 2, 3$ and for type 2 in period $t = 1$.

E.1.2 Hyperparameter Tuning

Q-Learning. We use Python’s Lasso from `sklearn.linear_model` to learn the Q -functions. For each model, we tune the regularization parameter α while keeping all other hyperparameters at their default values. For the models corresponding to types 1–3 in periods $t = 2, 3$ and type 1 in period $t = 1$, we set $\alpha = \text{None}$, which calls Lasso’s built-in 5-fold cross-validation procedure. For the remaining 12 models, which correspond to type 4 in periods $t = 2, 3$ and type 2 in period $t = 1$ (i.e., those who have not yet received a coupon or made a second purchase)⁴⁴, we tune α and the budget parameters $(\lambda_1, \lambda_2, \lambda_3)$ to optimize the 5-fold cross-validated estimates of the expected outcome and cost. Specifically, we first perform a grid search over $(\lambda_1, \lambda_2, \lambda_3)$ under the restriction $\lambda_1 = \lambda_2 = \lambda_3$, while setting $\alpha = \text{None}$, to determine a common λ value that yields a cross-validated cost close to the target budget. We then tune α for period $t = 3$ via grid search to maximize the cross-validated outcome subject to the budget constraint. Next, we tune α for periods $t = 2$ and $t = 1$ sequentially using the same procedure. Finally, we conduct a grid search over $(\lambda_1, \lambda_2, \lambda_3)$ again, this time allowing the parameters to vary across periods.

BOWL. We use Python’s `SGDClassifier` from `sklearn.linear_model` with L2 regularization and hinge loss to learn the rules for type 4 in periods $t = 2, 3$ and for type 2 in period $t = 1$ (i.e., those who have not yet received a coupon or made a second purchase). We tune the regularization parameter α while keeping all other hyperparameters at their default values. We conduct an extensive grid search to jointly tune the three α parameters (one for each period) and $(\lambda_1, \lambda_2, \lambda_3)$ to optimize the 5-fold cross-validated estimates of the expected outcome and cost.

E.2 Extension

We implement BOWL only for the extension.

⁴⁴There are four models $(Q_t^Y(h_t, 0), Q_t^Y(h_t, 1), Q_t^C(h_t, 0), Q_t^C(h_t, 1))$ for each period.

E.2.1 Model Specification

As in the baseline model, we split the users into four types for $t = 2, 3$ and two types for $t = 1$. For types 1–3 in periods $t = 2, 3$ and type 1 in period $t = 1$, the feasible actions are either sending only an email or sending nothing. For type 4 in periods $t = 2, 3$ and for type 2 in period $t = 1$, the feasible actions are sending an email with a coupon, sending only an email, and sending nothing. We learn the rules for these types separately. As discussed in Appendix F, we set the outcome to the third purchase for types 1 and 3 in periods $t = 2, 3$ and type 1 in period $t = 1$ (i.e., those who have already made a second purchase), and to the second purchase for types 2 and 4 in periods $t = 2, 3$ and for type 2 in period $t = 1$ (i.e., those who have not made a second purchase yet). Also, for type 4 in periods $t = 2, 3$ and for type 2 in period $t = 1$ (i.e., those who have not yet received a coupon or made a second purchase), we implement pairwise binary classification for each pair of actions and choose the action with two wins, breaking ties at random.

E.2.2 Hyperparameter Tuning

As in the baseline model, we use Python's `SGDClassifier` with L2 regularization and hinge loss to learn the rules. For the models corresponding to types 1 and 3 in periods $t = 2, 3$ and type 1 in period $t = 1$, we set $\alpha = 0.0001$, the default value. For the remaining 11 models, which correspond to types 2 and 4 in periods $t = 2, 3$ and for type 2 in period $t = 1$ (i.e., those who have not made a second purchase yet)⁴⁵, we jointly tune α and $(\lambda_1, \lambda_2, \lambda_3)$ to optimize the 5-fold cross-validated estimates of the expected outcome and cost. We perform the tuning using Python's `BayesianOptimization` from `bayes_opt`.

F Backward Outcome Weighted Learning with Multiple Actions

In the case of binary actions, BOWL uses the fact that $\mathbf{1}[A_t = d_t(H_t)] = 1 - \mathbf{1}[A_t \neq d_t(H_t)]$ to transform the problem of maximizing the average outcome into a binary classification problem. With multiple actions, we cannot simply use such a trick to transform the original problem into a multiclass classification problem. We instead use the transformation into multiple binary classification problems, which is operationalized as in Definition 5.⁴⁶

⁴⁵There are one binary classifier for type 2 in periods $t = 2, 3$ and three binary classifiers for type 4 in in periods $t = 2, 3$ and type 2 in period $t = 1$.

⁴⁶This is in the same spirit with the one-versus-one reduction of multiclass classification, which is implemented by many computer software packages available in R and Python.

Definition 5 (BOWL for Threshold DTRs with Multiple Actions). Construct an estimator $\hat{\mathbf{d}}(\boldsymbol{\lambda}) = (\hat{d}_t(\cdot; \bar{\boldsymbol{\lambda}}_t))_{t=1}^T$ for the threshold DTR $\mathbf{d}(\boldsymbol{\lambda}) = (d_t(\cdot; \bar{\boldsymbol{\lambda}}_t))_{t=1}^T$ by iterating the following three steps backward from $t = T$ to $t = 1$:

1. For each pair of actions $\{a_t, a'_t\} \subset \mathcal{A}_t$, apply a binary weighted classification method to the subsample for which both actions are feasible and either of the two actions is chosen in period t (i.e. $\{a_t, a'_t\} \in \mathcal{A}_t(H_t^{(i)})$ and $A_t^{(i)} \in \{a_t, a'_t\}$), using $A_t^{(i)}$ as the true class, $H_t^{(i)}$ as the feature vector, and $\frac{Y^{(i)} - \lambda_t C^{(i)}}{d_t^0(A_t^{(i)} | H_t^{(i)})}$ (when $t = T$) or $\frac{(Y^{(i)} - \lambda_t C^{(i)}) \prod_{j=t+1}^T \mathbf{1}[A_j^{(i)} = \hat{A}_j^{(i)}]}{\prod_{j=t}^T d_j^0(A_j^{(i)} | H_j^{(i)})}$ (when $t < T$) as the weight for observation i . Here, $\hat{A}_{t+1}^{(i)}, \dots, \hat{A}_T^{(i)}$ are defined in Step 3 of the iterations prior to t . Let $\tilde{d}_t(h_t; \{a_t, a'_t\})$ denote the constructed classification rule.
2. Construct a rule that chooses the action with the largest number of wins in a round-robin tournament.

$$\hat{d}_t(h_t; \bar{\boldsymbol{\lambda}}_t) \in \arg \max_{a_t \in \mathcal{A}_t(h_t)} \sum_{a'_t \in \mathcal{A}_t(h_t) \setminus \{a_t\}} \mathbf{1}[\tilde{d}_t(h_t; \{a_t, a'_t\}) = a_t].$$

In the case of ties, randomly choose one of the tied action.

3. Let $\hat{A}_t^{(i)} = \hat{d}_t(H_t^{(i)}; \bar{\boldsymbol{\lambda}}_t)$ for $i = 1, \dots, n$.

Implementation Details for Application in Section 7.2. For each given $\boldsymbol{\lambda}$, we use BOWL to learn a threshold policy by setting the outcome to the second purchase indicator if the user has not made a second purchase yet, and to the third purchase indicator if the user has already made a second purchase. Specifically, let Y_{2nd} and Y_{3rd} denote the second purchase and the third purchase within 2 months, respectively, and second_t denote whether the user has made a second purchase before the beginning of period t , which is included in H_t . We implement the algorithm in Definition 5 where the weight for observation i is given by $\frac{Y_{2nd}^{(i)} - \lambda_t C^{(i)}}{d_t^0(A_t^{(i)} | H_t^{(i)})}$ if $\text{second}_T = 0$ and $\frac{Y_{3rd}^{(i)}}{d_t^0(A_t^{(i)} | H_t^{(i)})}$ if $\text{second}_T = 1$ (when $t = T$), or $\frac{(Y_{2nd}^{(i)} - \lambda_t C^{(i)}) \prod_{j=t+1}^T \mathbf{1}[A_j^{(i)} = \hat{A}_j^{(i)}]}{\prod_{j=t}^T d_j^0(A_j^{(i)} | H_j^{(i)})}$ if $\text{second}_t = 0$ and $\frac{(Y_{3rd}^{(i)}) \prod_{j=t+1}^T \mathbf{1}[A_j^{(i)} = \hat{A}_j^{(i)}]}{\prod_{j=t}^T d_j^0(A_j^{(i)} | H_j^{(i)})}$ if $\text{second}_t = 1$ (when $t < T$). We do not penalize the cost if $\text{second}_t = 1$, since whether to send a no-incentive email to the user does not significantly affect the cost.

G Mathematical Appendix

Notation. For $t = 1, \dots, T$, let $\beta_t(h_t; \bar{\mathbf{d}}_{t+1}) = Q_t^Y(h_t, 1; \bar{\mathbf{d}}_{t+1}) - Q_t^Y(h_t, 0; \bar{\mathbf{d}}_{t+1})$ and $\gamma_t(h_t; \bar{\mathbf{d}}_{t+1}) = Q_t^C(h_t, 1; \bar{\mathbf{d}}_{t+1}) - Q_t^C(h_t, 0; \bar{\mathbf{d}}_{t+1})$. For convenience, we interpret $(\underline{\mathbf{d}}'_{t-1}, \bar{\mathbf{d}}_t)$ as $\mathbf{d}$ when $t = 1$ and interpret $(\underline{\mathbf{d}}_t, \bar{\mathbf{d}}'_{t+1})$ as $\mathbf{d}$ when $t = T$.

G.1 Proof of Proposition 1

Let $\mathbf{d}^*$ solve (6.1), and let $\boldsymbol{\lambda}^* \in \mathbb{R}_+^T$ satisfy that $E_{\mathbf{d}(\boldsymbol{\lambda}^*)}[C] = B$ and that $E_{\underline{\mathbf{d}}_t^*, \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*)}[C] = B$ for all $t = 1, \dots, T-1$. Then, $\mathbf{d}(\boldsymbol{\lambda}^*)$ is shown to solve (6.1) by induction from the following lemma (note that the lemma shows that $(\underline{\mathbf{d}}_{T-1}^*, \bar{\mathbf{d}}_T(\bar{\boldsymbol{\lambda}}_T^*))$ solves (6.1) without any assumption).

Lemma 1. *Let $t \in \{1, \dots, T\}$. If $t = T$, then $(\underline{\mathbf{d}}_{T-1}^*, \bar{\mathbf{d}}_T(\bar{\boldsymbol{\lambda}}_T^*))$ solves (6.1). If $t < T$ and $(\underline{\mathbf{d}}_t^*, \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*))$ solves the problem (6.1), then $(\underline{\mathbf{d}}_{t-1}^*, \bar{\mathbf{d}}_t(\bar{\boldsymbol{\lambda}}_t^*))$ solves (6.1).*

Proof. Let $t \in \{1, \dots, T\}$. For a rule d_t , let $d_t(h_t) \in [0, 1]$ denote the probability of choosing action 1 given h_t , regardless of whether d_t is deterministic or not. If $t < T$, suppose that $(\underline{\mathbf{d}}_t^*, \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*))$ solves (6.1).

Since $(\underline{\mathbf{d}}_t^*, \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*))$ is optimal (for $t = T$, this holds by the optimality of $\mathbf{d}^*$), d_t^* solves

$$\max_{d_t} E_{\underline{\mathbf{d}}_{t-1}^*, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*)}[Y] \text{ s.t. } E_{\underline{\mathbf{d}}_{t-1}^*, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*)}[C] \leq B \quad (\text{G.1})$$

among rules satisfying the period- t feasibility constraint. Observe that for any d_t ,

$$\begin{aligned} E_{\underline{\mathbf{d}}_{t-1}^*, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*)}[Y] &= E_{\underline{\mathbf{d}}_{t-1}^*}[Q_t^Y(H_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*)) + d_t(H_t)\beta_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*))], \\ E_{\underline{\mathbf{d}}_{t-1}^*, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*)}[C] &= E_{\underline{\mathbf{d}}_{t-1}^*}[Q_t^C(H_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*)) + d_t(H_t)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*))]. \end{aligned}$$

The problem (G.1) is then equivalent to

$$\max_{d_t} E_{\underline{\mathbf{d}}_{t-1}^*}[d_t(H_t)\beta_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*))] \text{ s.t. } E_{\underline{\mathbf{d}}_{t-1}^*}[d_t(H_t)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*))] \leq \bar{B}, \quad (\text{G.2})$$

where $\bar{B} = B - E_{\underline{\mathbf{d}}_{t-1}^*}[Q_t^C(H_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\boldsymbol{\lambda}}_{t+1}^*))]$.

Below, we show that $d_t(\cdot; \bar{\boldsymbol{\lambda}}_t^*)$ solves (G.2), which implies that $(\underline{\mathbf{d}}_{t-1}^*, \bar{\mathbf{d}}_t(\bar{\boldsymbol{\lambda}}_t^*))$ solves (6.1). Let

$$S_0 = \{h_t \in \mathcal{H}_t : d_t(h_t; \bar{\boldsymbol{\lambda}}_t^*) = 0, d_t^*(h_t) > 0\},$$

$$S_1 = \{h_t \in \mathcal{H}_t : d_t(h_t; \bar{\boldsymbol{\lambda}}_t^*) = 1, d_t^*(h_t) < 1\}.$$

Note that $\mathcal{A}_t(h_t) = \{0, 1\}$ for any $h_t \in S_0 \cup S_1$, since otherwise the feasibility constraint is violated by either $d_t(\cdot; \bar{\lambda}_t^*)$ or d_t^* . Observe that

$$\begin{aligned} 0 &\geq E_{\underline{\mathbf{d}}_{t-1}^*}[d_t^*(H_t)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] - E_{\underline{\mathbf{d}}_{t-1}^*}[d_t(H_t; \bar{\lambda}_t^*)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] \\ &= E_{\underline{\mathbf{d}}_{t-1}^*}[(d_t^*(H_t) - d_t(H_t; \bar{\lambda}_t^*))\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] \\ &= E_{\underline{\mathbf{d}}_{t-1}^*}[\mathbf{1}[H_t \in S_0]d_t^*(H_t)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] - E_{\underline{\mathbf{d}}_{t-1}^*}[\mathbf{1}[H_t \in S_1](1 - d_t^*(H_t))\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))], \end{aligned}$$

where the inequality in the first line holds since $E_{\underline{\mathbf{d}}_{t-1}^*}[d_t^*(H_t)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] \leq \bar{B}$ and $E_{\underline{\mathbf{d}}_{t-1}^*}[d_t(H_t; \bar{\lambda}_t^*)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] = \bar{B}$ by assumption. We then obtain that

$$\begin{aligned} &E_{\underline{\mathbf{d}}_{t-1}^*}[d_t^*(H_t)\beta_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] - E_{\underline{\mathbf{d}}_{t-1}^*}[d_t(H_t; \bar{\lambda}_t^*)\beta_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] \\ &= E_{\underline{\mathbf{d}}_{t-1}^*}[\mathbf{1}[H_t \in S_0]d_t^*(H_t)\beta_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] - E_{\underline{\mathbf{d}}_{t-1}^*}[\mathbf{1}[H_t \in S_1](1 - d_t^*(H_t))\beta_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] \\ &\leq \lambda_t^*(E_{\underline{\mathbf{d}}_{t-1}^*}[\mathbf{1}[H_t \in S_0]d_t^*(H_t)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))] - E_{\underline{\mathbf{d}}_{t-1}^*}[\mathbf{1}[H_t \in S_1](1 - d_t^*(H_t))\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))]) \\ &\leq 0, \end{aligned}$$

where the inequality in the third line holds since

$$\begin{aligned} d_t(h_t; \bar{\lambda}_t^*) &\in \arg \max_{a_t \in \{0, 1\}} \{Q_t^Y(h_t, a_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*)) - \lambda_t^* Q_t^C(h_t, a_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))\} \\ &= \arg \max_{a_t \in \{0, 1\}} a_t \{\beta_t(h_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*)) - \lambda_t^* \gamma_t(h_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}^*))\} \end{aligned}$$

for all $h_t \in S_0 \cup S_1$, $d_t(h_t; \bar{\lambda}_t^*) = 0$ for all $h_t \in S_0$, and $d_t(h_t; \bar{\lambda}_t^*) = 1$ for all $h_t \in S_1$. Therefore, $d_t(\cdot; \bar{\lambda}_t^*)$ solves (G.2), and hence $(\underline{\mathbf{d}}_{t-1}^*, \bar{\mathbf{d}}_t(\bar{\lambda}_t^*))$ solves (6.1). ■

G.2 Proof of Proposition 2

Let $t \in \{1, \dots, T\}$. Observe that for any deterministic rule d_t ,

$$\begin{aligned} E_{\underline{\mathbf{d}}_{t-1}, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})}[Y] &= E_{\underline{\mathbf{d}}_{t-1}}[Q_t^Y(H_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})) + d_t(H_t)\beta_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}))], \\ E_{\underline{\mathbf{d}}_{t-1}, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})}[C] &= E_{\underline{\mathbf{d}}_{t-1}}[Q_t^C(H_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})) + d_t(H_t)\gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}))], \end{aligned}$$

and hence

$$\begin{aligned} E_{\underline{\mathbf{d}}'_{t-1}, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})}[Y - \lambda_t C] &= E_{\underline{\mathbf{d}}'_{t-1}}[Q_t^Y(H_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})) - \lambda_t Q_t^C(H_t, 0; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}))] \\ &\quad + E_{\underline{\mathbf{d}}'_{t-1}}[d_t(H_t)\{\beta_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})) - \lambda_t \gamma_t(H_t; \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1}))\}], \end{aligned}$$

given any arbitrary $\underline{\mathbf{d}}'_{t-1}$, in particular $\underline{\mathbf{d}}_{t-1}^0$. It follows from the definition of $d_t(\cdot; \bar{\lambda}_t)$ that $d_t(\cdot; \bar{\lambda}_t)$ maximizes $E_{\underline{\mathbf{d}}'_{t-1}, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})}[Y - \lambda_t C]$ among rules d_t satisfying the period- t feasibility constraint. Under Assumption 1,

$$E_{\underline{\mathbf{d}}_{t-1}^0, d_t, \bar{\mathbf{d}}_{t+1}(\bar{\lambda}_{t+1})}[Y - \lambda_t C] = E \left[\frac{(Y - \lambda_t C) \prod_{j=t+1}^T \mathbf{1}[A_j = d_j(H_j; \bar{\lambda}_j)]}{\prod_{j=t}^T d_j^0(A_j | H_j)} \mathbf{1}[A_t = d_t(H_t)] \right],$$

where we set $\prod_{j=t+1}^T \mathbf{1}[A_j = d_j(H_j; \bar{\lambda}_j)] = 1$ when $t = T$. Since the action is binary and hence $\mathbf{1}[A_t = d_t(H_t)] = 1 - \mathbf{1}[A_t \neq d_t(H_t)]$, maximizing the right-hand side is equivalent to minimizing

$$E \left[\frac{(Y - \lambda_t C) \prod_{j=t+1}^T \mathbf{1}[A_j = d_j(H_j; \bar{\lambda}_j)]}{\prod_{j=t}^T d_j^0(A_j | H_j)} \mathbf{1}[A_t \neq d_t(H_t)] \right].$$

Therefore, $d_t(\cdot; \bar{\lambda}_t)$ minimizes the above among rules d_t satisfying the period- t feasibility constraint.