https://pubsonline.informs.org/journal/mksc

MARKETING SCIENCE
Vol. 42, No. 4, July–August 2023, pp. 637–658
ISSN 0732-2399 (print), ISSN 1526-548X (online)

Dynamic Coupon Targeting Using Batch Deep Reinforcement Learning: An Application to Livestream Shopping

Xiao Liu[a]

[a] Stern School of Business, New York University, New York, New York 10012
Contact: xl23@stern.nyu.edu, https://orcid.org/0000-0002-7093-8534 (XL)

Received: September 3, 2020
Revised: September 5, 2021; April 17, 2022
Accepted: June 23, 2022
Published Online in Articles in Advance: October 20, 2022

https://doi.org/10.1287/mksc.2022.1403

Abstract. We present an empirical framework for creating dynamic coupon targeting strategies for high-dimensional and high-frequency settings, and we test its performance using a large-scale field experiment. The framework captures consumers' intertemporal tradeoffs associated with dynamic pricing and does not rely on functional form assumptions about consumers' decision-making processes. The model is estimated using batch deep reinforcement learning (BDRL), which relies on Q-learning, a model-free solution that can mitigate model bias. It leverages deep neural networks to represent the high-dimensional state space and alleviate the curse of dimensionality. The empirical application is in a multibillion-dollar livestream shopping context. Our BDRL solution increases the platform's revenue by twice as much as static targeting policies and by 20% more than the model-based solution. The comparative advantage of BDRL comes from more effective and automatic targeting of consumers based on both heterogeneity and dynamics, using exceptionally rich, nuanced differences among consumers and across time. We find that price skimming, reducing discounts for attractive hosts, and increasing the coupon discount level at a faster rate for low spenders are effective strategies based on dynamics, consumer heterogeneity, and the two combined, respectively.

History: K. Sudhir served as the senior editor and John Hauser served as associate editor for this article.
Funding: Partial financial support was received from the NYU Center for Global Economy and Business.
Supplemental Material: The data files and online appendices are available at https://doi.org/10.1287/mksc.2022.1403.

Keywords: dynamic pricing • coupon • deep reinforcement learning • reference price • livestream shopping • targeting

1. Introduction

This paper presents a new solution to the dynamic coupon targeting problem based on batch deep reinforcement learning, and we conduct a large-scale field experiment to demonstrate its comparative advantages against state-of-the-art benchmarks.

In the e-commerce era, dynamic and personalized pricing strategies can improve market efficiency, allowing firms to serve more customers and generate higher profits (Furman et al. 2019). To ensure fairness and protect customer trust, companies often do not implement personalized pricing directly but instead use personalized discounts or coupons (UK Competition and Markets Authority 2018). Companies send coupons very frequently, and coupon redemptions account for tens of billions of dollars in annual marketing spending. For example, Whole Foods sends at least one coupon to each Amazon Prime member every week.[1] The aggregate value of digital coupon redemption is forecasted to surge from $47 billion in 2017 to $91 billion by 2022.[2] The high frequency of coupon delivery makes the coupon allocation problem a dynamic targeting problem, and fine-grained personalized targeting is more attractive than ever as modern marketers increasingly have access to detailed data on individual consumers and their shopping patterns.

However, having detailed consumer data are only half of the equation, as one must also have a feasible method of processing that data to generate effective targeting strategies. In this paper, we aim to create dynamic coupon targeting strategies for high-dimensional and high-frequency settings. We address three research questions. (1) How can we develop a theoretical framework that incorporates the intertemporal tradeoffs in dynamic coupon targeting to design a policy that maximizes revenue? (2) How can we evaluate the performance of dynamic coupon targeting policies? (3) What are the gains, if any, from using a dynamic targeting framework relative to a static benchmark, and what explains the differences in performance?

We face three technical challenges in answering these questions. First, we need to determine the optimal

dynamic pricing strategy (e.g., price penetration, price skimming, or cyclical) when the type(s) and magnitude of demand dynamics are unknown and when firms' repeated coupon interactions with consumers may also shape consumers' behaviors over time. Second, we need to predict consumer responses to coupons without using a parametric formulation that can introduce model bias. Third, we need to overcome the curse of dimensionality problem because firms tend to collect high-dimensional data on consumer and contextual features for targeting.

We overcome these three challenges by developing a framework to design and evaluate dynamic personalized pricing strategies. Our solution is a model-free, batch deep reinforcement learning (BDRL) algorithm. It is based on deep reinforcement learning techniques similar to those that were used to create artificial intelligence agents such as AlphaGo and the Deep Q-Network (DQN) algorithm, which defeated the best professional human players in the games of Go and Atari 2600, respectively (Mnih et al. 2015).

First, we formulate the dynamic coupon targeting problem as a Markov decision process (MDP) and build on the dynamic pricing literature (see Seetharaman 2009 for a review) to incorporate consumer intertemporal tradeoffs. We empirically test three dynamic pricing theories—the reference price effect, loyalty/inertia, and variety-seeking—and we identify the optimal coupon allocation sequence for each. Second, we use a model-free reinforcement learning solution, Q-learning, to solve the MDP. Our solution mitigates model bias because it does not rely on any functional form assumptions to model consumer purchase behaviors (the reward function) and the state transition process. Third, we use deep neural networks (DNNs) to approximate the action-specific value functions, thereby alleviating the curse of dimensionality problem.

We apply the BDRL framework, to a novel, multibillion-dollar livestream shopping context. Livestream shopping is a new format of e-commerce that allows brands and social media influencers to reach consumers directly through live interactive video sessions, in which hosts showcase and sell just about every product under the sun, from fresh fruits to fine jewelry. During a livestream session, consumers can ask hosts questions and make purchases in real time. High-frequency coupons are widely used by livestream shopping platforms to incentivize consumer engagement. Our data come from one of the largest livestream shopping platforms in the world. Because livestream shopping is a relatively new product (launched in 2016), the platform was still in an experimental mode at the time of our study, and it was testing the performance of coupon strategies using randomized trials. However, the platform was seeking an algorithmic solution for more efficient coupon allocation. Specifically, the platform was looking for an algorithm to decide who should receive coupons of which value (i.e., targeting based on consumer heterogeneity) and how the value should vary as each consumer gains purchase experience (i.e., targeting based on dynamics). Due to the firm's concern regarding potential negative effects of experimental policies, we use batch reinforcement learning in training and the doubly robust off-policy policy evaluation method to assess the performance of the proposed policy offline, against a comprehensive set of benchmarks, including both static targeting and model-based dynamic targeting policies. After demonstrating the effectiveness of BDRL offline, we run a large-scale field experiment to provide out-of-sample validation.

The results show that our solution increases the platform's gross merchandise value by 63%. The BDRL algorithm is approximately twice as effective as static targeting strategies, suggesting the importance of demand dynamics. BDRL is also 1.5 times more effective than a dynamic structural model, indicating possible model bias and a strong economic incentive for the implementation of model-free solutions. The reason behind the superiority of the seemingly black-box BDRL method is rather intuitive and explainable. BDRL can more effectively and automatically identify when to target consumers (dynamics) and who to target (heterogeneity) based on exceptionally rich, nuanced differences among consumers and across time. Regarding dynamics, static policies suffer from myopia—they ignore the long-term negative consequences of the reference price effect. By contrast, BDRL recommends small-discount coupons at the beginning, and the recommended discount level increases gradually to avoid the negative reference price effect in the long run. The advantage regarding heterogeneity is best demonstrated with an example: BDRL suggests smaller-discount coupons when the consumer is visiting the channel of a more attractive host. This granular consumer behavioral factor would be easy to overlook with nonscalable human approaches. Last, BDRL organically combines dynamics and heterogeneity. For instance, the strength of the reference price effect may vary across individuals and across time within each individual. BDRL can detect these nuances and recommend different pricing trends for different consumers, specifically, a faster increase in the coupon discount level for low spenders than for high spenders.

The paper makes three contributions. First, we make a methodological contribution by developing a new framework with which managers and researchers can design and evaluate high-frequency and high-dimensionality dynamic targeting strategies for coupons and pricing. Second, from a managerial perspective, we use a large-scale field experiment to demonstrate the real-world effectiveness of a robust machine-learning application in an area of practical importance (livestream shopping) and

theoretical importance (optimal coupon allocation strategies). Third, substantively, we contribute to the dynamic pricing literature by conducting empirical tests of the three dynamic pricing theories, and we contribute to the personalized pricing literature by comprehensively comparing static targeting, structural model–based targeting, and model-free dynamic targeting strategies. Our empirical finding that a flexible functional form does better out-of-sample/off-policy is an important data point in the debate about the raison d'être for structural models.

The rest of the paper is organized as follows. Section 2 reviews the literature. Section 3 introduces the business context and data. Section 4 and Section 5 describe our model and the benchmarks. Section 6 presents the results, and Section 7 concludes.

2. Literature

2.1. Dynamic Pricing

Because coupons provide price discounts, a dynamic coupon targeting strategy is essentially a personalized dynamic pricing strategy (Van Heerde and Neslin 2017). Therefore, to design dynamic coupons, we draw on the dynamic pricing literature, which has documented three dynamic effects of price promotions on demand (Seetharaman 2009): (1) reference price, (2) loyalty/inertia, and (3) variety-seeking.

Reference price refers to the idea that consumers may use historical prices to construct a reference price (Winer 1986). If the firm provides a high discount now, then consumers might adopt this low price as the reference price and become disinclined to purchase in the future, as the price almost certainly will be higher than the reference price. In the future, the firm would have to provide an even higher discount to entice consumers to make purchases. The pricing implication of the reference price effect is that a firm should start with a low discount and gradually increase the discount amount over time.

Inertia or loyalty, also known as state dependence (Jeuland 1979, Dubé et al. 2010), refers to the idea that consumers might become increasingly loyal as they gain purchase experience because of switching costs or learning effects. If so, then a low discount (relative to a high discount) will increase the paid amount per transaction and short-term profits, but it also will attract and lock in a relatively small customer base, thereby securing fewer repeated purchases in the future. The recommended strategy based on inertia is penetration pricing, namely, providing an initial high discount and decreasing the discount over time.

Variety-seeking, also known as negative state dependence (Kahn et al. 1986), refers to the idea that consumers become satiated with the product after a purchase and prefer to switch products in subsequent purchases. It implies that a higher discount will increase short-term purchases but decrease long-term purchase intentions for the same product. Therefore, the variety-seeking framework recommends that firms provide a consistent, low discount (Seetharaman and Che 2009).

We add to this literature by providing empirical tests of the three intertemporal tradeoffs and generalizing to individual-specific (i.e., *personalized*) dynamic pricing strategies.

2.2. Personalized Pricing

Our paper also relates to the empirical literature on personalized pricing and first-degree price discrimination (Rossi et al. 1996, Dubé and Misra 2022), which we extend by considering pricing dynamics.

2.3. Model-Based and Model-Free Reinforcement Learning

One popular approach to solving the dynamic coupon targeting problem is to formulate a parametric model of the consumer's responses to coupons, that is, a structural model in which the parameters are assumed to be policy invariant.[3] This approach is advantageous in that it provides a means of predicting how customers will behave in scenarios (i.e., states and/or actions) that do not arise in the historical data. For example, even if the platform sent high-discount coupons only to new consumers, the model could predict how loyal consumers would respond to the same coupons. This approach may also better account for changes in customer behavior that result from changes in the targeting policy. For instance, if consumers made myopic decisions in the historical data but become forward-looking under the new policy, the structural model could be used to estimate consumers' shifted responses.

These benefits come with some costs. First, structural models are sensitive to the specification of the consumer utility function, which is challenging when the consumer decision-making process is complex and difficult to measure, and the environment is high-dimensional and information rich. Model misspecification can result in model bias and suboptimal policy designs. We discuss the rationale for and evidence of model bias in Section 3.4.3. Moreover, the computation difficulty associated with structural model approaches could render them infeasible for deployment in high-frequency business production systems. By contrast, our reinforcement learning solution does not require the estimation of structural models and thus avoids the potential issues relating to misspecifying utility functions, model bias, and the curse of dimensionality.

Our paper contributes to the literature that uses reinforcement learning to solve sequential policy design problems in marketing (Hauser et al. 2009, Misra et al. 2019). Instead of a stateless multiarmed bandits problem,

we consider full reinforcement learning where the action can change future state transitions.

2.4. Livestream and Video Marketing

Our dynamic coupon targeting problem comes from a new business context: livestream shopping. Livestreaming and video marketing is an emerging area. Existing papers emphasize the importance of leveraging unstructured data in livestreams or videos to enhance the accuracy of predictions (Zhang et al. 2019). Similarly, we leverage summary statistics generated from image and audio features in livestreams to represent the information-rich environment experienced by consumers.

3. Data Patterns and Stylized Facts

3.1. Setting

Our empirical setting is livestream shopping. In the United States, livestream shopping is still a relatively new concept, but it has seen rapid adoption by industry giants in e-commerce, social networking, and traditional brick-and-mortar retail. Amazon launched Amazon Live in 2019, followed by competing livestream shopping initiatives from Wayfair, Facebook, YouTube, and Walmart in 2020 and 2021. In China, where livestream shopping is more established, the industry was forecasted to generate more than 500 billion RMB (USD 71 billion) in annual sales transactions in 2021.[4] Major Chinese livestream shopping platforms include Taobao Live, Kuaishou, and TikTok.

Our data set comes from one of the largest livestream shopping platforms in the world. The platform was launched in 2016 and has attracted more than two million hosts and 30 million daily active users. Figure 1 displays the user interface. The typical consumer performs the following sequence of actions.

1. Land on the search page (Figure 1(a)). When a consumer arrives on the livestream shopping platform, the landing page resembles a newsfeed. The consumer can browse the thumbnails of many livestream channels, each accompanied by a topic, seller name, average price, number of viewers, and number of likes for the livestream session. The thumbnails are ordered by a ranking algorithm similar to Google's PageRank; the algorithm considers factors such as whether a channel is currently live or not, the popularity of the host, and the preference match between the consumer and the channel.

2. Navigate to a channel page (Figure 1(b)). If interested, the consumer will click the thumbnail of a channel to go to the channel page, where the consumer can watch the livestream content and interact with the host. Interactions may take multiple forms: add items to the cart, purchase, chat, patronize, report the host for bad behavior, share the livestream on social media, and like/upvote the livestream session.

3. Receive a coupon (Figure 1(c)). Consumers are highly engaged on the platform. An average user watches two livestream videos and spends 20 minutes on the platform per week. High engagement is attributable partly to the high frequency of coupon distribution. Figure 1(c) presents an example scenario: Shortly after the consumer navigates to the channel page, a coupon for ¥15 off ¥99 pops up on the screen. (In Section 3.3, we discuss

Figure 1. (Color online) User Interface of Livestream Shopping

(a) Search Page

(b) Host Page

(c) Coupon Page

Source. https://m.znj.com/news/56292.html; https://www.seozhh.com/17429.html; http://www.chuangyejia.com/article-12355989.html.

all available coupons.) The consumer can click the coupon to claim it; the coupon will be applied automatically if the consumer makes a purchase before leaving the channel. Because livestream shopping is a relatively new product, at the time of our study the platform was still in the experimentation stage and used simple randomization to allocate coupons to customers.

4. Exit the channel. The consumer can exit the channel by clicking the "x" button in the top right corner of the screen, thereby returning to the search page. The coupon received within the channel is invalid after exit.

3.2. Summary Statistics

We collect data on 1,020,898 consumers during three months in 2019. These consumers received 25,886,094 coupons and completed 1,539,424 transactions while watching 200,568 distinct livestreams, created by 11,926 hosts who sell products in five categories: men's apparel, women's apparel, children's apparel, cosmetics, and jewelry. All consumers in the data set were "active," which the platform defines as receiving at least 10 coupons during the sample period. We focus exclusively on active users because the platform wants to create strategies to optimize the revenue from these consumers.

3.2.1. Raw Variables. Our data are at the coupon reception incidence level. Table 1 defines the variables in each observation.

3.3. Coupon Effect

The platform can send various kinds of coupons to consumers. Each coupon is characterized by two attributes: the discount value and threshold value. For example, a "¥15 off ¥99" coupon[5] has a discount value of ¥15 and a threshold value of ¥99. Because the value of the coupon is relative to the price of the product, we operationalize the coupons by calculating two metrics: the discount ratio (ratio of the discount value to the threshold value) and the threshold ratio (ratio of the threshold value to the average price in the livestream[6]). For example, if the average price is ¥120, then a "¥15 off ¥99" coupon has a discount ratio of 15.15% (= 15/99) and a threshold ratio of 82.5% (= 99/120). Consumers prefer higher discount ratios and lower threshold ratios.

Both discount ratio and threshold ratio are continuous variables, but we discretize both variables to five levels: LL (extra low), L (low), M (middle), H (high), and HH (extra high), based on the quintiles in their distributions. This approach reduces dimensionality and makes the solution more tractable. The quintiles are displayed in Table 2. In the running example of a "¥15 off ¥99" coupon, the 15.15% discount ratio falls in the LL discount level, whereas the 82.5% threshold ratio falls in the HH threshold level. In sum, we consider the platform's choice set to include 25 types of coupons (5 discount levels × 5 threshold levels).

Figure 2 displays the coupon redemption rate for each discount level and threshold level. As the discount level increases (from LL to HH), the redemption rate increases (Figure 2(a)) because a higher discount (i.e., lower price) gives a stronger incentive to purchase. As the threshold level increases, the redemption rate decreases (Figure 2(b)) because a higher threshold makes it harder for a transaction to qualify for the coupon, making the coupon less attractive to consumers.

3.4. Model-Free Evidence

Compared with one-size-fits-all coupon strategies, the benefits of moving to dynamic targeting depend on two conditions: consumer-level heterogeneity in the effectiveness of coupons, and intertemporal tradeoffs in the impacts of the coupon strategy between the

Table 1. Data Overview

Category	Variable	Definition
User ID	User ID	The user's unique identifier
Time ID	Time ID	The coupon reception incidence, that is, the number of times that the consumer has received a coupon during the sample period
Targeting variables	Consumer	A vector of static and dynamic consumer characteristics, to be introduced in Section 4.2.1
	Product	A vector of static and dynamic characteristics for each product promoted in the livestream videos watched by the consumer, to be introduced in Section 4.2.1
	Host	A vector of static and dynamic characteristics for each host of the livestream videos watched by the consumer, to be introduced in Section 4.2.1
	Livestream channel	A vector of static and dynamic video characteristics for each livestream channel watched by the consumer, to be introduced in Section 4.2.1
Coupon information	Coupon type	A vector of indicator variables that specify the type of coupon received in the incidence. The coupon types are defined by the discount and threshold ratios introduced in Section 3.3
Revenue	Revenue	The revenue generated from the coupon reception incidence. If the consumer does not purchase anything, the revenue is zero. If the consumer makes a purchase, the revenue equals the payment amount (after applying the coupon)

Table 2. Coupon Type Discretization by Discount Ratio and Threshold Ratio

	Ratio level				
	LL	L	M	H	HH
Discount ratio	0%–15%	16%–25%	26%–45%	46%–70%	71%–100%
Threshold ratio	0%–20%	21%–35%	36%–60%	61%–70%	70%–Infinity

Notes. The support of the discount ratio is from 0% to 100% and that of the threshold ratio is from zero to infinity. When the threshold ratio is greater than 100%, the consumer has to purchase multiple items to qualify for the discount.

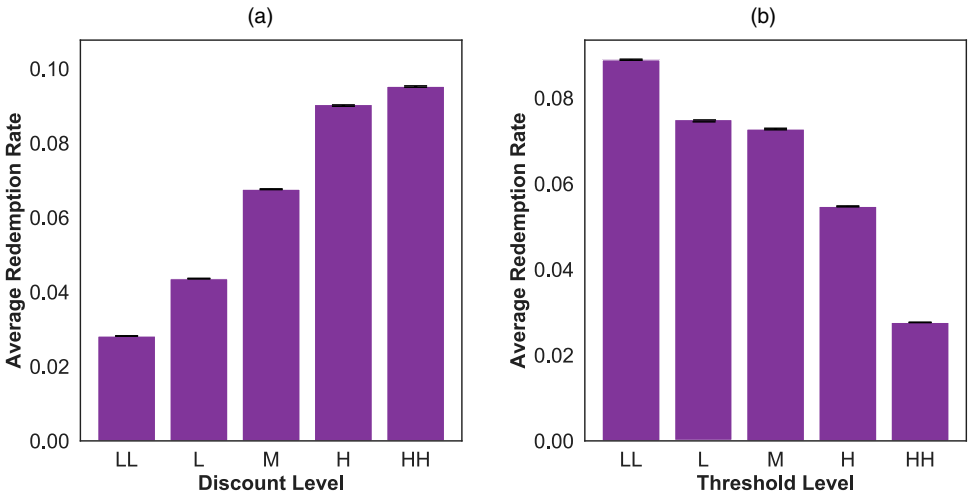
current period and future periods. Section 3.4.1 and Section 3.4.2 provide evidence of both conditions in the data. Moreover, to motivate our choice of a model-free reinforcement learning approach, we provide descriptive evidence of the possibility of model bias in Section 3.4.3.

3.4.1. Heterogeneity in Sensitivity to Discounts. Consumers have different price sensitivities, so the optimal coupon discount level likely varies across consumers. In Figure 3, we graph the effectiveness of two coupons, one with a low discount level (L, dark color) and the other with a high discount level (H, light color), for two segments of consumers, high spenders (the top 10% in the spending amount) and low spenders (the bottom 10% in the spending amount). A coupon's effectiveness is measured by the revenue it generates (i.e., the transaction amount). For high spenders, the low discount level coupon generates more revenue than the high discount level coupon (¥36.4 versus ¥31.1, $p < 0.001$); for low spenders, the pattern reverses: the high discount level coupon generates more revenue (¥3.8 versus ¥1.7, $p < 0.001$). The opposite patterns might be driven by different price sensitivities.[7] It is possible that high spenders are less price-sensitive than low spenders, so the existence of a discount may nudge the high spender toward a purchase, but the magnitude of the discount does not matter much. If so, for high spenders, low discounts may achieve the optimal balance between enticing purchases and maintaining high profit margins. For low spenders, however, their heightened price sensitivity may require deeper discounts (rather than the mere existence of a discount) to entice purchases. As these summary graphs illustrate, the consumers in our data set are heterogeneous in their sensitivities to different coupons, fulfilling the first criterion for a setting that would benefit from a dynamic targeting strategy.

3.4.2. Intertemporal Tradeoffs. This section tests whether consumers in our setting exhibit the three types of demand dynamics documented in the literature (Section 2.1). We find that consumers in our setting are affected by reference prices, but their purchase decisions do not appear to be state dependent or forward-looking.

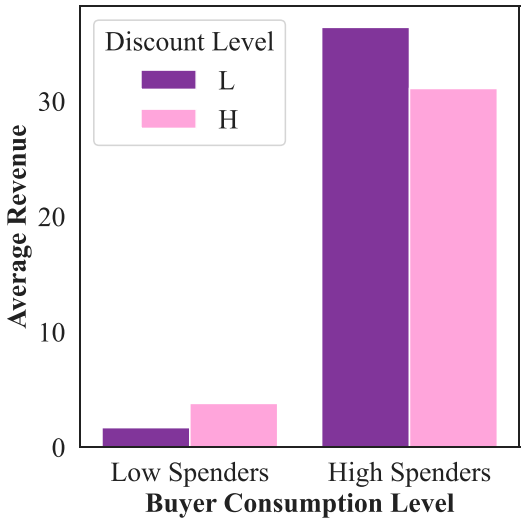
3.4.2.1. Reference Price. Reference price theory states that consumers often adopt past prices as reference points for evaluating future prices. Prices higher than the reference points are thought of as losses, and prices lower than the reference points are thought of as gains.

Figure 2. (Color online) Redemption Rate by Coupon Type



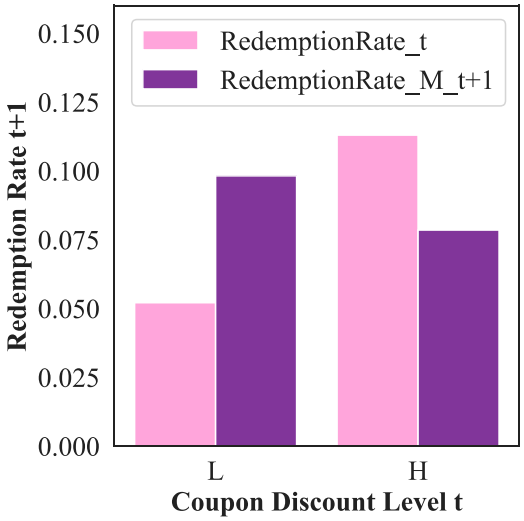
Notes. (a) By discount level. (b) By threshold level.

Figure 3. (Color online) Heterogeneity



Thus, reference price theory predicts that increasing discounts over time would lead to higher sales in later periods, whereas decreasing discounts over time would lead to lower sales. This is precisely the pattern that we see in the data. Figure 4 plots the redemption rate for two coupons, one with a low discount level (L), and the other with a high discount level (H),[8] over two consecutive coupon reception incidences: period t in light color and period $t + 1$ in dark color. Period t is the current period, that is, the period in which the coupon with either a high or low discount level was allocated; period $t+1$ is the next period, in which a middle discount level (M) was allocated. Figure 4 shows that the redemption rate in the current period is higher with a high discount than with a low discount (11.30% versus 5.21%, $p < 0.001$), but in the next period, the redemption rate is higher among consumers who received the low discount, not the high discount, in period t (7.85% versus 9.82%, $p < 0.001$). The reversal is consistent with the theory that consumers adopt the coupon level in the current period as the reference price for the next period, so the middle discount in the next period is more attractive than the reference price established by the low-discount coupon but less attractive than the reference price established by the high-discount coupon.

Figure 4. (Color online) Reference Price



3.4.2.2. State Dependence (Loyalty/Inertia and Variety-Seeking).

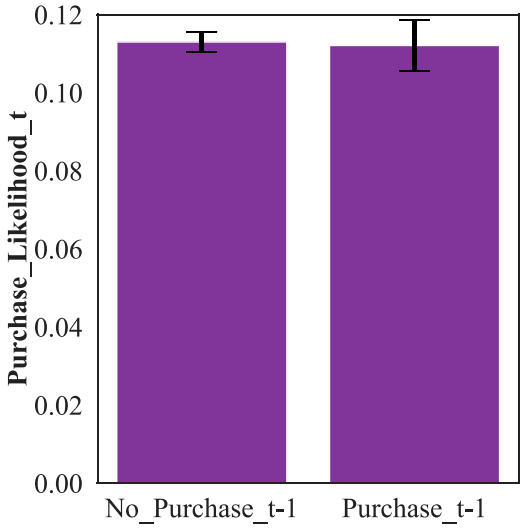
Another type of demand dynamics is the state dependence effect. A positive state dependence effect captures customer loyalty/inertia, whereas a negative state dependence effect captures variety-seeking. In our setting, state dependence would occur if a consumer who made a purchase (versus who did not make a purchase) during incidence $t - 1$ is either more or less likely to make a purchase during incidence t (Dubé et al. 2010).[9] Following the literature, we evaluate state dependence by comparing the purchase likelihoods between consumers who did and did not make a purchase in the previous period after receiving a coupon of a given discount level. In Online Appendix B, we provide additional evidence that helps disentangle structural state dependence from the possible confounds of unobserved heterogeneity.

We do not find evidence of state dependence in our setting. Figure 5 includes the subset of consumers who received high-discount coupons in two consecutive periods. Some consumers made a purchase in incidence $t - 1$, whereas others did not, but the purchase decision at $t - 1$ did not affect the purchase likelihood at t (11.30% versus 11.21%, $p = 0.58$). In other words, consumers who purchased in the previous period did not appear to have been locked in or disincentivized from making purchases in subsequent periods, as would be predicted by loyalty/inertia and variety-seeking, respectively. We speculate that this setting might lack of state dependence[10] because the most popular categories in livestream shopping are apparel and cosmetics, which are often associated with low customer loyalty.[11]

3.4.2.3. Forward-Looking.

The three demand dynamics tested in the previous sections assume that consumers are backward-looking, that is, a consumer's reaction to a coupon in the present is affected by coupons received in the past. Another form of demand dynamics involves forward-looking consumers, whose objective is to maximize not current utility but rather total discounted utility from now on. Forward-looking consumers form expectations about the platform's future targeting policies when making current decisions. For example, if a consumer foresees that future coupons will be more generous, she might forego purchases now and instead engage in strategic

Figure 5. (Color online) State Dependence



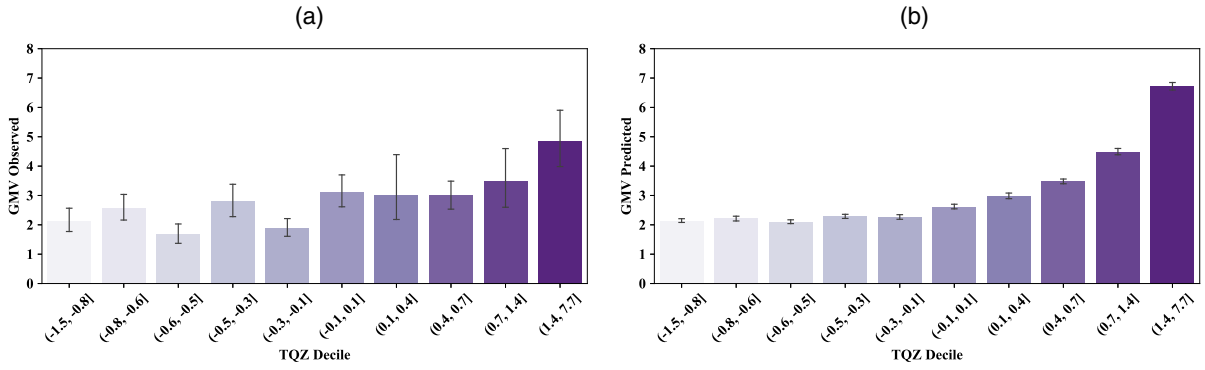
waiting. Or, if the consumer anticipates that future coupons will be less generous, she might stockpile now.

We formally test whether consumers are forward-looking in the livestream shopping setting in Online Appendix D. In short, we find no evidence of forward-looking behaviors, and we build the model framework (Section 4) under the assumption that consumers are backward-looking, only. We offer two explanations for the absence of forward-looking behaviors. First, the popular product categories on the platform are seasonal and nondurable, so they are incompatible with either strategic waiting or stockpiling. Second, the platform allocated coupons randomly during the sample period. If the platform instead used strategies such as price penetration or price skimming, consumers might have learned to behave strategically and become forward-looking in the long run. We present an extension of the model framework that allows forward-looking behaviors in Online Appendix D.2, and in Section D.3, we discuss potential pricing strategies that platforms could adopt if facing forward-looking consumers.

3.4.3. Model Bias. Modeling consumer purchase behaviors in a complex shopping environment is challenging and may be subject to model bias.[12] Model bias could emerge from two sources: distributional mismatch (i.e., covariate shift) and wrong functional forms. In distribution mismatch, the model is unbiased in the training data, which was collected under the current policy, but it becomes biased under the new policy because the two policies have different state-action distributions. Distribution mismatch is inevitable because the state-action distributions under the two policies are always different. The second source of model bias due to wrong functional forms could also exist in our data. Figure 6 shows the relationship between the consumer activity level (TQZ)[13] and gross merchandise value (GMV). Figure 6(a) shows that, in the observed data, the relationship between TQZ (discretized into deciles) and GMV is nonmonotonic, with many ups and downs. In Figure 6(b), however, a widely used machine learning model, gradient boosting decision tree (GBDT; Friedman 2001), predicts a monotonic relationship. A comparison of the observed and predicted values suggests that commonly used models often are mis-specified, thereby creating bias and leading to suboptimal policy decisions.

The previous model-free evidence indicates that to effectively implement dynamic coupon targeting in our setting, we need a model that can incorporate rich consumer heterogeneity and intertemporal tradeoffs and can avoid model bias. Rich consumer heterogeneity, intertemporal tradeoffs, and difficult-to-parameterize consumer preferences are common not only in livestream shopping settings but also for any marketer in the digital age. Before we introduce the model framework, in the next section, we discuss the generalizability of the findings from our setting to e-commerce.

Figure 6. (Color online) Relationship Between TQZ and GMV: Comparison of Observed and Predicted Values



Notes. (a) Observed. (b) Predicted.

3.5. Generalizability of the Setting

In Figure 7, we present a stylized framework that can be used to extend our model of the dynamic coupon targeting problem to a more generalized e-commerce setting.[14] Our BDRL approach analyzes the livestream shopping decision process in terms of the consumer's decisions about churn, search, and purchase, which is a framework that applies to most e-commerce settings. In our stylized sequence of events, each period t comprises three stages. In Stage 1, the consumer makes a churn decision: whether to leave the e-commerce platform permanently. If not, then the consumer visits the platform and starts the search process by choosing and visiting a product or seller's web page (in our context, the consumer clicks on a thumbnail to access a livestream channel). The churn and search decisions jointly determine the shopping context, which is defined by the features of the consumer, seller (i.e., host), product, and web page (in our context, the livestream channel). In Stage 2, the platform decides which coupon to send to the consumer or, more generally, determines the price for the consumer. In Stage 3, the consumer makes the purchase decision based on the shopping context and the coupon/price information. This concludes period t. When the next period, $t + 1$, arrives, the consumer will return to the churn decision and repeat the process. In sum, the consumer's decision process can be characterized by the churn, search, and purchase decisions.

The stylized consumer decision process applies to many e-commerce contexts including livestream shopping, traditional online shopping, customer relationship management (CRM), and mobile apps for consumer services such as Uber. In the next section, we introduce a general model framework to solve the dynamic coupon targeting problem.

4. Model

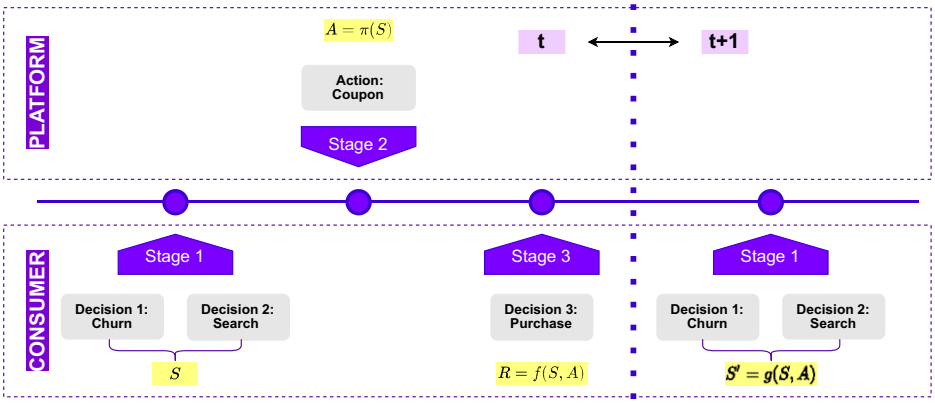
4.1. Problem Definition

We formally define the dynamic targeting problem as follows. Each consumer i ($i = 1, 2, \ldots, I$) on the e-commerce platform[15] could visit many sellers' product pages and receive a total of T_i coupons within a window of time. Consumers can receive only one coupon each time she visits a seller's page, so we treat every consumer visit to a seller's product page as a coupon reception incidence.[16] The platform chooses the coupon; in each coupon reception incidence t, the platform sends a coupon $A_{it} \in \mathbb{A}$ to consumer I, where A is the number of available coupon types and $\mathbb{A}$ is the set of all coupons.[17] The platform's decision of which coupon to send during each consumer visit is based on context information including characteristics of the consumer, seller, web page (i.e., virtual store), and product. These context characteristics are denoted as the state variables, $\mathbf{S}_{it} \in \mathbb{S}$. The platform aims to create a targeting policy $\pi(A_{it} \mid \mathbf{S}_{it}) : \mathbb{S} \times \mathbb{A} \to [0, 1]$, which is a mapping from the state and action space to probabilities (i.e., $\pi(A_{it} \mid \mathbf{S}_{it})$ is the probability of choosing action A_{it} in state $\mathbf{S}_{it}$, $0 \le \pi(A_{it} \mid \mathbf{S}_{it}) \le 1$)[18] for the purpose of maximizing its reward, $R_{it} \in \mathbb{R}$. The reward of the platform is defined as the GMV, which is the total revenue generated by consumers' purchases.[19] Intuitively, revenue is equal to the after-coupon price if the consumer stays on the platform and makes a purchase and zero otherwise. Thus, the revenue function is

$$R_{it}(\mathbf{S}_{it}, A_{it}) = \begin{cases} price(\mathbf{S}_{it}) * (1 - discount_{A_{it}}) * \mathbb{1}\{Buy_{it}(\mathbf{S}_{it}, A_{it})\} \\ \qquad\qquad\qquad\qquad \text{if } \mathbf{S}_{it} \neq Churned \\ 0 \qquad\qquad\qquad\qquad \text{othewise,} \end{cases} \tag{1}$$

where *Churned* is the absorbing state, when a consumer permanently leaves the platform[20]; $\mathbb{1}\{Buy_{it}(\mathbf{S}_{it}, A_{it})\}$ is an indicator function that takes the value of one when consumer i makes a purchase and zero otherwise; and $discount_{A_{it}}$ is the discount ratio associated with action A_{it}. The discount ratio has two competing effects on revenue: a higher discount ratio lowers the consumer payment amount but increases the purchase likelihood. The optimal discount ratio balances the two competing effects to achieve the highest revenue. The platform's

Figure 7. (Color online) Structural Model Sequence of Events



objective function is to maximize the expected, total discounted GMV across consumers and across time by creating the optimal targeting policy π^*:

$$\pi^* = argmax_\pi E_\pi\left[\sum_{i=1}^{I}\sum_{t=0}^{T_i}\delta^t R_{it}(\mathbf{S}_{it}, A_{it})\right]$$
$$= argmax_\pi E\left[\sum_{i=1}^{I}\sum_{t=0}^{T_i}\delta^t \sum_{A_{it}\in\mathbb{A}}\pi(A_{it}\mid \mathbf{S}_{it})R_{it}(\mathbf{S}_{it}, A_{it})\right],$$

where δ is the discount factor,[21] and the expectation is taken over the initial state probabilities $p(\mathbf{S}_{i0})$, the state transition probabilities $p(\mathbf{S}_{it+1}\mid \mathbf{S}_{it}, A_{it})$, and the action distribution $\pi(A_{it}\mid \mathbf{S}_{it})$.

To solve the optimization problem, we collect historical data in batch mode, meaning that all the data become available to the econometrician at the same time; the econometrician cannot collect new data on the go. The data are in a panel format where, for each consumer, we observe the {state, action, reward} tuple repeating multiple times. Formally, the data are denoted as $\mathscr{H} = \{\mathbf{S}_{i0}, A_{i0}, R_{i0}, \mathbf{S}_{i1}, A_{i1}, R_{i1}, \ldots, \mathbf{S}_{iT_i}, A_{iT_i}, R_{iT_i}\}_{i=1}^{I}$.

We note several assumptions behind our definition of the dynamic targeting problem.

- The decision maker (of coupon allocation) is the platform, not the sellers. Although each seller can determine the product price on her own web page,[22] the seller has no control over the platform's decisions about coupon allocation. This assumption might seem restrictive, but it applies to most e-commerce platforms and offline retailers that sell many products, where the coupon allocation decision is made by a centralized system.
- The action set is fixed, and there is no capacity constraint for products or coupons. As described briefly in Section 3.3 and in more detail in Section 6.1.2, we discretize the action space to $\bar{A}$ coupon types. When a consumer arrives, the platform can send any of the $\bar{A}$ coupons in the action space to the consumer. Moreover, the platform can send unlimited coupons of every type. We solve only the problem of coupon allocation as the coupon supply process is outside the scope of the current study. We assume that the coupon supply process is exogenously given and, thus, is not affected by the coupon targeting policy.
- We do not model the consumer's choice of when to use the coupon. Coupons expire quickly and can be used to purchase only the products promoted on the corresponding web page; consumers cannot accumulate many coupons over time and compare them across webpages before making a purchase decision.
- The consumer's search behavior (decision of which web page to visit and in which sequence) is exogenous and not affected by coupons.[23]
- There are no fairness concerns. Consumers are familiar with the platform's targeting practices. Although different consumers receive different coupons, they do not feel they have been treated unfairly. The ubiquity of personalized coupons from different brands justifies this assumption.[24]
- Consumers are not forward-looking. As discussed in Section 3.4.2, in each period, the consumer's objective is to maximize current utility and not future utility. Consumers cannot anticipate the platform's future strategy. In Online Appendix D, we discuss how to extend the current model to the forward-looking scenario.

4.2. BDRL Framework

As illustrated in Section 3.4.2, the coupon targeting policy likely affects both immediate revenue and future revenue because consumers may use historical coupon levels as reference points for evaluating future coupons. To solve for an optimal coupon targeting policy that has the flexibility to account for such intertemporal tradeoffs, we use a deep reinforcement learning framework (Figure 8). The deep reinforcement learning problem is defined as an MDP: $(\mathbb{S}, \mathbb{A}, p, R, \delta)$, where

- $\mathbb{S}$ and $\mathbb{A}$ denote the state and action spaces, respectively.
- At time step t, the agent of the reinforcement learning system (the e-commerce platform) takes action $A_t \in \mathbb{A}$ in state $\mathbf{S}_t \in \mathbb{S}$, receives the reward R_t, and observes the next state $\mathbf{S}_{t+1}$, according to the transition process $p(R_t, \mathbf{S_{t+1}} \mid \mathbf{S_t}, A_t)$. The transition process captures the environment that the agent faces.
- The agent's goal is to maximize the total discounted reward, which is also called the return: $Return = \sum_{t=\tau+1}^{\infty}\delta^t R_t(\mathbf{S}_t, A_t, \mathbf{S}_{t+1})$ where the discount factor is δ.

In our setting, the reward R_t is the GMV from the consumer, which is determined by how the coupons affect the consumer's ultimate purchase decision. Within the reinforcement learning framework, the "environment" is the process that generates the reward and state changes based on the agent's action and current state values. As discussed in Section 3.5, the environment is the combination of three consumer decisions: (1) churn, that is, whether to leave the platform and never come back (which affects state transitions); (2) search, that is, which web page to visit (which affects state transitions);

Figure 8. Reinforcement Learning Framework

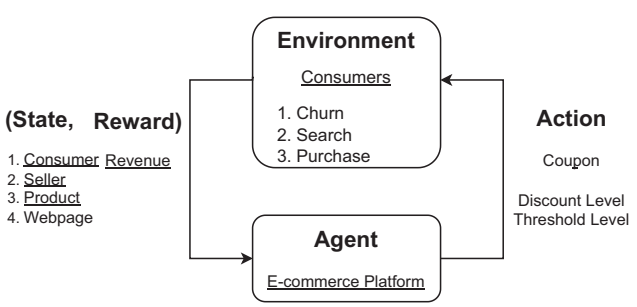


Table 3. Static State Variables

Group	State variables
Consumer	Demographics (e.g., gender), behavioral variables (e.g., purchasing power), product category preference, the consumer's preference for specific sellers
Seller	Demographics, the seller's popularity (e.g., monthly revenue, quality rating)
Product	Product category, price, market share, rating, etc.
Web page	Engagement scores, e.g., the average number of consumers who visit the web page and who add a product to the cart; unstructured data such as the web page esthetics features

and (3) purchase, that is, whether to purchase after receiving the coupon (which affects the reward and state transitions).

We provide the details for the primitives of the MDP in the following sections.

4.2.1. States. The states contain the contextual information collected by the platform for targeting purposes. In the shopping context, states can be categorized into four groups of characteristics: consumer, seller, product, and web page. Furthermore, we divide states within each group into static states (Table 3) and dynamic states (Table 4) because only some state variables are affected by actions. The distribution of a dynamic state changes after an action is performed, whereas the distribution of a static state variable is independent of actions.

4.2.1.1. Static States. Static states are provided in Table 3.[25]

4.2.1.2. Dynamic States. Our BRDL approach uses dynamic states based on the recency, frequency, and monetary value (RFM) framework (Fader et al. 2005), which is commonly used by industry practitioners to quantify customer transaction histories. Recency refers to how much time has passed since the customer has performed the activity of interest (e.g., making a purchase or visiting a web page); frequency refers to how many times the customer has performed the activity within a set period, and monetary value refers to the dollar value associated with the activity.[26] Dynamic states are categorized into five groups: consumer related, seller related, product related, web page related, and the terminal state.

The dynamic states can incorporate the intertemporal tradeoffs discussed in Section 2.1 and Section 3.4.2. Because our model-free evidence shows that state dependence (inertia and variety-seeking) is not present in our setting, we focus on the reference price effect alone.[27] We capture the reference price effect using the monetary value associated with the coupon (monetary_coupon): specifically the average, minimum, and maximum of the discount ratio and threshold ratio of the coupons received by the consumer since the beginning of the sample period.[28]

4.2.2. State Transition. We discuss the state transition process for static state variables and dynamic state variables separately.

Static states have two types of transition processes: fixed and stationary. The values of some static states (e.g., the consumer's gender) are fixed throughout the sample period. The values of other static variables change from time to time, but their distributions are stationary, not affected by the platform's actions. For example, the web page that a consumer visits, and the associated seller and product characteristics, change in each consumer visit. However, based on the data patterns in Online Appendix A, we assume that the coupon a consumer receives in the current period does not affect her web page choice in the next period. In other words, the consumer's web page choice is independent of the

Table 4. Dynamic State Variables

Group	State variables
Consumer	Related to coupon reception behaviors: the number of days since the consumer last received a coupon (recency_coupon), the number of coupons the consumer has received since the beginning of the sample period (frequency_coupon), and the average, minimum, and maximum of the discount ratio and threshold ratio of the coupons received since the beginning of the sample period (monetary_coupon)
Seller	The total number of sellers visited by the consumer since the beginning of the sample period (frequency_seller)
Product	The number of periods (coupon reception incidences) since the consumer last purchased a product (recency_product), the number of products purchased since the beginning of the sample period (frequency_product), the average, maximum, and minimum prices of the products purchased (monetary_product), and the cumulative spending since the beginning of the sample period (monetary_product)
Web page	The number of periods since the consumer last visited a web page (recency_webpage) and the number of webpages visited since the beginning of the sample period (frequency_webpage)
Terminal	Once the consumer stops coming back to the platform, the terminal state occurs. This state captures consumer churn.

coupon targeting policy. Under this assumption, we allow the state transition process for the seller, product, and web page characteristics to be stationary.

The transitions of dynamic states are all stochastic. Table 5 illustrates the product dynamic states as examples. For instance, recency_product measures the number of periods since a consumer purchased a product last time. If a consumer purchases a product in incidence t, then recency resets to zero; otherwise, recency increases by one. This transition is stochastic from one period (which consists of three stages; Figure 7) to the next because whether the consumer purchases or not is stochastic. Once the terminal state, churned (i.e., the consumer permanently leaves the platform) is reached, and there are no subsequent state transitions.

4.2.3. Action. As explained in Section 3.3, we discretize the platform's coupon choice action space using two metrics: the discount ratio and the threshold ratio. After discretization, the total number of actions is $\bar{A}$.

4.2.4. Reward. The reward function R, as defined in Equation (1), is a mapping from the state and action space to a real-valued number, $\mathbb{S} \times \mathbb{A} \to \mathbb{R}$.

4.3. Policy Learning

To solve the MDP problem described in Section 4.2, we use the batch-constrained Q-learning algorithm (BCQ; Fujimoto et al. 2019).

Algorithm 1 (Q-Learning (Watkins 1989))

1. Algorithm parameters: learning rate $\alpha \in (0,1]$, small $\varepsilon > 0$
2. Initialize $Q(\mathbf{S}, A)$, for all $\mathbf{S} \in \mathbb{S}, A \in \mathbb{A}(s)$, arbitrarily except $Q(terminal, \cdot) = 0$
3. Loop for each ε:
4. Initialize $\mathbf{S}$
5. Loop for each step of the episode:
6. Choose A from $\mathbf{S}$ using policy derived from Q (e.g., ϵ-greedy)
7. Take action $\boldsymbol{A}$, observe, R, $\mathbf{S}'$
8. $Q^{new}(\mathbf{S}, A) \leftarrow (1-\alpha)\underbrace{Q(\mathbf{S}, A)}_{old\,value} + \alpha * \underbrace{(R + \delta \max_A Q(\mathbf{S}', A))}_{update}$
9. $\mathbf{S} \leftarrow \mathbf{S}'$
10. until $\mathbf{S}$ is terminal

To understand how this algorithm works, we first introduce the concept of Q-learning (Algorithm 1). "Q" stands for the "quality" of an action taken in a given state; "Q-learning" involves a function that calculates the return (total discounted rewards) used to provide the reinforcement feedback in the learning process.[29] Before learning begins, Q is initialized to a possibly arbitrary fixed value (line 2 of algorithm 1). Then, at each time t (line 5), the agent (the platform, in our case) selects an action A (line 6), observes a reward R and a new state $\mathbf{S}'$ (which may depend on both the previous state and the selected action, line 7). Given the new reward R and new state $\mathbf{S}'$, Q is updated to Q^{new} using a weighted average of the Q value from the previous iteration and the new information $(R + \delta \max_A Q(\mathbf{S}', A))$; the weight α is the learning rate (a hyperparameter), and δ is the discount factor (line 8).

Q-learning is model-free because it does not rely on any functional form assumptions about the reward function or the state transition function. Rather, Q-learning uses samples collected from the environment (in the historical data, in our case) to directly observe the reward and the next state. The sampled values (i.e., R and $\mathbf{S}'$ in step 8 of Algorithm 1)[30] enable the Q-function to update even though we know only the reward value and not the functional form of the reward as a function of the state and action.

The original Q-learning algorithm is appropriate when the state and action spaces are small. For enormous state and action spaces (e.g., with continuous state values), however, we cannot expect to find an

Table 5. Transition Processes of Dynamic State Variables (RFM)

Variable	Transition
Recency_product	$r\_product_{t+1} = \begin{cases} 0 & \text{if } purchase_t = 1 \\ r\_product_1 + 1 & \text{otherwise} \end{cases}$
Frequency_product	$f\_product_{t+1} = \begin{cases} f\_productt_t + 1 & \text{if } purchase_t = 1 \\ f\_productt_t & \text{otherwise} \end{cases}$
Monetary_product	$avg\_product_{t+1} = \begin{cases} \dfrac{avg\_product_t + price_t}{f\_product_t + 1} & \text{if } purchase_t = 1 \\ avg\_product_t & \text{otherwise} \end{cases}$
	$max\_product_{t+1} = \begin{cases} price_t & \text{if } price_t > max\_product_t \\ max\_product_t & \text{otherwise} \end{cases}$
	$min\_product_{t+1} = \begin{cases} price_t & \text{if } price_t < min\_product_t \\ min\_product_t & \text{otherwise} \end{cases}$

Notes. This table presents only the transition process for product-related dynamic states. The corresponding transition process for consumer-related, seller-related, and web page-related dynamic states are derived similarly.

optimal value function within the limit of infinite time and data. Instead, the value function needs to be a parameterized function, for example, DNNs with many hidden layers (Mnih et al. 2015) or the random forest (RF) algorithm (Kim et al. 2021). We choose DNNs, a parametric model with a large number of parameters, as the functional approximator.[31]

Another important feature of our solution is that we use batch reinforcement learning, which means that our data set is fixed; no further interactions between our policy and the environment can occur, and our problem does not involve the exploration-exploitation tradeoff. We choose to use batch reinforcement learning because we face a high-stakes problem involving millions of consumers and billions of dollars in revenue. A suboptimal proposed policy could cause a significant profit loss for the platform and could hurt customer experience and satisfaction. Batch reinforcement learning enables us to ensure the safety of our proposed policy before launching a field experiment. (In practice, one would need to train the model with the newest available data to ensure that the model does not become stale.)

Although batch analysis using fixed historical data sounds like the standard practice in econometric modeling, batch reinforcement learning algorithms have been shown to be susceptible to extrapolation error (Fujimoto et al. 2019), induced by generalization from the neural network function approximator. When selecting action A, such that the pair $(\mathbf{S}, A)$ is distant from the data contained in the batch, the estimate $Q(\mathbf{S}, A)$ may be arbitrarily poor, introducing extrapolation error. We address this extrapolation error problem by using a BCQ algorithm (Fujimoto et al. 2019), which favors a state-action visitation similar to some subset of the provided batch. Specifically, the algorithm will adjust a threshold value τ for a state-action pair and will allow only the actions whose relative probability is above the threshold τ (line 5). The probability of actions in each state is calculated by training a generative model G_ω (lines 2 and 7). We use the GBDT model as G_ω.

Algorithm 2 (BCQ)

1. Input: Batch data $\mathcal{B}$, horizon T, target network update rate Υ, mini-batch size N, threshold τ.
2. Initialize Q-networks Q_θ, generative model G_ω (trained in a standard supervised learning fashion, with cross-entropy loss), and target networks $Q_{\theta'}$, with $\theta' \leftarrow \theta$.
3. for $t = 1$ to T, do:
4. Sample mini-batch M of N transitions $(\mathbf{S}, A, R, \mathbf{S}')$ from $\mathcal{B}$
5. $A' = \arg\max_{A' | \frac{G_\omega(A'|\mathbf{S}')}{\max_{\hat{A}} G_\omega(\hat{A}|\mathbf{S}')} > \tau} Q_\theta(\mathbf{S}', A')$
6. $\theta \leftarrow \arg\min_\theta \sum_{(\mathbf{S},A,R,\mathbf{S}')\in M} l_\kappa(R + \gamma Q_{\theta'}(\mathbf{S}', A') - Q_{\theta'}(\mathbf{S}, A))$
7. $\omega \leftarrow \arg\min_\omega - \sum_{(\mathbf{S},A)\in M} \log G_\omega(A \mid \mathbf{S})$
8. If $t \bmod \Upsilon = 0$: $\theta' \leftarrow \theta$
9. end for

4.4. Evaluation

4.4.1. In-Sample Policy Evaluation. After the model learns a new policy, the next question is whether the policy is effective. Because we have only historical data, we need to solve an in-sample off-policy policy evaluation problem, where the policy to be evaluated is different from the policy used to generate the historical batch data. Formally, given a new policy, π^e, we want to calculate the value function associated with this policy, V^{π_e}:

$$V^{\pi_e} = E_{P,\pi_e}\left[\sum_{t=0}^{T} \delta^t R_t\right]. \tag{2}$$

Equation (2) also represents the net present value of a customer or, loosely speaking, the customer lifetime value (CLV; Fader et al. 2005) (associated with coupon redemption) during time frame T. For ease of interpretation, we report the policy performance in terms of the CLV in later sections.

We estimate the value of the new policy using the doubly robust estimator (Dudík et al. 2014). Most studies in the reinforcement learning literature use one of two classes of methods for policy evaluation. One class is the inverse propensity weighting (IPW) method, also known as the importance sampling (IPS) method. The idea behind IPS is that, when making counterfactual predictions for a new policy that generates a different action distribution than the distribution observed in the data, we can use propensity scores to reweigh the original observations such that we can use the reweighed observations as a control group for estimating the value of the new policy. In other words, IPS uses importance weighting to correct for the incorrect proportions of actions in the historical data. The second class of approaches for policy evaluation is called the direct method, which uses the historical data to learn the functional mapping between the state-action space and the reward. Then, the estimated reward function can be used to calculate the value function. The doubly robust estimator combines the two methods, so it is unbiased if either the propensity score estimator is correct or the direct method estimator is correct (i.e., the two methods do not have to be correct at the same time).

Mathematically, the doubly robust estimator is

$$V^{\pi_e} = E_n\left[\sum_{t=0}^{T-1} \delta^t \left(\underbrace{\xi_{0:t}[R_t - m_t(\mathbf{S}_t, A_t)]}_{\text{Importance Sampling: low bias}} + \underbrace{\xi_{0:t-1}\{\Sigma_{A\in\mathcal{A}} m_t(\mathbf{S}_t, A)\pi_e(A \mid \mathbf{S}_t)\}}_{\text{Direct Method: low variance}} \right)\right], \tag{3}$$

where $\xi_{0:t} = \prod_{\tau=0}^{t} \frac{\pi_e(A_\tau|\mathbf{S}_\tau)}{\pi_b(A_\tau|\mathbf{S}_\tau)}$ are the inverse propensity weights, and $\pi_b(A_\tau \mid \mathbf{S}_\tau)$ is the behavioral policy. In our case, the behavioral policy is known because the platform used a random allocation policy with predetermined action probabilities; $m_t(\mathbf{S}_t, A_t)$ is the direct method estimator, and for the functional form, we chose GBDT in which the Q value is the dependent variable and the state action variables $(\mathbf{S}, A)$ are the independent variables. When the behavioral policy is known, both the IPS and the doubly robust methods produce unbiased evaluations, but the doubly robust estimator has lower variance than IPS (Dudík et al. 2011). We use the doubly robust estimator in our application to compare BDRL against a set of benchmarks, introduced in Section 5.

4.4.2. Field Experiment.

We also provide out-of-sample evaluation of the new policy using a field experiment, in which we divide all the users into conditions with different coupon targeting policies: a fully random allocation policy in the control condition, the policy learned using the BDRL algorithm in one treatment condition, and a benchmark policy in another treatment condition.

5. Benchmark Policies

Now we introduce the benchmark algorithms. Table 6 presents the design matrix, which has three components: whether the objective is to design a static or dynamic targeting policy, whether the treatment is homogeneous or heterogeneous across consumers, and whether the solution is model-based or model-free. To generate the static homogeneous treatment policy benchmark, we use a linear regression without the interaction between states and actions (Section 5.1). We generate the static heterogeneous treatment policy benchmarks using three alternative models: GBDT, DNN, and orthogonal random forest (ORF). We generate the dynamic heterogeneous treatment policy using a structural approach (Section 5.3). Finally, BDRL is the model-free dynamic heterogeneous treatment approach (Section 4.2).

5.1. Static Policy with Homogeneous Treatment

Our first benchmark is the static targeting policy with homogeneous treatments across consumers. We use a linear regression model without the interaction between states and actions (Equation (4)). The dependent variable is the revenue from one coupon reception incidence, and the independent variables are states and actions. The model estimates the average treatment effect of each coupon (action A) and chooses the optimal coupon, that is, the coupon that generates the highest predicted reward, $\hat{R}(\mathbf{S}, A)$ (Equation (5)). Although states are used to estimate the model coefficients, states do not affect the choice of the optimal action because there is no interaction between states and actions in Equation (4). In fact, the optimal action is the same across all states. In other words, all consumers will receive the same coupon, which is chosen to maximize revenue.

$$R_{it} = \beta_0 + \mathbf{S}_{it}\boldsymbol{\beta}_s + \beta_A A_{it} + \epsilon_{it}, \epsilon_{it} \sim N(0, \sigma^2) \tag{4}$$

$$\pi(A \mid \mathbf{S}) = \begin{cases} 1 & A = \arg\max_{a \in \mathbb{A}} \hat{R}(\mathbf{S}, A) \\ 0 & o.w. \end{cases} \tag{5}$$

5.2. Static Policy with Heterogeneous Treatments

The homogeneous treatment approach is hardly optimal because it assumes that one coupon is most effective for every consumer in every context. By contrast, static targeting policies that allow for heterogeneous treatments recognize differences among consumers, although these policies still optimize for each period separately. This type of static targeting strategy is prevalent in the industry and can be categorized into two groups: indirect or direct methods. Most indirect methods are machine learning algorithms that use the state variables and actions as inputs to predict the reward output. After the model has been trained, a postselection exercise loops over all the actions and picks the one that is predicted to generate the highest reward. We use GBDT, which is used by Facebook (He et al. 2014), and DNN as the indirect method benchmarks. The direct methods[32] were developed more recently in the causal inference literature. In contrast to indirect methods, direct methods calculate the heterogeneous treatment effect of each action and identify the optimal action as the one with the highest treatment effect. We choose the ORF model (Oprescu et al. 2019) as the direct method benchmark because it is less sensitive than others with respect to the estimation error of nuisance parameters. In the extant literature, it

Table 6. Benchmark Models

		Objective	
Treatment	Model	Static	Dynamic
Homogeneous		Regression without interaction (Section 5.1)	
Heterogeneous	Model-based	GBDT with interaction (Section 5.2)	
		DNN with interaction (Section 5.2)	
		ORF (Section 5.2)	Structural (Section 5.3)
	Model-free		BDRL

is unclear how direct methods perform compared with indirect methods, so we empirically test both.

5.3. Dynamic Policy with Heterogeneous Treatments: Structural Model

Model-based dynamic algorithms, unlike the methods in Section 5.2, consider the intertemporal tradeoffs created by targeting policies. We take a structural approach in which we construct a function f for the reward (Equation (6)) and a function g for the state transition process (Equation (7)).

Both the reward function and the state transition function depend on the consumer decision process. As explained in Figure 7, each period t comprises three stages. In Stage 1, the consumer makes a churn decision (i.e., whether to continue using the platform or to leave permanently); if the consumer chooses to continue using the platform, she starts the search process. The churn and search decisions jointly determine the state variables $\mathbf{S}$, which comprise the consumer, seller, product, and web page features. In Stage 2, the platform decides which action to take based on a policy function $\pi(\mathbf{S})$. In Stage 3, the consumer makes the purchase decision based on the state and action she observes. The purchase decision results in a reward for the platform, and the reward function is specified as $R = f(\mathbf{S}, A)$. This concludes period t. In the next period $(t + 1)$, the consumer again makes the churn and search decisions, which jointly determine the new state $\mathbf{S}'$. The state transition is specified by function g, so $\mathbf{S}' = g(\mathbf{S}, A)$.

The existing marketing literature has proposed many models for the search, purchase, and churn behaviors. In Online Appendix G, we present the sequential search model in a full structural fashion.[33] We also consider a simplified version in which both the f and g functions are formulated as GBDT models (Equations (6) and (7)), which are more flexible than the full structural model. Once we fit the reward function $(f(\mathbf{S}, A))$ and the state transition function $(g(\mathbf{S}, A))$, we use backward induction to calculate the Q function (i.e., the policy-specific value function; Equation (8)). The optimal policy is the one that maximizes the Q function (Equation (9)).

$$\hat{R}_{it}(\mathbf{S_{it}}, A_{it}) = \sum_{k=1}^{K} f_k(\mathbf{S_{it}}, A_{it}), f_k \in \mathcal{F} \tag{6}$$

$$\mathbf{S}_{it+1} = \sum_{l=1}^{l} g_l(\mathbf{S_{it}}, A_{it}), g_l \in \mathcal{G} \tag{7}$$

$$Q(\mathbf{S}, A) = \hat{E}\left[\hat{R}_A + \delta \max_{A'} Q(\mathbf{S}', A') \,\middle|\, \mathbf{S}\right] \tag{8}$$

$$\pi(A \mid \mathbf{S}) = \begin{cases} 1 & A = \arg\max_{A \in \mathbb{A}} \hat{Q}(\mathbf{S}, A) \\ 0 & o.w. \end{cases} \tag{9}$$

6. Empirical Application and Results

6.1. Empirical Application

This section discusses the empirical application of the modeling framework presented in Section 4. First, in Section 6.1.1 and Section 6.1.2, we introduce the primitives of the MDP (namely, the states and action) in our empirical context of livestream shopping. Then, in Section 6.1.3, we present the train-test split and hyperparameters.

6.1.1. States. States can be categorized into four groups: consumer, seller (host), product, and web page characteristics. We provide the specific variables used in the livestream shopping setting and their summary statistics in Online Appendix H.

6.1.2. Action. As explained in Section 3.3, we discretize the action space into 25 coupons, each characterized by two metrics, the discount ratio and the threshold ratio, with five levels each based on the quintiles in their respective distributions. Importantly, the action set does not change over time and can be set ex ante, exogenously. Although the cutoff points depend on the data, which may vary over time, the entire support of the two dimensions can be fully covered by the five levels. For instance, for the threshold ratio, the HH level spans 70% to infinity; for the discount ratio, the HH level spans 70%–100%. We chose the two dimensions carefully, with sufficiently broad numerical support to accommodate the anticipated range of experimentation. For example, the largest threshold ratio observed in our data are 300%, but if a coupon with a 400% threshold ratio is required in the future, it would still be categorized as the HH type.

6.1.3. Train-Test Split. We need to train a reward prediction model and then derive the corresponding policy for the three benchmark models: the linear regression model (Section 5.1), GBDT, and DNN (Section 5.2). To assess the prediction accuracy of the three models, we create the training and test sets with an 80–20 split (i.e., the training set contains 80% of the consumers, and the test set contains the other 20%). All observations from the same consumer belong to the same subset. We provide the model prediction accuracy in Online Appendix I and the hyperparameters for the different estimation approaches in Online Appendix J.

6.2. Policy Evaluation

We measure the "in-sample" performance of the dynamic coupon targeting policy using the doubly robust estimator (Equation (3)) introduced in Section 4.4.1.

Table 7 compares the CLV estimates from our model-free dynamic targeting policy (BDRL) with those from

Table 7. Model Comparison Based on the Doubly Robust Estimator

		1. Static policy with homogeneous treatments	2. Static policy with heterogeneous treatments			3. Model-based dynamic policy with heterogeneous treatments	4. Model-free dynamic policy with heterogeneous treatments
		A: Regression	B: GBDT	C: DNN	D: ORF	E: Structural[a]	F: Proposed BDRL
CLV	Mean	6.53	7.52	6.95	7.49	8.37	9.53
(Return)	Std	2.00	2.29	2.27	2.29	2.77	3.23
Gain	Mean	11%	28%	18%	28%	43%	62%
t test[b]		222.99	524.25	346.60	515.47	714.27	945.57
p value		<0.001	<0.001	<0.001	<0.001	<0.001	<0.001

Notes. All the gains are compared with the mean CLV of ¥5.87. The total sample size is 1,020,898 consumers.
[a]The result is for the full structural model. The result from the simplified version is qualitatively similar and available upon request.
[b]The t tests compare the mean CLV of different policies. Online Appendix P reports distributional differences measured by the Kolmogorov-Smirnov test.

the benchmark policies. The mean CLV observed in the data are ¥5.87[34]; the value is relatively small because only 6% of the incidences result in a purchase (i.e., 94% of the incidences have a reward of zero). The static policy with homogeneous treatments, estimated using linear regression, increases the CLV by 12% compared with the original coupon allocation rule, which was purely random without any optimization. The three static policies with heterogeneous treatments, estimated using GBDT, DNN, and ORF, all increase the CLV by around 20%; the improvement over the static policy with homogeneous treatments is evidence of the importance of personalization. Interestingly, our results corroborate existing findings that there is no strict hierarchy in the relative effectiveness of indirect methods (GBDT and DNN) and direct methods (ORF). The dynamic policy with heterogeneous treatments, estimated using a structural model, increases the CLV by 43%, indicating that demand dynamics play an important role in coupon effectiveness, so it is important to account for how coupon targeting strategies affect consumers over time. Finally, the dynamic targeting policy generated using BDRL increases the CLV by 63%, which translates to an increase of $13.1 million (91.8 million RMB) from the consumers in our sample and $18 billion (126 billion RMB) from the entire customer base during the three-month sample period.[35]

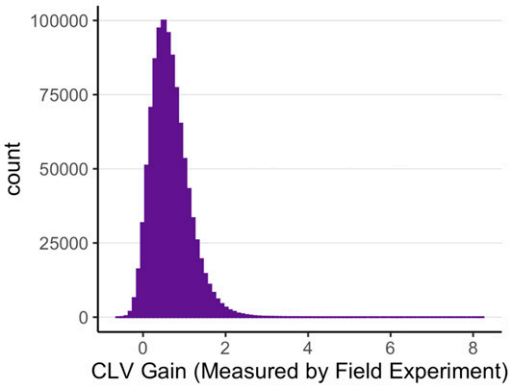
6.3. Field Experiment

We further test the performance of the new policy using an "out-of-sample" field experiment on the platform. The consumers who participated in the field experiment are the same as those in our batch data (1,020,898 consumers). To minimize the risk of losing revenue, we test only three coupon allocation strategies: (1) random allocation, (2) model-based dynamic targeting based on the structural model, and (3) model-free dynamic targeting based on BDRL.[36] Consumers in our sample were assigned to conditions (1), (2), or (3) with probabilities of 80%, 10%, and 10%, respectively.[37] The experiment lasted two weeks in January 2020. Table 8 reports the results. The return value (total discount rewards during the experiment period, or CLV) is rescaled for privacy protection purposes. As shown, the model-based dynamic targeting policy (structural model) was 39% more effective than the random allocation policy, confirming the importance of incorporating demand dynamics and using optimization to determine the best coupons. The model-free dynamic targeting policy (BDRL) performed even better,

Table 8. Field Experiment Results

		Random allocation	Model-based dynamic policy with heterogeneous treatments (structural)	Model-free dynamic policy with heterogeneous treatments (BDRL)
CLV	Mean	6.98	9.70	11.16
(Return)	Std	2.43	3.15	3.86
Gain	Mean	~	+39%	+60%
t test		~	690.80	925.95
p value		~	<0.001	<0.001

Figure 9. (Color online) Histogram of the CLV Gain



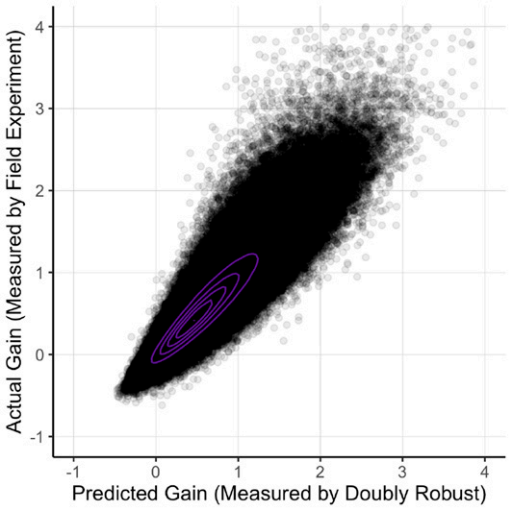
with a 60% increase in the CLV. In Figure 9, we plot the histogram of the CLV gain (the percentage increase in the CLV from random allocation to the BDRL policy). The distribution has a log-normal shape with a mean of 60%. For some consumers, the CLV gain is as large as 800%.

The model-free approach outperforms the model-based approach by avoiding model bias. We were able to use the model-free approach in this case because two conditions were met: data from the firm's previous randomized coupon experiments provided sufficient stochasticity such that we do not need to rely on functional form assumptions, and the empirical evidence suggests an absence of structural shifts in consumer behaviors (e.g., forward-looking) under alternative policies. As such, whether our model-free BDRL approach or a structural model approach would provide better results depends on the specifics of the setting. However, when large amounts of randomized historical data are available and structural shifts in consumer preferences are unlikely, our BDRL approach provides substantial advantages.

Tables 7 and 8 show that the findings from the field experiment are consistent with the policy evaluation results. For cross-reference, in Figure 10, we plot the predicted CLV gain of each consumer using the doubly robust estimator against the actual CLV gain in the field experiment. The predictions visually align with the realized values.

The dynamic targeting policy generated using our BDRL approach performs the best of the three strategies tested in the field experiment, but we warn readers not to overgeneralize this finding because of multiple caveats. First, our policy change did not "systematically alter the structure" (Lucas 1976, p. 279), so structural models might not have clear advantages in our setting. It is possible that our model's superiority comes from the similarity between the environment in the field experiment and the environment in which we collected the estimation data. If the estimation data had been collected under an alternative pricing policy, then the model-free BDRL approach might not have done better. Moreover, the experiment was a very short-run intervention during which consumers might not have had time to adjust their preferences or behaviors. Another caveat is that we tested BDRL against only two alternatives (the structural model policy and random allocation). We encourage future research to consider other policies such as static targeting (with either indirect or direct methods), which is widely used in the e-commerce and advertising industry, or a homogeneous dynamic policy that includes only intertemporal price discrimination, such as markdown pricing or markup pricing. Future research may also consider alternative structural model specifications (see one example that does not model search behaviors in Online Appendix L).

Figure 10. (Color online) Actual vs. Predicted CLV Gain (%)

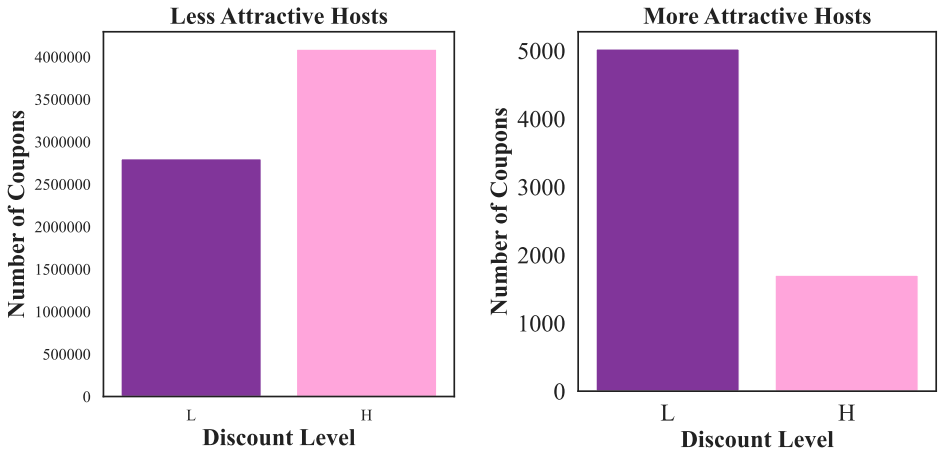


6.4. Targeting Rules

This section provides substantive insights into the targeting rules created by BDRL. The next three sections describe how our targeting rules can determine who to target (heterogeneity), when to target (dynamics), and the interaction between who and when to target.

6.4.1. Whom to Target (Heterogeneity). Our targeting policy recommends whom to target, that is, which coupon is the most effective for each consumer, given the consumer's context. One example of a context variable that is unique to livestream shopping is the host's attractiveness. Figure 11 displays the coupon frequency by the discount level and host attractiveness. The left panel is for less-attractive hosts (below the median of attractiveness), and the right panel is for more-attractive hosts. Our targeting policy recommends sending more high-discount (low-discount) coupons to

Figure 11. (Color online) Targeting Rule Under BDRL: Host Attractiveness



consumers who are watching livestream content created by less-attractive hosts (more-attractive hosts), perhaps reflecting a compensatory effect in which consumers derive more utility from attractive hosts and hence demand less monetary reward.

6.4.2. When to Target (Dynamics). Our targeting policy also determines when to target, which can help us understand why the BDRL algorithm performs better than the static benchmarks. We plot the action distributions under the two policies in Figure 12: the static policy (GBDT) in the left plot and the dynamic policy (BDRL) in the right plot. The horizontal axis is the N_th coupon reception incidence. We consider a new customer (incidence $n = 1$) versus a loyal customer ($n = 5$). For new customers, GBDT recommends more high-discount coupons, whereas BDRL recommends more low-discount coupons. More generally, BDRL recommends giving increasing discounts over time, consistent with the reference price effect: if the platform gives high-discount coupons to new consumers, they may use the highly discounted prices as references in the future, making them less likely to respond to future incentives. The dynamic policy appears to recognize the long-term negative consequence of providing high-discount coupons to new consumers and thus adopts a more conservative approach of gradually increasing the discount level as consumers become more loyal.[38]

6.4.3. When to Target Whom. Finally, our targeting policy recommends combining cross-sectional and intertemporal price discrimination, that is, modifying the coupon allocation strategy based on the consumer's experience on the platform. Figure 13 plots the average discount ratio by the coupon reception incidence (from the 1st to the 25th incidences) for two segments of consumers: high spenders (square symbol) and low spenders (triangle symbol). The policy recommends giving low spenders (relative to high spenders) slightly higher discounts initially and increasing the discount at a faster rate as the consumers gain experience. This coupon policy may be responding to higher price-sensitivity among low spenders than high spenders, providing larger and larger discounts to these customers to keep them continuously engaged.

Figure 12. (Color online) Comparison of Static and Dynamic Targeting Policies

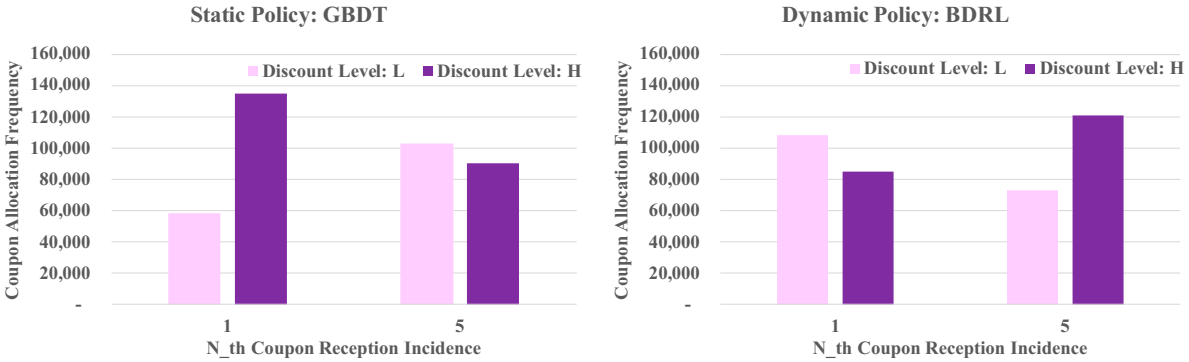
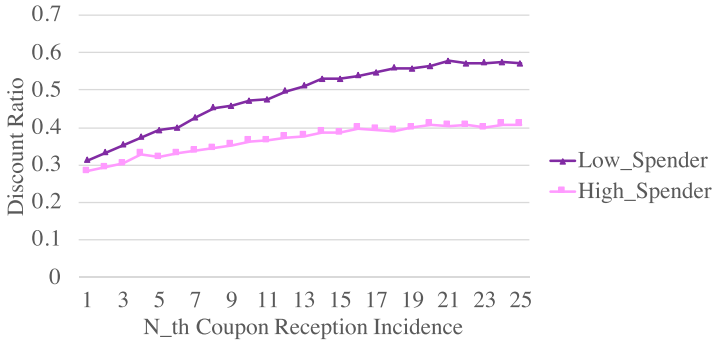


Figure 13. (Color online) Targeting Rule: When to Target Whom



7. Conclusion and Future Directions

We design a dynamic theoretical framework, BDRL, which incorporates the intertemporal tradeoffs in dynamic pricing and coupon targeting to create a policy that maximizes a platform's revenue. We empirically evaluate the performance of the dynamic coupon targeting policy relative to diverse benchmarks: based on static and dynamic objectives, using model-based and model-free estimation, and recommending heterogeneous and homogenous treatments across consumers. Using a large-scale field experiment, we demonstrate that our BDRL approach can increase the livestream shopping platform's GMV by more than 60%—far superior to the approximately 30% increase from static targeting policies and the 40% increase from a dynamic structural model-based policy. Our BDRL framework is suitable for the high-frequency and high-dimensional pricing problems that are common in e-commerce settings.

Although our findings illustrate that BDRL can be effective at solving dynamic pricing and coupon targeting problems, we acknowledge several limitations that open the door for future research.

First, we assume that consumers are not forward-looking because we did not observe evidence of forward-looking behaviors in our historical data. However, such behaviors could emerge in other settings. What would be the optimal pricing strategy for forward-looking consumers? Future research can test a number of predictions from the theoretical literature on dynamic pricing, as we discuss in Online Appendix D.3.

Another limitation of our framework is the lack of unobserved heterogeneity, that is, consumers' heterogeneous preferences and sensitivities to coupons beyond the observed features. Our data cover millions of consumers but for only a limited time (three months), so the identification relies heavily on pooling across customers and the assumption of no unobserved heterogeneity.[39] Future research could explore two solutions: formulate the problem as a partially observable MDP (Hauser et al. 2009) or use multiagent reinforcement learning and treat each consumer segment as a separate agent.

One other limitation in our study is sample selection because we examine only "active" users. Although we conducted a sensitivity test in Online Appendix M, it would be interesting to explore whether our conclusions generalize to less-active or inactive users, where the zero-inflated nature of the purchase frequency distribution could make the estimation challenging and require new methodological solutions.

Future research could also explore demand dynamics on a longer horizon. It is possible that the loyalty effect or variety-seeking effect would dominate the reference price effect in the long run.

Finally, due to data limitations, we could not directly incorporate unstructured data, such as image, audio, and video features, as state variables. Instead, we used summary statistics that were predefined by the company. However, livestream shopping involves an information-rich environment, so we hope that future research will test the informational values of both structured and unstructured data.

We see BDRL as a powerful framework for solving many marketing problems associated with sequential decision making, such as ad targeting, chatbot conversation design, recommender systems, and dynamic targeted pricing in both online and offline settings. We encourage future marketers to experiment with and apply BDRL.

Acknowledgments

The author thanks seminar participants at The University of Arizona, Bocconi University, Central European University, University of Chicago, Dartmouth College, Korea Advanced Institute of Science & Technology, London Business School, London School of Economics, Nanyang Technological University, New York University, Peking University, Spotify, Stanford University, and University of Southern California

for helpful comments; and Jiawen Yan, Zexi Ye, Xinyu Wei, Gaomin Wu, and Qi Zhao for excellent research assistance.

Endnotes

[1] Statista (2017 October). Retrieved from https://bit.ly/3CXIunS.

[2] Juniper Research (November 2017): shorturl.at/ehCEN.

[3] For example, Bertsekas (2019) outlines a method that involves estimating two structural models (one for reward and another for state transition) solving the exact dynamic programming problem using the Bellman Equation and then using the solution to inform the new coupon policy.

[4] Hallanan L (2019 March 15). Retrieved from https://bit.ly/2UwavBC.

[5] The displayed coupons include only these two values with no other variation in message content.

[6] The "average price" includes all products displayed in the same livestream video. On the platform, a consumer receives only one coupon per livestream video, but the coupon can be applied to any of the products featured in the video. We calculate the threshold ratio with the average price because we need to assign a threshold value to every coupon reception incidence, even when we do not know which product the coupon was applied to, for instance, when a consumer did not purchase any products. Usually, all products within the same livestream have similar prices, so the threshold ratio based on the average price represents the consumer's overall impression of the difficulty of meeting the coupon's threshold.

[7] We interpret the findings in Figure 3 as the causal impact of the coupon type on revenue by consumption level. Although high spenders and low spenders may choose different livestream channels or products, within each channel, the coupons are randomly allocated (according to the platform's current strategy). Thus, the interpretation of the plot is not subject to selection bias; that is, consumers do not self-select into different treatments (coupon types).

[8] The same effect holds when we use any other pair of the five discount levels, LL, L, M, H, and HH.

[9] We test state dependence at the level of the platform, not the brands, because the decision maker in our problem is the platform, not brand manufacturers or hosts. Even if consumers develop loyalty to specific hosts (we show evidence of this in Online Appendix A), this behavior does not affect the dynamic pricing decision for the platform.

[10] Prior research (Gedenk and Neslin 1999) also found mixed evidence of state dependence.

[11] Evidence of low customer loyal in apparel and cosmetics can be found here (https://bit.ly/2UwavBC) and here (https://bit.ly/3zHzvFi). One possible explanation for the absence of variety-seeking is that only the frequent buyers become satiated enough to seek variety. We test this hypothesis by dividing consumers into two groups: infrequent buyers and frequent buyers. Results in Online Appendix C indicate no variety-seeking even for the frequent buyers.

[12] See Online Appendix E for an example.

[13] TQZ is a score that the platform uses to indicate the consumer's activity level (higher TQZ is more active), calculated from the consumer's recent search and purchase histories, payment amounts, and review posting activity.

[14] The mathematical notations in Figure 7 are explained in Section 5.3.

[15] In the general framework, we use a single term, "the e-commerce platform," to represent all the entities that face the dynamic targeting problem. Examples include livestream shopping platforms, generic e-commerce platforms, and firms using CRM apps.

[16] If a consumer visits more product pages, she receives more coupons. There are no browsing or purchase occasions without a coupon.

[17] For example, in our field setting, the set of coupons $\mathbb{A}$ that is available to the platform consists of coupons that reflect any combination of discount values and threshold values that the platform would like to implement.

[18] We use the specification of stochastic policies, where each action has a probability between zero and one, because the platform used stochastic allocation policies to generate our training data. The notation for stochastic policies is more general and all-inclusive than that for deterministic policies, where the probability of each action is either zero or one.

[19] Managers at the company and other e-commerce companies told us that the GMV usually is the objective function of choice for pricing decisions on e-commerce platforms. Our BDRL method can easily be adapted to maximize other reward objectives such as gross or net profits.

[20] The platform defines a "churned consumer" as one who has not used the platform for X consecutive days, where X is a number between 1 and 365. We cannot disclose the exact value X because it is a trade secret. This type of churn definition is an industry common practice and consistent with the "silent" churn definition in Ascarza et al. (2018). We acknowledge that our definition of churn is only an approximation because, in our non-contractual setting, there is no mechanism that would prevent a consumer from leaving and then returning to the platform. However, customer attrition from the platform is a real phenomenon, and management recognizes that if a customer remains inactive for a long time, then she is statistically unlikely to return to the platform. We use a simplified assumption in which consumers choose between staying and permanently leaving because it allows us to formulate the problem as an episodic MDP (with a terminal state instead of a continuing one), which "is mathematically easier because each action affects only the finite number of rewards subsequently received during the episode" (Sutton and Barto 2018). Of course, the policy learned under this assumption may be more myopic than the optimal one. If some consumers can come back to the platform after a "hibernation" period (X or more days), then our policy overlooks the impact of a before-the-hibernation coupon (action) on the after-the-hibernation revenue (reward). Underestimating the future impact may motivate the platform to be aggressive and provide coupons that are more generous than necessary, hence reducing the platform's total revenue. Future research could explore more sophisticated estimates of churn to relax our assumption.

[21] The discount factor applies to future incidences, which may not occur at fixed intervals of calendar time. That said, at the time of each coupon offer decision, although the platform does not know the precise time intervals between the consumer's future visits, the platform knows the expected intervals. As such, the discount factor in this setting can be thought of as the discount rate based on the expected time intervals between each consumer's visits. Our discount rate specification is also consistent with Sutton and Barto (2018, p. 49): "The MDP framework is abstract and flexible and can be applied to many different problems in many ways. For example, the time steps need not refer to fixed intervals of real time; they can refer to arbitrary successive stages of decision making and acting." Online Appendix O provides more discussion of the implications of not adjusting the discount rate by the time intervals between consumer visits.

[22] The livestream shopping market is characterized by monopolistic competition; each seller can determine the product price on her own web page, and the products are differentiated from one another. There are millions of sellers in the market, and each seller independently sets prices to maximize her own profit. The

platform, however, makes the coupon allocation decision using a centralized system. The coupons affect the end price that consumers pay but not the list price, which is controlled by sellers.

[23] The assumption of exogenous search behavior is informed by empirical evidence, but it also can be relaxed. On the one hand, we tested whether coupons affect a consumer's subsequent web page choices by comparing the distributions of seller characteristics (e.g., seller attractiveness) in a store visit, conditional on the type of coupon received in the previous incidence. The data pattern fails to reject the null hypothesis (see Online Appendix A). On the other hand, we could relax the assumption of exogenous search behavior by allowing the seller-related and web page-related state variables to be dynamic instead of stationary. See more details in Section 4.2.1.

[24] Although livestreaming creates an interactive environment in which consumers and hosts can chat with each other, we found no evidence that consumers publicly share individual coupon information in the chat.

[25] Static features are defined as those not affected by actions. Static features may come from a stationary distribution instead of being a fixed number.

[26] The RFM variables could lead to a potential endogeneity problem if firms previously used these variables to determine how to target consumers. Although our input data do not suffer from this problem because the platform allocated the coupons randomly (i.e., without targeting), future researchers need to exercise caution in scenarios with the endogeneity issue and consider solutions such as the instrumental variable approach (Gönül et al. 2000) or the latent trait approach (Rhee and Russell 2009).

[27] We also consider a specification with state dependence. The results remain qualitatively unchanged (see Online Appendix N).

[28] Identifying references price effects is typically challenging in scanner data applications, where one does not know whether a customer observed the price (Rajendran and Tellis 1994). In livestream shopping, however, the identification of reference price effects is less ambiguous for two reasons. First, the livestream shopping app records the livestream channel visited by each consumer, the product the consumer clicked, and the targeted coupon received, so the econometrician knows whether the consumer observed the price. Second, our training data were generated using a random targeting policy, so no targeting endogeneity exists.

[29] The Q-function is often defined as a table, called the Q-table, which stores one state-action pair and the associated Q-value in each row.

[30] Although both Q-learning and the conditional choice probability (CCP) estimator (Hotz and Miller 1993) use sample observations (the so-called cell estimators) in the estimation procedure, they are fundamentally different algorithms. We compare the two algorithms in Online Appendix F.

[31] Other function approximators such as Chebyshev polynomials were previously used in the dynamic programming literature (Rust 1996). Chebyshev polynomials are preferred for smooth value functions (Cai and Judd 2010), but DNNs can approximate non-smooth functions effectively (Imaizumi and Fukumizu 2019). Recent theoretical work (Fan et al. 2020) has proved the accuracy of the deep learning approximation in reinforcement learning settings.

[32] Here, "direct methods" belong to a different class of treatment estimation algorithms than the policy evaluation methods in Section 4.4.1.

[33] Moreover, to account for the reference price effect, we follow the literature (Bell and Lattin 2000) and add two additional state variables in this model that represent the gain and loss effects of the reference price.

[34] To protect the privacy of the platform, the amounts have been changed by an affine transformation.

[35] Caveat: our data include only active users, defined as at least 10 coupon reception incidences in the three-month sample period. If the proposed policy is more effective for active users than for the average customer, then the projected increase in the GMV would be an upper bound. The projection to the entire customer base is based on the reported annual sales of $28 billion (200 billion RMB) in 2019. Pak J (2020 April 6). Retrieved from https://https://bit.ly/3ANCY5i.

[36] We assume that BCQ finished learning the optimal coupon allocation policy after training on the historical batch data. It is possible, however, that the learning process is incomplete, and as new data come in, parameters in the model will keep updating. Future research could explore another experimental condition with an online reinforcement learning algorithm (e.g., DQN) where the initial values of parameters come from the converged parameter values in the batch mode (e.g., BCQ) and continue updating in each iteration, as new data arrive.

[37] The number of consumers in each condition is 816,718, 102,090, and 102,090, respectively.

[38] One concern about a policy that increases the discount level with loyalty is that the discount level has an upper bound (HH). It is unclear how to solve the dynamic targeting problem once the upper bound is reached. It is possible that there is a steady state at the upper bound such that the platform can achieve stable revenue with HH coupons. During our short sample period of three months, only 1.18% of the consumers reached the steady state, defined as receiving HH for at least two consecutive periods). We do not have sufficient data to study consumer behaviors after the upper bound is reached, so we leave it for future research.

[39] Similarly, Dubé and Misra (2022) assumed that heterogeneity in customers' price sensitivities can be characterized by an observed, high-dimensional vector containing a sparse subset of observable customer characteristics.

References

Ascarza E, Netzer O, Hardie BGS (2018) Some customers would rather leave without saying goodbye. *Marketing Sci.* 37(1):54–77.

Bell DR, Lattin JM (2000) Looking for loss aversion in scanner panel data: The confounding effect of price response heterogeneity. *Marketing Sci.* 19(2):185–200.

Bertsekas D (2019) *Reinforcement Learning and Optimal Control* (Athena Scientific, Belmont, MA).

Cai Y, Judd KL (2010) Stable and efficient computational methods for dynamic programming. *J. Eur. Econom. Assoc.* 8(2–3): 626–634.

Dubé JP, Misra S (2022) Personalized pricing and customer welfare. *J. Political Econom.* 131(1):131–189.

Dubé JP, Hitsch GJ, Rossi PE (2010) State dependence and alternative explanations for consumer inertia. *RAND J. Econom.* 41(3):417–445.

Dudík M, Langford J, Li L (2011) Doubly robust policy evaluation and learning. Getoor L, Scheffer T, eds. *Proc. 28th Internat. Conf. on Machine Learn.* (Omnipress, Madison, WI), 1097–1104.

Dudík M, Erhan D, Langford J, Li L (2014) Doubly robust policy evaluation and optimization. *Statist. Sci.* 29(4):485–511.

Fader PS, Hardie BGS, Lee KL (2005) RFM and CLV: Using iso-value curves for customer base analysis. *J. Marketing Res.* 42(4): 415–430.

Fan J, Wang Z, Xie Y, Yang Z (2020) A theoretical analysis of deep q-learning. Bayen AM, Jadbabaie A, Pappas G, Parrilo PA, Recht B, Tomlin C, Zeilinger M, eds. *Learning for Dynamics and Control* (JMLR, Cambridge, MA), 120:486–489.

Friedman JH (2001) Greedy function approximation: A gradient boosting machine. *Ann. Statist.* 29(5):1189–1232.

Fujimoto S, Conti E, Ghavamzadeh M, Pineau J (2019) Benchmarking batch deep reinforcement learning algorithms. Preprint, submitted October 3, https://arxiv.org/abs/1910.01708.

Furman J, Coyle D, Fletcher A, McAules D, Marsden P (2019) Unlocking digital competition: Report of the digital competition expert panel. Report, The National Archives, Kew, London. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/785547/unlocking_digital_competition_furman_review_web.pdf.

Gedenk K, Neslin SA (1999) The role of retail promotion in determining future brand loyalty: Its effect on purchase event feedback. *J. Retailing* 75(4):433–459.

Gönül FF, Kim BD, Shi M (2000) Mailing smarter to catalog customers. *J. Interactive Marketing* 14(2):2–16.

Hauser JR, Urban GL, Liberali G, Braun M (2009) Website morphing. *Marketing Sci.* 28(2):202–223.

He X, Pan J, Jin O, Xu T, Liu B, Xu T, Shi Y, et al. (2014) Practical lessons from predicting clicks on ads at Facebook. Saka E, Shen D, Lee K, Li Y, eds. *Proc. 8th Internat. Workshop on Data Mining for Online Advertising* (Association for Computing Machinery, New York), 1–9.

Hotz JV, Miller RA (1993) Conditional choice probabilities and the estimation of dynamic models. *Rev. Econom. Stud.* 60(3):497–529.

Imaizumi M, Fukumizu K (2019) Deep neural networks learn non-smooth functions effectively. Chaudhuri K, Sugiyama M, eds. *Proc. 22nd Internat. Conf. on Artificial Intelligence and Statist.*, vol. 89 (JMLR, Cambridge, MA), 869–878.

Jeuland AP (1979) Brand choice inertia as one aspect of the notion of brand loyalty. *Management Sci.* 25(7):671–682.

Kahn BE, Kalwani MU, Morrison DG (1986) Measuring variety-seeking and reinforcement behaviors using panel data. *J. Marketing Res.* 23(2):89–100.

Kim M, Sudhir K, Uetake K (2021) A structural model of a multitasking salesforce: Multidimensional incentives and plan design. *Management Sci.* 68(6):4602–4630.

Lucas RE (1976) Econometric policy evaluation: A critique. *Carnegie-Rochester Conference Series on Public Policy*, vol. 1, 19–46.

McCall JJ (1970) Economics of information and job search. *Quart. J. Econom.* 84(1):113–126.

Misra K, Schwartz EM, Abernethy J (2019) Dynamic online pricing with incomplete information using multiarmed bandit experiments. *Marketing Sci.* 38(2):226–252.

Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, et al. (2015) Human-level control through deep reinforcement learning. *Nature* 518(7540):529.

Oprescu M, Syrgkanis V, Wu ZS (2019) Orthogonal random forest for causal inference. Chaudhuri K, Salakhutdinov R, eds. *Internat. Conf. Machine Learn.* vol. 97 (PMLR, Cambridge, MA), 4932–4941.

Rajendran KN, Tellis GJ (1994) Contextual and temporal components of reference price. *J. Marketing* 58(1):22–34.

Rhee E, Russell GJ (2009) Forecasting household response in database marketing: A latent trait approach. Lawrence KD, Klimberg RK, eds. *Advances in Business and Management Forecasting*, vol. 6 (Emerald, Bingley, UK), 109–131.

Rossi PE, McCulloch RE, Allenby GM (1996) The value of purchase history data in target marketing. *Marketing Sci.* 15(4): 321–340.

Rust J (1996) Numerical dynamic programming in economics. Amman HM, Kendrick DA, Rust J, eds. *Handbook of Computational Economics*, vol. 1 (Elsevier, North Holland Publishing Co., Amsterdam, Netherlands), 619–729.

Seetharaman PB, Che H (2009) Price competition in markets with consumer variety seeking. *Marketing Sci.* 28(3):516–525.

Seethu Seetharaman PB (2009) 17 dynamic pricing. Rao VR, ed. *Handbook of Pricing Research in Marketing* (Edward Elgar Publishing, Cheltenham, UK), 384.

Seiler S (2013) The impact of search costs on consumer behavior: A dynamic approach. *Quant. Marketing Econom.* 11(2): 155–203.

Sutton RS, Barto AG (2018) *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, MA).

UK Competition and Markets Authority (2018) Pricing algorithms: Economic working paper on the use of algorithms to facilitate collusion and personalised pricing. Working paper, UK Competition and Markets Authority, UK.

Urban GL, Liberali G, MacDonald E, Bordley R, Hauser JR (2013) Morphing banner advertising. *Marketing Sci.* 33(1):27–46.

Van Heerde HJ, Neslin SA (2017) Sales promotion models. *Handbook of Marketing Decision Models* (Springer, Berlin), 13–77.

Watkins CJCH (1989) Learning from delayed rewards. PhD thesis, Cambridge University, Cambridge, UK.

Wen H, Zhang J, Lin Q, Yang K, Huang P (2019) Multi-level deep cascade trees for conversion rate prediction in recommendation system. *Proc. Conf. AAAI Artificial Intelligence* 33:338–345.

Winer RS (1986) A reference price model of brand choice for frequently purchased products. *J. Consumer Res.* 13(2):250–256.

Zhang Q, Wang W, Chen Y (2019) In-consumption social listening with moment-to-moment unstructured data: The case of movie appreciation and live comments. *Marketing Sci.* 39(2): 285–295.