Churning while Learning: Maximizing User Engagement in a Recommendation System

(Authors' names blinded for peer review)

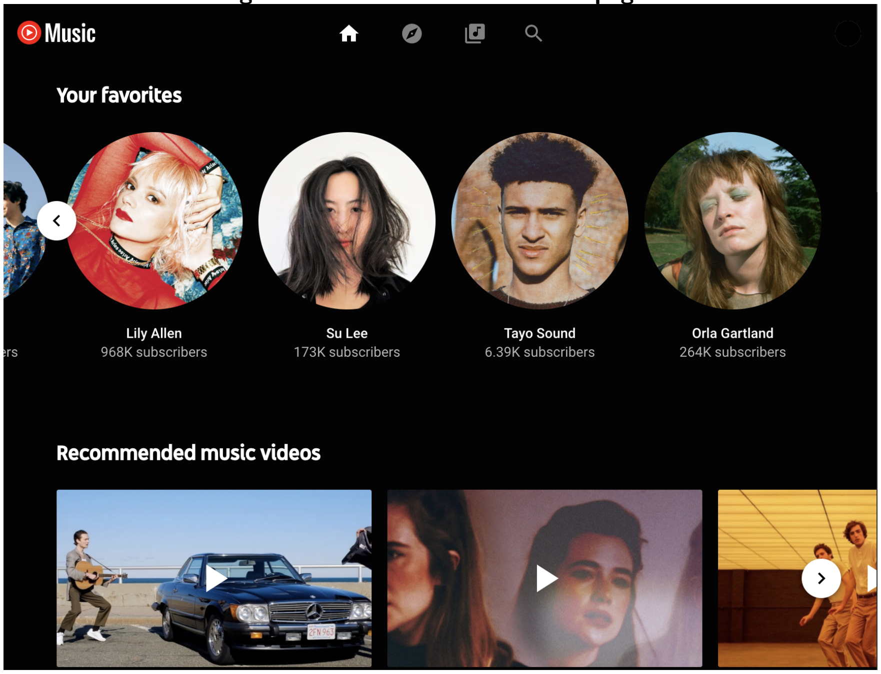
Online media platforms such as Spotify, YouTube Music, and NetEase Cloud Music rely on long-term user engagement for revenue generation. The primary operational level under their control is content recommendation (i.e., what content to recommend to various users), where the right recommendation can induce users to further interact with the platform, sign-up for premium memberships, view ads, and more. However, given constantly updating supply (i.e., new content being created) and heterogeneous user behavior, optimal recommendation is challenging. It requires a careful balance between exploration (understanding the efficacy of new content) and exploitation (recommending well understood existing content). Motivated by a real-world impressions level dataset from NetEase Cloud Music (Zhang et al. 2020), we propose a two-period model for the platform recommendation problem, with the aim of maximizing long term user engagement. Our model captures two key features of this marketplace: (1) supply-side learning (i.e., platform learning the efficacy of new content) and (2) demand-side heterogeneous churning (i.e., different users churning at different rates as a function of both their engagement and their types). We use our two-period model to understand the interplay between churning while learning and how it impacts the long-term user engagement. In addition to characterizing the structure of the optimal learning policy for the platform recommendation problem (i.e., which users to experiment on), we numerically illustrate our key findings on a wide range of parameters. We find that accounting for heterogeneous churning behavior while learning the efficacy of new content can lead to improvements of up to 14% in the long-term user engagement.

Key words: recommendation system; learning; churning; beta-binomial; first impression effect

1. Introduction

Online media platforms have grown to dominate the music industry over the last decade, with revenue from digital music accounting for more than half of the revenue generated by the industry in 2018 (Watson 2018), and with digital media's market-share continuing to grow every year between 2015-2019 (IFPI 2019). As these media platforms grow, they are faced with increasingly complex operational challenges. At a high-level, online media platforms such as Spotify, YouTube Music, and Tencent need to coordinate interactions between content creators who supply new media (e.g. music and music videos), and users who engage with the content and from whom the platform can generate revenue directly by paid subscriptions, or indirectly by display advertising. In this rich ecosystem, we focus on a particular aspect of coordination between content creators and platform users we term the *platform recommendation problem*, in which the platform decides how to recommend new content to users. Fig. 1 shows a screen grab of the YouTube Music homepage along with recommendations of new content (recommended music videos) for a user to listen to/watch.

Figure 1 YouTube Music Homepage.



Note. Depicted is the homepage for the online media platform YouTube Music. Note the bottom row of recommended music videos (sometimes referred to as *cards*).

The platform recommendation problem (formally defined in Section 3) is complicated by a number of factors related to the nature of the content (supply), user behavior (demand), and the platform's information about each. In terms of the supply, the specific catalog of media the platform can recommend is always changing as content creators produce and upload new creatives.

Furthermore, there is significant heterogeneity in the quality of such content. For content that has already been widely displayed on the platform, it may be possible to correctly "estimate" its click through rate (CTR) and optimize recommendations accordingly. However, such content quickly becomes stale, and with new uploaded content, there's again little information to inform the platform's recommendation decision. Further, gathering information on new content requires care, since showing a low quality creative to too many users can decrease their engagement with the platform. Adding to the complexity, such user engagement can itself be heterogeneous depending on the "type" of the user. For instance, as we demonstrate by analyzing the NetEase dataset (c.f. Section 2), new users are more likely to churn (i.e., leave the platform) as a result of one "poor" interaction as compared to regular users. How can the platform maximize user engagement (e.g., expected number of clicks) while learning the CTR of new creatives in the presence of heterogeneous churn behavior of users?

A natural framework to maximize user engagement while learning content clickability is *multi-armed bandits* (Sutton and Barto 2018), where different new creatives represent the various "arms" of the bandit and a learning algorithm (e.g., Thompson sampling (Thompson 1933) or UCB (Lattimore and Szepesvári 2020)) dictates the recommendations. Such learning paradigms have seen a recent surge in the literature (e.g., see Slivkins (2019) for a modern introduction of the area). However, given the underlying "one-period" horizon in bandits, such models are rather myopic and do not capture long-term effects of the recommendations (e.g., a poor recommendation causing a user to churn and hence, eliminating future clicks), which is critical in our application. In this work, we complement such learning-based approaches by explicitly optimizing for both immediate and future rewards via a two-period model. In particular, we consider the well-known exploration vs. exploitation trade-off (explore a new creative with uncertainty over its CTR or exploit an existing creative with known CTR) in combination with the *first impression effect*, wherein the future engagement of new users with the platform depends disproportionately on the outcomes of their first interaction.

1.1. Our Contributions

We study the inherent tension between exploration and exploitation in the presence of the first impression effect, using data provided by NetEase Cloud Music (Zhang et al. 2020), the second largest music media platform in China. A summary of our key contributions and findings is as follows:

1. We explore the NetEase Cloud Music dataset and find evidence of a first impression effect. Namely, new users who do not engage with the recommended creative are around 5 percentage points less likely to return to the platform than new users who do engage with (click) the

recommendations. On the other hand, we find this number to be significantly lower (around 1 to 2 percentage points) for "regular" users. Furthermore, we find evidence that in spite of such heterogeneous user behavior, NetEase Cloud Music shows new creatives to users regardless of how long they've used the platform (which we refer to as the users *type*).

2. Based on our exploratory data analysis, we propose a two-period model with the goal of maximizing long-term user engagement. Our model captures the two features in this marketplace: (1) supply-side learning (uncertainty over the CTR of a newly created content) and (2) demand-side heterogeneous churning (e.g., regular vs. new users). To the best of our knowledge, this is the first model in the literature that optimizes for the long-term user engagement while learning under the presence of heterogeneous churning behavior.
3. We then solve the proposed model by introducing an easily computable user index we term the *Experimentation Coefficient*. We prove the optimal policy in our model is to experiment on users in order of their experimentation coefficient (c.f. Theorem 1). Using this characterization, we give tight worst-case characterizations of the loss incurred by type-blind policies (c.f. Theorem 2) and perform comparative statics.
4. Finally, we supplement our analytical developments by performing numerics over a wide range of parameters. We find that accounting for heterogeneous churning behavior while learning the CTR of new content can lead to 0-14% improvements in the long-term user engagement.

1.2. Literature Review

Our work intersects with several streams of literature in machine learning, statistics, operations management, and behavioral psychology. Here, we overview some of these streams and explain how our work contributes to and/or differs from each.

Recommendation systems. Given the broad applicability of the recommendation problem (Alexander 2020), there exist several works in this domain (Schafer et al. 2007, White et al. 2009, Melville and Sindhwani 2010, Jannach et al. 2010, Lü et al. 2012, Breese et al. 2013, Figueiredo et al. 2016, Mehrotra et al. 2019), including the well-known literature on the "Netflix prize" (Netflix 2006, Bennett et al. 2007). The high-level idea in these works is to use historical data to predict the behavior of the next user and optimize the recommendation accordingly. A key limitation of much work in this area is that they are myopic in nature, i.e., their objective is to maximize "immediate reward" (e.g., probability user clicks on the recommendation) and they do not necessarily capture the "long-run reward". This is in contrast with our work, where we explicitly accommodate the long-run value of a recommendation via a two-stage model.

Estimation. A key parameter in our recommendation platform is the click through rate (CTR) of a piece of creative content, i.e., if a specific creative (e.g., video) is shown to a random user, what is the probability the user will click on it? There is a widespread statistical literature on parameter estimation (Silvey 2013), including works such as Rubin (1974), Kirk (2012) and Bhat et al. (2020) along with a mix of machine learning tools (James et al. 2013). However, such ideas usually focus on optimizing some statistical metric (e.g., maximize the "power" of an experiment or the "precision" of the estimate) and the resulting impact on the business metric of interest (e.g., user engagement) is unclear. On the contrary, our framework explicitly focuses on optimizing user engagement and implicitly *learns* the underlying parameters over time. In this way, our framework resembles work in "joint estimation and optimization" (Kao et al. 2009, Feit and Berman 2019, Elmachtoub and Grigas 2021, Zhu et al. 2021), where one formulates an overall optimization problem to simultaneously estimate the parameters and optimize the underlying business objective. The key dimension we differ from this idea is that our approach is learning-based, as opposed to estimation-based, i.e., we maintain a Bayesian belief over the parameters of interest. Furthermore, we leverage a real-world dataset to capture micro-level features (e.g., the first impression effect) in the content recommendation ecosystem and hence, our work is tailored to the application at hand.

Learning. There is a recent surge in learning-based approaches in the operations community (Harrison et al. 2012, Besbes and Zeevi 2015, Keskin and Zeevi 2017, Feng et al. 2018, Bastani et al. 2018, Negoescu et al. 2018, Baardman et al. 2019, Agrawal and Jia 2019, Bimpikis and Markakis 2019, Nambiar et al. 2020, Gijsbrechts et al. 2020, Agrawal et al. 2021, Chen et al. 2021). Such approaches leverage general learning theory (Sutton and Barto 2018) and tailor it to a specific operations problem of interest, which allows one to extract application-specific insights. Our work follows a similar template in that we focus on learning in the presence of churning for the platform recommendation problem, which enables us to characterize the optimal experimentation policy in closed-form. Though there exist a few learning-based approaches for content recommendation (Kveton et al. 2015, McInerney et al. 2018, Dragone et al. 2019), most of them are myopic ("bandits") and our focus on learning under churning while capturing the long-term impact is unique.

First impressions and churning. As mentioned earlier, a key feature in our model is the first impression effect, i.e., new users are more likely to churn than regular users in the event of one poor interaction. Though not directly related to the problem of "churning while learning" tackled in this work, we mention in passing the growing literature that models churning and the importance of first platform interaction with a user (Liu et al. 2016, Padilla and Ascarza 2017, Martins 2017, Yang et al. 2018, Kanoria et al. 2018, Kostić et al. 2020). It is also worth mentioning works in behavioral psychology that document the importance of first impression for humans (Asch 1946, Kelley 1950, Rabin and Schrag 1999, Agnew et al. 2018). Our work is one of the first to leverage this theory in the context of content recommendation.

The rest of this paper is organized as follows. In Section 2, we perform an exploratory analysis of the NetEase Music dataset and highlight some of the salient features. In Section 3, we build on our data analysis and propose a two-period model, which captures the key features at play (supply-side learning, demand-side heterogeneous churning, and optimizing long-term engagement). In Section 4, we analyze the proposed model. Specifically, we characterize the optimal policy (Theorem 1) and compare it with a policy that ignores heterogeneity among users (Theorem 2). We numerically illustrate our findings in Section 5, followed by concluding remarks in Section 6.

2. Preliminary Data Analysis

In this section, we discuss three key components of the platform recommendation problem, and connect our discussion to practice using the data from NetEase Cloud Music (Zhang et al. 2020). First, in Section 2.1, we touch upon the supply side, i.e., the digital media (cards) produced by content creators. Next, in Section 2.2, we study the demand side of the market, i.e., the users who interact with the cards. Finally, in Section 2.3, we discuss the platform, the operational levers under its control, and its objectives. Our goal in this section is to communicate the key features of the digital platform, which will motivate and inform our model development in Section 3.

2.1. Content Creators

The supply in this market corresponds to the various cards created by content creators, as alluded to in Section 1. Each card can have various intrinsic dimensions such as the song/video it contains and the underlying artist. All such dimensions ultimately affect the "clickability" of the card, which can be summarized by its *click through rate* (CTR), i.e., the probability it will be clicked if shown to a random user. For our modeling purposes, we primarily focus on the CTR of each card. Of course, not every card is created equal and there is heterogeneity among the CTRs of various cards. Furthermore, the CTR of a card is a parameter the platform learns over time, by recommending the card to a number of users and observing whether they click or not. In Fig. 2, we plot the evolution of a running sample *estimate* of the CTR for four randomly chosen cards from the NetEase dataset.

Note the clear heterogeneity among the cards CTRs, which stabilize between 0% and 40% (Fig. 3a), confirming that some creatives are more effective than others. Naturally, learning this underlying "true" CTR for each card is valuable to the platform, as it can use such information to recommend a better portfolio of cards to the users, resulting in more clicks and in a higher overall activity.

In order to learn the CTR of each card, a key quantity to understand is the prior distribution over CTRs. In Fig. 3a, we show the empirical histogram obtained via a sample estimate of each

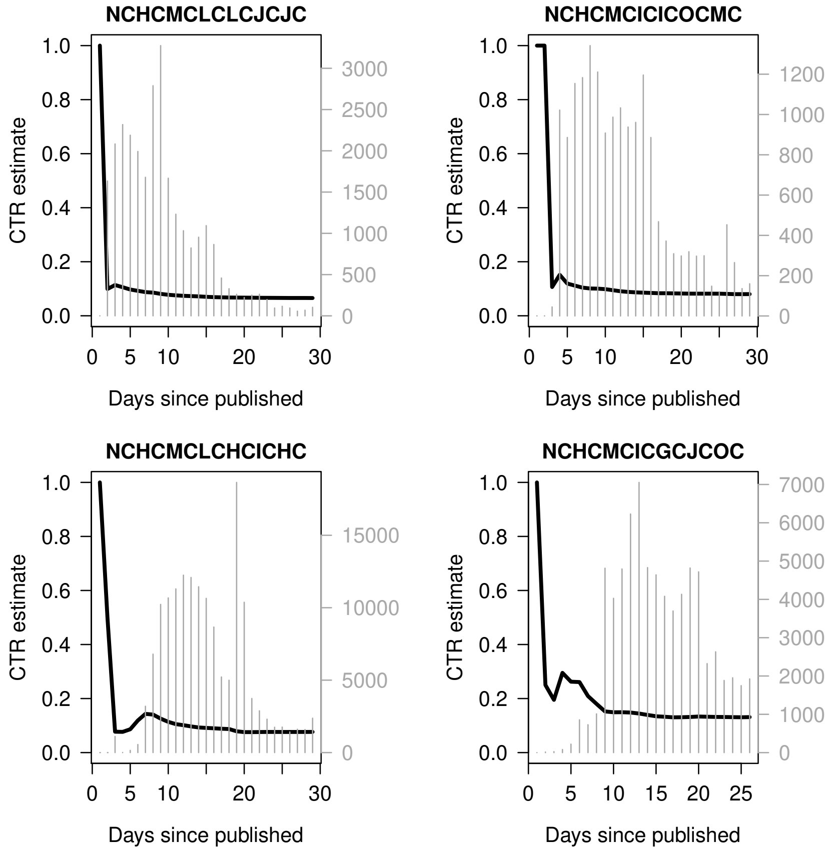
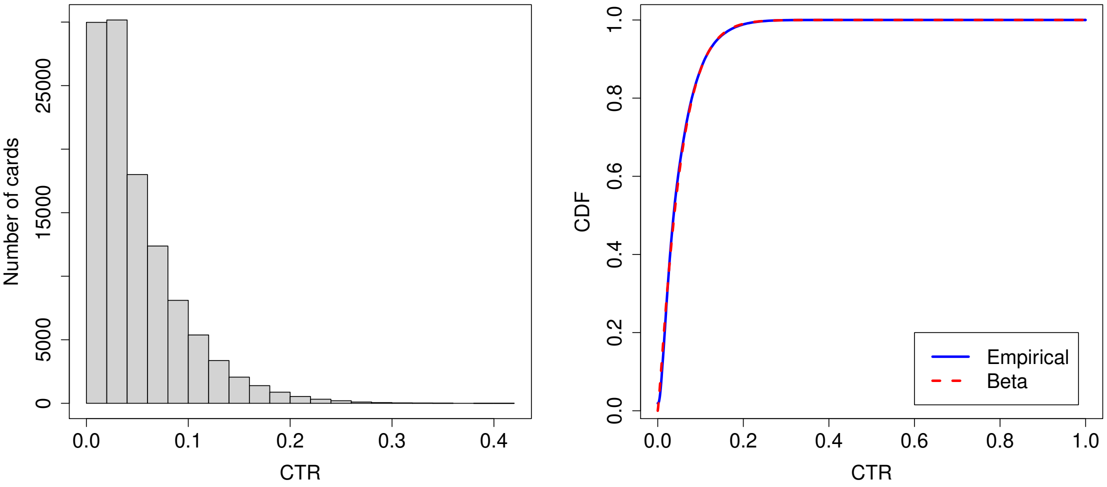


Figure 2 **The evolution of the CTR estimate for four randomly chosen cards (ID given at the top of each plot) from the NetEase dataset. The x-axis denotes the days since the card was published. The left side of the y-axis and the thick black line within each plot correspond to the empirical CTR estimate, i.e., the number of times the card was clicked divided by the number of times the card was shown over the period (accounting for all the impressions till this day). The right side of the y-axis and the gray vertical bars within each plot denote the number of times the card was shown on a given day.**

card's CTR in the NetEase dataset that was shown at least 100 times (around 110,000 such cards). The sample mean across all such cards equals $\mu_0 = 0.0507$, and the sample standard deviation equals $\sigma_0 = 0.0441$.

Visually, the histogram for the empirical CTR resembles a Beta distribution. In fact, as we show in Fig. 3b, the Beta distribution explains the data quite well. We obtain the fit by calibrating the first two moments of the Beta(α_0, β_0) distribution to the data-driven sample mean μ_0 and standard deviation σ_0:

$$\mu_0 = \frac{\alpha_0}{\alpha_0 + \beta_0}$$
$$\sigma_0^2 = \frac{\alpha_0 \beta_0}{(\alpha_0 + \beta_0)^2(\alpha_0 + \beta_0 + 1)}.$$



(a) Empirical histogram of CTR (b) Beta fit to the empirical distribution

Figure 3 The empirical histogram (subplot (a)) is plotted using cards with at least 100 impressions (around 110,000 such cards). For each card, we estimate the CTR as the proportion of times it was clicked. The sample mean equals $\mu_0 = 0.0507$ and the sample standard deviation equals $\sigma_0 = 0.0441$. To fit the Beta distribution (subplot (b)), we calibrate to the first two moments of the data, as in Eq. (1).

Rearranging gives us the following closed-form expressions for α_0 and β_0:

$$\alpha_0 = \left(\frac{1-\mu_0}{\sigma_0^2} - \frac{1}{\mu_0}\right)\mu_0^2 \tag{1a}$$

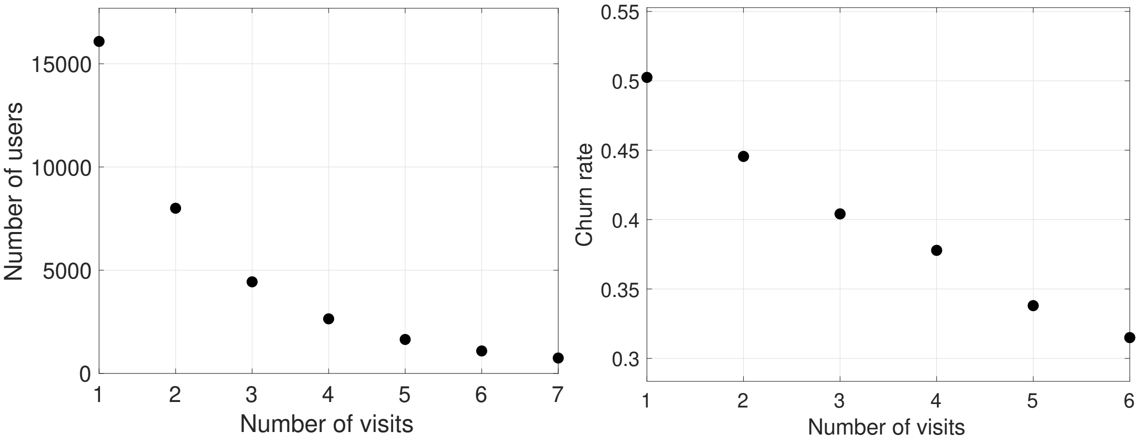
$$\beta_0 = \alpha_0\left(\frac{1}{\mu_0} - 1\right). \tag{1b}$$

Given the quality of the fit on the sample data, we will assume from here on that the CTR of each card is drawn from a Beta distribution with known parameters. Building on this observation, we further model the user's click behavior using a Bernoulli distribution (whether a user clicks or not). Accordingly, in our model development and the corresponding analysis, we will leverage the Beta-Bernoulli conjugacy to maintain a tractable belief over the CTR of each card.

2.2. Platform Users

The demand in this market corresponds to the platform's users. At a high-level, as discussed in Section 1, one way to think about the user base is to split it into two categories: (1) regular users and (2) new users. Regular users are the ones who have already spent considerable time on the platform and are less likely to churn than new users. To substantiate this intuition, in Fig. 4a, we plot the churn behavior of a set of new users in the NetEase dataset. Out of the 16083 new users, only 8002, 4437, 2644, 1645, 1089, and 746 returned to the platform for a second, third, fourth, fifth, sixth, and seventh visit, respectively. This suggests a churn rate of around 50% $(1 - 8002/16083)$

after the first visit, monotonically decreasing to around 30% $(1-746/1089)$ after the sixth visit (Figure 4b), meaning the churn likelihood decreases as a user becomes older.



(a) Churn progression

(b) Churn rate

Figure 4 We consider the churn behavior of users whose first visit was between days 7 and 10 (inclusive) in the NetEase dataset (16083 such users). We do not include users with a later first visit since the dataset only contains information for 30 days and hence, including a user with a first visit on day 25 for example might not give them enough time to re-visit the platform multiple times. Subplot (a) shows the number of users (y-axis) who visited the platform at least x times. Subplot (b) displays the corresponding churn rate after each visit. For example, out of the 8002 users who visited the platform twice, only 4437 returned for a third visit, implying a churn rate of $1-4437/8002 \approx 45\%$ after visit 2.

A churn rate of 50% is quite high; half the potential business is lost after just one interaction. Such a high churn rate presents an opportunity to boost the number of active users by mitigating this churn. In this direction, in order to formulate the underlying operational problem that aims at maximizing long-term user engagement, we first explore whether there is an explanation for the churn behavior of the new users. In particular, we investigate if there is a systematic difference in some of the observables between the new users who churn and the ones who do not. Given our discussion in Section 2.1, we focus on the how the churn probability of a new user varies as a function of whether or not they click on a card in their first visit (recall first impression effect). Out of the 16083 new users plotted in Fig. 4, 2581 clicked on a card during their first visit whereas 13502 did not. Among the 2581 users who clicked, 1202 churned (i.e., did not return for a second visit), implying a churn rate of around 46.57%. On the other hand, among the 13502 users who did not click, around 50.95% of the users churned, suggesting an increase of 4.38 percentage points in the churn likelihood. Of course, our model-free analysis on the observational data is not necessarily causal in nature since there might be an underlying confounding variable (e.g., users who are "excited" about the platform might be more likely to click and more likely to stay). However, given the absence of experimental data[1], it seems reasonable to assume that a user who has a positive experience in their initial visit is more likely to return than a user who does not. As a robustness check, we repeat our calculation on an additional sample of 8846 users (users whose first visits were on days 11, 12, and 13) and find the churn proportions to be similar (47.27% if a user clicks and 52.39% if they do not), suggesting an increase of around 5 percentage points if a user does not click in their first visit. Furthermore, in Appendix A, we repeat our analysis and find the churn rate in "regular" users to be much lower (around 5% given a click), with an increase of around 1 to 2 percentage points in the absence of a click, lending additional evidence to the existence of first impression effect. Armed with this observation, in the next subsection, we consider the objectives and challenges of the platform along with the operational levers under its control.

2.3. Platform Design

The platform is primarily responsible for matching supply (cards) with demand (users) by recommending cards to the users when they visit the platform. As stated in Zhang et al. (2020), a long-term goal of the platform is to maximize the user activity (e.g., number of clicks per day). Given heterogeneity in both supply (Section 2.1) and demand (Section 2.2), optimizing this coordination is challenging. In terms of the supply, learning the CTR of newly created cards is vital. Ideally, the platform would not show too many impressions of low CTR cards but quickly identify and recommend high CTR cards. With regards to the demand, controlling the churn rate is critical. All else being equal, a higher churn rate results in a lower number of clicks in the *future* and hence, the platform needs to control the churn if it wishes to maximize clicks in the long-term. Otherwise, if the platform optimizes myopically, it might end up maximizing the clicks in the short-term and lose out on the long-term benefits due to churning. This represents a key temporal trade-off for the platform. Furthermore, given the first impression effect, accounting for the "type" of the user (e.g., regular or new) is important. How can the platform design the recommendation system that maximizes the expected number of long-term clicks while learning the CTR of new cards and accommodating the heterogeneous churn behavior of the users?

In this direction, we discuss the key operational decision under the control of platform. Given the problem of learning the CTR of any new card, the platform needs to experiment on the users by showing them the new card and observing their click behavior. Hence, the decision we focus on in this work is the *experimentation policy* employed by the platform. Given the heterogeneous user behavior discussed in Section 2.2, it is natural to split the experimentation policy as a function of the user type. In particular, given a new card and a corresponding *experimentation budget* (i.e.,

[1] It might be of interest to the platform to conduct an appropriate randomized control trial to validate such causation.

number of times to show the card in order to learn its CTR), the platform needs to decide on how to optimally split this budget between the regular and new users.

A simple experimentation policy is *blind randomization*, where the allocation of the experimentation budget is independent of the user type. As an illustration, consider the following example. Suppose there are 100 regular and 100 new users and 1 new card. Given an exogenous experimentation budget of 50 impressions, the blind experimentation policy allocates 25 impressions to the regular users and 25 impressions to the new users. In other words, 25 regular and 25 new users are shown the new card whereas the remaining 150 users are shown some other card (an old card for which the platform has already learned the CTR). Intuitively, given the type-specific churn behavior discussed above, such a blind randomization seems sub-optimal[2]. However, it is worth highlighting that any experimentation policy that does not account for the type of the user would result in blind randomization automatically, and to the best of our knowledge, all existing relevant works do not model type-specific churning. In fact, by analyzing the NetEase dataset (details in Appendix B), we find evidence that blind randomization (or perhaps *more* experimentation on new users than on regular users) is being used in practice. Hence, in this work, we focus on designing optimal experimentation policies in the presence of heterogeneous churning behavior, with the goal of maximizing long-term expected number of clicks.

3. Model and Preliminaries

We now build on our data exploration in Section 2 to propose a model capturing two key features in this marketplace with the goal of optimizing long-term engagement: (1) supply-side learning (uncertainty over the CTR of a newly created content) and (2) demand-side heterogeneous churning (e.g., regular vs. new users). In order to focus on these salient features, we abstract away some peculiarities of the NetEase platform (e.g., a user seeing multiple recommendations simultaneously). We discuss possible extensions of our model in Section 6.

Before formally defining our model in Section 3.1 and discussing preliminary observations in Section 3.2, we provide a high-level overview. Our model zooms in on the case of single new *card* being introduced into the marketplace. Following the analysis of Section 2.1 (c.f. Fig. 3), we model the new card as having a Beta prior distribution over its CTR, and have it compete with a pool of well understood cards for whom the CTR is fixed and known[3]. Associated with the new card is an exogenously determined *experimentation budget* B, where B can be thought of as decided by some pacing or optimization algorithm (e.g. Agarwal et al. (2014) and Xu et al. (2015)), and that must

[2] It will become clear in Section 4 that blind randomization is almost always strictly sub-optimal.

[3] One can think of the platform maintaining a large collection of existing cards, with the *expected* CTR of this collection fixed and known. To recommend an old card, the platform samples a card from this collection.

be spent by showing the card B times. Though exogenous, we emphasize that B is arbitrary in our model and hence, our results follow for all values of B. This experimentation allows the platform to learn the CTR of the new card by updating the Beta prior. The task then is to decide which types of users should be shown the new card given that, as discussed in Section 2.2, different user types vary in their associated churn rates, i.e., their click-then-churn rate and no-click-then-churn rate. The objective of the platform is to maximize the total number of expected clicks (over two stages) by coordinating user-card interactions while satisfying the budget for experimentation. In this direction, we briefly discuss the two stages of interest.

To study the effects of users churning at different rates depending on whether or not they clicked in the previous period, we consider the problem as consisting of two stages. In the first stage, the platform decides which users to show the new card and which users to show the old card. Each user interacts with their recommendation as governed by the CTR of the recommended card, and churns as a function of their type and whether they interacted or not. Hence, only a subset of stage 1 users proceed to stage 2. In the second stage, the platform utilizes the outcomes of the B experimental recommendations in stage 1 to update its belief over the new card, and either discards the new card and uses *another* old card for all users, or embraces the new card and shows it to all remaining users who have not yet seen it. We explicitly disallow showing the same card to the same user in both periods, which is a desirable feature from a practical point-of-view.

3.1. Model and Notation

Formally, we model users who interact with the platform in two stages. In the first stage, each user is shown a card and the outcome of the user-card interaction is observed by the platform (e.g., whether the user clicks or not). The user then either returns to the platform in stage 2 where they will be shown another card to interact with, or they churn and leave the platform forever. We model each user as having an observable type $k \in [K]$ (e.g., new vs. regular users), where type is defined by the pair (q_k, δ_k) which governs the user's interaction dependent churn rates. Specifically, a user of type k churns between stage 1 and stage 2 with probability q_k if they interact with/click their recommended card, and churn with probability $q_k + \delta_k$ if they do not interact with their recommendation; so $\delta_k < 1 - q_k$ represents the incremental increase in churn rate. In Fig. 5, we overview a user's interaction with the platform in our two-period model.

We model the platform as choosing between two types of cards (new and old) to display to each user in stage 1 and stage 2, based on "current" knowledge of the underlying CTRs. Let $p^{\text{old}} > 0$ be the fixed and known CTR of the old card[4], and let $p^{\text{new}} \sim \text{Beta}(\alpha_0, \beta_0)$ be the uncertain CTR and

[4] As in Footnote 3, one can think of the platform sampling an old card from an existing large collection, where p^{old} denotes the expected CTR of this collection. Such a view allows for the old cards to have different CTRs from each other as long as the average equals p^{old}.

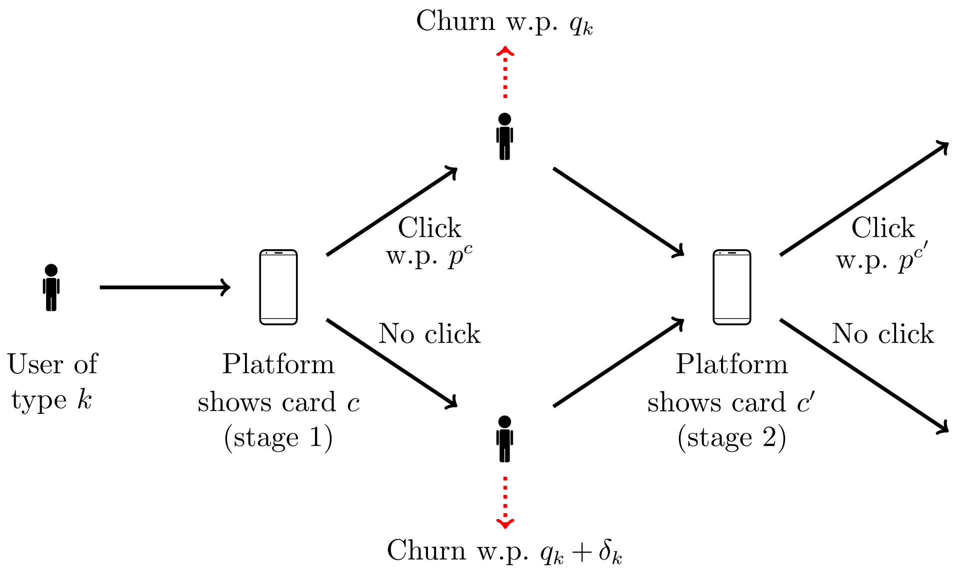

Churn w.p. q_k
Click w.p. p^c
No click
Click w.p. $p^{c'}$
No click
User of type k
Platform shows card c (stage 1)
Platform shows card c' (stage 2)
Churn w.p. $q_k + \delta_k$

Figure 5 **Our two-period model of user behavior. Note that the churn behavior (both the "base" rate q_k and the "increment" due to low engagement δ_k) depends on the user type k and hence, we explicitly capture such heterogeneity in user behavior.**

distributional prior of the new card before stage 1. By definition of a Beta distribution, the prior mean of p^{new} equals $\mu_0 := \frac{\alpha_0}{\alpha_0+\beta_0}$. In the first stage, for a given user, the platform chooses a card (either new or old) to display, where the platform is constrained to show the new card B times (over all users in stage 1). Based on observed clicks in stage 1, the platform updates its belief over p^{new} via the Beta-Binomial conjugacy (Gelman et al. 2013). In particular, if the platform observes N^{new} clicks of the new card (out of B interactions), then the stage 2 belief (posterior) over p^{new} becomes Beta(α_1, β_1) where

$$\alpha_1 := \alpha_0 + N^{\text{new}} \tag{2a}$$

$$\beta_1 := \beta_0 + B - N^{\text{new}}. \tag{2b}$$

The updated expectation of p^{new} after stage 1 is $\mu_1 := \frac{\alpha_1}{\alpha_1+\beta_1}$. Using the posterior belief, the platforms decides on the user-specific optimal card to show in stage 2 (end of horizon). In this direction, we define

$$y := \begin{cases} \text{new} & \text{if } \mu_1 > p^{\text{old}} \\ \text{old} & \text{otherwise.} \end{cases} \tag{3}$$

For the objective of maximizing expected number of total clicks (across both stages) and stage 2 being the end of horizon, the stage 2 optimal policy is straightforward. When y equals old, an old card (different from the one in stage 1) is shown to all users in stage 2. When y equals new, the new card is shown to all users in stage 2 who did not see the new card in stage 1. Users who saw the new card in stage 1 are instead shown an old card in stage 2 since we explicitly disallow showing the same card to the same user twice.

As the optimal decision in stage 2 is fully determined based on the outcome of the interactions in stage 1, the only decision facing the platform in our model is the choice of which users are shown the new card in stage 1. Let Λ_k be the number of type k users present at the beginning of stage 1, and $\Lambda := \sum_{k\in[K]} \Lambda_k$ be the total number of initial users. We assume Λ is greater than or equal to the experimentation budget B. To describe the platform's decision, let Λ_k^{new} be a decision variable representing the number of users of type k shown the new card in stage 1, and $\Lambda_k^{\mathrm{old}} := \Lambda_k - \Lambda_k^{\mathrm{new}}$ be the number shown an old card in stage 1. We denote the policy of the platform by $\pi(\cdot)$, which maps the parameters of the model to type-level allocations of the new card in stage 1, i.e.,

$$\pi\left(\{\Lambda_k\}_{k=1}^K, \{(q_k,\delta_k)\}_{k=1}^K, p^{\mathrm{old}}, \alpha_0, \beta_0\right) = (\Lambda_1^{\mathrm{new}}, \dots, \Lambda_K^{\mathrm{new}}). \tag{4}$$

To describe the platforms objective, let $\mathcal{R}(\pi)$ be the expected number of clicks across the two stages under policy π. In Fig. 6, we give an overview of our model from the platform's perspective.

The platform's objective is to find a policy that maximizes $\mathcal{R}(\pi)$, subject to the constraint that the policy must satisfy the exogenous budget for experimentation, $\sum_{k=1}^K \Lambda_k^{\mathrm{new}} = B$. We will term this the *Platform Recommendation Problem.* In the next subsection, we highlight some simple properties and nuances of the platform recommendation problem, before characterizing the optimal policy in Section 4.

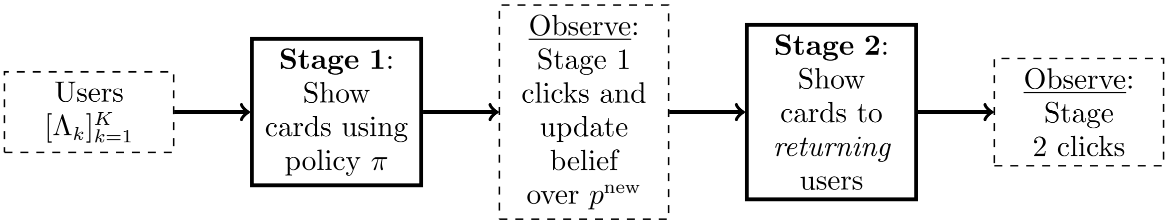

Users
$[\Lambda_k]_{k=1}^K$
Stage 1: Show cards using policy π
Observe: Stage 1 clicks and update belief over p^{new}
Stage 2: Show cards to *returning* users
Observe: Stage 2 clicks

Figure 6 Timeline from the platform's perspective. The objective of the platform is to maximize the total number of expected clicks (over both the stages) and hence, our model captures the long-term effect of churning ("churning while learning").

3.2. Preliminary Observations

Given the setup above, we can make a few observations about the platform recommendation problem, which aid the characterization of the optimal policy in Section 4. Observe that the three key quantities of interest are (1) the number of clicks in stage 1, (2) the number of users who churn after stage 1, and (3) the number of clicks in stage 2. The three observations below shed light on each of these quantities.

Observation 1: Expected number of clicks in stage 1 is the same for every policy. To see the expected number of clicks in the first stage is independent of the policy, note that the number of stage 1 clicks can be written as the sum of two binomial random variables. Let

$$n_{i,k}^{\text{new}} \sim \text{Bern}(p^{\text{new}}),$$
$$n_{i,k}^{\text{old}} \sim \text{Bern}(p^{\text{old}})$$

be Bernoulli random variables for whether or not the i^{th} user of type k clicks if they are shown a new or old card, and let $\pi_{i,k}$ be the indicator random variable for whether or not the i^{th} user of type k was shown a new card under policy π. Then, for any policy π, the number of clicks in stage 1 is

$$\sum_{k\in[K]}\sum_{i\in[\Lambda_k]}\left(\pi_{i,k}n_{i,k}^{\text{new}}+(1-\pi_{i,k})n_{i,k}^{\text{old}}\right)\sim \text{Binom}(B,p^{\text{new}})+\text{Binom}(\Lambda-B,p^{\text{old}}),$$

since, under any policy, B users are shown a new card, and click probability depends on the independent sum of the card specific CTRs. Hence, the expected number of stage 1 clicks equals $B\mu_0+(\Lambda-B)p^{\text{old}}$, which is clearly policy independent.

While the expected number of stage 1 clicks is policy independent, the same cannot be said for the number of users who churn between stage 1 and stage 2.

Observation 2: Number of users who churn can be written as a sum of binomials. For each user type, the number of users who churn is policy dependent and distributed as the sum of two binomials. Specifically, for a fixed type k, the number of users who churn under π is

$$\sum_{i\in[\Lambda_k]}\left\{\text{Bern}(q_k)\mathbb{I}\left(\pi_{i,k}n_{i,k}^{\text{new}}+(1-\pi_{i,k})n_{i,k}^{\text{old}}=1\right)+\text{Bern}(q_k+\delta_k)\mathbb{I}\left(\pi_{i,k}n_{i,k}^{\text{new}}+(1-\pi_{i,k})n_{i,k}^{\text{old}}=0\right)\right\}. \tag{5}$$

Letting

$$N_k^{\text{new}}(\pi):=\sum_{i\in[\Lambda_k]}\pi_{i,k}n_{i,k}^{\text{new}},$$
$$N_k^{\text{old}}(\pi):=\sum_{i\in[\Lambda_k]}(1-\pi_{i,k})n_{i,k}^{\text{old}}$$

be the number of users of type k who click the new card and old card, respectively, the expression in Eq. (5) is distributed as

$$\text{Binom}\left(N_k^{\text{new}}+N_k^{\text{old}},q_k\right)+\text{Binom}\left(\Lambda_k-N_k^{\text{new}}-N_k^{\text{old}},q_k+\delta_k\right), \tag{6}$$

which clearly depends on how cards are allocated by type since both N_k^{new} and N_k^{old} are functions of the policy π.

Finally, having characterized the stage 1 clicks, the users who churn, and the choice of card in stage 2 (recall Eq. (3)), we can now describe the number of clicks in stage 2 among the remaining users (i.e., the users who do not churn).

Observation 3: Stage 2 clicks can be written as a sum of at most two binomials. From Eq. (6), it follows that the number of type k users who remain in the system after stage 1 equals

$$M_k(\pi) := \Lambda_k - \mathrm{Binom}(N_k^{\mathrm{new}}(\pi) + N_k^{\mathrm{old}}(\pi), q_k) - \mathrm{Binom}(\Lambda_k - N_k^{\mathrm{new}}(\pi) - N_k^{\mathrm{old}}(\pi), q_k + \delta_k).$$

There are two cases depending on the choice of card in stage 2. If $y = \mathrm{old}$ (recall Eq. (3)), then all users are shown another old card with identical (expected) CTR to the old card show in stage 1 and the number of stage 2 clicks (across all returning users) follows a binomial:

$$\mathrm{Binom}\left(\sum_{k\in[K]} M_k(\pi), p^{\mathrm{old}}\right).$$

On the other hand, if $y = \mathrm{new}$, the policy should favor the new card and show it to all users who have not yet seen it in stage 1. Users who had seen the new card in stage 1 are shown an old card. Letting $M_k^{\mathrm{new}}(\pi)$ and $M_k^{\mathrm{old}}(\pi)$ be the number of users of type k who were previously shown the new card or old card, respectively, the number of stage 2 clicks is distributed as the sum of two binomials:

$$\mathrm{Binom}\left(\sum_{k\in[K]} M_k^{\mathrm{new}}(\pi), p^{\mathrm{old}}\right) + \mathrm{Binom}\left(\sum_{k\in[K]} M_k^{\mathrm{old}}(\pi), p^{\mathrm{new}}\right).$$

In the next section, we characterize the optimal policy for the platform recommendation problem, which, due to Observation 1 (expected number of stage 1 clicks is the same for every policy), is equivalent to maximizing expected number of stage 2 clicks. Given Observations 2 and 3, it should be clear that this is a rather non-trivial exercise, involving a careful analysis of understanding the interaction between learning and churning.

Before doing so, we briefly discuss a quantity that will be useful in our analysis. Recalling the definition of the card y (Eq. (3)), we are interested in the expected value of its CTR, i.e., $\mathbb{E}[p^y]$. By definition of y and the structure of the Beta-Bernoulli conjugacy as in Eq. (2), it follows that

$$\mathbb{E}[p^y] = \mathbb{E}_{p^{\mathrm{new}}}\left[\mathbb{E}_{N^{\mathrm{new}}}\left[\max\left\{p^{\mathrm{old}}, \frac{\alpha_0 + N^{\mathrm{new}}}{\alpha_0 + \beta_0 + B}\right\}\right] \middle| p^{\mathrm{new}}\right],$$

where $p^{\mathrm{new}} \sim \mathrm{Beta}(\alpha_0, \beta_0)$ is the CTR of the new card and $N^{\mathrm{new}} \sim \mathrm{Binom}(B, p^{\mathrm{new}})$ denotes the number of stage 1 clicks on the new card (out of B interactions). It is easy to see through this expression that $\mathbb{E}[p^y]$ is independent of the policy π and purely determined by the model primitives p^{old}, (α_0, β_0), and B. In fact, it admits the following closed-form characterization since the expectation over p^{new} integrates out (for details see Appendix C),

$$\mathbb{E}[p^y] = \frac{\Gamma(\alpha_0+\beta_0)}{\Gamma(\alpha_0)\Gamma(\beta_0)\Gamma(\alpha_0+\beta_0+B)} \sum_{b=0}^{B} \binom{B}{b} \max\left\{p^{\mathrm{old}}, \frac{\alpha_0+b}{\alpha_0+\beta_0+B}\right\} \Gamma(\alpha_0+b)\Gamma(\beta_0+B-b). \quad (7)$$

Where $\Gamma(\cdot)$ denotes the standard gamma function. It follows from Eq. (7) that we can exactly compute (as opposed to estimate) $\mathbb{E}[p^y]$ in a tractable manner (via a discrete sum over $B+1$ terms). Hence, in what follows, we will treat $\mathbb{E}[p^y]$ as a given.

4. Optimal Policy for Platform Recommendation Problem

In this section, we characterize the optimal policy for the platform recommendation problem. To build intuition, we first present a warm-up analysis in Section 4.1 for a setting with $B=1$ and two user types. This allows us to illustrate our model and some of its potentially counter-intuitive properties in a simple setting. We then generalize our intuition and characterize the structure of the optimal policy in Section 4.2, and compare the optimal policy against the blind randomization policy in Section 4.3.

4.1. Warm-up Analysis

Consider a setting with experimentation budget $B=1$ and two user types $K=2$ such that $(q_1,\delta_1)=(0,0)$ ("regular" user with no churning at all) and (q_2,δ_2) arbitrary ("new" user). Suppose there is one user of each type, i.e., $\Lambda_1=\Lambda_2=1$ and let all other parameters (p^{old}, α_0, and β_0) be arbitrary, except that $p^{\text{old}}>\mu_0$ (recall this prior mean of p^{new} is $\mu_0=\frac{\alpha_0}{\alpha_0+\beta_0}$). Hence, the decision here is which user to show the new card. To make the setting interesting, we further impose that the Beta(α_0,β_0) prior over p^{new} is such that if the new card is clicked in stage 1 (i.e., $N^{\text{new}}=1$), then the posterior mean of p^{new} is greater than p^{old}, i.e.,

$$\mu_1^+ := \frac{\alpha_0+1}{\alpha_0+\beta_0+1} > p^{\text{old}}.$$

Otherwise, the setup is rather trivial since y (recall Eq. (3)) would always equal "old" irrespective of the stage 1 data. Accordingly, our setup implies the following regarding y:

$$y=\begin{cases}\text{new} & \text{if new card is clicked in stage 1}\\ \text{old} & \text{otherwise.}\end{cases} \tag{8}$$

Recalling from Eq. (4) that $\pi=(\Lambda_1^{\text{new}},\Lambda_2^{\text{new}})$ denotes the platforms policy (i.e., who to show the new card to in stage 1), our goal here is to compare the following two policies: (1) $\pi_1=(1,0)$ (experiment on "regular" user) and (2) $\pi_2=(0,1)$ (experiment on "new" user). We can write out the expected numbers of clicks under both policies (see Appendix D for the details),

$$\begin{aligned}\mathcal{R}(\pi_1)&=\mu_0+p^{\text{old}}+p^{\text{old}}+\mathbb{E}[p^y]\left\{1-q_2-\delta_2(1-p^{\text{old}})\right\},\\ \mathcal{R}(\pi_2)&=\mu_0+p^{\text{old}}+\mathbb{E}[p^y]+p^{\text{old}}\left\{1-q_2-\delta_2(1-\mu_0)\right\}.\end{aligned}$$

Naturally, policy 1 is optimal if $\mathcal{R}(\pi_1)\geq\mathcal{R}(\pi_2)$. In fact, the optimal policy in this simple setup can be either π_1 or π_2, depending on the parameter regime. To illustrate this suppose $(\alpha_0,\beta_0)=(1,2)$, implying $\mu_0 = 1/3$ and $\mu_1^+ = 1/2$. Then, our setup in this subsection requires $p^{\text{old}} \in (1/3, 1/2)$. With $(q_2, \delta_2) = (0, 0.05)$, trivial algebra (along with the expression of $\mathbb{E}[p^y]$ as in Appendix D) implies that the optimal policy π^* is as follows:

$$\pi^* = \begin{cases} \pi_1 & \text{if } p^{\text{old}} > \frac{1}{8}(\sqrt{17} - 1) \approx 0.39 \\ \pi_2 & \text{otherwise.} \end{cases}$$

Hence, in this example, as p^{old} decreases, it is more desirable to experiment on the new user. It is worth highlighting a key underlying trade-off here. By definition of y (see Eq. (3)), $\mathbb{E}[p^y] \geq p^{\text{old}}$ (since y corresponds to the "max"), thus even though the new user is more likely to churn under policy 2, the platform enjoys a higher "reward" in stage 2 from the old user under policy 2 ($\mathbb{E}[p^y] \geq p^{\text{old}}$). This means that an optimal policy needs to carefully balance between controlling the churn rate of "new" users and ensuring the learning from experimentation is used on as many "regular" users as possible (since they are the ones who are more likely to return). Note that experimenting on regular users in stage 1 constrains the platform to not exploit its learning on them since the same card is not allowed to be shown twice to the same user.

The stylized analysis in this subsection shows that intuitive policies such as of experimenting on "regular" users first are not necessarily optimal. This raises the question of what structure does the optimal policy exhibit in general. Is it possible to characterize it in terms of the problem primitives? As we establish in the next subsection, the answer is yes.

4.2. Optimal Policy

We now show that the optimal policy for deciding which users receive the new card in stage 1 exhibits a crisp characterization. To this end, we first define a coefficient for each user type that we term the *Experimentation Coefficient* (EC), which plays a key role in our analysis.

Definition 1 (Experimentation Coefficient (EC)). Given model primitives B, p^{old}, (α_0, β_0) and the corresponding $\mathbb{E}[p^y]$ as in Eq. (7), we define the experimentation coefficient for a user type $k \in [K]$ with churn parameters (q_k, δ_k) as

$$\rho_k := (q_k + \delta_k)\left(\mathbb{E}[p^y] - p^{\text{old}}\right) - \delta_k p^{\text{old}}(\mathbb{E}[p^y] - \mu_0).$$

As we will see in Appendix E (Eq. (EC.14) in particular), the EC represents the effective difference in the probability of a stage 2 click between showing a new card versus an old card to the user in stage 1. We emphasize that the EC can be computed entirely from model primitives.

Without loss of generality, from here on out, we will reorder the user types in decreasing order of the EC, i.e., $\rho_k \geq \rho_{k+1}$ for all $k \in [K-1]$. We now define a policy (*rank*) in which we experiment on the users based on their EC, i.e., a user with a higher EC is given priority.

DEFINITION 2 (RANK (π^{RANK})). The *rank* policy sorts the user types by their experimentation coefficient and then experiments in descending order until the experimentation budget is spent, i.e.,

$$\pi^{\text{rank}} := \left(\min\{B, \Lambda_1\}, \min\{(B-\Lambda_1)^+, \Lambda_2\}, \ldots, \min\left\{ \left(B - \sum_{j=1}^{k-1} \Lambda_j \right)^+, \Lambda_k \right\}, \ldots \right).$$

For example, if $\Lambda_1 \geq B$, then $\pi^{\text{rank}} = (B, 0, \ldots, 0)$. Next, in Theorem 1, we establish that π^{rank} uniquely maximizes total expected clicks. We defer the proof to Appendix E.

THEOREM 1 (**π^{rank} is uniquely optimal**). *For any experimentation budget B and model parameters $[\{\Lambda_k\}_{k=1}^K, \{(q_k, \delta_k)\}_{k=1}^K, p^{old}, \alpha_0, \beta_0]$, π^{rank} is the unique policy that maximizes the total number of expected clicks across the two stages.*

Theorem 1 characterizes the optimal policy the platform should employ, and allows us to compare it with current practices. It implies that, to understand which users should be shown new cards, the platform should take into account a mixture of both the supply and demand side data: stage 1 and stage 2 CTRs (p^{old}, μ_0, $\mathbb{E}[p^y]$) as well as heterogeneous churn information ($[(q_k, \delta_k)]_{k\in[K]}$), all of which is packaged into the EC.

Interpreting Theorem 1 via ECs. As Theorem 1 reduces the platform recommendation problem to essentially computing the EC, we can understand the nature of the optimal policy by interpreting the experimentation coefficient in terms of model primitives. Recall the EC of type k is:

$$\rho_k = q_k \left(\mathbb{E}[p^y] - p^{\text{old}} \right) + \delta_k \left((\mathbb{E}[p^y] - p^{\text{old}}) - p^{\text{old}}(\mathbb{E}[p^y] - \mu_0) \right).$$

The first term, $q_k(\mathbb{E}[p^y] - p^{\text{old}})$, is always non-negative (since $\mathbb{E}[p^y] \geq p^{\text{old}}$) and hence, is non-decreasing in q_k. If δ_k equals 0 for all $k \in [K]$, i.e., there is no change in churn rate based on whether or not a user clicks in stage 1, then the optimal policy would be to simply experiment on the users who churn at the highest rates first. This is perhaps surprising as we showed in Section 2, new users churn at higher rates and thus without additional click interaction information, Theorem 1 implies that the platform should experiment on its newest users first! However, the intuition behind such a policy aligns with our warm-up analysis in Section 4.1. In particular, exploring the new users to learn (since they will churn independently of the recommendation) and exploiting the learning on the regular users in stage 2 (since they are more likely to return). Hence, the first term favors experimenting on the new users.

The second term, $\delta_k \left((\mathbb{E}[p^y] - p^{\text{old}}) - p^{\text{old}}(\mathbb{E}[p^y] - \mu_0) \right)$, is more complicated than the first and may be positive or negative. It encodes the interaction between clicking in the first stage, and being present in the second state. To illustrate this, suppose p^{new} was such that it could never exceed p^{old}, so $\mathbb{E}[p^y] = p^{\text{old}}$ a.s., the experimentation is fruitless. Then this second term is negative (and the first term is zero and $p^{\text{new}} \leq_{st} p^{\text{old}}$ implies $\mathbb{E}[p^{\text{new}}] = \mu_0 \leq p^{\text{old}}$), and the optimal policy is to experiment in ascending order of δ_k, i.e., experimenting on regular users so as to minimize expected churn as intuitively described at the beginning of this section. It is only when the experimentation may indeed reveal a superior card that the optimal policy requires a more careful accounting. Thus, the interaction between churning and learning is captured via the second term, which can possibly negate the effect of the first term and favor experimenting on the regular users.

Having characterized the optimal policy, in the next subsection, we compare it against the blind randomization policy as discussed in Section 2.3.

4.3. Cost of Blind Randomization

In Section 2.3, we introduced the *blind randomization* policy and provided evidence (Appendix B) that NetEase Music might be employing such a strategy. In this subsection, we explore the gap between the optimal policy π^{rank} and the blind randomization policy π^{blind}, which allocates the experimentation budget uniformly across all user types (weighted by the type frequencies):

$$\pi^{\text{blind}} := \left(\frac{\Lambda_1}{\Lambda} B, \ldots, \frac{\Lambda_K}{\Lambda} B \right).$$

Note that our definition of π^{blind} allows for the allocations to be non-integers, this can be handled in practice by simply rounding the numbers but we proceed without doing so for notational convenience.

An immediate corollary of Theorem 1 is that unless all user types are the same[5] in terms of the ECs $\{\rho_k\}_{k=1}^K$, blind randomization is strictly sub-optimal. Naturally, it is of interest to quantify this degree of sub-optimality as the *cost of blind randomization.* In the next theorem, we show that the normalized difference in clicks between π^{rank} and π^{blind} can be written as a weighted sum of the experimentation coefficients. We then use this characterization to give a type-agnostic upper bound on the difference in expected clicks. For ease of exposition, we assume without loss of generality that there is some k^* such that $\sum_{k=1}^{k^*} \Lambda_k = B$ since if not, then we can split the group of type k^* users into two types k_1^* and k_2^* with identical ECs.

THEOREM 2 ($\pi^{\textbf{rank}}$ **vs.** $\pi^{\textbf{blind}}$). *For any experimentation budget B, model parameters $[\{\Lambda_k\}_{k=1}^K, \{(q_k, \delta_k)\}_{k=1}^K, p^{old}, \alpha_0, \beta_0]$, and k^* such that $\sum_{k=1}^{k^*} \Lambda_k = B$, the normalized expected difference in clicks between the optimal policy and blind randomization is*

$$\frac{\mathcal{R}(\pi^{rank}) - \mathcal{R}(\pi^{blind})}{\Lambda} = \sum_{k=1}^{k^*} \frac{\Lambda_k}{\Lambda} \left(1 - \frac{B}{\Lambda}\right) \rho_k - \sum_{k=k^*+1}^{K} \frac{\Lambda_k}{\Lambda} \frac{B}{\Lambda} \rho_k. \tag{9}$$

[5] If all the ECs equal each other, then every policy is an instance of the rank policy and hence, is optimal.

Further, this difference can be upper bounded by

$$\frac{\mathcal{R}(\pi^{rank}) - \mathcal{R}(\pi^{blind})}{\Lambda} \leq \frac{\mathbb{E}[p^y] - p^{old} + p^{old}|\mathbb{E}[p^y] - \mu_0|}{4}. \tag{10}$$

A proof is presented in Appendix E. We remark that the inequality in Eq. (10) is tight for certain parameter regimes (as discussed at the end of the proof). For a more involved expression that is tight for arbitrary supply-side parameters, see Eq. (EC.18) in the proof. The results in Theorem 2 can serve as a useful guide for platform designers. First, the exact characterization in Eq. (9) explains when type-aware policies are most beneficial, i.e., when the user types are heterogeneous where heterogeneity is measured through the EC. If the platform has access to all parameters of our model, it can use the expression in Eq. (9) directly to gauge the potential gains of type-aware policies.

Of course, type parameters are a function of how the platform decides to separate users into types, which is a non-trivial task in itself. In the NetEase data set, we identified a natural characterization of users in terms of their prior engagement with the platform; however, we note that such a type delineation is not the only one possible. It may be the case that other distinguishing information offers a stronger characterization of groups with more dichotomous churn rates. In the second part of Theorem 2, we give a upper bound on the difference regardless of type. This upper bound holds for any platform currently engaged in blind randomization, and bounds the worst-case loss that could be incurred by foregoing some potentially clever type characterization of the user space. When the supply-side parameters are such that the right hand side of Eq. (10) is small, it implies that blind randomization is sufficient for the marketplace (irrespective of the demand-side parameters), making it an important decision tool for platform designers considering how to optimize their recommendation engines.

5. Numerical Study

We now illustrate our analytical results of Section 4 via a numerical study. To do so, motivated by our exploratory data analysis of Section 2, we consider two types of users: "regular" and "new". For the new user, based on our analysis in Section 2.2, we set the churn parameters to be $(q_2, \delta_2) = (0.47, 0.04)$ whereas for the regular user, we set $(q_1, \delta_1) = (0, 0)$ (meaning they do not churn at all)[6]. In terms of the supply-side parameters, we experiment with $p^{\text{old}} \in \{0, 0.01, \ldots, 0.99\}$ and $\mu_0 \in \{0, 0.01, \ldots, 0.99\}$. Such a wide range captures various scenarios of interest, e.g., a new card from a famous artist might have a high prior mean. We fix the sum of prior parameters[7] $\alpha_0 + \beta_0 = 23.79$

[6] In our data analysis (Appendix A), we found these parameters to be close to 0 ($\approx .05$ and $\approx .02$). Our results presented here are robust to such minor fluctuations.

[7] Intuitively, for a Beta(α_0, β_0) distribution, the sum $\alpha_0 + \beta_0$ captures the precision: higher the sum, lower the variance.

as determined by our data analysis in Section 2.1. Note that the prior parameters (α_0, β_0) are uniquely determined given μ_0 and $\alpha_0 + \beta_0$ as $\alpha_0 = \mu_0 \times (\alpha_0 + \beta_0)$. Finally, we fix experimentation budget at $B = 100$, which seems just high enough to "learn" a Beta parameter. Note that given the model primitives, we can perform all computations exactly (without any approximations) in a tractable fashion.

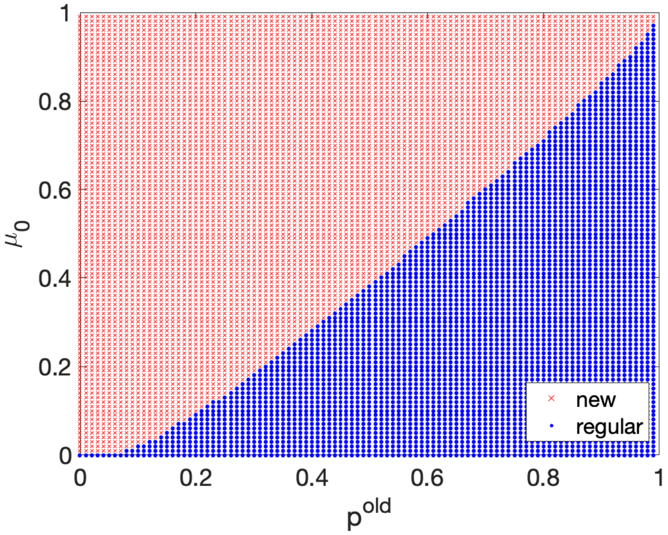
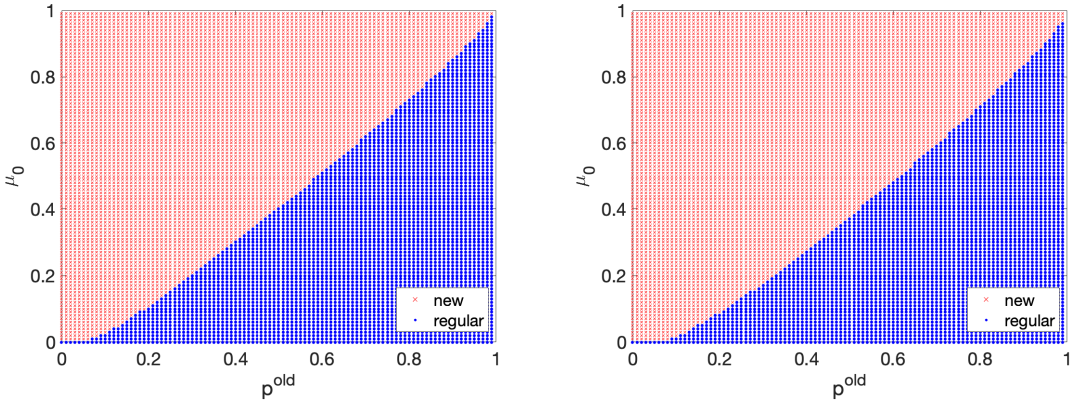


Figure 7 **Optimal policy for $B = 100$ as we vary p^{old} and μ_0. The blue dots (red crosses) represent points where the EC of the regular user (new user) is higher than that of the new user (regular user) and hence, it is optimal to experiment on the regular users (new users) first (c.f. Theorem 1).**

Given two user types, there are two rank-based policies of interest: (1) experiment on regular users first and (2) experiment on new users first. In Fig. 7, we plot the optimal policy as a function of the two parameters we experiment with: p^{old} and μ_0. As in the warm-up analysis of Section 4.1, it is optimal to experiment on the regular users first as p^{old} increases. Furthermore, the opposite is true for μ_0, i.e., it is optimal to experiment on the new users first as μ_0 increases. There is a clear separating boundary, defining regions where either of the two policy is optimal. It is worth highlighting that this boundary is non-symmetric and lies below the $y = x$ line, meaning it is sometimes optimal to first experiment on the new users even if $\mu_0 < p^{\text{old}}$ (recall Section 4.1).

In Fig. 8, we perform a sensitivity analysis with respect the experimentation budget B, by varying between 30 and 300 (recall the base value is 100). On visual inspection, the boundary shifts to the right as B increases but the sensitivity is low, suggesting robustness of the optimal policy to the value of B.

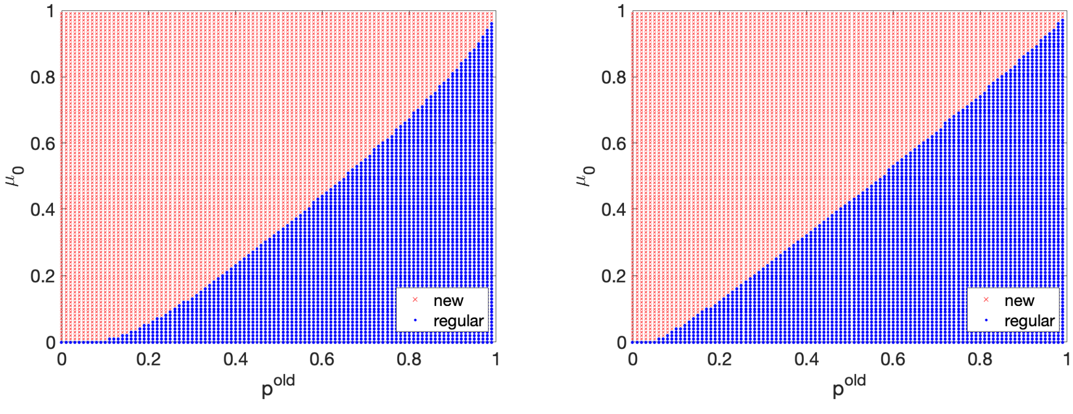
Similarly, in Fig. 9, we perform a sensitivity analysis with respect to the prior variance by scaling the sum of prior counts $\alpha_0 + \beta_0$ by a factor of 2 (both up and down). Roughly speaking, a



(a) $B = 30$ (b) $B = 300$

Figure 8 **Sensitivity analysis with respect to B.**

higher value of $\alpha_0 + \beta_0$ means a higher precision and hence, a lower variance. As $\alpha_0 + \beta_0$ increases, the separating boundary shifts to the left, getting closer to the $y = x$ line[8]. This suggests that a higher prior variance favors experimentation on new users. We also experimented with the churn parameters by increasing (q_1, δ_1) and decreasing (q_2, δ_2) and found the optimal policy to be very similar to as in Fig. 7.



(a) $\alpha_0 + \beta_0 = 11.895$ (b) $\alpha_0 + \beta_0 = 47.58$

Figure 9 **Sensitivity analysis with respect to the prior "precision" $\alpha_0 + \beta_0$.**

In addition to the structure of the optimal policy, we quantify the cost of blind randomization in the NetEase dataset (recall Theorem 2) in Table 1. We use the same parameters as above ($q_1 = 0$, $\delta_1 = 0$, $q_2 = 0.47$, $\delta_2 = 0.04$, $\alpha_0 + \beta_0 = 23.79$) with $\Lambda_1 = \Lambda_2 = B$, and experiment with various

[8] Though not shown here for brevity, we also tested the case of scaling $\alpha_0 + \beta_0$ by a factor of 10 and found the separating boundary to still lie below the $y = x$ line, but almost overlapping it.

values of p^{old} and μ_0. We report the percentage increase in the expected number of clicks when the platform employs the optimal policy as opposed to blind randomization, i.e.,

$$\frac{\mathcal{R}(\pi^{\text{rank}}) - \mathcal{R}(\pi^{\text{blind}})}{\mathcal{R}(\pi^{\text{blind}})}.$$

Clearly, the rank policy can result in significant gains in terms of user engagement (3% to 14% in the regime where p^{old} is low and μ_0 is high). For regions where p^{old} is high and μ_0 is not, the gains seem modest (0% to 2%). It is worth highlighting that even small gains would be amplified (in absolute value) when one multiplies them with the scale of operation of large platforms such as NetEase and YouTube, where thousands of new cards are released every day. Though omitted for brevity, we found the numbers to be of the similar magnitude for the various sensitivity analysis discussed above (with respect to both B and $\alpha_0 + \beta_0$).

		μ_0								
		0.01	**0.05**	**0.10**	**0.15**	**0.20**	**0.25**	**0.30**	**0.35**	**0.40**
p^{old}	**0.01**	3.55%	9.75%	11.95%	12.78%	13.22%	13.49%	13.67%	13.80%	13.90%
	0.05	0.18%	2.04%	5.06%	7.31%	8.76%	9.73%	10.43%	10.95%	11.35%
	0.10	0.05%	0.29%	1.47%	3.22%	4.91%	6.26%	7.29%	8.10%	8.75%
	0.15	0.11%	0.02%	0.35%	1.19%	2.42%	3.71%	4.87%	5.83%	6.63%
	0.20	0.15%	0.10%	0.02%	0.35%	1.00%	1.95%	3.00%	3.99%	4.86%
	0.25	0.19%	0.15%	0.09%	0.04%	0.33%	0.87%	1.64%	2.52%	3.38%
	0.30	0.23%	0.19%	0.14%	0.08%	0.05%	0.31%	0.77%	1.42%	2.17%
	0.35	0.27%	0.23%	0.18%	0.13%	0.07%	0.05%	0.28%	0.69%	1.25%
	0.40	0.31%	0.27%	0.22%	0.17%	0.13%	0.07%	0.04%	0.26%	0.62%

Table 1 Cost of blind randomization.

6. Conclusions

Recommendation platforms must coordinate interactions between users and content while carefully juggling a litany of challenges corresponding to both sides of this market. In this paper, we zoomed in on a particular aspect of the this coordination, balancing the first impression effect, in which new users who interact with recommended creatives are retained by the platform at higher rates, against the platform's need to experiment with new creatives. Our primary contribution is in the clean modeling approach, which abstracts away complicating factors to lay this problem bare, capturing the key features of this marketplace (supply-side learning and heterogeneous user behavior). Our model is directly inspired by real-world data. We explored the NetEase Music dataset and found evidence (a) of heterogeneous card-specific CTRs and them resembling a Beta distribution, (b) of a first impression effect and heterogeneous churn rates among users (depending on a user's prior engagement with the platform), and (c) that the platform appears to ignore heterogeneous user behavior and instead opts for a blind randomization policy.

Leveraging our model, we theoretically characterized the optimal policy using type specific experimentation coefficients (ECs), which synthesize the relevant model parameters into a single and interpretable number. We then quantified (both theoretically and numerically) the potentially loss in expected clicks the platform incurs by ignoring the underlying heterogeneity in its user population. Experimenting on a wide range of parameters, we showed the platform can increase the platform activity by around 0-14% by accounting for user heterogeneity in its experimentation policy.

For platform managers, our modeling approach yields a framework through which to think about retention of the platform's user-base, namely via the ECs, which can guide managers in understanding the users that are most sensitive to experimentation while capturing the long-term value. Armed with ECs, which may be based on a type classification using users level of prior engagement with the platform (e.g., new vs. regular), or with a different classification (e.g., mobile vs. desktop users), our work gives a simple, easy to implement priority rule for who should receive experimental content. Given that there is limited information on new content and users behavior is heterogeneous, establishing priority via ECs is a first order effect and, as we have shown theoretically and numerically, can significantly improve the efficiency of the platform.

Finally, to the best of our knowledge, our work is the first to integrate heterogeneous churning behavior of users with learning theory in order to maximize the long-run activity on a recommendation platform. We believe it opens the door for many possible avenues of future research. First, as mentioned at the beginning of Section 3, our model abstracts away some peculiarities of the real world. Naturally, it is of interest to study the extensions of our framework that capture such peculiarities. One feature that seems common across most recommendation platforms is the user seeing multiple cards during a visit, as opposed to just one. Our intuition is that it is possible to capture this via a cascade model (Craswell et al. 2008, Kveton et al. 2015) and extract similar managerial insights to ours (e.g., optimality of the rank policy). Second, in our framework, we model the CTR of a card as being independent of the user. With the rise of micro-level data on both the user-specific and card-specific features, it seems worthwhile to accommodate such contextual information. Though there exists some recent works in the space of contextual bandits (Chu et al. 2011, Agrawal and Goyal 2013, Agrawal et al. 2017, Oh and Iyengar 2019), accommodating contextual information in a non-myopic setting is an active area of research (Hallak et al. 2015, Dann et al. 2019, Modi and Tewari 2020). Third, given the objective of maximizing long-run engagement, we proposed a two-stage model in order to capture the long-term effects of churning. A fruitful exercise would be to generalize our two-stage model to multiple stages and understand the market outcome in steady-state. Finally, in this work, we primarily focused on the behavior of the users as a function of platform's experimentation policy and assumed the content generation process to be exogenous. Understanding the effects of platform's actions on content creators' motivation to publish new content and capturing such feedback loop in the model itself would be useful. We hope to pursue some of these directions in future research.

References

Agarwal, Deepak, Souvik Ghosh, Kai Wei, Siyu You. 2014. Budget pacing for targeted online advertisements at linkedin. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1613–1619.

Agnew, Julie R, Hazel Bateman, Christine Eckert, Fedor Iskhakov, Jordan Louviere, Susan Thorp. 2018. First impressions matter: An experimental investigation of online financial advice. *Management Science* **64**(1) 288–307.

Agrawal, Shipra, Vashist Avadhanula, Vineet Goyal, Assaf Zeevi. 2017. Thompson sampling for the mnl-bandit. *Conference on Learning Theory*. PMLR, 76–78.

Agrawal, Shipra, Navin Goyal. 2013. Thompson sampling for contextual bandits with linear payoffs. *International Conference on Machine Learning*. PMLR, 127–135.

Agrawal, Shipra, Randy Jia. 2019. Learning in structured mdps with convex cost functions: Improved regret bounds for inventory management. *Proceedings of the 2019 ACM Conference on Economics and Computation*. 743–744.

Agrawal, Shipra, Steven Yin, Assaf Zeevi. 2021. Dynamic pricing and learning under the bass model. *arXiv preprint arXiv:2103.05199* .

Alexander, Julia. 2020. Recommendation is one of the biggest issues facing streamers like Netflix, HBO Max, and more URL `https://www.theverge.com/2020/1/9/21058599/netflix-streaming-farewell-recommendations-lulu-wang-hbo-max-quibi`.

Asch, Solomon E. 1946. Forming impressions of personality. *The Journal of Abnormal and Social Psychology* **41**(3) 258.

Baardman, Lennart, Elaheh Fata, Abhishek Pani, Georgia Perakis. 2019. Learning optimal online advertising portfolios with periodic budgets. *Available at SSRN 3346642* .

Bastani, Hamsa, Pavithra Harsha, Georgia Perakis, Divya Singhvi. 2018. Learning personalized product recommendations with customer disengagement. *Available at SSRN 3240970* .

Bennett, James, Stan Lanning, et al. 2007. The Netflix Prize. *Proceedings of KDD cup and workshop*, vol. 2007. Citeseer, 35.

Besbes, Omar, Assaf Zeevi. 2015. On the (surprising) sufficiency of linear models for dynamic pricing with demand learning. *Management Science* **61**(4) 723–739.

Bhat, Nikhil, Vivek F Farias, Ciamac C Moallemi, Deeksha Sinha. 2020. Near-optimal ab testing. *Management Science* **66**(10) 4477–4495.

Bimpikis, Kostas, Mihalis G Markakis. 2019. Learning and hierarchies in service systems. *Management Science* **65**(3) 1268–1285.

Breese, John S, David Heckerman, Carl Kadie. 2013. Empirical analysis of predictive algorithms for collaborative filtering. *arXiv preprint arXiv:1301.7363* .

Chen, Xi, Zachary Owen, Clark Pixton, David Simchi-Levi. 2021. A statistical learning approach to personalization in revenue management. *Management Science* .

Chu, Wei, Lihong Li, Lev Reyzin, Robert Schapire. 2011. Contextual bandits with linear payoff functions. *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 208–214.

Craswell, Nick, Onno Zoeter, Michael Taylor, Bill Ramsey. 2008. An experimental comparison of click position-bias models. *Proceedings of the 2008 international conference on web search and data mining*. 87–94.

Dann, Christoph, Lihong Li, Wei Wei, Emma Brunskill. 2019. Policy certificates: Towards accountable reinforcement learning. *International Conference on Machine Learning*. PMLR, 1507–1516.

Davis, Philip J. 1959. Leonhard euler's integral: A historical profile of the gamma function: In memoriam: Milton abramowitz. *The American Mathematical Monthly* **66**(10) 849–869.

Dragone, Paolo, Rishabh Mehrotra, Mounia Lalmas. 2019. Deriving user-and content-specific rewards for contextual bandits. *The World Wide Web Conference*. 2680–2686.

Elmachtoub, Adam N, Paul Grigas. 2021. Smart "predict, then optimize". *Management Science* .

Feit, Elea McDonnell, Ron Berman. 2019. Test & roll: Profit-maximizing a/b tests. *Marketing Science* **38**(6) 1038–1058.

Feng, Yifan, Rene Caldentey, Christopher Thomas Ryan. 2018. Learning customer preferences from personalized assortments. *Available at SSRN* .

Figueiredo, Flavio, Bruno Ribeiro, Jussara M Almeida, Christos Faloutsos. 2016. Tribeflow: Mining & predicting user trajectories. *Proceedings of the 25th international conference on world wide web*. 695–706.

Gelman, Andrew, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, Donald B Rubin. 2013. *Bayesian data analysis*. CRC press.

Gijsbrechts, Joren, Robert N Boute, Jan A Van Mieghem, Dennis Zhang. 2020. Can deep reinforcement learning improve inventory management? performance on dual sourcing, lost sales and multi-echelon problems. *Performance on Dual Sourcing, Lost Sales and Multi-Echelon Problems (October 6, 2020)* .

Hallak, Assaf, Dotan Di Castro, Shie Mannor. 2015. Contextual markov decision processes. *arXiv preprint arXiv:1502.02259* .

Harrison, J Michael, N Bora Keskin, Assaf Zeevi. 2012. Bayesian dynamic pricing policies: Learning and earning under a binary prior distribution. *Management Science* **58**(3) 570–586.

IFPI. 2019. Ifpi global music report 2019 URL `https://www.ifpi.org/ifpi-global-music-report-2019/`.

James, Gareth, Daniela Witten, Trevor Hastie, Robert Tibshirani. 2013. *An introduction to statistical learning*, vol. 112. Springer.

Jannach, Dietmar, Markus Zanker, Alexander Felfernig, Gerhard Friedrich. 2010. *Recommender systems: an introduction*. Cambridge University Press.

Kanoria, Yash, Ilan Lobel, Jiaqi Lu. 2018. Managing customer churn via service mode control. *Columbia Business School Research Paper* (18-52).

Kao, Yi-hao, Benjamin Roy, Xiang Yan. 2009. Directed regression. *Advances in Neural Information Processing Systems* **22** 889–897.

Kelley, Harold H. 1950. *The warm-cold variable in first impressions of persons*.

Keskin, N Bora, Assaf Zeevi. 2017. Chasing demand: Learning and earning in a changing environment. *Mathematics of Operations Research* **42**(2) 277–307.

Kirk, Roger E. 2012. Experimental design. *Handbook of Psychology, Second Edition* **2**.

Kostić, Stefan M, Mirjana I Simić, Miroljub V Kostić. 2020. Social network analysis and churn prediction in telecommunications using graph theory. *Entropy* **22**(7) 753.

Kveton, Branislav, Csaba Szepesvari, Zheng Wen, Azin Ashkan. 2015. Cascading bandits: Learning to rank in the cascade model. *International Conference on Machine Learning*. PMLR, 767–776.

Lattimore, Tor, Csaba Szepesvári. 2020. *Bandit algorithms*. Cambridge University Press.

Liu, Guimei, Tam T Nguyen, Gang Zhao, Wei Zha, Jianbo Yang, Jianneng Cao, Min Wu, Peilin Zhao, Wei Chen. 2016. Repeat buyer prediction for e-commerce. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 155–164.

Lü, Linyuan, Matúš Medo, Chi Ho Yeung, Yi-Cheng Zhang, Zi-Ke Zhang, Tao Zhou. 2012. Recommender systems. *Physics reports* **519**(1) 1–49.

Martins, Helder. 2017. Predicting user churn on streaming services using recurrent neural networks.

McInerney, James, Benjamin Lacker, Samantha Hansen, Karl Higley, Hugues Bouchard, Alois Gruson, Rishabh Mehrotra. 2018. Explore, exploit, and explain: personalizing explainable recommendations with bandits. *Proceedings of the 12th ACM conference on recommender systems*. 31–39.

Mehrotra, Rishabh, Mounia Lalmas, Doug Kenney, Thomas Lim-Meng, Golli Hashemian. 2019. Jointly leveraging intent and interaction signals to predict user satisfaction with slate recommendations. *The World Wide Web Conference*. 1256–1267.

Melville, Prem, Vikas Sindhwani. 2010. Recommender systems. *Encyclopedia of machine learning* **1** 829–838.

Modi, Aditya, Ambuj Tewari. 2020. No-regret exploration in contextual reinforcement learning. *Conference on Uncertainty in Artificial Intelligence*. PMLR, 829–838.

Nambiar, Mila, David Simchi-Levi, He Wang. 2020. Dynamic inventory allocation with demand learning for seasonal goods. *Production and Operations Management* .

Negoescu, Diana M, Kostas Bimpikis, Margaret L Brandeau, Dan A Iancu. 2018. Dynamic learning of patient response types: An application to treating chronic diseases. *Management science* **64**(8) 3469–3488.

Netflix. 2006. The Netflix Prize Rules URL `https://www.netflixprize.com/assets/rules.pdf`.

Oh, Min-hwan, Garud Iyengar. 2019. Thompson sampling for multinomial logit contextual bandits. *NeurIPS*. 3145–3155.

Padilla, Nicolas, Eva Ascarza. 2017. First impressions count: Leveraging acquisition data for customer management.

Rabin, Matthew, Joel L Schrag. 1999. First impressions matter: A model of confirmatory bias. *The quarterly journal of economics* **114**(1) 37–82.

Rubin, Donald B. 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology* **66**(5) 688.

Schafer, J Ben, Dan Frankowski, Jon Herlocker, Shilad Sen. 2007. Collaborative filtering recommender systems. *The adaptive web*. Springer, 291–324.

Silvey, Samuel. 2013. *Optimal design: an introduction to the theory for parameter estimation*, vol. 1. Springer Science & Business Media.

Slivkins, Aleksandrs. 2019. Introduction to multi-armed bandits. *arXiv preprint arXiv:1904.07272* .

Sutton, Richard S, Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Thompson, William R. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* **25**(3/4) 285–294.

Watson, Amy (November-14). 2018. Digital music - statistics and facts URL `https://www.statista.com/topics/1386/digital-music/`.

White, Ryen W, Peter Bailey, Liwei Chen. 2009. Predicting user interests from contextual information. *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. 363–370.

Xu, Jian, Kuang-chih Lee, Wentong Li, Hang Qi, Quan Lu. 2015. Smart pacing for effective online ad campaign optimization. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2217–2226.

Yang, Carl, Xiaolin Shi, Luo Jie, Jiawei Han. 2018. I know you'll be back: Interpretable new user clustering and churn prediction on a mobile social application. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 914–922.

Zhang, Dennis J, Ming Hu, Xiaofei Liu, Yuxiang Wu, Yong Li. 2020. Netease cloud music data. *Manufacturing & Service Operations Management* .

Zhu, Taozeng, Jingui Xie, Melvyn Sim. 2021. Joint estimation and robustness optimization. *Management Science* .

Appendix

Appendix A: Further Details on the First Impression Effect

In this appendix, we discuss the churn rate of "regular" users. Recall from Section 2.2 that for a new user, the churn rate is around 45% if they click and 50% if they do not click, implying a "delta" of around 5 percentage points. For a "regular" user, we found the churn rate to be around 5% if they click, with a delta of around 1 to 2 percentage points. For illustrative purposes, we defined a regular user as follows. We considered the set of users who have been on the platform for at least six months. Intuitively, a regular user is someone who visits the platform frequently. To enforce this, we considered day 10 in the data and filtered for users (six months plus) who visited the platform on both days 9 and 10. There were 36,429 such users. We analyzed their interaction on day 10 and whether they churned (i.e., never returned) as a function of whether they clicked or not on day 10. Out of the 36,429 users, 6161 clicked during their visit on day 10, with 238 (out of 6161) churning, i.e., a churn rate of around 3.86% given a click. Of the remaining 30,268 users who did not click, 1705 churned (5.63%), implying a delta of 1.77 percentage points. As a robustness check, in addition to day 10, we repeated the calculation for days 7, 8, and 9 and found the numbers to be similar: $(3.56\%, 5.46\%)$, $(5.01\%, 6.26\%)$, and $(5.80\%, 6.90\%)$, respectively. For each day, there were around 40,000 users (high sample size). Note that our definition of a regular user is just one possibility, and it is possible to construct alternative "types" of users (perhaps with even lower delta and hence, a stronger first impression effect). Our model and analysis in Section 3 and Section 4 respectively are flexible enough to allow for an arbitrary number of user types with corresponding base churn rate and delta.

Appendix B: Evidence of Blind Randomization in NetEase Music Dataset

In order to understand whether the NetEase dataset supports the hypothesis of blind randomization, we consider the new cards in the dataset (cards published in November, the month for which the data is released). Though there are around 100,000 such cards, it is possible that the first impression of these cards was to a user outside the 2 million users in the dataset. Accordingly, in order to be sure we only use the new cards for which we know the first impression, we focus on the cards that were shown exactly once in November (over *all* users at NetEase and not just the 2 million users in the released dataset). We inferred this using the `mlog_stats.csv` file provided in the dataset (Zhang et al. 2020). We found 1024 such cards.

We conduct a Bayesian analysis with a null hypothesis that blind randomization exists. Mathematically, given a new card and two user types (regular and new), blind randomization is equivalent to:

$$\mathbb{P}\{\text{show new card} \mid \text{user is new}\} = \mathbb{P}\{\text{show new card} \mid \text{user is regular}\}. \tag{EC.1}$$

Under the null hypothesis, both probabilities in (EC.1) equal $\mathbb{P}\{\text{show new card}\}$ (i.e., experimentation is type-independent), which we denote by p_{blind}. For inference, we consider the ratio

$$\frac{\mathbb{P}\{\text{show new card} \mid \text{user is new}\}}{\mathbb{P}\{\text{show new card} \mid \text{user is regular}\}} = \frac{\mathbb{P}\{\text{user is new} \mid \text{show new card}\}}{\mathbb{P}\{\text{user is regular} \mid \text{show new card}\}} \times \frac{\mathbb{P}\{\text{user is regular}\}}{\mathbb{P}\{\text{user is new}\}} \tag{EC.2}$$

as a test statistic, where the equality follows Bayes' theorem. Under the null hypothesis, our test statistic should be close to 1 whereas a value less than 1 (greater than 1) indicates new users receive less (more) experimentation than regular users.

We classify a user to be "new" if they registered on the platform within x months of November and we play with $x \in \{0,1,2\}$ as a robustness check. We estimate $\mathbb{P}\{\text{user is new} \mid \text{show new card}\}$ as the empirical ratio

$$\sum_{c=1}^{1024} \frac{\mathbb{I}\{\text{first impression of card } c \text{ shown to a new user}\}}{1024}, \tag{EC.3}$$

where the sum is over the 1024 new cards discussed above. Trivially, $\mathbb{P}\{\text{user is regular} \mid \text{show new card}\} = 1 - \mathbb{P}\{\text{user is new} \mid \text{show new card}\}$. Similarly, our estimate of $\mathbb{P}\{\text{user is new}\}$ equals the empirical proportion

$$\sum_{i=1}^{57{,}750{,}012} \frac{\mathbb{I}\{\text{impression } i \text{ shown to a new user}\}}{57,750,012}, \tag{EC.4}$$

where the sum is over all the 57,750,012 impressions in the dataset. Trivially, $\mathbb{P}\{\text{user is regular}\} = 1 - \mathbb{P}\{\text{user is new}\}$. We plug these estimates to evaluate our test statistic from (EC.2) and present a summary in Table EC.1.

x	0	1	2
Test statistic	0.72	1.82	1.24

Table EC.1 Test statistic as in (EC.2) **corresponding to the 1024 cards with exactly one impression. Recall that** $x \in \{0,1,2\}$ **defines the class of new users (**x **months since registration). In terms of the raw quantities, number of impressions to new users equals approximately 0.4 million, 2.4 million, and 3.8 million for** $x = 0,1,2$**, respectively. In addition, the number of new cards (out of 1024) with first impression to a new user equals 5, 76, and 82 for** $x = 0,1,2$**, respectively.**

The estimate of the test statistic for $x \in \{1,2\}$ is bigger than 1 whereas for $x = 0$, the estimate equals 0.74. By themselves, these numbers are hard to interpret. To quantify their statistical significance, we use the concept of Bayesian p-values (Gelman et al. 2013). As mentioned above, under the null hypothesis, the experimentation probability is independent of the type and we denote it by p_{blind}. Under the null hypothesis, it seems reasonable to estimate p_{blind} as the empirical ratio

$$\frac{\text{number of new cards}}{\text{number of impressions}} = \frac{1024}{57,750,012}. \tag{EC.5}$$

Further, as before, we use the ratio

$$\frac{\text{number of impressions to new users}}{\text{number of impressions}} = \frac{n_{\text{new}}}{57,750,012} \tag{EC.6}$$

to capture $\mathbb{P}\{\text{user is new}\}$. Accordingly, we get

$$\frac{\mathbb{P}\{\text{user is regular}\}}{\mathbb{P}\{\text{user is new}\}} = \frac{n_{\text{regular}}}{n_{\text{new}}}, \tag{EC.7}$$

where $n_{\text{regular}} = 57,750,012 - n_{\text{new}}$ denotes the number of impressions to the regular users. Recall from the caption of Table EC.1 that n_{new} equals approximately 0.4 million, 2.4 million, and 3.8 million for $x = 0,1,2$, respectively.

Given our setup, under the null hypothesis, the number of new cards (out of 1024) with first impression to a new user equals

$$y \sim \text{Binomial}(p_{\text{blind}}, n_{\text{new}}). \tag{EC.8}$$

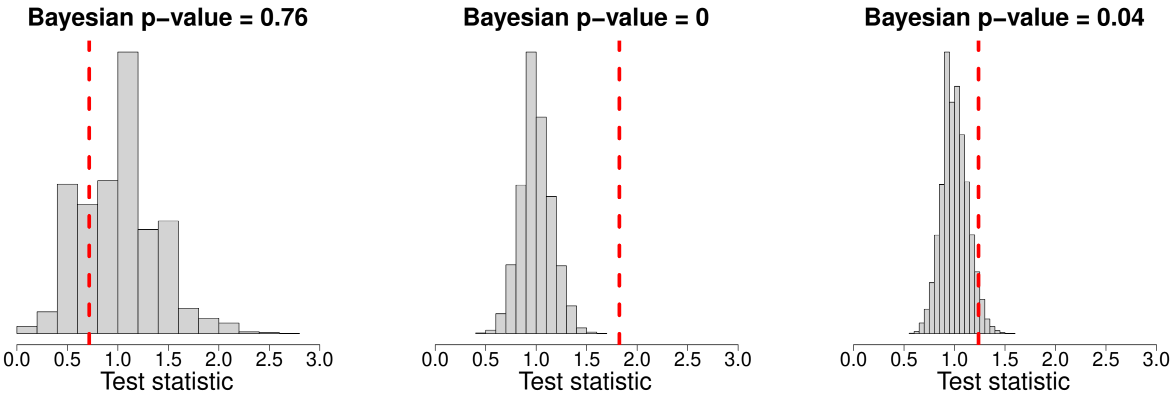
It is possible for y to exceed 1024 (though extremely unlikely given our estimates) and in that case, one can simply cap it at 1024. Then, the number of new cards (out of 1024) with first impression to an regular user equals $1024 - y$. This implies

$$\frac{\mathbb{P}\{\text{user is new} \mid \text{show new card}\}}{\mathbb{P}\{\text{user is regular} \mid \text{show new card}\}} = \frac{y}{1024 - y}. \tag{EC.9}$$

Plugging (EC.7) and (EC.9) in (EC.2) gives us the distribution of our test statistic under the null hypothesis:

$$\frac{y}{1024 - y} \times \frac{n_{\text{regular}}}{n_{\text{new}}} \text{ where } y \text{ as in (EC.8).} \tag{EC.10}$$

We use Monte-Carlo to understand this distribution (by simulating y). Furthermore, we know the value of the test statistic in real-data (see Table EC.1). Hence, we can estimate the Bayesian p-value for each $x \in \{0, 1, 2\}$. In Figure EC.1, we report these Bayesian p-values. For $x = 0$, even though the test statistic is below 1, the Bayesian p-value equals 0.76, indicating the data is consistent with the null hypothesis. Moreover, for $x \in \{1, 2\}$, the Bayesian p-value is close to zero, indicating more experimentation on new users than suggested by the null hypothesis.



(a) $x = 0$ (b) $x = 1$ (c) $x = 2$

Figure EC.1 Bayesian p-values corresponding to the 1024 cards for $x \in \{0, 1, 2\}$. For each $x \in \{0, 1, 2\}$, we compute the distribution of our test statistic under the null hypothesis (as in (EC.10)) via Monte-Carlo (10,000 samples of y as in (EC.8)). Then, we check where the ratio computed using real-data (recall Table EC.1) lies on this distribution (dotted red line) and compute the area to its right (Bayesian p-value). By definition, Bayesian p-value lies between 0 and 1 and a value around 0.5 suggests the data is consistent with the null hypothesis. On the other hand, a Bayesian p-value closer to 0 (1) suggests more experimentation on new (regular) users.

Appendix C: Characterization of $\mathbb{E}[p^y]$

Recall from Eq. (3) that

$$y = \begin{cases} \text{new} & \text{if } \mu_1 > p^{\text{old}} \\ \text{old} & \text{otherwise,} \end{cases}$$

where the posterior mean $\mu_1 = \frac{\alpha_1}{\alpha_1+\beta_1}$. Furthermore, recall the structure of the Beta-Bernoulli conjugacy as in Eq. (2):

$$\alpha_1 = \alpha_0 + N_{\text{new}}$$
$$\beta_1 = \beta_0 + B - N_{\text{new}},$$

where $N^{\text{new}} \sim \text{Binom}(B, p^{\text{new}})$ denotes the number of stage 1 clicks on the new card out of B interactions. Then, it follows that

$$\mathbb{E}[p^y] = \mathbb{E}_{p^{\text{new}}}\left[\mathbb{E}_{N^{\text{new}}}\left[\max\left\{p^{\text{old}}, \frac{\alpha_0 + N^{\text{new}}}{\alpha_0+\beta_0+B}\right\}\right] \middle| p^{\text{new}}\right],$$

where $p^{\text{new}} \sim \text{Beta}(\alpha_0, \beta_0)$. Plugging in the corresponding probability density/mass functions of beta and binomial distributions, we get

$$\begin{aligned}
\mathbb{E}[p^y] &= \int_p \frac{p^{\alpha_0-1}(1-p)^{\beta_0-1}}{\mathsf{B}(\alpha_0,\beta_0)} \sum_{b=0}^{B} \binom{B}{b} p^b (1-p)^{B-b} \max\left\{p^{\text{old}}, \frac{\alpha_0+b}{\alpha_0+\beta_0+B}\right\} dp \\
&= \sum_{b=0}^{B} \frac{\binom{B}{b}}{\mathsf{B}(\alpha_0,\beta_0)} \max\left\{p^{\text{old}}, \frac{\alpha_0+b}{\alpha_0+\beta_0+B}\right\} \int_p p^{\alpha_0+b-1}(1-p)^{\beta_0+B-b-1} dp \\
&= \sum_{b=0}^{B} \frac{\binom{B}{b}}{\mathsf{B}(\alpha_0,\beta_0)} \max\left\{p^{\text{old}}, \frac{\alpha_0+b}{\alpha_0+\beta_0+B}\right\} \frac{\Gamma(\alpha_0+b)\Gamma(\beta_0+B-b)}{\Gamma(\alpha_0+\beta_0+B)} \\
&= \frac{1}{\mathsf{B}(\alpha_0,\beta_0)\Gamma(\alpha_0+\beta_0+B)} \sum_{b=0}^{B} \binom{B}{b} \max\left\{p^{\text{old}}, \frac{\alpha_0+b}{\alpha_0+\beta_0+B}\right\} \Gamma(\alpha_0+b)\Gamma(\beta_0+B-b) \\
&= \frac{\Gamma(\alpha_0+\beta_0)}{\Gamma(\alpha_0)\Gamma(\beta_0)\Gamma(\alpha_0+\beta_0+B)} \sum_{b=0}^{B} \binom{B}{b} \max\left\{p^{\text{old}}, \frac{\alpha_0+b}{\alpha_0+\beta_0+B}\right\} \Gamma(\alpha_0+b)\Gamma(\beta_0+B-b).
\end{aligned}$$

Note that $\mathsf{B}(\cdot)$ and $\Gamma(\cdot)$ denote the standard beta and gamma functions, respectively.

Appendix D: Further Details on the Warm-up Analysis

Under π_1, in stage 1, the click behavior is governed as $n_{1,1}^{\text{new}} \sim \text{Bern}(p^{\text{new}})$ and $n_{1,2}^{\text{old}} \sim \text{Bern}(p^{\text{old}})$, meaning the expected number of stage 1 clicks equals $\mu_0 + p^{\text{old}}$. In terms of churning, since $q_1 = \delta_1 = 0$, regular user returns in stage 2 w.p. 1 whereas the new user returns w.p. $q_2 + \delta_2(1 - n_{1,2}^{\text{old}})$. In stage 2, the regular user is shown an old card (since they have already seen the new card) whereas the new user is shown card y, implying expected number of stage 2 clicks equals $p^{\text{old}} + \mathbb{E}[p^y]\left\{1 - q_2 - \delta_2(1-p^{\text{old}})\right\}$ (note that all the random variables in this expression are independent and hence, decouple). This implies the total expected clicks under policy 1 equals

$$\mathcal{R}(\pi_1) = \mu_0 + p^{\text{old}} + p^{\text{old}} + \mathbb{E}[p^y]\left\{1 - q_2 - \delta_2(1-p^{\text{old}})\right\}.$$

A similar argument implies

$$\mathcal{R}(\pi_2) = \mu_0 + p^{\text{old}} + \mathbb{E}[p^y] + p^{\text{old}}\left\{1 - q_2 - \delta_2(1-\mu_0)\right\}.$$

Furthermore, note that the expression for $\mathbb{E}[p^y]$ in this stylized setup is straightforward. In particular, combining Eq. (8) with Eq. (7) and plugging in $B=1$ implies

$$\begin{aligned}
\mathbb{E}[p^y] &= \frac{\Gamma(\alpha_0+\beta_0)}{\Gamma(\alpha_0)\Gamma(\beta_0)\Gamma(\alpha_0+\beta_0+B)} \sum_{b=0}^{B} \binom{B}{b} \max\left\{p^{\text{old}}, \frac{\alpha_0+b}{\alpha_0+\beta_0+B}\right\} \Gamma(\alpha_0+b)\Gamma(\beta_0+B-b) \\
&= \frac{\Gamma(\alpha_0+\beta_0)}{\Gamma(\alpha_0)\Gamma(\beta_0)\Gamma(\alpha_0+\beta_0+1)} \left\{\Gamma(\alpha_0)\Gamma(\beta_0+1)p^{\text{old}} + \Gamma(\alpha_0+1)\Gamma(\beta_0)\mu_1^+\right\} \\
&= \frac{\Gamma(\alpha_0+\beta_0)\Gamma(\alpha_0)\Gamma(\beta_0+1)}{\Gamma(\alpha_0)\Gamma(\beta_0)\Gamma(\alpha_0+\beta_0+1)} p^{\text{old}} + \frac{\Gamma(\alpha_0+\beta_0)\Gamma(\alpha_0+1)\Gamma(\beta_0)}{\Gamma(\alpha_0)\Gamma(\beta_0)\Gamma(\alpha_0+\beta_0+1)} \mu_1^+ \\
&= \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_0)} \frac{\Gamma(\alpha_0+\beta_0)}{\Gamma(\alpha_0+\beta_0+1)} \frac{\Gamma(\beta_0+1)}{\Gamma(\beta_0)} p^{\text{old}} + \frac{\Gamma(\beta_0)}{\Gamma(\beta_0)} \frac{\Gamma(\alpha_0+\beta_0)}{\Gamma(\alpha_0+\beta_0+1)} \frac{\Gamma(\alpha_0+1)}{\Gamma(\alpha_0)} \mu_1^+ \\
&= \frac{\beta_0}{\alpha_0+\beta_0} p^{\text{old}} + \frac{\alpha_0}{\alpha_0+\beta_0} \mu_1^+.
\end{aligned}$$

The last equality is true since $\frac{\Gamma(z+1)}{\Gamma(z)} = z$ for $z > 0$ (see (19) in Davis (1959)).

Appendix E: Omitted Proofs

Proof of Theorem 1. Recall from Section 3.2 that the total number of clicks in stage 1 is identical in expectation across policies, thus we will focus exclusively on stage 2 clicks. Fix a user u_k of type k and suppose they are shown an old card in stage 1. The probability that u_k does not churn can be computed by conditioning on whether or not they click in stage 1,

$$\begin{aligned}
\mathbb{E}[\mathbb{I}\{\sim \text{Churn}, \text{Old}\}] &= \mathbb{E}[\mathbb{E}[\mathbb{I}\{\sim \text{Churn}, \text{Old}\}|\text{Click}] \\
&= p^{\text{old}}(1-q_k) + (1-p^{\text{old}})(1-q_k-\delta_k) \\
&= 1-q_k-\delta_k\left(1-p^{\text{old}}\right).
\end{aligned}$$

Then, in stage 2, u_k is shown either *another* old card, or the new card. Thus, the probability of clicking in stage 2 is independent of the u_k's behaviour in stage 1, and the expected probability of a stage 2 click having been shown a old card is the product of the chance the user does not churn and the CTR of the optimal stage 2 card (y in this case since the user was shown old card in stage 1),

$$\lambda_k^{\text{old}} := \mathbb{E}[p^y]\left(1-q_k-\delta_k\left(1-p^{\text{old}}\right)\right). \tag{EC.12}$$

Now, suppose instead u_k was shown a new card in stage 1. Again, the probability that u_k does not churn is

$$\begin{aligned}
\mathbb{E}[\mathbb{I}\{\sim \text{Churn}, \text{New}\}] &= \mathbb{E}[\mathbb{E}[\mathbb{I}\{\sim \text{Churn}, \text{New}\}|\text{Click}] \\
&= \mu_0(1-q_k) + (1-\mu_0)(1-q_k-\delta_k) \\
&= 1-q_k-\delta_k(1-\mu_0).
\end{aligned}$$

Then, in stage 2, u_k is always shown an old card to avoid redundancy. Thus, the probability of clicking in stage 2 is again independent of the u_k's behaviour in stage 1 and the probability of stage 2 click having been shown a new card in stage 1 is

$$\lambda_k^{\text{new}} := p^{\text{old}}\left(1-q_k-\delta_k(1-\mu_0)\right). \tag{EC.13}$$

Now, we can compute the marginal cost (in clicks) of showing a new card to u_k instead of an old card by taking the difference between Eq. (EC.13) and Eq. (EC.12):

$$\begin{aligned}\lambda_k^{\text{new}} - \lambda_k^{\text{old}} &= p^{\text{old}}\left(1-q_k-\delta_k\left(1-\mu_0\right)\right) - \mathbb{E}[p^y]\left(1-q_k-\delta_k\left(1-p^{\text{old}}\right)\right)\\ &= \left(p^{\text{old}}-\mathbb{E}[p^y]\right) + q_k\left(\mathbb{E}[p^y]-p^{\text{old}}\right) + \delta_k\left(\left(\mathbb{E}[p^y]-p^{\text{old}}\right) - p^{\text{old}}(\mathbb{E}[p^y]-\mu_0)\right)\\ &= \left(p^{\text{old}}-\mathbb{E}[p^y]\right) + (q_k+\delta_k)\left(\mathbb{E}[p^y]-p^{\text{old}}\right) - \delta_k p^{\text{old}}(\mathbb{E}[p^y]-\mu_0)\\ &= \left(p^{\text{old}}-\mathbb{E}[p^y]\right) + \rho_k, \end{aligned} \tag{EC.14}$$

where ρ_k equals the experimentation coefficient (recall Definition 1 in Section 4.2). Finally, since Eq. (EC.14) is the expected difference in clicks between showing a new card versus an old card (in stage 1) to a user of type k, and the first term $(p^{\text{old}}-\mathbb{E}[p^y])$ is a constant with respect to the experimentation policy π, maximizing total expected clicks is equivalent to the following optimization:

$$\begin{aligned}\max_{\pi} & \sum_{k\in[K]}\sum_{i\in[\Lambda_k]}\{\pi_{i,k}\rho_k-(1-\pi_{i,k})\rho_k\}\\ \text{s.t.} & \sum_{k\in[K]}\sum_{i\in[\Lambda_k]}\pi_{i,k}=B\\ & \pi_{i,k}\in\{0,1\},\ k\in[K],\ i\in[\Lambda_k].\end{aligned}$$

By examination of the objective function, the optimal policy (in stage 1) is to sort users by ρ_k, and show the new card in descending order until the budget B is exhausted. This implies the optimality and uniqueness of π^{rank}. The proof is now complete. □

Proof of Theorem 2. For blind randomization, the expected number of clicks can be easily computed as the sum of the policy-independent stage 1 clicks (c.f. Observation 1 in Section 3.2), and the stage 2 clicks from users shown old/new cards, where the allocation is proportionally split across types,

$$\mathcal{R}(\pi^{\text{blind}}) = \mathbb{E}[\text{stage 1 clicks}] + \sum_{k=1}^{K}\left\{\frac{B\Lambda_k}{\Lambda}\lambda_k^{\text{new}} + \left(\Lambda_k - \frac{B\Lambda_k}{\Lambda}\right)\lambda_k^{\text{old}}\right\}, \tag{EC.15}$$

where λ_k^{new} and λ_k^{old} are as defined in the proof of Theorem 1 (see Eq. (EC.12) and Eq. (EC.13)). Similarly, using the definition of k^*, for the rank policy, the expected number of clicks equals

$$\mathcal{R}(\pi^{\text{rank}}) = \mathbb{E}[\text{stage 1 clicks}] + \sum_{k=1}^{k^*}\Lambda_k\lambda_k^{\text{new}} + \sum_{k=k^*+1}^{K}\Lambda_k\lambda_k^{\text{old}}. \tag{EC.16}$$

Hence, the difference between these policies is

$$\begin{aligned}\mathcal{R}(\pi^{\text{rank}}) - \mathcal{R}(\pi^{\text{blind}}) &= \sum_{k=1}^{k^*}\Lambda_k\left(1-\frac{B}{\Lambda}\right)(\lambda_k^{\text{new}}-\lambda_k^{\text{old}}) + \sum_{k=k^*+1}^{K}\Lambda_k\frac{B}{\Lambda}(\lambda_k^{\text{old}}-\lambda_k^{\text{new}})\\ &= \sum_{k=1}^{k^*}\Lambda_k\left(1-\frac{B}{\Lambda}\right)(p^{\text{old}}-\mathbb{E}[p^y]+\rho_k) + \sum_{k=k^*+1}^{K}\Lambda_k\frac{B}{\Lambda}(-p^{\text{old}}+\mathbb{E}[p^y]-\rho_k)\\ &= \sum_{k=1}^{k^*}\Lambda_k\left(1-\frac{B}{\Lambda}\right)\rho_k - \sum_{k=k^*+1}^{K}\Lambda_k\frac{B}{\Lambda}\rho_k,\end{aligned} \tag{EC.17}$$

where the second equality follows from plugging in Eq. (EC.14), and the third equality follows from recalling that k^* is such that $\sum_{k=1}^{k^*}\Lambda_k = B$ and simplifying. Dividing through by Λ gives Eq. (9) as desired.

Armed with Eq. (EC.17), we now consider parameter regimes which maximize this difference. To that end, fix p^{old}, μ_0, and $\mathbb{E}[p^y]$, and note the maximum the EC can be is

$$\rho_U := \max_{q,\delta} \left\{ (q+\delta)\left(\mathbb{E}[p^y] - p^{\text{old}}\right) - \delta p^{\text{old}}(\mathbb{E}[p^y] - \mu_0) \right\} = \begin{cases} \mathbb{E}[p^y] - p^{\text{old}} & \text{if } \mathbb{E}[p^y] \geq \mu_0 \\ \mathbb{E}[p^y] - p^{\text{old}} - p^{\text{old}}(\mathbb{E}[p^y] - \mu_0) & \text{if } \mathbb{E}[p^y] < \mu_0, \end{cases}$$

depending on whether or not $p^{\text{old}}(\mathbb{E}[p^y] - \mu_0)$ is positive or negative. Similarly, the minimum the EC can be is

$$\rho_L := \min_{q,\delta} \left\{ (q+\delta)\left(\mathbb{E}[p^y] - p^{\text{old}}\right) - \delta p^{\text{old}}(\mathbb{E}[p^y] - \mu_0) \right\} = \begin{cases} \left((\mathbb{E}[p^y] - p^{\text{old}}) - p^{\text{old}}(\mathbb{E}[p^y] - \mu_0)\right)^- & \text{if } \mathbb{E}[p^y] \geq \mu_0 \\ 0 & \text{if } \mathbb{E}[p^y] < \mu_0, \end{cases}$$

where the superscript minus means $(x)^- := \min\{x, 0\}$. Upper bounding types $k \leq k^*$ with ρ_U and lower bounding types $k > k^*$ with ρ_L in Eq. (EC.17) yields,

$$\begin{aligned}
\frac{\mathcal{R}(\pi^{\text{rank}}) - \mathcal{R}(\pi^{\text{blind}})}{\Lambda} &\leq \frac{1}{\Lambda}\left\{ \sum_{k=1}^{k^*} \Lambda_k \left(1 - \frac{B}{\Lambda}\right)\rho_U - \sum_{k=k^*+1}^{K} \Lambda_k \frac{B}{\Lambda}\rho_L \right\} \\
&= \frac{1}{\Lambda}\left\{ B\left(1 - \frac{B}{\Lambda}\right)\rho_U - (\Lambda - B)\frac{B}{\Lambda}\rho_L \right\} && \left(\sum_{k=1}^{k^*} \Lambda_k = B\right) \\
&\leq \frac{1}{\Lambda}\left\{ (\rho_U - \rho_L) \max_{B \in [0,\Lambda]} \left(B - \frac{B^2}{\Lambda}\right) \right\} \\
&= \frac{\rho_U - \rho_L}{4} \\
&= \begin{cases} \frac{\mathbb{E}[p^y] - p^{\text{old}} - \left((\mathbb{E}[p^y] - p^{\text{old}}) - p^{\text{old}}(\mathbb{E}[p^y] - \mu_0)\right)^-}{4} & \text{if } \mathbb{E}[p^y] \geq \mu_0 \\ \frac{\mathbb{E}[p^y] - p^{\text{old}} - p^{\text{old}}(\mathbb{E}[p^y] - \mu_0)}{4} & \text{if } \mathbb{E}[p^y] < \mu_0 \end{cases} && \text{(EC.18)} \\
&\leq \frac{\mathbb{E}[p^y] - p^{\text{old}} + p^{\text{old}}|\mathbb{E}[p^y] - \mu_0|}{4} && \text{(EC.19)}
\end{aligned}$$

as desired. For tightness, consider an instance with and types $(q_1, \delta_1) = (0, 1)$, $(q_2, \delta_2) = (0, 0)$, and supply $\Lambda_1 = \Lambda_2 = B$ for some B arbitrary. If $\mu_0 > \mathbb{E}[p^y]$, then $\rho_1 \geq \rho_2$ and plugging directly into Eq. (EC.17) yields tightness. For the case when $\mu_0 \leq \mathbb{E}[p^y]$, Eq. (EC.19) is no longer tight, however the more involved expression in the preceding Eq. (EC.18) is tight both the cases. This completes the proof. □