

# Personalized Algorithms and the Virtue of Learning Things the Hard Way

Omid Rafieian\*

Cornell Tech and Cornell University

Si Zuo\*

Cornell University

Preliminary and incomplete, please do not cite

For Most Updated Version, Click [Here](#)

## Abstract

Recommendation systems are now an integral part of the digital ecosystem. However, the increased dependence of users on recommendation systems has heightened concerns among consumer protection advocates and regulators. Past studies have documented various threats personalization algorithms pose to different aspects of consumer welfare, through violating consumer privacy, unfair allocation of resources, or creating filter bubbles that can lead to increased political polarization. In this work, we bring a consumer learning perspective to this problem and examine whether personalized recommendation systems hinder consumers’ ability to learn their own preferences. We develop a utility framework where consumers learn their preference parameters in the presence of a recommendation system. We introduce a notion of regret which is defined as the regret when consumers make decisions on their own. We theoretically show that the presence of the recommendation system acts as a barrier to consumer learning. We then empirically investigate this phenomenon using the MovieLens data and a fully randomized lab experiment. Finally, we discuss a variety of consumer protection policies that help improve consumer learning and document the welfare implications of each.

**Keywords:** recommendation system, consumer learning, consumer protection, privacy, reinforcement learning

---

\*Please address all correspondence to: or83@cornell.edu and sz549@cornell.edu. The experiment is approved by the Cornell Institutional Review Board (IRB) under approval number IRB0147472.

# 1 Introduction

Recommendation systems are now an integral part of the digital ecosystem. Digital platforms use massive amounts of user-level data to deliver personalized recommendations to their users. One of the canonical examples of recommendation systems is the Netflix movie recommendation algorithm, which reportedly saves the company over one billion dollars annually by reducing the churn rate [Gomez-Uribe and Hunt, 2015]. Other examples include Facebook and Twitter’s news feed personalization, Amazon’s product recommendation, and YouTube’s video recommendation algorithm.

In today’s digital age, the online marketplace is saturated with many options, presenting consumers with the challenge of sifting through too many options to find what they truly want. Personalized recommendation systems have emerged as a solution to this problem with the intent to make consumers’ choices easier by reducing their search costs. These systems are designed to effectively narrow down options in real time and guide consumers towards products or services that best align with their preferences and needs. By doing so, a personalized recommendation system acknowledges the overwhelming nature of online choices and offers a tailored approach, ensuring that consumers can select a fitting item without the need to exhaustively explore the vast digital landscape.

However, as the adoption and reliance on personalized recommendation systems grow, there is increasing scrutiny regarding their potential pitfalls. One pressing concern revolves around privacy. To deliver tailored recommendations, these systems need to track consumer behavior, raising significant concerns among users and leading to stringent regulations such as the General Data Protection Regulation (GDPR) [Johnson, 2022]. Furthermore, the very purpose of personalized algorithms is to differentiate between users, which runs the risk of inadvertent discrimination [Lambrecht and Tucker, 2019]. As such, a large body of work in the recent literature focuses on the fairness implications of personalized recommendation systems to minimize the potential harm of these systems for underrepresented minorities [Barocas et al., 2019]. Another alarming issue is the role of personalized recommendation systems in fostering polarization. Critics argue that these systems create echo chambers by continuously feeding users content that aligns with their existing beliefs or preferences. This can amplify existing biases and deepen divisions, particularly in politically charged environments [Dandekar et al., 2013, Flaxman et al., 2016].

In this work, we bring a consumer learning perspective to this problem and examine how the presence of recommendation systems affects consumer learning. By consistently following the personalized recommendations, users may inadequately explore items, thereby

not learning sufficiently about their own preference parameters. For instance, a consumer with a taste for action movies may rarely explore other types of movies if the recommendation system correctly identifies this taste and only exposes the consumer to movies of this genre. This over-reliance on personalized recommendation systems creates a paradoxical situation. While these systems aim to refine user choices based on perceived preferences, they may also limit users' exposure to diverse options, hindering the organic process of preference learning.

Insufficient learning presents many challenges and potential harms from a consumer protection standpoint. Primarily, it destabilizes consumers' primary decision-making tool — their inherent preferences and beliefs. With a distorted understanding of what they truly enjoy or need, consumers become more vulnerable to manipulation by platforms that have insights into their behaviors and can strategically curate content. Further, miscalibrated beliefs about preferences can lead to misguided decisions, detracting from the overall user experience. Alarming, in situations where a recommendation system becomes unavailable due to system glitches or privacy regulations, consumers who rely heavily on these systems are prone to making more mistakes. Together, it is essential to understand the challenges recommendation systems pose to consumer learning.

In this paper, we study the interplay between personalized recommendation systems and consumer learning and aim to answer the following questions:

1. How can we develop a unified theoretical framework to study consumer learning in the presence of a personalized recommendation system? What are the metrics that we can use to quantify the impact of recommendation systems on consumer learning?
2. Empirically, how does the presence of a personalized recommendation system affect consumer learning? What is the underlying mechanism?
3. What consumer protection policies can we consider? How can we improve the recommendation systems to improve consumer learning?

To answer these questions, we face several challenges. First, we need a theoretical framework that allows for dynamic preference learning through experience. In particular, we want our framework to capture learning by both the user and the recommendation system in a way that the recommendation system will have an information advantage as it uses the data of many other users. Second, we need an empirical framework that allows us to evaluate outcomes under different policies and test our theoretical predictions and assumptions using real data. Specifically, consumer learning is an internal process that is generally not observable by the researcher, so we need to validate the process assumed in the empirical analysis.

To address our first set of challenges, we build a general linear utility framework where consumers have some preference parameters over the space of item features. To model consumers’ decision-making and learning in this domain, we turn to the literature on adaptive learning and specifically use the Thompson Sampling approach as the way users make decisions on their own and update the posterior distribution of their preference parameters. As such, consumers update the posterior distribution of their preference parameters after experiencing an item and realizing the utility of this experience. We then characterize the personalized recommendation system and its decision-making process as a low-rank model where the algorithm has a lower-dimensional representation of items. This low-rank assumption mimics the reality of personalized algorithms used by platforms and parsimoniously accounts for the platform’s information advantage over a single user. Lastly, to quantify the impact of recommendation systems on consumer learning, we use two measures that are commonly used in the literature on Thompson Sampling: (1) Shannon entropy that accounts for the amount of learning by the user, and (2) expected regret, which is the overall welfare loss compared to the first-best optimal choice throughout.

For the second set of challenges, we develop an empirical strategy based on two separate approaches: (1) an empirical analysis of large-scale data of movie ratings based on our theoretical framework, and (2) a lab experiment that allows us to validate our theoretical assumptions and predictions. In the first part, we use the large-scale MovieLens data set commonly used as a benchmark for developing personalized recommendation systems. We perform matrix factorization to obtain ratings for all user-movie pairs and treat them as the ground-truth utility outcomes. We then use these ground-truth utility outcomes to identify consumers’ preference parameters given any set of item covariates in the utility specification. Once we have such consumer-specific preference parameters, we can simulate consumer learning for any sequence of movies consumed and evaluate measures of both consumer learning and expected regret. This allows us to quantify to what extent the presence of recommendation systems acts as a barrier to consumer learning.

In the second part of our empirical framework, we drop all the assumptions on consumer learning and design a lab experiment to test the learning assumptions and the predictions of our theoretical framework. In particular, we designed a 20-period game where participants chose boxes with different attributes in each period and received some reward based on a linear underlying utility. We randomly assigned half of the participants to a treatment condition in which they received box recommendations for the box with the highest utility in the first 10 periods but did not receive any recommendations from periods 11 to 20. Participants

in the control condition did not receive any recommendations throughout. Our experiment allows us to test if users in the control condition learned more in the first 10 periods, where they explored on their own and made better choices from period 11 to 20, compared to the treatment group who relied on the recommendation system and did not sufficiently explore in the first 10 periods.

To illustrate the impact of recommendation systems on consumer learning, we focus on two types of consumers: (1) self-exploring consumers who make decisions on their own as though there is no recommendation system, and (2) recommendation-system-dependent (RS-dependent henceforth) users who follow the recommendations provided by the recommendation system. Both groups start with the same utility specification and priors over the distribution of preference parameters and update their parameters according to their experience. To isolate the welfare impact of relying on the recommendation systems, we introduce a notion of post-shutdown regret, which assumes a point where the recommendation system is no longer available and measures the expected cumulative regret from that point onward. This metric allows us to see how the difference in their own preference learning can manifest itself in consumers’ decision-making.

From a theoretical standpoint, the regret bounds established in Russo and Van Roy [2016] reveal two important facts about our analysis. First, Russo and Van Roy [2016] find that the upper bound for the expected cumulative regret depends on the square root of the dimensionality of the item space. Hence, the extent to which an RS-dependent user achieves a lower expected cumulative regret than the self-exploring user scales with the extent to which the recommendation system can reduce the dimensionality of the item space by using the full matrix of consumers and items, i.e., the rank of the full matrix. Second, Russo and Van Roy [2016] show that the square root of Shannon entropy of the prior distribution of optimal action appears in the upper bound for the expected cumulative regret. We link that finding to the analysis of post-shutdown regret. That is, the extent to which a self-exploring consumer incurs a lower regret than the RS-dependent consumer depends on the entropy of the prior distribution of optimal action at the point of shutdown. Intuitively, we expect the entropy to be higher for the RS-dependent user because these users explored actions less than the self-exploring users. However, the extent to which there is a discrepancy between these two groups is an inherently empirical question.

Next, we take our theoretical framework to data with real underlying consumer preferences. We first estimate the underlying consumer preferences for movies using MovieLens data and set those parameters as ground-truth parameters. We then evaluate different outcomes for

self-exploring and RS-dependent consumers, in terms of Shannon entropy and per-period expected regret. We show that the self-exploring user reduces the Shannon entropy at a higher rate, which implies that the user learns more through exploration. This finding suggests that relying on the recommendation system acts as a barrier to consumer learning. We then define an arbitrary shutdown point for the recommendation system and examine expected regret measures before and after the shutdown of the recommendation system. Our results show that prior to shutdown, the RS-dependent consumer exhibits lower expected regret as the recommendation system learns the preference parameters faster than the user. However, after the recommendation system shutdown, the self-exploring user incurs lower expected regret in decision-making. This finding further suggests that the dependence on the recommendation system limits consumer learning, thereby resulting in worse decisions in the absence of the recommendation system.

An immediate concern that emerges from our findings is the potential susceptibility of RS-dependent consumers to platform or third-party manipulations. That is, if the consumers have great uncertainty about their preference parameters, adversarial players can use this information to manipulate these consumers. This concern gives rise to the potential consumer protection policies to mitigate such issues. To that end, we evaluate a series of viable consumer protection policies in terms of regret before and after the shutdown. In particular, we consider a class of policies where the recommendation system is always available but only available for a proportion of time periods. Notably, we find that there are policies in this class of policies that perform better than the self-exploring user in terms of regret both before and after the recommendation system shutdown. Thus, there exist some policies that perform better than a full ban on consumer tracking.

Lastly, we use a lab experiment to validate the findings from our analysis with the MovieLens data in a controlled environment without the set of assumptions imposed on the structure of consumer learning. We designed a 20-period box-choosing game and recruited participants through Mechanical Turk to participate in our experiment. The results from our experiment confirm both our assumptions and predictions. We find strong empirical evidence for consumer learning. The rate at which consumers choose the right answer increases over time. More specifically, we show that participants in the condition where the recommendation system was present for the first 10 periods did significantly worse in the absence of the recommendation system than those in the control condition who did not access the recommendation system. Further, participants in the control condition (self-exploring) show more precise learning compared to participants in the treatment condition when specifically asked to estimate

the linear utility parameters. Together, our experimental findings provide complementary proof-of-concept supporting our main hypothesis that recommendation systems act as a barrier to consumer learning, even when we do not make specific modeling assumptions about the process of consumer learning.

In summary, our paper provides several contributions to the literature. Substantively, we present a comprehensive study of the effect of personalized recommendation systems on consumer learning through a series of theoretical analyses, counterfactual simulations, and experiments. We document that even when RS enhances consumer welfare by increasing the match value between consumers and recommended products, it can negatively affect consumers’ learning by limiting the degree to which they explore their own preferences. While concerns related to privacy, fairness, and polarization are more thoroughly studied in the past literature on personalized recommendation systems, the impact of these systems on consumer learning has been overlooked. The impact of RS on consumer learning is particularly important as learning is the primary tool consumers have for independent decision-making. As such, our work extends the policy debate on the societal impact of personalized recommendation systems by bringing a consumer learning perspective, which helps us develop more fundamental policies to empower consumers. Methodologically, we build a framework that allows us to quantify the potential welfare loss due to underexploration in the context of personalized recommendation systems. Our framework is general and can be applied to a variety of domains that involve sequential decision-making.

## 2 Related Literature

First, our paper relates to the literature on personalization. Prior methodological work in this domain has offered a variety of methods to generate personalized policies, such as low-rank matrix factorization models for collaborative filtering [Linden et al., 2003, Mazumder et al., 2010, Koren et al., 2021], models to estimate Conditional Average Treatment Effects [Athey and Imbens, 2016, Shalit et al., 2017, Wager and Athey, 2018, Nie and Wager, 2021], and personalized policy learning methods [Swaminathan and Joachims, 2015]. Applied work in this domain has focused on two themes of (1) empirical gains from personalization in a variety of domains [Ascarza, 2018, Simester et al., 2020a,b, Rafeian and Yoganarasimhan, 2021, Yoganarasimhan et al., 2022, Rafeian, 2023, Dubé and Misra, 2023], and (2) the implications personalization has for consumer welfare in terms of privacy [Goldfarb and Tucker, 2011a,b, Tucker, 2014, Johnson et al., 2020], fairness [Sweeney, 2013, Lambrecht and Tucker, 2019], and political polarization [Bakshy et al., 2015, Hosseinmardi et al., 2021]. Our work adds to this stream by bringing a consumer learning view to this problem, which has been largely

ignored in the prior work on personalized algorithms. We demonstrate how the negative impact of personalized algorithms on consumer learning can result in poor decision-making by consumers in the absence of algorithms.

Second, our paper relates to the literature on consumer learning. Understanding consumer learning dynamics has been of great interest to researchers in marketing [Roberts and Urban, 1988]. Ever since the seminal paper by Erdem and Keane [1996] who modeled forward-looking consumers who make decisions under uncertainty and engage in an exploration-exploitation trade-off, numerous studies have focused on choice contexts where dynamic learning plays an important role [Akerberg, 2003, Crawford and Shum, 2005, Erdem et al., 2005, Hitsch, 2006, Erdem et al., 2008]. An important issue in this stream of work is computational complexity, which has made its application infeasible in more high-dimensional domains [Ching et al., 2013]. More recently, Lin et al. [2015] have shown that using heuristic-based index strategies for learning yield similar performance while having the advantage of computational and cognitive simplicity. We extend this stream of literature by offering a Thompson Sampling approach for determining consumer choice and belief updating process. We further demonstrate how the increased flexibility offered by the Thompson Sampling approach can help researchers study settings with high-dimensional learning.

Third, our work relates to the vast literature on adaptive learning and multi-armed bandits. Prior research in this domain has offered a variety of algorithms to use [Lattimore and Szepesvári, 2020]. Although Thompson sampling has been around since the work by Thompson [1933], it has only recently gained traction after providing a remarkable empirical performance better than state-of-the-art benchmarks [Chapelle and Li, 2011]. Since then, many researchers have attempted to provide a variety of theoretical guarantees on Thompson sampling for a variety of adaptive learning problems [Agrawal and Goyal, 2012, 2013, Russo and Van Roy, 2014, 2016]. For a comprehensive review of Thompson sampling, please see Russo et al. [2018]. Most of the literature in this domain focuses on a single learner that optimizes the action and updates parameters upon experience. Our work extends this single-agent framework to a setting with both a learning recommendation system and an agent, offering new insights for modeling general principal-agent problems in contexts with decision-making under uncertainty.



### 3 Theoretical Framework

#### 3.1 Motivation

The main task for a personalized recommendation system is to learn individual-level user preferences and provide each user with effective recommendations. In the current digital landscape with numerous items available, personalized recommendation systems offer great convenience for users in decision-making. However, an important concern about recommendation systems is the low diversity of recommended products [Lee and Hosanagar, 2019, Chen et al., 2023]. In other words, these recommendation systems over-exploit certain strategies that work well and under-explore the rest of the item space. Thus, consistently following the algorithmic recommendations can have important implications for users, particularly in term of learning their own preference parameters.

Understanding the impact of personalized recommendation systems on consumer learning is important from a consumer protection standpoint because it provides insights into the primacy tool digital consumers can use for decision-making. In particular, if consumers lack a comprehensive understanding of their preferences, they would make worse decisions in the absence of recommendation systems (e.g., due to privacy regulations). This results in a feedback loop wherein consumers overly rely on recommendation systems and learn less about their own preferences. More importantly, consumers without sufficient learning of their preferences are more susceptible to digital manipulations by adversarial players. An adversarial player or platform can take advantage of consumers’ insufficient learning and exploit them in a variety of welfare-reducing ways. Together, we view consumer learning as an important yet understudied component of consumer welfare that allows us to design better policies around personalized algorithms

In this section, we aim to develop a model of consumer learning, where a consumer learns their own preference parameters by interacting with the system. Our goal is to understand how the presence of a personalized recommendation system interferes with consumer learning. To illustrate the impact of the recommendation system, we focus on two types of users: (1) self-exploring users who ignores the recommendation system and makes their own decisions, and (2) RS-dependent users who follow the algorithmic recommendations. Both groups learn their preferences through experience. We hypothesize that RS-dependent users will be exposed to a less diverse set of items, which, in turn, hinder their preference learning.

### 3.2 Illustrative Example

Suppose there is a user who is watching one movie per period. For simplicity, we assume there are two movie attributes: (a) action intensity  $x_1$ , and (2) emotional depth  $x_2$ . We further assume that there are five types of movies with the following attributes:

Movie Type	$x_1$ (Action Intensity)	$x_2$ (Emotional Depth)
1	10	0
2	7	3
3	5	5
4	3	7
5	0	10

Table 1: Attributes of the five movies in the simple example.

As shown in Table 1, Movie Type 1 has a high level of action intensity but lacks emotional depth, whereas Movie Type 5 has no action intensity but a high level of emotional depth. The user’s utility of watching a movie with attributes  $x_t$  at time  $t$  is as below:

$$u(x_t) = \theta_1 x_{1t} + \theta_2 x_{2t} + \epsilon_t,$$

where  $\epsilon_t$  follows the standard normal distribution,  $x_{1t}$  and  $x_{2t}$  are the two features of the movie that the user chooses at period  $t$ , and  $\theta_1$  and  $\theta_2$  are the preference parameters of the user.

In our example, we assume  $\theta_1 = 0.6$  and  $\theta_2 = 0.4$ . We want to illustrate how users learn their preferences through their movie-watching experience. For this purpose, we different kinds of movie-watching sequences. We consider scenarios wherein the user only watches a subset of all five movie types. Let  $\mathcal{J}$  denote all subsets of the set of all five movies. For any subset  $\mathcal{S} \in \mathcal{J}$ , we simulate learning for consumers as follows:

1. Nature randomly selects the set  $\mathcal{S}$ .
2. At the beginning of period  $t$ , the user randomly selects one movie from the set  $\mathcal{S}$  to watch.
3. After watching each movie, the user observes the realized utility  $u_{jt}$  and updates her beliefs on  $\theta_1$  and  $\theta_2$  using Ordinary Least Squares (OLS).

We continue the process for ten periods for each user and measure how accurately users can estimate their own preference parameters  $\theta_1$  and  $\theta_2$ . The size of subset  $\mathcal{S}$  controls the amount of exploration by the user, so we present the results for different sizes of the subset  $\mathcal{S}$  from which users select movies to watch in Table 2. As shown in this table, the larger the size of the set is, the lower the overall Root Mean Squared Error (RMSE) for preference parameters, which implies a more calibrated preference learning by users. The results from this table highlight the idea that diversifying choices or reducing over-exploitation can enhance the user’s learning about their own preferences. As personalized recommendation systems are often characterized as diversity-reducing algorithms [Lee and Hosanagar, 2019], the findings from the simple example above suggests that these systems likely act as a barrier to learning. In next sections, we formalize this intuition and present some theoretical analysis.

Number of Movies in Set $\mathcal{S}$	Combined RMSE	$\theta_1$ RMSE	$\theta_2$ RMSE
2	0.164203	0.114636	0.117564
3	0.113749	0.079482	0.081372
4	0.097327	0.068326	0.069312
5	0.090282	0.063523	0.064153

Table 2: RMSE for 1000 simulations,  $\theta_1 = 0.6, \theta_2 = 0.4$

### 3.3 Model Setup

We consider a user who sequentially chooses actions from an action set  $\mathcal{A}$ . Each action is characterized by a  $d$ -dimensional set of attributes, i.e.,  $\mathcal{A} \subset \mathbb{R}$ . For example, an action can be a movie with  $d$  attributes (e.g., emotional depth of a movie). Let  $a_t \in \mathcal{A}$  denote the action chosen by the user in period  $t$ . After experiencing the product, the user receives a utility that is linear in the attributes of the action as follows:

$$u_{t,a_t} = a_t^T \theta^* + \epsilon_t, \quad (1)$$

where  $\theta^* \in \mathbb{R}^d$  is the vector of true preference parameters by the user, and  $\epsilon_t$  denotes the error term that comes from a Normal distribution with known variance, which implies that  $E[u_{t,a_t}] = a_t^T \theta^*$ . We use the simple and general utility framework in Equation (1) to study how consumers choose actions and update their parameters both in the absence and presence of the recommendation system in the following sections.

### 3.3.1 Consumer Choice and Learning without Recommendation System

Given the utility framework in Equation (1), we want to understand the user’s decision-making process. A forward-looking utility-maximizing user wants to optimize the overall utility over  $T$  periods. This naturally motivates users to learn their preference parameters through experience and balance good decision-making with proper exploration of their own preference parameters. A common way to represent the objective function of the forward-looking utility-maximizing user is to use the notion of expected cumulative regret that is developed in the machine learning theory literature, which is defined as follows:

$$\min R_{a_1, a_2, \dots, a_T}^T = \sum_{t=1}^T \mathbb{E}_t (u_{t, a_t^*} - u_{t, a_t}), \quad (2)$$

where  $a_t$ ’s are the actions chosen by the user and each element in the summation above captures the expected regret that is defined as the expected difference between the first-best optimal action at a given time period ( $a_t^*$ ) and the action chosen by the user ( $a_t$ ). Typical approaches to find the optimal sequence of choices by users involve solving a dynamic programming problem, which is known as an NP-hard problem in the literature. The lack of cognitive simplicity of dynamic programming solutions has motivated researchers to study the simpler heuristic-based strategies as the underlying learning process [Lin et al., 2015]. We draw inspiration from this stream of literature and assume that consumers employ a Thompson Sampling approach that is a simple and intuitive heuristic-based strategy with excellent empirical guarantees [Chapelle and Li, 2011].

Thompson Sampling aims to find the right balance in the exploration-exploitation trade-off in decision-making scenarios. It operates by initializing the user’s prior belief distribution about the preference weights. It then draws from this distribution and computes the utility for all items using these draws of the preference weights. Once the action is chosen and the utility of this action is realized by the user, the user will update the distribution of preference weights according to the observed outcome. The details for the process are provided in Algorithm 1. The user starts with a normal prior on  $\theta^*$  and then samples a draw to form her beliefs  $\hat{\theta}_t$ . Based on her beliefs, she chooses the action (movie) that maximizes her current utility, and then she updates her belief based on the history of observed data  $\mathcal{F}_t$ , where  $\mathcal{F}_t = (a_\tau, u_\tau)_{\tau=1,2,\dots,t}$

This approach of assuming that the self-exploring user uses Thompson Sampling has several key advantages. First, it is a commonly used heuristic strategy for this dynamic problem, and early literature has shown it to be nearly optimal [Chapelle and Li, 2011].

---

**Algorithm 1** Linear-Gaussian Thompson Sampling Choice and Updating Process for Self-Exploring User

---

**1. Sample Distribution**

$$\hat{\theta}_t \sim N(\mu_{t-1}, \Sigma_{t-1})$$

**2. Select Action**

$$a_t \in \arg \max_{a \in \mathcal{A}} \langle a, \hat{\theta}_t \rangle$$

**3. Update Beliefs**

$$\mu_t \leftarrow \mathbb{E}[\theta^* | \mathcal{F}_t]$$

$$\Sigma_t \leftarrow \mathbb{E}[(\theta^* - \mu_t)(\theta^* - \mu_t)^T | \mathcal{F}_t]$$

**4. Increment  $t$  and Go to Step 1**

---

Second, it is computationally light, making it advantageous our later empirical analysis using the MovieLens data set.<sup>1</sup> Third, due to its simplicity, it is easy incorporate it in cases where there is a recommendation system present in the problem. We discuss this issue in the following section.

### 3.3.2 Consumer Choice and Learning with Recommendation System

We now introduce a personalized recommendation system that aims to simplify the user's decision-making problem. Since we want to quantify the impact of recommendation systems on consumer learning, we assume that the recommendation system's objective is the same as that of user.<sup>2</sup> A natural difference between the recommendation system and a single user is the fact that the system has access to the data of all users. An intuitive way to mathematically characterize the difference between the recommendation system and a single user is to note that the recommendation system has access to the matrix  $Y_{N \times J}$ , where  $N$  represents the number of all users and  $J$  represents the number of all items, but the user access the data only for their corresponding row in this matrix. Hence, the recommendation system has data advantage over user  $i$  as it can pool the data of all users to estimate underlying preferences for

---

<sup>1</sup>As a robustness check, we show that our qualitative insights will not change if we use an approximate dynamic programming solution to this problem.

<sup>2</sup>It is worth emphasizing that the only reason we make this assumption is to ensure that the impact of the recommendation system is not driven by the misalignment in objectives. It is generally easy to show that in cases where the objectives are misaligned, the extent of harm by the recommendation system will be larger [Kleinberg et al., 2022]. In that sense, our results will provide a lower bound for the welfare loss due to recommendation system.

user  $i$ . To reflect the data advantage of the recommendation system, we make the following assumption:

**Assumption 1.** *The recommendation system has access to an  $r$ -dimensional representation of the action space that is sufficient for utility estimation, such that  $r \leq d$ .*

The existence of such  $r$ -dimensional representation is essentially the same as the low-rank assumption that is commonly made in the recommendation systems literature. Under the low rank assumption, we can decompose the matrix  $Y_{N \times J}$  into two matrices  $U_{N \times R}$  and  $V_{J \times R}$ , where  $Y \approx UV^T$ . As such, one could treat matrix  $V$  as an  $r$ -dimensional representation of the action space. It is easy to verify that  $r \leq d$  because if the matrix of outcomes is generated by a  $d$ -dimensional utility model as in Equation (1), the upper bound for the rank of matrix  $Y_{N \times J}$  is  $d$  by construction. However, in most settings, since both users and actions exhibit high degrees of similarity, it is easy to find a substantially lower-dimensional representation of the action space by pooling all the data and using the large matrix  $Y$ , such that  $r \ll d$ .

Let  $g : \mathbb{R}^d \rightarrow \mathbb{R}^r$  denote the mapping that maps the  $d$ -dimensional action from our action space to the  $r$ -dimensional representation. The utility from choosing this action is presented as follows:

$$u_{t,a_t} = f(a_t)^T \gamma^* + \epsilon_t, \quad (3)$$

where  $\gamma^*$  represents the true utility parameters at the  $r$ -dimensional space. Algorithm 2 shows the Linear-Gaussian Thompson Sampling process for the recommendation system that operates at an  $r$ -dimensional action space and need to learn preference parameters  $\gamma^*$ . The basic structure is the same as Algorithm 1, and the main differences are the lower-dimensional representation and the updating process.

Practically, one could think of Algorithm 2 as a continuous matrix factorization as the recommendation system receives more information about user  $i$ . In our empirical analysis with the MovieLens data set, we specifically use this approach to highlight the differences in the learning process by the recommendation system and a single user.

When the recommendation system is present, we assume that the user follows the recommended item every time period. It is easy to rationalize this choice in a variety of ways due to the search cost associated with exploration and the fact that the recommendation system learns faster than the user because of its information advantage. However, we abstract away from this possibility and simply assume that the user always follow the recommendation system to illustrate the differences between the self-exploring and RS-dependent user. Algorithm 3 present the choice and updating process for the RS-dependent user. Since the RS-dependent

---

**Algorithm 2** Linear-Gaussian Thompson Sampling for Fast-Updating Recommendation System with Data Advantage

---

**1. Sample Distribution for User  $i$ :**

$$\hat{\gamma}_{ti} \sim N(\mu_{t-1,i}^\gamma, \Sigma_{t-1,i}^\gamma)$$

**2. Select Action for User  $i$ :**

$$a_{ti} \in \arg \max_{a \in \mathcal{A}} \langle a, \hat{\gamma}_{ti} \rangle$$

**3. Update Beliefs for User  $i$ :**

$$\mu_{t,i}^\gamma \leftarrow \mathbb{E}[\gamma^* | \mathcal{F}_{t,i}, \mathcal{F}_{t,j_1}, \dots, \mathcal{F}_{t,j_N}]$$

$$\Sigma_{t,i}^\gamma \leftarrow \mathbb{E}[(\gamma^* - \mu_{t,i}^\gamma)(\gamma^* - \mu_{t,i}^\gamma)^T | \mathcal{F}_{t,i}, \mathcal{F}_{t,j_1}, \dots, \mathcal{F}_{t,j_N}]$$

**4. Increment  $t$  and Return to Step 2**


---

user follows a different movie-watching path from the self-exploring user, the belief updating process evolves differently for the RS-dependent. It is important to notice that although the recommendation system operates at a lower-dimensional item space, the user still updates in the  $d$ -dimensional space.<sup>3</sup>

---

**Algorithm 3** Linear-Gaussian Thompson Sampling Choosing and Updating Process for RS-dependent User

---

**1. Follow RS's Suggestion**

$$a_t = a_{RS}$$

**2. Update Beliefs**

$$\mu_t \leftarrow \mathbb{E}[\theta^* | u_1, u_2, \dots, u_t]$$

$$\Sigma_t \leftarrow \mathbb{E}[(\theta^* - \mu_t)(\theta^* - \mu_t)^T | \mathcal{F}_{t,i}]$$

**3. Increment  $t$  and Go to Step 1**


---



---

<sup>3</sup>An implicit assumption we make here is that the RS-dependent user cannot learn from the recommendations beyond their own experience. This assumption is reasonable as recommendation systems are often very complex and it is not realistic to assume that users can learn further by observing that a product is recommended. We later empirically test this possibility in a lab experiment and demonstrate the validity of this assumption.

### 3.4 Welfare Analysis

We now conduct a welfare analysis of the two types of users we defined earlier: (1) Self-exploring users who make decisions on their own following Algorithm 1, and (2) RS-dependent users who follow the recommendations provided through Algorithm 2 and update their preference parameters following Algorithm 3. To perform this analysis, we introduce a shutdown event whereby the recommendation system becomes unavailable at a certain time period  $\tau$ . Comparing the welfare loss of the two groups after shutdown is illuminating as it shows which group can make better decisions in the absence of a recommendation system. Motivated by this shutdown event, we introduce a new notion of expected cumulative regret that is defined similar to Equation (2) from the point of recommendation system shutdown as follows:

$$R_{\text{PSD}}^{\tau, T} = \sum_{t=\tau}^T \mathbb{E}_t (u_{t, a_t^*} - u_{t, a_t}), \quad (4)$$

where  $R_{\text{PSD}}^{\tau, T}$  is the regret post shutdown (PSD). Our goal is to compare the expected cumulative regret  $R^T$  and  $R_{\text{PSD}}^{\tau, T}$  for both self-exploring and RS-dependent groups. Since a key difference between self-exploring and RS-dependent users is in the experiential paths and belief updating processes, we need regret bounds that depend on the characteristics of the prior distribution used by the user. To that end, we heavily borrow from the seminal paper by Russo and Van Roy [2016] who find the information-theoretic regret bounds for the Thompson Sampling algorithm.

We first state the main finding in Russo and Van Roy [2016] and then discuss how it relates to our setting. In their work, Russo and Van Roy [2016] demonstrates that the expected cumulative regret of Thompson sampling up to time  $T$  is bounded by

$$\sqrt{\frac{H(a^*)dT}{2}}, \quad (5)$$

where  $H(a^*)$  is the entropy of the prior distribution of the optimal action  $a^*$ , and  $d$  is the dimension of the true parameter  $\theta^*$ .

In the absence of a recommendation system shutdown, the regret bound of Equation (5) allows us to directly compare the performance of self-exploring and RS-dependent users. In particular, the regret bounds for these two groups differ in the dimensionality of the problem. The self-exploring user has a regret bound of  $\sqrt{H(a^*)dT/2}$ , whereas the RS-dependent user has a regret bound of  $\sqrt{H(a^*)rT/2}$ . As such, the difference between the overall regret bounds boil down to the difference between  $d$  and  $r$ . The following remark summarizes this point as



follows:

**Remark 1.** *To the extent that the recommendation system reduces the dimensionality of the item space, the RS-dependent user has a lower expected cumulative regret than the self-exploring user in the absence of recommendation system shutdown.*

Now, if there is a recommendation system shutdown at time period  $\tau$ , we have two types of users with the same dimensionality of the item space who will be making decisions on their own from time period  $\tau$  onward. However, these two user types will differ in their entropy of the prior distribution of the optimal action  $a^*$  at time period  $\tau$ . Let  $a_{SE,\tau}^*$  and  $a_{TS,\tau}^*$  denote the posterior distribution of optimal action at the start of period  $\tau$ . The regret bound for the self-exploring user will be  $\sqrt{H(a_{SE,\tau}^*)d(T-\tau)/2}$ , whereas the regret bound for the RS-dependent user will be  $\sqrt{H(a_{TS,\tau}^*)d(T-\tau)/2}$ . As such, the difference between the two depends on the difference in the Shannon entropy of the prior distribution of optimal action at point  $\tau$ . Intuitively, since a self-exploring user is likely to explore more than the RS-dependent user, we expect the entropy of the prior distribution of optimal action at point  $\tau$  to be lower for the self-exploring user compared to the RS-dependent user. This results in the following remark:

**Remark 2.** *To the extent that the self-exploring user has lower Shannon entropy of the prior distribution of optimal action at point  $\tau$ , the self-exploring user has a lower expected cumulative regret post shutdown than the RS-dependent user.*

Although we intuitively expect a lower entropy for self-exploring user at the point of shutdown, the extent of it is an empirical question. This motivates our empirical analysis with the illustrative example below and the MovieLens data in §4.

### 3.5 Synthetic Experiment with the Illustrative Example

Continuing from the illustrative example in §3.2, we aim to investigate whether recommendation system leads to increased exploitation. We hypothesize that this could result in insufficient learning for users, which leads to a higher regret once the recommendation system is shut down. To delve deeper, we outline the following timeline:

1. We consider three users: (1) self-exploring user, (2) RS-dependent user (with fast updating), and (3) RS-dependent user with perfect information. All three users are identical in having the same utility function represented as  $u(x_t) = \theta_1^*x_{1t} + \theta_2^*x_{2t} + \epsilon_t$ . We introduce the RS-dependent user with perfect information as a benchmark, as it

provides the first-best recommendation system. In this scenario, the RS possesses flawless information about the user’s preferences, meaning it’s fully aware of  $\theta_1^*$  and  $\theta_2^*$ .

2. In every period, each user is presented with the same subset of movies, denoted as  $\mathcal{S}$ . This subset contains two movies randomly selected from the whole set  $\mathcal{J}$ , which contains all 5 movies.
3. All three users choose one movie to watch during each period and subsequently update their beliefs.

First, we examine the exploration paths of the three types of users. Intuitively, the RS-dependent user (Perfect Information) should exhibit more exploitation (repeated item suggestions) than the RS-dependent user (Fast Updating). Conversely, the self-exploring user should show the most exploration. We set  $\theta_1^* = 0.6$  and  $\theta_2^* = 0.4$ , implying that the expected utilities for watching all movies are ranked as: Movie Type 1 > Movie Type 2 > Movie Type 3 > Movie Type 4 > Movie Type 5. As illustrated in Table 3, RS-dependent users focus more heavily on movies types 1 and 2. On the other hand, the self-exploring user distributes their attention across all five movies more evenly. This pattern aligns with our hypothesis that RS tends to offer repeated suggestions.

User Type	Movie 1 Count	Movie 2 Count	Movie 3 Count	Movie 4 Count	Movie 5 Count
self-exploring user	2929	2531	1959	1514	1067
RS-dependent user (perfect information)	3993	3049	1983	975	0
RS-dependent user (fast updating)	3443	2806	1954	1261	536

Table 3: Movie Counts for Self Exploring and RS Dependent Users, 1000 simulations,  $\theta_1^* = 0.6, \theta_2^* = 0.4$

Related to this, we aim to determine whether over-exploitation truly leads to insufficient learning. Table 4 reveals that the self-exploring user possesses the lowest RMSE compared to the RS-dependent users in the final period. Transitioning to another measure of learning, the Shannon entropy, Figure 1 indicates that the self-exploring user exhibits the lowest entropy. By synthesizing the data from both the table and figure, it becomes evident that the self-exploring user achieves superior learning and exhibits lower entropy, a result of their broader exploration.

User Type	Combined RMSE	$\beta_1$ RMSE	$\beta_2$ RMSE	Avg. Regret
self-exploring user	0.1005	0.0619	0.0792	0.2554
RS-dependent user (perfect information)	0.1237	0.0513	0.1126	0.0000
RS-dependent user (fast updating)	0.1089	0.0517	0.0958	0.1298

Table 4: RMSE and Regret for Self Exploring and RS Dependent Users

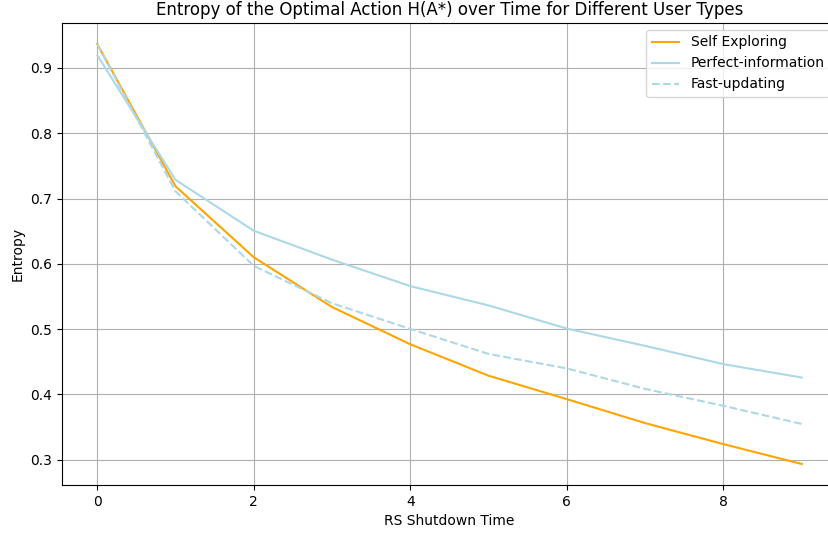


Figure 1: Shannon Entropy for Self Exploring and RS Dependent Users

Now, we present evidence supporting our two remarks:

1. The RS-dependent user has a lower expected cumulative regret than the self-exploring user when there is no recommendation system shutdown.
2. Given the self-exploring user exhibits a lower Shannon entropy for the prior distribution of the optimal action at point  $\tau$ , this user subsequently has a lower expected cumulative regret post-shutdown than the RS-dependent user.

Assuming the RS is scheduled for a shutdown from period 5, Figure 2 portrays the average expected regret both before and after the RS shutdown for the three user types. Notably, the self-exploring user incurs the highest regret prior to the shutdown (as indicated in Remark 1), but experiences reduced regrets post-shutdown (as suggested by Remark 2).

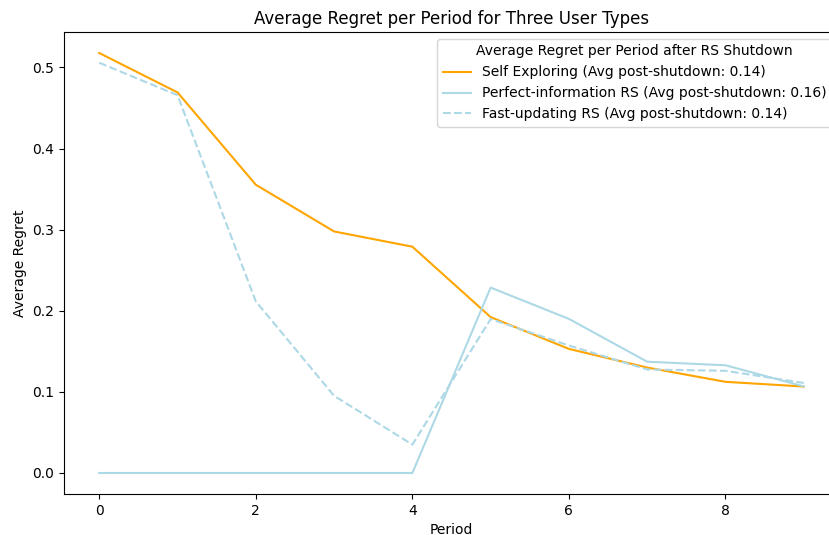


Figure 2: Average Expected Regret Before and After RS Shutdown

While our elementary example supports our intuition, recommendation systems in real-world applications can be more complex, often influenced by a myriad of factors. Our simplistic model, which is based on a two-dimensional framework encompassing only 5 movies, only scratches the surface of the actual complexity involved in large-scale platforms. These platforms cater to thousands of users, each with distinct preferences and viewing histories, and deal with vast arrays of ratings that influence the recommendation system algorithms. Recognizing this leap in complexity, in Section 4, we examine our theoretical predictions using the MovieLens data set that contains real consumer preference data. This data set presents a more realistic scenario, where RS does not just rely on straightforward metrics but employ advanced matrix decomposition techniques, capturing the nuances of user preferences and interactions to provide movie recommendations.

## 4 Empirical Analysis: Case of MovieLens Data

In this section, we perform our empirical analysis using the MovieLens 1M data set.<sup>4</sup> This data set serves as a benchmark for personalized recommendation systems and is frequently used by researchers and practitioners in this field. There are over one million ratings that correspond to a total of 6,040 distinct users and 3706 unique movies, allowing researchers to form large-scale matrices needed for matrix factorization tasks. In addition to ratings,

<sup>4</sup>This dataset is publicly accessible and can be obtained from <https://grouplens.org/datasets/movielens/1m/>

the data provide us with some information about the movies, such as the movie genre and themes as well as a large array of tags associated with each movie. At the user level, we observe demographic characteristics of users such as gender, age, occupation, and zip code.

#### 4.1 Empirical Strategy

Building on our earlier synthetic experiment, we aim to validate the two remarks from our theoretical session. Specifically, we want to examine if the RS-dependent user indeed experiences a lower regret prior to the RS shutdown and a higher regret post-shutdown in a setting where underlying preferences come from true consumer preferences. A significant challenge arises in understanding how both users and the RS make choices and adapt based on the sophisticated matrix decomposition technique. In addition, we need a specific learning context where neither the user nor the RS knows about the user’s preferences. Therefore, we consider the cold-start problem setting, where there is a new user  $i$  with no historical rating information. Consistent with our analysis in §3.3, we define the following types of users:

- **Self-exploring user:** Self-exploring user  $i$  chooses one action (movie)  $a_t$  to watch from action set  $\mathcal{A}$ . Each movie is characterized by a  $d$ -dimensional set of attributes (movie features). Similar to Equation 1, self-exploring user  $i$ ’s utility (rating)<sup>5</sup> from watching movie  $a_t$  follows:

$$y_{i,t,a_t} = a_t^T \theta_i^* + \epsilon_{it}, \quad (6)$$

where  $y_{i,t,a_t}$  denotes user  $i$ ’s rating on movie  $a_t$  on period  $t$ . The top 10 mostly used tags in MovieLens data are: *sci-fi*, *atmospheric*, *action*, *comedy*, *surreal*, *based on a book*, *twist ending*, *funny*, *visually appealing*, and *dystopia*. For the remaining empirical exercise, we assume the self-exploring user uses the top 100 most frequently used movie tags so  $d = 100$ . The user use Thompson sampling to choose movies ( $a_t$ ) and update her beliefs  $\hat{\theta}$  following Algorithm 1. For our simulation practice, we need to estimate the ground-truth vector of preference weights from the data and then use them as model primitives to generate the data under different movie-watching sequences.

- **RS-dependent user:** RS-dependent user always follows the recommendation provided by the RS. Upon consuming the item, the user updates her beliefs on the  $d$ -dimensional space similar to the self-exploring. As mentioned in Assumption 1, RS has access to an  $r$ -dimensional representation of the action space. To empirically operationalize this insight, we first illustrate the fundamental principles behind this low-rank methodology.

---

<sup>5</sup>Directly observing user’s realized utility from watching a movie is challenging. Based on the Movie Lens data, we believe rating is a reasonable approximation of user’s satisfaction after watching a movie.

In the realm of recommendation systems, a common approach involves decomposing a ratings matrix  $Y_{N \times J}$  ( $N$  users and  $J$  movies) into user and item factors:

$$Y_{N \times J} \approx U_{N \times R} V_{J \times R}^T + \mu, \quad (7)$$

where  $Y_{N \times J}$  is the matrix of ratings, with users represented by rows and items by columns. Each element denotes a user's rating for a specific item,  $U_{N \times R}$  is the matrix of user factors, reflecting latent attributes for each user,  $V_{J \times R}$  signifies the matrix of item factors, and  $\mu$  denotes the mean rating across all items. Graphically, the matrices can be illustrated as:

$$U = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix}, \quad V = \begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{bmatrix}, \quad \mu = \begin{bmatrix} \mu & \mu & \mu \\ \mu & \mu & \mu \\ \mu & \mu & \mu \end{bmatrix}.$$

To reconstruct matrix  $Y$ , one can utilize the user and item factors. For illustration, the entry at the top-left of matrix  $Y$  can be computed by:

$$u_{11}v_{11} + u_{12}v_{21} + u_{13}v_{31} + \mu$$

By using a similar methodology, the entire matrix  $Y$  can be reconstituted, where each entry is a result of multiplying the corresponding row of  $U$  with the respective column of  $V$ , followed by the addition of  $\mu$ . When it comes to the cold-start problem, RS will use the Sequential Matrix Completion (SMC) technique to generate new  $U$  and  $V$  matrix as the new user  $i$  watches one more movie. The evolution of  $Y_{(N+1) \times J}$ ,  $U_{(N+1) \times R}$  and  $V_{J \times R}$  matrices over time  $t = 0, \dots, T$  is as below:

$$\begin{array}{ccc} Y^{(t=0)} & Y^{(t=1)} & \dots & Y^{(t=T)} \\ \begin{bmatrix} y_{11} & y_{12} & \dots & y_{1J} \\ y_{21} & y_{22} & \dots & y_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ y_{N1} & y_{N2} & \dots & y_{NJ} \\ \textcolor{red}{0} & \textcolor{red}{0} & \dots & \textcolor{red}{0} \end{bmatrix} & \begin{bmatrix} y_{11} & y_{12} & \dots & y_{1J} \\ y_{21} & y_{22} & \dots & y_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ y_{N1} & y_{N2} & \dots & y_{NJ} \\ \textcolor{red}{y_{i1}} & \textcolor{red}{0} & \dots & \textcolor{red}{0} \end{bmatrix} & \dots & \begin{bmatrix} y_{11} & y_{12} & \dots & y_{1J} \\ y_{21} & y_{22} & \dots & y_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ y_{N1} & y_{N2} & \dots & y_{NJ} \\ \textcolor{red}{y_{i1}} & \textcolor{red}{y_{i2}} & \dots & \textcolor{red}{y_{(iJ)}} \end{bmatrix} \\ U_{(N+1) \times R, t=0} V_{J \times R, t=0}^T & U_{(N+1) \times R, t=1} V_{J \times R, t=1}^T & \dots & U_{(N+1) \times R, t=T} V_{J \times R, t=T}^T \end{array}$$

For the new user  $i$ , RS decomposes the row vector  $y_i$  into the product of user  $i$ 's preference

row vector  $u_i$  and item matrix  $V_{J \times R}$ . Mapping to Assumption 1, we could think the  $\gamma^* = u_i \in R^r$  and the  $r \leq d$ .

For the RS-dependent user, she chooses and updates her beliefs of  $\theta^* \in R^d$  following Algorithm 3. It is important to note that even though RS operates in  $r$  dimensions, the user could only observe the tag information and therefore updates her belief in the  $d$ -dimensional space.

Together, we need to obtain a personalized  $d$ -dimensional set of preference weights from our data, such that  $\theta_i^*$  and  $\theta_j^*$  can differ. We then need to re-generate the data using these primitives and simulate how a self-exploring and RS-dependent user would behave.

## 4.2 Data-Driven Simulation Exercise

In our data-driven simulation exercise, we need to first obtain a ground-truth set of individual-level preference weights that are informed by real data and then re-generate the data for each user and measure their utility under counterfactual paths, not the ones observed in the data.

To obtain the ground truth data, our approach can be summarized as follows:

- We complete the matrix of rating  $Y_{N \times J}$  using the matrix completion approach. The outcome of this approach is matrix  $\tilde{Y}_{N \times J}$ , where all entries have some values according to the low-rank decomposition approach.
- Let  $A_{J \times d}$  denote the matrix of movie tags in the  $d$ -dimensional space. For each user  $i$ , we use the ratings from the complete matrix  $\tilde{Y}$  for that user and regress it on the tags  $A_{J \times d}$  to obtain the individual-level preference weights  $\theta_i^* \in \mathbb{R}^d$ . Performing this for all users gives us the matrix  $\Theta_{N \times d}^*$ .
- We use the set of individual-level preference weights to regenerate the data of movies and ratings, such that  $Y_{N \times J}^* = \Theta_{N \times d}^* A_{J \times d}^T$ .

Once we generate the ground truth matrix of ratings, we split our data into two parts: training and test. For test users, we remove all ratings to create a cold-start setting. For training users, we use the same missingness pattern as our data, but we use the regenerated ratings. Let  $Y^{\text{train}}$  and  $Y^{\text{test}}$  denote these two matrices. For each user in the test data, we can now simulate the movie-watching sequence in both self-exploring and RS-dependent cases as follows:

- *Self-exploring user:* For self-exploring user  $i$  in the test data, the user starts with a prior about their own preference weights  $\Theta_i^{(0)} \in \mathbb{R}^d$  and chooses a random movie in the beginning. The true utility is realized based on the  $Y_{N \times J}^*$  matrix. In every time period  $t$ , the user updates the  $d$ -dimensional set of preference weights based on the realized utility

from  $\Theta_i^{(t-1)}$  to  $\Theta_i^{(t)}$ .

- *RS-dependent user:* For RS-dependent user  $i$  in the test data, the movie-watching sequence will be different as the user relies on the RS. The RS uses the training data  $Y^{\text{train}}$  to obtain a low-dimensional embedding of the movie features as in Equation (7). This gives the RS an embedding  $V_{J \times r}$  for movie features instead of  $A_{J \times d}$  to work with. In our setting, we find that  $r = 10$ , which is substantially lower than the number of tags  $d = 100$ . For this  $r$ -dimensional feature embedding, the platform learns a set of features for each user  $i$ , as denoted by  $U_i \in \mathbb{R}^r$ . The platform starts with a prior  $U_i^{(0)}$  and recommends a random movie in the first period. Like before, the user starts with a prior about their own preference weights  $\Theta_i^{(0)} \in \mathbb{R}^d$ . In every step  $t$ , the RS updates this  $r$ -dimensional vector of parameters from  $U_i^{(t-1)}$  to  $U_i^{(t)}$  using the procedure in Algorithm 2. User updates on the original  $d$ -dimensional vector space according to Algorithm 3.

We set the total periods at 200 and run for 100 simulations to obtain our results.

### 4.3 Results

Now we want to test the two remarks in our theory section. Remark 1 indicates that the RS-dependent user should have a lower regret compared with the self-exploring user when no RS shutdown occurs. Figure 3 plots the expected regret over time for both self-exploring and RS-dependent user. As shown in this figure, RS-dependent user has lower regret than the self-exploring user in almost every period. On average, RS-dependent user has only 25% (0.03/0.12) of the expected regret compared to the self-exploring user, showing the powerful prediction capabilities of the low-rank technique employed by the RS.

To test Remark 2, we need to illustrate two points: First, we want to test if the self-exploring user has lower Shannon entropy of the optimal action (movie) than the RS-dependent user when there is no RS shutdown. If this intuition is correct, when the RS shutdown occurs at period  $\tau$ , the RS-dependent user faces greater uncertainty about the prior distribution of optimal movies. In other words, RS-dependent user learns less over time than the self-exploring user. Figure 4 shows how the Shannon entropy evolves over time for both users. As illustrated in this figure, the entropy for both users decreases over time, indicating the learning for both users. As users become more certain about their preferences on movies, the uncertainty surrounding the optimal movies drops.

Second, at early periods, the entropy levels for the two users are relatively close; but, over time, the difference grows. This is because the entropy for the self-exploring user drops more rapidly than that of the RS-dependent user. This suggests that the self-exploring user learns at a faster rate. Finally, consistent with our hypothesis, the self-exploring user has



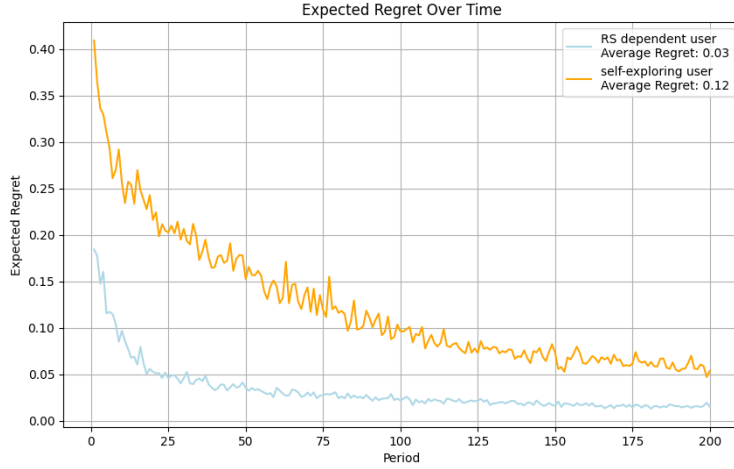


Figure 3: Expected regret over time for self-exploring user and RS-dependent user with RS shutdown at Period 50

lower entropy over time. At the final period, the self-exploring user's entropy is about 67% ( $2.40/3.59$ ) of the Shannon entropy of the RS-dependent user.

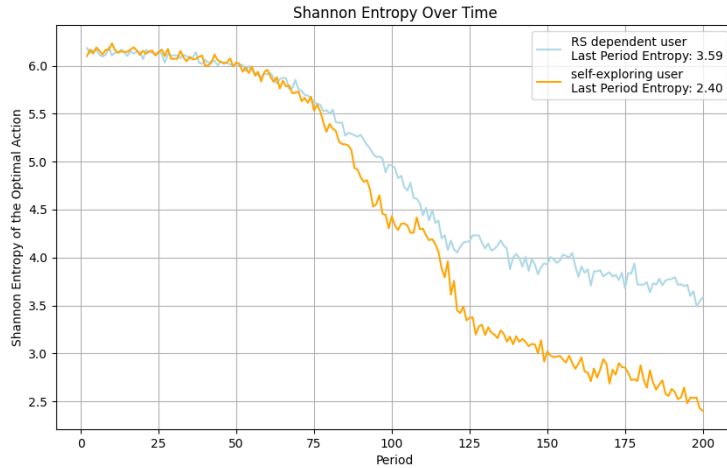


Figure 4: Shannon Entropy over time for self-exploring user and RS-dependent user without RS shutdown

The second-part of Remark 2 is about the post-shutdown regret for two users. Figure 5 clearly illustrates this point. Although the regret is lower for the RS-dependent user when the recommendation system is available, this user has a higher regret after shutdown, which is due to insufficient learning. In subsequent analyses, we aim to assess counterfactual policies

to determine if alternative RS configurations could potentially enhance the current system’s performance.

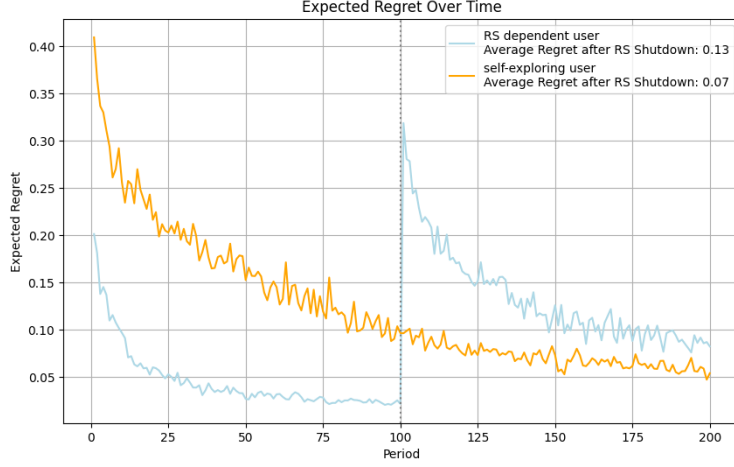


Figure 5: Expected regret over time for self-exploring user and RS-dependent user with RS shutdown at Period 50

#### 4.4 Potential Regulations

Our results in §4.3 show that personalized recommendation systems can act as a barrier to consumer learning. In particular, we demonstrate that when the recommendation system is not present, users who relied heavily on the recommendation system make worse decisions. The negative impact of recommendation systems on consumer learning motivates consumer protection policies. A typical candidate for such policies is privacy regulation whereby the recommendation system cannot use personal user-level data. Our analysis of self-exploring user reflects the potential outcomes under privacy regulations than ban user tracking and personalization. Although this approach has good regret performance in the absence of the recommendation system, our results suggest a better performance by the recommendation system prior to the shutdown. Thus, we want to examine if there are alternative policies that can achieve sufficient learning without completely losing the benefits of a recommendation system.

**Random Availability Policies** We consider a specific class of policies that add randomness to the availability of the recommendation system. As a result of this random availability, RS-dependent users need to make their own decisions in some time periods. This likely results in an increase in consumer learning without fully eliminating the benefits of the recommendation

systems. Specifically, we operationalize these policies using a single parameter  $p$  that controls the probability by which the recommendation system will be unavailable. It is noteworthy that our extant RS embodies a policy where  $p = 0$ , indicating its full availability.

We simulate the outcomes under each random availability policy and present the results in Figure 6. This figure clearly illustrates the trade-off between regret before and after shutdown, which has a close connection with the well-known exploration-exploitation trade-off. We show the Pareto Frontier of different random availability policies. Notably, we find some random availability policies that Pareto dominate the self-exploring policy. This finding suggests that random availability policies can serve as better alternatives than data protection policies that entirely ban remove the recommendation system.

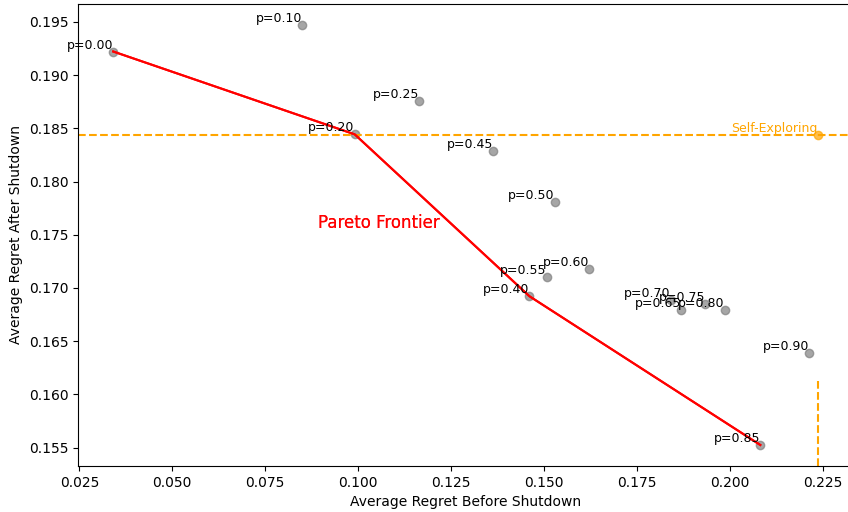


Figure 6: The performance of random availability policies in terms of regret before and after recommendation system shutdown.

**Self-regulated Policies** Another class of policies we consider is the self-regulated RS. As shown in Figure 4, RS-dependent learner has insufficient learning compared with the self-exploring user. So the first policy we test is adding a regularization term on Shannon entropy. The idea is that different movies will decrease the Shannon entropy differently, so the RS could suggest movies that would reduce the Shannon entropy more. In other words, these movies could reduce the uncertainty and aid consumer learning. The regularization weight is represented by  $\lambda$ . A higher  $\lambda$  means a higher punishment for high entropy (high uncertainty), thereby resulting in more learning. Figure 7 shows the outcomes based one

different  $\lambda$  values. Similar to Figure 6, we could find policies that Pareto dominates the self-exploring user.

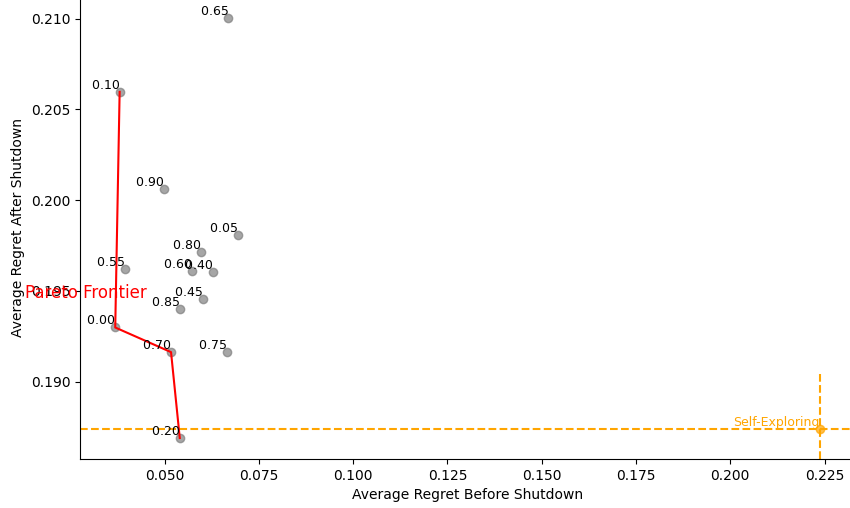


Figure 7: The performance of self-regulated policies in terms of regret before and after recommendation system shutdown (numbers in the graph represent values of  $\lambda$ ).

## 5 Empirical Analysis: Lab Experiment

Both our theoretical analysis and empirical analysis of the MovieLens data make explicit assumptions about the dynamics of consumer learning. However, one could argue that the real learning behavior of users can substantially deviate from these model-based assumptions. In this section, we want to relax these model-based assumptions and verify the validity of our results. As such, we aim to design an experiment to see if the real learning dynamics of users generate the same patterns that are predicted by our theoretical framework.

### 5.1 Experiment Design

Our goal is to design an experiment that mimics our empirical analysis in §4. As such, we want a multi-period experiment where one group receive recommendations until a certain period (treatment) and the other group do not receive any recommendations and have to make decisions on their own (control). Specifically, our treatment group reflects the RS-dependent user who relies on the recommendation system for a number of periods and observe a recommendation system shutdown at some point, whereas the control group creates a situation similar to the self-exploring user who does not receive any help from

recommendations and need to make decisions and learn the environment on their own.

We face several challenges in designing our experiment. First, we need an environment where users have a clear set of preference parameters and are incentivized to learn these parameters. Second, given the short span of a lab experiment, the task of learning has to be simple enough that users can feasibly learn their preference parameters in that time period. If the underlying preference parameters are overly complex, it is not possible to observe any tangible learning by users in the short span of a lab experiment. Third, the learning task needs to have some level of difficulty to prevent users from hacking the recommendation system. Our third challenge is almost the opposite of the second challenge, which suggests that our experiment needs to find a balance in terms of the over complexity of the learning task.

In our experiment, participants engage in a strategic game of choice, set in a stylized environment consisting of four distinct boxes. Each box contains a unique combination of red and blue balls, with each ball type having an associated utility parameter, denoted by  $a$  for red and  $b$  for blue. These parameters are unknown to participants, thereby incentivizing them to learn them over repeated rounds of play. Their objective is to consistently select the box that offers the highest payoff, which is computed based on the number of balls and their hidden values, with a small noise component denoted by  $e$  added to each round's payoff. Learning the true values of  $a$  and  $b$  is the key to mastering the game and reaping maximum rewards. Figure 8 shows the first question page for the control group. The treatment group will see the same page but with the recommended choice as show in Figure 9

1. Imagine you're playing a game where you get to pick one of four boxes: Box 1, Box 2, Box 3 and Box 4. Each box has a number of red and blue balls inside. Let's say Box 1 has 1 red ball and 6 blue balls, while Box 2 has 7 red balls and 1 blue ball. You can only choose one box each time.

Each type of ball is worth a certain amount of points: each red ball has a value "a" points and each blue ball has a value "b" points. Both "a" and "b" are two fixed number in range of zero to one. However, the exact values of "a" and "b" are unknown to you at first. You can learn about these values over time as you play the game more and more.

Your payoff from one box is:  $\text{num.red.ball} * a + \text{num.blue.ball} * b + e$ .

e is a small surprise. This "e" could either give you bonus points or subtract some points, and it changes every time you play. Overall, e is very close to 0.

So, for example, if you choose Box 1, your total score would be "a"\*1 (for the 1 red ball) + "b"\*6 (for the 6 blue balls), plus the small surprise "e".

The goal of this game is to choose the box with the highest possible points in each round. **If you choose the box with the highest payoff, you get the exact amount of points; otherwise, you get 0.** The numbers "a" and "b" will stay the same throughout this entire experiment. Trying to learn these two numbers will help you make better choices and potentially maximize your rewards!

Box 1: 1 red balls and 6 blue balls
Box 2: 7 red balls and 1 blue balls
Box 3: 8 red balls and 0 blue balls
Box 4: 2 red balls and 2 blue balls

Figure 8: Control Group First Question Example

The goal of this game is to choose the box with the highest payoff in each round. **If you choose the box with the highest payoff, you get the exact amount of points; otherwise, you get 0.** The numbers "a" and "b" will stay the same throughout this entire experiment. Trying to learn these two numbers will help you make better choices and potentially maximize your rewards!

Note: The recommended choice is a suggestion made by the system, based on historical data, where users tend to win more money with this choice.

Box 1: 1 red balls and 6 blue balls
Box 2: 7 red balls and 1 blue balls
Box 3: 8 red balls and 0 blue balls (Recommended Choice!)
Box 4: 2 red balls and 2 blue balls

Figure 9: Treatment Group First Question Example

Regardless of the treatment assignment, if the participant picks the box with the highest payoff, s/he will see a message shown in Figure 10. However, if the participant does not choose the box with the highest payoff, they s/he see the message shown in Figure 11.


In this round, you chose Box 3: 8 red balls and 0 blue balls.
Congratulations! The box you chose has the highest points, you win 5.51.
You have won 5.51 points until now!


Figure 10: Optimal Box Message

In this round, you chose Box 1: 1 red balls and 6 blue balls.

Unfortunately, the box you chose is not the one with the highest points, you win 0.

You have won 0 points until now!



Figure 11: Non-optimal Box Message

**Experiment Timeline:** The experiment consisted of a structured process designed to gauge participant responses under specific conditions. The experiment’s timeline proceeded as follows:

1. Participants are presented with a sequence of 20 box-choosing questions.
2. After answering each question, participants are immediately informed of the points they received from their choices. Furthermore, a cumulative tally, updated in real-time, keep track of the total points.
3. Participants in the treatment group observe a recommended choice for the first 10 periods, but from the period 11, they receive no recommendations. Participants in the treatment group do not receive any recommendations.
4. The experiment ends after 20 periods, i.e., after the participants have responded to all 20 questions. Each participant receives a base payment of \$1, and an additional bonus of  $0.02 \times [\text{total points earned}]$ .

**Regret and Learning Measurement:** We quantify *regret* as the expected reward disparity between the participant’s acquired points and the points associated with the optimal choice. To evaluate *learning*, we pose belief-based queries to the participants: “Please utilize the slider to indicate your belief about the value of  $a$  (points corresponding to each red ball). The scale spans from 0 to 1:  $a$  value of 0 suggests your belief that  $a$  is approximating 0, while a value of 1 indicates your belief that  $a$  is near 1.” Furthermore, to gauge the participants’ confidence in their beliefs, we pose another question: “Using the slider, express your confidence regarding the value of  $a$  (points for each red ball). The scale extends from 0 to 10, where 0 represents ‘Very Unsure’ and 10 stands for ‘Very Sure’.” These questions are asked after questions 5, 10, 15, and 19.



**Experiment Hypothesis:** We present the following two hypotheses for our experiment:

1. **Low Regret (H1):** The control group has a higher correct answer rate and a lower regret than the treatment group after the recommendation system shutdown.
2. **Better Learning (H2):** The control group has a lower bias and higher confidence about  $a$  and  $b$  than the treatment group.

## 5.2 Experiment Implementation

We launched the experiment on MTurk and collected the 199 responses (Control  $N = 91$ , Treat  $N = 108$ ). Table 5 shows the basic information from the experiment.

Treatment	Duration (in seconds)	Question Difficulty	a b Difficulty
Control	546.56	3.11	2.44
Treat	447.91	2.45	2.82

Table 5: Experiment Basic Information

From the table, it is evident that participants in the treatment group took less time compared to those in the control group to answer the questions. Following their responses to the 20 box-choosing questions, participants were prompted to rate the difficulty of the questions as well as their grasp on the concepts of  $a$  and  $b$ . They used a 5-point Likert scale, with 1 signifying “very easy” and 5 being “very difficult”. The data suggests that while the treatment group found the questions to be relatively easier, they perceived the concepts of  $a$  and  $b$  to be more challenging compared to the control group.

## 5.3 Result

**Low Regret Hypothesis:** Figure 12 illustrates the evolution of the correct answer ratio and mean expected regret for both the treatment and control groups over time. Initially, a discernible learning pattern emerges for the control group, evident by their rising correct answer ratio and diminishing regret as time progresses. Notably, post the RS shutdown (starting from period 11), the control group consistently outperforms the treatment group by showing a higher correct answer rate and lower regret. This trend aligns with our Low Regret Hypothesis (H1).

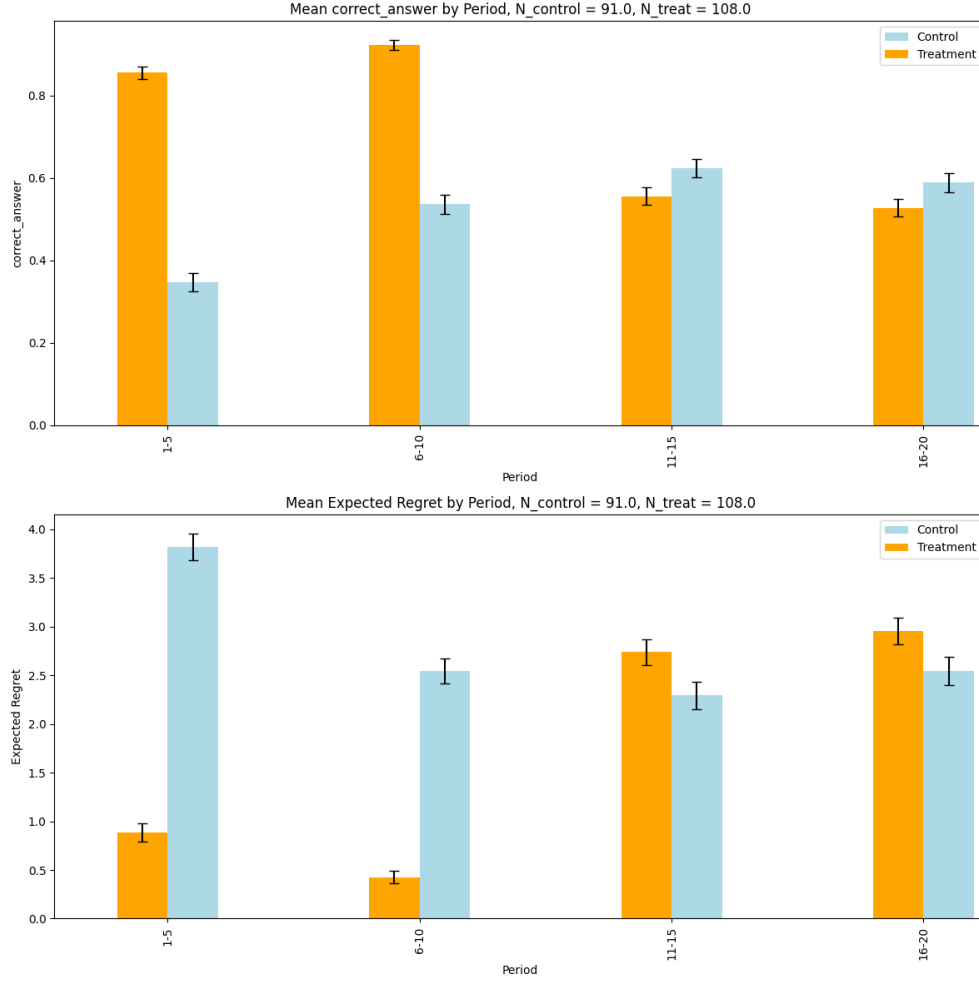


Figure 12: Expected regret over time for both user types.

**Better Learning Hypothesis** Learning is gauged through two metrics in our study: belief bias and confidence level. Figure 13 shows the temporal evolution of belief bias for parameters  $a$  and  $b$ . The markers  $bias_{a_1}$  through  $bias_{a_4}$  denote four checkpoints corresponding to the learning questions asked right after questions 5, 10, 15, and 19 respectively. For the control group, the learning trajectory for  $a$  (the more prominent parameter) is quite clear, while it remains ambiguous for  $b$ . Aligning with our anticipations, the initial checkpoint (post-question 5) reveals negligible variance in bias between the treatment and control groups. Yet, by the concluding checkpoint (post-question 19), a discernible difference emerges: the control group exhibits reduced bias for both  $a$  and  $b$  in comparison to the treatment group. This trend resonates with our “Better Learning Hypothesis”, positing that the control group’s understanding of parameters  $a$  and  $b$  is notably less biased than that of the treatment group.

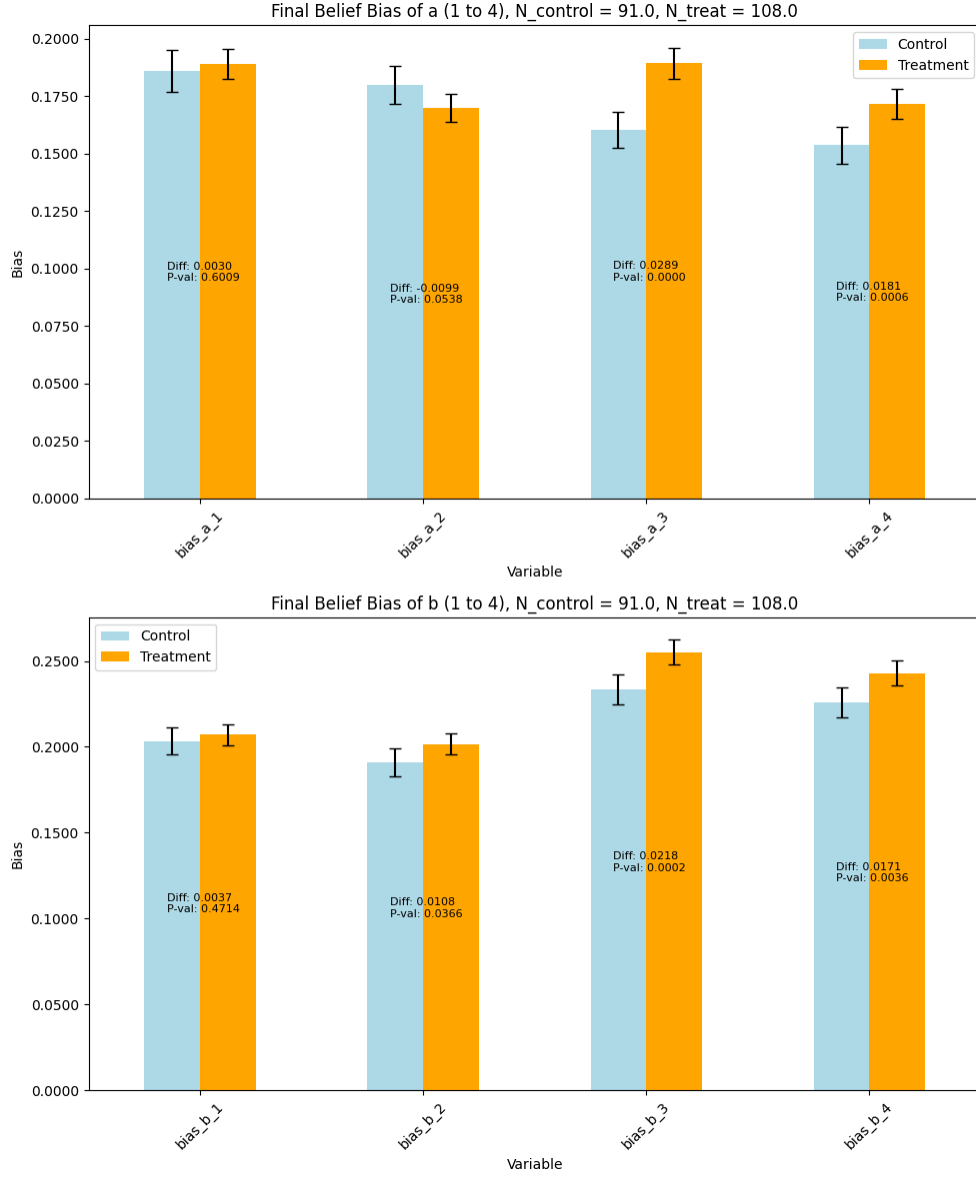


Figure 13: Belief Bias of a and b Over Time

Figure 14 illustrates the trajectory of participants' confidence in their beliefs about parameters  $a$  and  $b$  over time. The checkpoints correspond to the same intervals as those in the belief bias section. For the control group, a clear upward trend is evident: participants grow progressively more confident as time advances. Interestingly, we find that the treatment group is pattern shifts, making the control group more confident over time. This observed dynamic aligns with our "Better Learning Hypothesis," suggesting that the control group ultimately has greater confidence in their understanding of parameters  $a$  and  $b$  compared to

the treatment group.

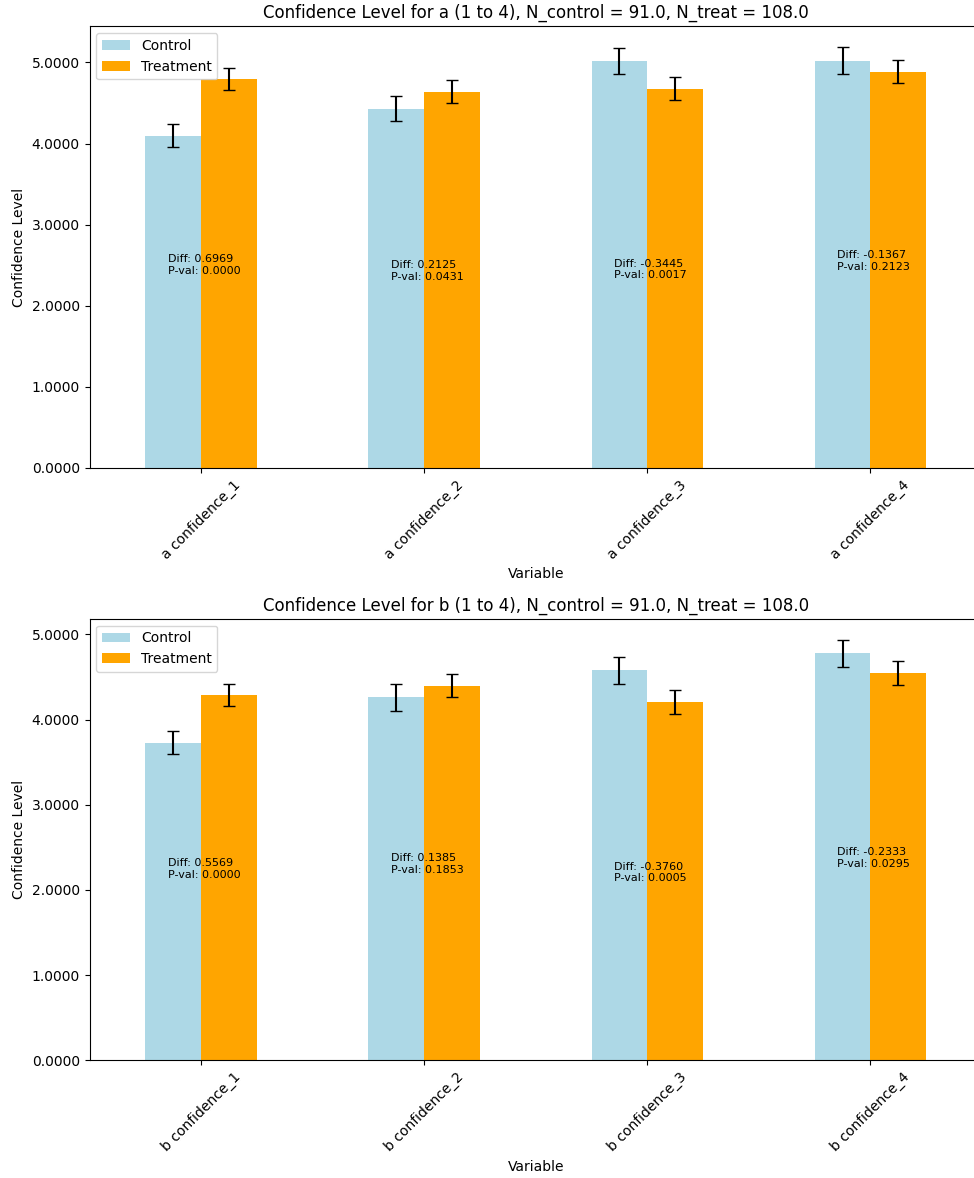


Figure 14: Belief Confidence of a and b Over Time

## 6 Discussion and Conclusion

Recommender systems are now an integral part of the digital ecosystem. However, the increased dependence of users on recommender systems has heightened concerns among consumer protection advocates and regulators. Past studies have documented various threats personalization algorithms pose to different aspects of consumer welfare, through violating

consumer privacy, unfair allocation of resources, or creating filter bubbles that can lead to increased political polarization. In this work, we bring a consumer learning perspective to this problem and examine whether personalized recommender systems hinder consumers' ability to learn their own preference preferences. We develop a linear framework where consumers learn their preference parameters in the presence of a recommender system. We introduce a notion of regret which is defined as the regret when consumers make decisions on their own. We theoretically show that the presence of the recommender system acts as a barrier to consumer learning. We then empirically investigate this phenomenon using the MovieLens data and a fully randomized lab experiment. Finally, we discuss different consumer protection policies and document the welfare implications of each.

## References

- D. A. Akerberg. Advertising, learning, and consumer choice in experience good markets: an empirical examination. *International Economic Review*, 44(3):1007–1040, 2003.
- S. Agrawal and N. Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on learning theory*, pages 39–1. JMLR Workshop and Conference Proceedings, 2012.
- S. Agrawal and N. Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International conference on machine learning*, pages 127–135. PMLR, 2013.
- E. Ascarza. Retention futility: Targeting high-risk customers might be ineffective. *Journal of Marketing Research*, 55(1):80–98, 2018.
- S. Athey and G. Imbens. Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences*, 113(27):7353–7360, 2016.
- E. Bakshy, S. Messing, and L. A. Adamic. Exposure to ideologically diverse news and opinion on facebook. *Science*, 348(6239):1130–1132, 2015.
- S. Barocas, M. Hardt, and A. Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2019. <http://www.fairmlbook.org>.
- O. Chapelle and L. Li. An empirical evaluation of thompson sampling. *Advances in neural information processing systems*, 24, 2011.
- G. Chen, T. Chan, D. Zhang, S. Liu, and Y. Wu. The effects of diversity in algorithmic recommendations on digital content consumption: A field experiment. *Available at SSRN 4365121*, 2023.
- A. T. Ching, T. Erdem, and M. P. Keane. Learning models: An assessment of progress, challenges, and new developments. *Marketing Science*, 32(6):913–938, 2013.
- G. S. Crawford and M. Shum. Uncertainty and learning in pharmaceutical demand. *Econometrica*, 73(4):1137–1173, 2005.
- P. Dandekar, A. Goel, and D. T. Lee. Biased assimilation, homophily, and the dynamics of polarization. *Proceedings of the National Academy of Sciences*, 110(15):5791–5796, 2013.
- J.-P. Dubé and S. Misra. Personalized pricing and consumer welfare. *Journal of Political Economy*, 131(1):131–189, 2023.
- T. Erdem and M. P. Keane. Decision-making under uncertainty: Capturing dynamic brand choice processes in turbulent consumer goods markets. *Marketing science*, 15(1):1–20, 1996.
- T. Erdem, M. P. Keane, T. S. Öncü, and J. Strebel. Learning about computers: An analysis of information search and technology choice. *Quantitative Marketing and Economics*, 3: 207–247, 2005.

- T. Erdem, M. P. Keane, and B. Sun. A dynamic model of brand choice when price and advertising signal product quality. *Marketing Science*, 27(6):1111–1125, 2008.
- S. Flaxman, S. Goel, and J. M. Rao. Filter bubbles, echo chambers, and online news consumption. *Public opinion quarterly*, 80(S1):298–320, 2016.
- A. Goldfarb and C. Tucker. Online Display Advertising: Targeting and Obtrusiveness. *Marketing Science*, 30(3):389–404, 2011a.
- A. Goldfarb and C. E. Tucker. Privacy Regulation and Online Advertising. *Management science*, 57(1):57–71, 2011b.
- C. A. Gomez-Urbe and N. Hunt. The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems (TMIS)*, 6(4):1–19, 2015.
- G. J. Hitsch. An empirical model of optimal dynamic product launch and exit under demand uncertainty. *Marketing Science*, 25(1):25–50, 2006.
- H. Hosseinmardi, A. Ghasemian, A. Clauset, M. Mobius, D. M. Rothschild, and D. J. Watts. Examining the consumption of radical content on youtube. *Proceedings of the National Academy of Sciences*, 118(32), 2021.
- G. Johnson. Economic research on privacy regulation: Lessons from the gdpr and beyond. 2022.
- G. A. Johnson, S. K. Shriver, and S. Du. Consumer privacy choice in online advertising: Who opts out and at what cost to industry? *Marketing Science*, 39(1):33–51, 2020.
- J. Kleinberg, S. Mullainathan, and M. Raghavan. The challenge of understanding what users want: Inconsistent preferences and engagement optimization. *arXiv preprint arXiv:2202.11776*, 2022.
- Y. Koren, S. Rendle, and R. Bell. Advances in collaborative filtering. *Recommender systems handbook*, pages 91–142, 2021.
- A. Lambrecht and C. Tucker. Algorithmic bias? an empirical study of apparent gender-based discrimination in the display of stem career ads. *Management science*, 65(7):2966–2981, 2019.
- T. Lattimore and C. Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.
- D. Lee and K. Hosanagar. How do recommender systems affect sales diversity? a cross-category investigation via randomized field experiment. *Information Systems Research*, 30(1):239–259, 2019.
- S. Lin, J. Zhang, and J. R. Hauser. Learning from experience, simply. *Marketing Science*, 34(1):1–19, 2015.

- G. Linden, B. Smith, and J. York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1):76–80, 2003.
- R. Mazumder, T. Hastie, and R. Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *The Journal of Machine Learning Research*, 11:2287–2322, 2010.
- X. Nie and S. Wager. Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, 108(2):299–319, 2021.
- O. Rafeian. Optimizing user engagement through adaptive ad sequencing. *Marketing Science*, 42(5):910–933, 2023.
- O. Rafeian and H. Yoganarasimhan. Targeting and privacy in mobile advertising. *Marketing Science*, 2021.
- J. H. Roberts and G. L. Urban. Modeling multiattribute utility, risk, and belief dynamics for new consumer durable brand choice. *Management Science*, 34(2):167–185, 1988.
- D. Russo and B. Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.
- D. Russo and B. Van Roy. An information-theoretic analysis of thompson sampling. *The Journal of Machine Learning Research*, 17(1):2442–2471, 2016.
- D. J. Russo, B. Van Roy, A. Kazerouni, I. Osband, Z. Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018.
- U. Shalit, F. D. Johansson, and D. Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *International Conference on Machine Learning*, pages 3076–3085. PMLR, 2017.
- D. Simester, A. Timoshenko, and S. I. Zoumpoulis. Efficiently evaluating targeting policies: Improving on champion vs. challenger experiments. *Management Science*, 66(8):3412–3424, 2020a.
- D. Simester, A. Timoshenko, and S. I. Zoumpoulis. Targeting prospective customers: Robustness of machine-learning methods to typical data challenges. *Management Science*, 66(6):2495–2522, 2020b.
- A. Swaminathan and T. Joachims. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning*, pages 814–823. PMLR, 2015.
- L. Sweeney. Discrimination in online ad delivery: Google ads, black names and white names, racial discrimination, and click advertising. *Queue*, 11(3):10–29, 2013.
- W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.



- C. E. Tucker. Social Networks, Personalized Advertising, and Privacy Controls. *Journal of Marketing Research*, 51(5):546–562, 2014.
- S. Wager and S. Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 0(0):1–15, 2018. doi: 10.1080/01621459.2017.1319839.
- H. Yoganarasimhan, E. Barzegary, and A. Pani. Design and evaluation of optimal free trials. *Management Science*, 2022.