Personalized Algorithms and the Virtue of Learning Things the Hard Way

Omid Rafieian[*]
Cornell Tech and Cornell University

Si Zuo[*]
Cornell University

For Most Updated Version, Click Here

Abstract

Personalized recommendation systems are now an integral part of the digital ecosystem. However, the increased dependence of users on these personalized algorithms 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 theoretically compare the expected regret for different types of consumers based on their usage of the personalized algorithm. Empirically, we demonstrate that consumers who rely more on personalized recommendations do not sufficiently learn their own preference parameters and make worse decisions in the absence of recommendation systems, compared to consumers who ignore personalized algorithms. 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.

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. Alarmingly, when 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 metrics can we 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.

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 that 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 using 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.

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.

Theoretically, the regret bounds established in Russo and Van Roy [2016] reveal two essential 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 through a low-rank factorization. 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 shutdown point. 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 the RS-dependent consumer exhibits lower expected regret before the shutdown 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.

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 and empirical analyses. 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, Rafieian and Yoganarasimhan, 2021, Yoganarasimhan et al., 2022, Rafieian, 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 [Ackerberg, 2003, Crawford and Shum, 2005, Erdem et al., 2005, Hitsch, 2006, Erdem et al., 2008, Jeziorski and Segal, 2015]. 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 terms 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 primary 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 ignore the recommendation system and make 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 Model of Consumer Learning

We consider a generic utility framework wherein user i receives utility from taking action A_j from action set $\mathcal{A}$. Each action is characterized by a d-dimensional set of attributes, i.e., $\mathcal{A} \subset \mathbb{R}^d$. For example, an action can be a movie with d attributes (e.g., genre, runtime). The user-specific vector of preferences $\theta_i \in \mathbb{R}^d$ characterize the utility from action $A_j \in \mathcal{A}$ as follows:

$$u_i(A_j) = A_j^T \theta_i + \epsilon_i, \tag{1}$$

where $\epsilon_{i,j}$ denotes the error term that comes from a mean-zero Normal distribution with known variance σ_ϵ^2, which implies that $E[u_i(a)] = A_j^T \theta_i$.

We extend our framework to sequential settings where users learn their preference parameters through experience. Let t denote each time period and $A_{i,t}$ the action chosen by user i in period t. For notation brevity, we define $U_{i,t} = u_i(A_{i,t})$ and let $\mathcal{H}_{i,t}$ denote the prior sequence of actions and utility outcomes up until period t, that is, $\mathcal{H}_{i,t} = (A_{i,1}, U_{i,1}, A_{i,2}, U_{i,2}, \ldots, A_{i,t}, U_{i,t})$. We assume θ_i is drawn from a Normal distribution $N(\mu_{i,0}, \Sigma_{i,0})$. The user starts with a prior $\hat{\theta}_{i,0} \sim N(\mu_{i,0}, \Sigma_{i,0})$ and update the preference parameters at the end of each time period t given the prior sequence $\mathcal{H}_{i,t}$ according to the following rule:

$$\mu_{i,t} = \mathbb{E}[\theta_i \mid \mathcal{H}_{i,t}] \tag{2}$$

$$\Sigma_{i,t} = \mathbb{E}\left[(\theta_i - \mu_{i,t})(\theta_i - \mu_{i,t})^T \mid \mathcal{H}_{i,t}\right] \tag{3}$$

The sequential nature of learning indicates that consumers update their parameters in every time period in a Bayesian fashion. Following the literature on Bayesian learning [Ching et al., 2013, Peleg et al., 2022, Tehrani and Ching, 2023], for any $t \geq 0$, we present the consumer parameter updating from $\mu_{i,t}$ and $\Sigma_{i,t}$ to $\mu_{i,t+1}$ and $\Sigma_{i,t+1}$ as follows:

Algorithm 1 Bayesian Updating

```
    Input: $\mu_{i,t}, \Sigma_{i,t}, A_{i,t}, U_{i,t}$
    Output: $\mu_{i,t+1}, \Sigma_{i,t+1}$
$\Sigma_{i,t+1} \leftarrow \left(\Sigma_{i,t}^{-1} + \frac{1}{\sigma_\epsilon^2} A_{i,t} A_{i,t}^T\right)^{-1}$
$\mu_{i,t+1} \leftarrow \Sigma_{i,t+1} \left(\Sigma_{i,t}^{-1} \mu_{i,t} + \frac{1}{\sigma_\epsilon^2} A_{i,t} U_{i,t}\right)$
```

Algorithm 1 determines consumer learning given any consumption sequence $\mathcal{H}_{i,t}$. See Appendix A for the proof. Our goal is to examine how different consumption sequences can result in different levels of learning.

3.3 Consumer Choice

We now discuss consumer's decision-making process that determine the consumption sequence $\mathcal{H}_{i,t}$. We characterize the consumer's decision-making process in the definition below:

Definition 1. *Let $\mathcal{I}_{i,t}$ denote all the information available to consumer i at time t. The consumer's decision-making process is characterized by the policy $\pi(\cdot \mid \mathcal{I}_{i,t})$, which is a probability distribution over actions conditional on the information available.*

To isolate the impact of personalized algorithms on consumer learning, we consider two types of consumers: (1) self-exploring consumer who makes decisions on their own in the absence of personalized recommendation system, and (2) RS-dependent consumer who follows the recommendation system (RS) in every time period. Both types have the same learning process as described in Algorithm 1. In what follows, we first characterize consumer choice for self-exploring consumers in §3.3.1. We then present how the recommendation system provides recommendations to characterize the consumption sequence for the RS-dependent consumer in §3.3.2.

3.3.1 Self-Exploring Consumer

In the absence of the recommendation system, the consumers make decision on their own. Given the utility framework in Equation (1), a forward-looking utility-maximizing consumer wants to optimize the overall utility over T periods. This naturally motivates consumers to learn their preference parameters through experience and balance good decision-making with proper exploration of their own preference parameters. We can define the objective function for a forward-looking consumer as maximizing the discounted expected utility stream as follows:

$$\underset{\pi}{\operatorname{argmax}}\, \mathbb{E}\left[\sum_{t=0}^{\infty} \delta^t U_{i,t} \mid \mu_{i,0}, \Sigma_{i,0}, \pi\right], \tag{4}$$

where δ is the discount factor and the expectation is taken over the randomness in actions $A_{i,t}$ and utilities $U_{i,t}$. Typical approaches to find the optimal sequence of choices by users involve solving a dynamic programming problem, which is known to be an NP-hard problem. 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 performance in terms of welfare [Chapelle and Li, 2011].[1] Thompson Sampling aims to find the right balance between exploration and exploitation in the decision-making process. The algorithm starts by initializing the consumer's prior belief distribution about the preference weights $N(\mu_{i,0}, \Sigma_{i,0})$. It then draws $\hat{\theta}_{i,0}$ from this distribution and computes the utility for all possible actions by plugging in $\hat{\theta}_{i,0}$ for $\theta_{i,0}$ in Equation (1). In the next step, the algorithm chooses the action that maximizes the estimated utility and observes the utility $U_{i,0}$ for that instance. Finally, the algorithm applies Bayesian updating procedure in Algorithm 1 to the new instance and updates the posterior distribution of preference weights. The Thompson Sampling algorithm continues this process for $\mathcal{T}$ periods.

Algorithm 2 Choice and Learning for the Self-Exploring Consumer

```
    Input: $\mu_{i,0}, \Sigma_{i,0}, \mathcal{A}, \mathcal{T}$
    Output: $\mathcal{H}_{i,\mathcal{T}}, \{\mu_{i,t}, \Sigma_{i,t}\}_{t=1}^{\mathcal{T}}$
for $t = 0 \to \mathcal{T}$ do
    $\hat{\theta}_{i,t} \sim N(\mu_{i,t}, \Sigma_{i,t})$                                          ▷ Distribution Sampling
    $A_{i,t} \in \text{argmax}_{A \in \mathcal{A}} A^T \hat{\theta}_{i,t}$                                   ▷ Action Selection
    $\Sigma_{i,t+1} \leftarrow \left(\Sigma_{i,t}^{-1} + \frac{1}{\sigma_\epsilon^2} A_{i,t} A_{i,t}^T\right)^{-1}$          ▷ Belief Updating (Same as Algorithm 1)
    $\mu_{i,t+1} \leftarrow \Sigma_{i,t+1} \left(\Sigma_{i,t}^{-1} \mu_{i,t} + \frac{1}{\sigma_\epsilon^2} A_{i,t} U_{i,t}\right)$     ▷ Belief Updating (Same as Algorithm 1)
end for
```

Assuming that the self-exploring consumer 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]. Second, it is computationally light, making it advantageous our later empirical analysis using the MovieLens data set.[2] 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.

e

3.3.2 RS-Dependent Consumer

We now focus on consumer choice in the presence of the recommendation system. To do so, we first introduce a personalized recommendation system that aims to simplify the consumer's decision-making problem. Since we want to quantify the impact of personalized

[1] The prior literature has documented lower regret for Thompson Sampling compared to the alternatives across several empirical domains.

[2] As a robustness check, we show that our qualitative insights will not change if we use an approximate dynamic programming solution to this problem.

recommendation systems on consumer learning, we assume that the recommendation system's objective is the same as that of consumer.[3] A natural difference between the personalized recommendation system and a single consumer is the fact that the system has access to the data of all other consumers. To understand how the recommendation system's data advantage manifests itself in better decision-making capabilities, we first introduce some notations. Let $\Theta_{[d\times N]}$ denote the matrix of preference weights for a group of N consumers, i.e., $\Theta_{[d\times N]} = [\theta_1 \mid \theta_2 \mid \dots \mid \theta_N]$. In all major platforms, N is a very large number. Similarly, let $A_{[d\times J]}$ denote the matrix of attributes for all J actions ($J = |\mathcal{A}|$) where each column is represent the vector of attributes for an action. We can define the matrix analog of the consumer utility in Equation 1 as follows:

$$U = \Theta^T A + E, \tag{5}$$

where $U_{N\times J}$ represent the utility for each pair of user and action and $E_{N\times J}$ is a matrix of i.i.d error term drawn from a mean-zero Normal distribution with known variance σ_ϵ^2. The recommendation system has access to $U^{\text{obs}}_{N\times J}$, which is an incomplete realization of matrix U as each consumer reveals utility for a subset of items. The main question is how the data from other consumers $U^{\text{obs}}_{N\times J}$ help the personalized algorithm learn about the new consumer i with vector of preferences θ_i.

In principle, if the prior data from consumers do not inform us about the new consumer, the recommendation system's strategy will be the same as the self-exploring consumer. However, the prior empirical work on personalized recommendation systems suggests otherwise as it documents extensive similarities in consumer preferences. The most common approach to characterize the similarities in consumer preferences is to use a factor model. That is, there is set of r independent d-dimensional factors $\{F_k\}_{k=1}^r$, where each factor $F_k \in \mathbb{R}^d$ presents a common type of preference for action attributes and each consumer's preferences is a linear combination of these r factors. We formally present this assumption as follows:

Assumption 1. *The matrix of consumers' preference weights $\Theta_{[d\times N]}$ can be decomposed as follows:*

$$\Theta_{[d\times N]} = F_{[d\times r]}\Gamma_{[r\times N]}, \tag{6}$$

[3] 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.

where $F_{[d\times r]} = [F_1 \mid F_2 \mid \ldots \mid F_r]$ is the matrix containing all r factors, and $\Gamma_{[r\times N]} = [\gamma_1 \mid \gamma_2 \mid \ldots \mid \gamma_N]$ presents the factor weights for all N consumers.

We can now view the recommendation system's data advantage in light of Assumption 1. Because the recommendation system has consumers' prior data, they have access to an accurate estimate of factor matrix F. As such, the recommendation system's task of learning consumer i's preference parameters will turn into the task of learning consumer i's weights for r factors because we have: $\theta_i = F\gamma_i$. Hence, the recommendation system's data advantage translates into learning only r parameters, as compared to self-exploring consumer's learning of d parameters.

We now present the procedure that determines choice and learning for the RS-dependent consumer in Algorithm 3. In this setting, not only is there a consumer who learns her preference parameters through experience, there is also a recommendation system that learns consumer preferences and offers recommendations. Both players learn consumer i's preference parameters, but they operate in different spaces: consumer i learns her own parameters θ_i in the d-dimensional space, whereas the recommendation system learns consumer i's preference weights for factors in the r-dimensional space. To distinguish between these two learning processes, we use superscripts (θ) and (γ) to refer to the parameters of both players' prior distributions: $\mu_{i,0}^{(\theta)}$, $\Sigma_{i,0}^{(\theta)}$, $\mu_{i,0}^{(\gamma)}$, and $\Sigma_{i,0}^{(\gamma)}$.

The recommendation system moves first in each time period. Since the recommendation system has access to $\hat{F}$, it only wants to learn γ_i. As such, it engages in a Linear-Gaussian Thompson Sampling procedure, whereby it first $\hat{\gamma}_{i,t}$ from the posterior distribution and then recommend the item with the highest expected utility (lines 2 and 3). The RS-dependent consumer always follows the recommended action (line 4). Once the utility is realized, both the consumer and recommendation system update parameters of their posterior distribution $\mu_{i,t+1}^{(\theta)}$, $\Sigma_{i,t+1}^{(\theta)}$, $\mu_{i,t+1}^{(\gamma)}$, and $\Sigma_{i,t+1}^{(\gamma)}$. The algorithm repeats this process for $\mathcal{T}$ periods.

A few points are worth noting about the RS-dependent consumer. First, we assume that the consumer 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 consumer because of its information advantage. In the main analysis, we abstract away from this possibility and simply assume that the consumer always follow the recommendation system to illustrate the differences between the self-exploring and RS-dependent consumer. [4] Second, the consumer learning

[4] If the RS-dependent user is rational and incurs a search cost when searching independently but no search cost when relying on the RS, then she faces a choice between searching on her own and following the RS

Algorithm 3 Choice and Learning for the RS-Dependent Consumer

```
    Input: $\mu_{i,0}^{(\theta)}, \Sigma_{i,0}^{(\theta)}, \mu_{i,0}^{(\gamma)}, \Sigma_{i,0}^{(\gamma)}, \hat{F}, \mathcal{A}, \mathcal{T}$
    Output: $\mathcal{H}_{i,\mathcal{T}}, \{\mu_{i,t}^{(\theta)}, \Sigma_{i,t}^{(\theta)}\}_{t=1}^{\mathcal{T}}, \{\mu_{i,t}^{(\gamma)}, \Sigma_{i,t}^{(\gamma)}\}_{t=1}^{\mathcal{T}}$
for $t = 0 \to \mathcal{T}$ do
    $\hat{\gamma}_{i,t} \sim N(\mu_{i,t}^{(\gamma)}, \Sigma_{i,t}^{(\gamma)})$    ▷ RS: Distribution Sampling
    $A_{i,t}^{RS} \in \text{argmax}_{A \in \mathcal{A}} A^T \hat{F} \hat{\gamma}_{i,t}$    ▷ RS: Recommendation Selection
    $A_{i,t} \leftarrow A_{i,t}^{RS}$    ▷ Consumer: Action Selection
    $\Sigma_{i,t+1}^{(\theta)} \leftarrow \left( \left( \Sigma_{i,t}^{(\theta)} \right)^{-1} + \frac{1}{\sigma_\epsilon^2} A_{i,t} A_{i,t}^T \right)^{-1}$    ▷ Consumer: Belief Updating
    $\mu_{i,t+1}^{(\theta)} \leftarrow \Sigma_{i,t+1}^{(\theta)} \left( \left( \Sigma_{i,t}^{(\theta)} \right)^{-1} \mu_{i,t}^{(\theta)} + \frac{1}{\sigma_\epsilon^2} A_{i,t} U_{i,t} \right)$    ▷ Consumer: Belief Updating
    $\Sigma_{i,t+1}^{(\gamma)} \leftarrow \left( \left( \Sigma_{i,t}^{(\gamma)} \right)^{-1} + \frac{1}{\sigma_\epsilon^2} A_{i,t} A_{i,t}^T \right)^{-1}$    ▷ RS: Belief Updating
    $\mu_{i,t+1}^{(\gamma)} \leftarrow \Sigma_{i,t+1}^{(\gamma)} \left( \left( \Sigma_{i,t}^{(\gamma)} \right)^{-1} \mu_{i,t}^{(\gamma)} + \frac{1}{\sigma_\epsilon^2} A_{i,t} U_{i,t} \right)$    ▷ RS: Belief Updating
end for
```

procedure in Algorithm 3 is identical to Algorithm 2, and the difference in learning only comes from the prior consumption sequence in these two settings. Finally, an implicit assumption we make here is that the RS-dependent consumer 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.

3.4 Welfare Analysis

We now conduct a welfare analysis of the two types of users we defined earlier: (1) self-exploring users who follow Algorithm 2, and (2) RS-dependent users who follow Algorithm 3. To conduct our welfare analysis, we use the notions of *regret* and *expected regret* that is defined as follows:

Definition 2. *Suppose that $A_i^* \in \text{argmax}_{a \in \mathcal{A}} \mathbb{E}[u_i(a) \mid \theta_i]$ is the optimal action given θ_i. For the sequence of actions $\{A_{i,t}\}_{t=0}^{T}$ chosen according to policy π, the* ***regret*** *is given as follows:*

$$\textit{Reg}(T; \pi, t = 0) = \sum_{t=0}^{T} \left(u_i(A_i^*) - u_i(A_{i,t}) \right), \tag{7}$$

every period. This dynamic decision process is complex, so we abstract away in the theoretical discussion. In §4.3, we extend our analysis to include the rational RS-dependent user with a search cost.

which is equal to the cumulative difference between the utility from always choosing optimal action and the utility from the sequence of actions taken till the end of period T. Based on the notion of regret in Equation (7)*, we can define the **expected regret** as follows:*

$$\mathbb{E}\left[Reg(T;\pi,t=0)\right]=\mathbb{E}\left[\sum_{t=0}^{T}\left(u_i(A_i^*)-u_i(A_{i,t})\right)\right], \tag{8}$$

where the expectation is taken over over the randomness in the actions, utilities, and the prior distribution over θ_i. This notion of expected regret is often referred to as the Bayes regret or Bayes risk.

As is clear from the definition, the notion of regret is closely tied to the notion of consumer welfare that we want to study in this section. One advantage of using regret is that we can directly use other established regret bounds from the literature. The paper we heavily borrow from for regret analysis is Russo and Van Roy [2016] that 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 policy π^{TS} up to time T is bounded by

$$\mathbb{E}\left[\mathrm{Reg}(T;\pi^{TS},t=0)\right]\leq\sqrt{\frac{H(A_i^*)dT}{2}}, \tag{9}$$

where $H(A_i^*)$ is the Shannon entropy of the prior distribution of the optimal action A_i^* [5], and d is the dimension of the true parameter θ. When the recommendation system is present, the regret bound of Equation (9) 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 $\sqrt{H(A_i^*)dT/2}$, whereas the RS-dependent user has a regret bound of $\sqrt{H(A_i^*)rT/2}$. 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*

[5] Shannon entropy is a measure of the uncertainty or randomness in a set of outcomes, quantifying the amount of information or surprise inherent in a variable's possible states. It is calculated using the formula $H(X)=-\sum_{i=1}^{n}p(x_i)\log_2 p(x_i)$, where $p(x_i)$ is the probability of each outcome x_i, summing across all possible outcomes n of the random variable X.

user in the absence of recommendation system shutdown.

Proof. Please see the formal proposition and proof in Appendix A. □

Now, if there is a recommendation system shutdown at time period τ, 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 τ onward. However, these two user types will differ in their entropy of the prior distribution of the optimal action A_i^* at time period τ. Let $A_{i,\tau,SE}^*$ and $A_{i,\tau,RS}^*$ denote the posterior distribution of optimal action at the start of period τ for self-exploring and RS-dependent users, respectively. The regret bound for the self-exploring user will be $\sqrt{H(A_{i,\tau,SE}^*)d(T-\tau)/2}$, whereas the regret bound for the RS-dependent user will be $\sqrt{H(A_{i,\tau,RS}^*)d(T-\tau)/2}$. As such, the difference between the two depends on the difference in the Shannon entropy of the prior distribution of optimal action at point τ. 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 τ 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 τ, the self-exploring user has a lower expected cumulative regret post shutdown than the RS-dependent user.*

Proof. Please see the formal proposition and proof in Appendix A. □

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

4 Empirical Analysis

As discussed in the previous section, our hypothesis about the impact of personalized recommendation systems on consumer learning depends on the distribution of consumer preferences. To provide an empirical proof-of-concept for our theoretical remarks, we need an empirical setting with real consumer preferences where a personalized recommendation system can learn complex consumer preferences and offer useful recommendations. We use the MovieLens 1M dataset, which is the benchmark dataset for personalized RS research[6]. It includes over one million ratings corresponding to a total of 6,040 distinct users and 3,706

[6] This dataset is publicly accessible and can be obtained from `https://grouplens.org/datasets/movielens/1m/`.

unique movies between 2000 and 2003, allowing researchers to form large-scale matrices needed for matrix factorization tasks. The median user rated 96 movies, and 63% of users rated all movies on the same day[7]. In addition to ratings, the data provide information about the movies, such as genre and themes, as well as a large array of tags associated with each movie. At the user level, we observe demographic characteristics such as gender, age, occupation, and zip code.

4.1 Empirical Strategy

To take our theoretical framework to data, we need to estimate two policy functions π^{RS} and π^{SE} which could map the consumer i's d-dimensional preference to the action A_t (the movie consumer i chooses to watch at period t). In other words, we need to simulate what movie consumer i would watch if she explores by herself or relies on the RS. To solve this problem, we face three key challenges. First, we do not directly observe consumers' d-dimensional vector of true preferences, so we must infer it from the data. We discuss our solution in § 4.1.1. Second, given consumers' d-dimensional preference, we need to figure out how RS could translate the d-dimensional preference into by r parameters where $r \leq d$. We provide the solution in § 4.1.2. Finally, we do not directly observe the learning behavior of users from Movie Lens Data, so we need to model and simulate how users choose movies and update beliefs over time. We describe our solution in § 4.1.3. Details of the simulation are provided in Appendix B.1.

4.1.1 Ground Truth Generation

Suppose there exists a user i who is new to the movie platform, and her preference for a new movie is based on d-dimensional movie attributes (e.g.,, originality, famous actress). In line with our theoretical framework, user i faces J movies on her main page and wants to choose a movie to watch. For our simulation exercise, we need to estimate her true preference parameter space $\theta_i \in \mathbb{R}^d$, representing user i's "taste" for d different movie attributes. We presume the user knows the movie j's attribute vector $A_j \in \mathbb{R}^d$ (each movie is represented by d attributes) but not her own preference vector $\theta_i \in \mathbb{R}^d$. This assumption is made for several reasons: first, the simulation focuses on the cold-start problem, implying the user is new to the platform and requires time to discover her preferences; second, movie attributes are generally observable and detailed on movie streaming platforms, especially combined with large number of reviews left by other users. We acknowledge that learning both preference

[7] Movie Lens is a website where users can rate movies to receive recommendations from the website. Therefore, many users rated all the movies at once.

and movie attributes could coexist in real settings. However, for experience-based products like movies and music, the preferences are probably more complex than movie attributes, as individuals often find it challenging to articulate their preferences. Therefore, for our main analysis, we focus on the learning of the user's own preference instead of the movie attributes. The way we estimate user i's preference θ_i is as below:

- In the MovieLens data, we can observe tags for all movies. We utilize the most frequently used d tags as attributes for the movies and generate $A_{[d\times J]}$, where J represents the number of movies, totaling 3080 in our dataset. Common tags include original, great ending, good soundtrack, and Oscar. We set $d = 100$ for the main simulation but allow for an arbitrary choice of d in §4.3.
- We use user i's rating on movie j to approximate user i's utility of watching movie j.[8] This yields a rating matrix $Y_{[N\times J]}$ with $N = 6040$ in the dataset (6040 users).
- Next, we aim to estimate all N users' preferences on all J movies: $\Theta_{[d\times N]}$, where column i represents user i's preference vector θ_i. We assume the ground truth $\Theta^T_{[d\times N]}$ remains constant in the dataset, as most users in the MovieLens dataset rated all movies within a single day. All J movies can be represented by an action matrix $A_{[d\times J]}$, where column j represents movie j. Our goal is to derive the preference matrix $\Theta_{[d\times N]}$ such that $\Theta^T_{[d\times N]} \times A_{[d\times J]} = Y_{[N\times J]}$.
- Given the sparsity of the rating matrix $Y_{[N\times J]}$, we complete it using matrix decomposition. The resulting matrix $\tilde{Y}_{[N\times J]}$, filled via low-rank decomposition, allows us to compute $\Theta_{[d\times N]^T} = \tilde{Y}_{[N\times J]} \times A^+$, where A^+ is the pseudo-inverse of $A_{[d\times J]}$[9].

4.1.2 RS's Factor Matrix

Our theory (Assumption 1) describes the RS's information advantage is the low-rank technique. That is, RS could use a low-dimension factor vector $\gamma_i \in \mathbb{R}^r$ to represent user i's preference vector θ_i such that $\theta_i = F\gamma_i$. Therefore, our second test is to figure out the low rank factor vector γ_i.

- The completion of the rating matrix $Y_{N\times J}$ using the matrix completion approach reveals $r = 10$ as the optimal low-rank dimension. See Appendix B.1 for details. Consequently, we set $r = 10$ for the RS's factor model. The implications of different r values are further discussed in § 4.3.
- After generating the ground truth matrix of ratings, we divide our data into two segments:

[8] Directly observing a user's realized utility from watching a movie is challenging. Based on the MovieLens data, we consider ratings a reasonable approximation of a user's satisfaction after watching a movie.

[9] A^+ is the least-squares solution to the equation $\Theta_{[d\times N]^T} \times A_{[d\times J]} = \tilde{Y}_{[N\times J]}$

training ($N_{\text{train}} = 2624$) and test ($N_{\text{test}} = 1000$).[10] For test users, we remove all ratings to create a cold-start setting. For training users, we use the regenerated ratings. Let Y^{train} and Y^{test} denote these two matrices.

- The ground truth $\Theta_{[d\times N_{train}]}$ in the train sample is decomposed into the product of $F_{[d\times r]}$ and $\Gamma_{[r\times N_{train}]}$, with $d = 100$ and $r = 10$.

4.1.3 Consumer's Updating and Learning Process

After obtaining the ground truth for all users and the RS's factor matrix $F_{[d\times r]}$, we proceed with the simulation and analysis of two types of users: the self-exploring user and the RS-dependent user. We set the total periods $T = 200$ and repeat the simulation process for every user in Y^{test}. For each user i from Y^{test}, we select a random list of 500 movies available for selection over 200 periods. The effect of movie list size is discussed in §4.3. Once a movie is chosen, it is removed from the list. We assume that both RS and self-exploring users have an uninformative prior, so the movie for the first period is randomly chosen. We allow users to have more prior information than RS in §4.3. The realized utility (rating) for user i watching movie j is modeled as $\Theta^T_{[d\times 1]} \times A_{[d\times 1]} + e$, where $e \sim N(0, \sigma^2)$. [11] We then follow §3.3.1 and §3.3.2 to simulate self-exploring user and RS-dependent users' choosing and updating behavior.

4.2 Main Results

4.2.1 Impact of Recommendation Systems on Consumer Welfare

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 1 plots the expected regret over time for both self-exploring and RS-dependent users. In our simulation exercise, for user i in the test dataset ($N = 1000$), we simulate two scenarios: (i) user i is a self-exploring user; (ii) user i is an RS-dependent user; and calculate the expected utility and expected regret respectively. Then, we aggregate all 1000 users in the test dataset and plot the mean of the expected regrets in Figure 1.

[10] There are 6040 users in the Movie Lens data, we split the rating data $Y_{[N\times J]}$ into three subsets: $Y^1_{N_1\times J}$, $Y^2_{N_2\times J}$, and $Y^3_{N_3\times J}$, where Y^3 is used for cold-start problem simulation. The subsets correspond to 40%, 43%, and 17% of the total users in the Movie Lens 1M dataset, with $N_1 = 2416$, $N_2 = 2624$, and $N_3 = 1000$, respectively. Y^1 is mainly used to identify the best matrix decomposition model for imputing all ratings so we do not include it in the cold-start problem simulation exercise. Please see Appendix B.1 for details.

[11] The RMSE is calculated as the square root of the average of the squared differences between the observed ratings in the test set $Y_{N_{\text{test}}}$ and the ratings in the imputed rating matrix $\tilde{Y}_{[N_{\text{test}}\times J]}$.In other words, we try to use e to capture the difference between our regenerated rating matrix and the real rating matrix. In the empirical exercise, $\sigma = 0.86$ and the average rating of $Y_{N_{\text{test}}}$ is 3.56.

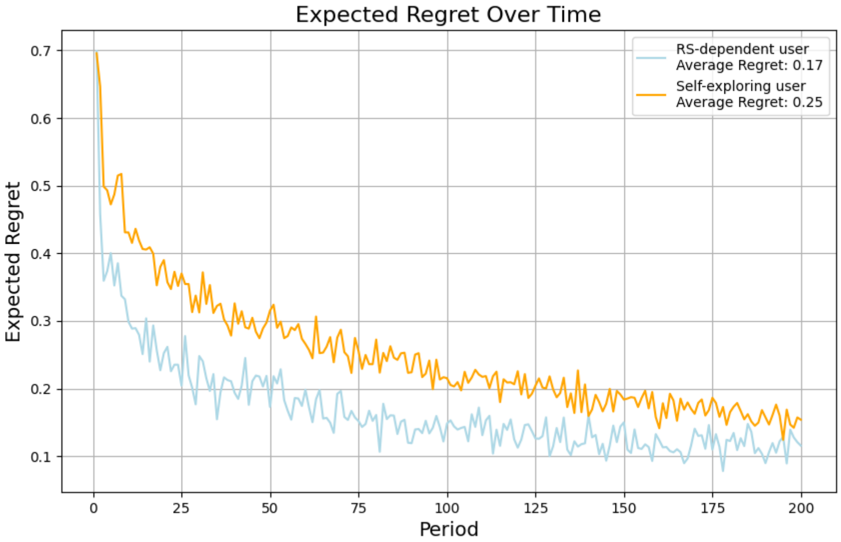

Figure 1: Expected regret over time for self-exploring user and RS-dependent user without RS Shutdown

The x-axis in Figure 1 represents the period number, and the y-axis represents the average expected regret for all users in the test dataset when they are (i) self-exploring users or (ii) RS-dependent users. We interpret the difference between a self-exploring user and an RS-dependent user as the value RS creates for users by offering them recommendations.

As shown in this figure, both self-exploring users' and RS-dependent users' regrets decrease over time, indicating a learning pattern for both types of users. In addition, the RS-dependent user's expected regret drops faster than that of the self-exploring user at the beginning. Across all 200 periods, the RS-dependent user experiences 68% (0.17/0.25) of the expected regret compared to the self-exploring user, demonstrating the powerful prediction capabilities of the low-rank technique employed by the RS.

4.2.2 Impact of Recommendation Systems on Consumer Learning

To test Remark 2, we examine whether the self-exploring user has higher learning than the RS-dependent user. We use Shannon entropy $H(A_i^*)$ to measure learning as it is related to our regret bound calculation. The regret bound increases at the square root speed of $H(A_i^*)$, which is the entropy of the prior distribution of the optimal action A_i^*, and measures the user's uncertainty about the optimal action A_i^*. If user i has perfect information about her preference parameter θ_i, then $H(A_i^*)$ should be zero; if user i has the uniform prior on all actions, then $H(A_i^*) = \log |\mathcal{A}|$, which is the log of the cardinality of the action set $\mathcal{A}$. Hence,

$H(A_i^*) \in [0, \log |\mathcal{A}|]$.

When a user watches more and more movies over time, the entropy of the posterior distribution of the optimal action should drop as she updates her belief on θ_i and learns about the optimal action. Therefore, we use the entropy of the posterior action to quantify the amount of learning for self-exploring users and RS-dependent users until period t. Both self-exploring and RS-dependent users update their belief on $(\mu_{i,t}, \Sigma_{i,t})$ (mean and variance of the posterior distribution on θ_i) according to Algorithm 1. Given $(\mu_{i,t}, \Sigma_{i,t})$, we sample θ_i from $N(\mu_{i,t}, \Sigma_{i,t})$ for n times and count the frequency of events where movie A_j is the optimal movie. We then use the probability ($p_{i,j,t} = \frac{count_{A_j}}{n}$) to calculate the Shannon entropy $H(A_{i,t}^*) = -\sum_{j \in \mathcal{A}} p_{i,j,t} \log_2(p_{i,j,t})$. We set $n = 1000$ in our simulation. See Appendix B.2 for details. [12]

For user i in the test dataset ($N = 1000$), we simulate two scenarios: (i) user i is a self-exploring user; (ii) user i is an RS-dependent user; and calculate the Shannon entropy respectively. We then aggregate all 1000 users in the test dataset and plot the mean of Shannon entropy in Figure 2. The x-axis in Figure 2 represents the period number, and the y-axis represents the average Shannon entropy for all users. As illustrated in this figure, the entropy for both self-exploring and RS-dependent users decreases over time, indicating the learning for both users. As users become more certain about their preferences for movies, the uncertainty surrounding the optimal movie drops.

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. Consistent with our hypothesis, the self-exploring user has lower entropy over time. At the final period, the self-exploring user's entropy is about 80% (2.49/3.11) of the Shannon entropy of the RS-dependent user.

In addition to entropy measurement, we also use another measure for learning: the KL-divergence from the uninformative prior. The KL-divergence measures the difference between two distributions. Since both self-exploring and RS-dependent users start with the same uninformative prior, we can use the KL-divergence to measure how much the posterior belief has been updated (changes from the prior). We find that the self-exploring user's posterior distribution on the preference parameter has a larger KL-divergence from the prior compared with that of the RS-dependent user, implying higher learning. Details are in

[12] In order to reduce the number of samples needed to calculate the entropy, we randomly select a small subset $\mathcal{S} \subset \mathcal{A}$ for the entropy calculation, where $\mathcal{S}$ is of size s and we set $s = 20$. Therefore, $H(A_i^*) \in [0, 4.32]$ as $\log_2(20) = 4.32$. For comparison, $\log_2(500) = 8.96$.

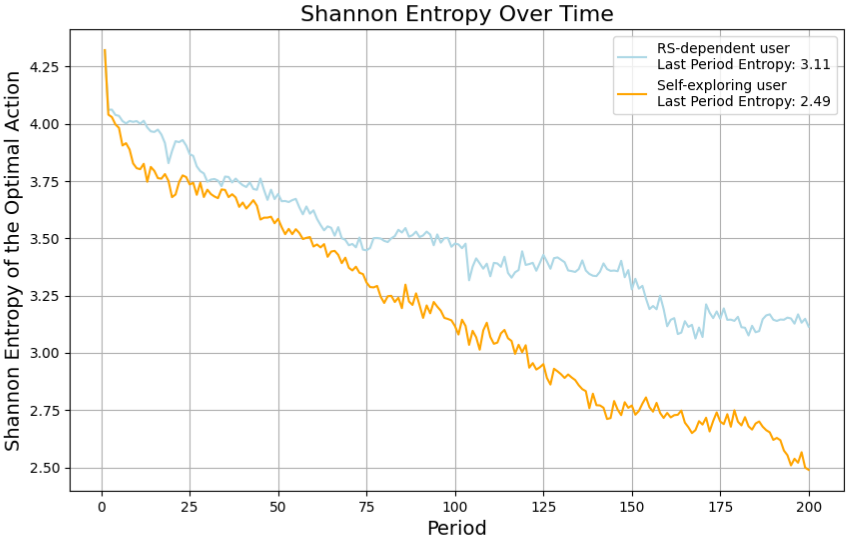

Figure 2: Shannon Entropy over time for the self-exploring user and RS-dependent user without RS shutdown.

Appendix B.2.

The learning/entropy result plays an important role in our story. First, it is the direct measurement of learning we use in the paper. Second, it connects our Remark 1 and Remark 2 as the Shannon entropy on the posterior distribution of the optimal action would become the Shannon entropy on the prior distribution of the optimal action when RS becomes unavailable at period τ.

4.2.3 Impact of Recommendation System Shutdown on Consumer Welfare

The second part of Remark 2 is about the post-shutdown regret for two users. We introduce the shutdown event at period τ and use the expected regret post-shutdown to measure the welfare loss associated with the insufficient learning. There are several reasons why we need the shutdown event of RS. First, the shutdown event allows us to map the insufficient learning (high Shannon entropy on the posterior distribution of the optimal action) to a welfare measure. Second, the absence of RS, or the distortion of RS, is likely due to some privacy or data protection regulations. As such, measuring the outcomes under this event shares important policy-related insights. Third, this analysis quantifies consumers' raw decision-making ability, which is immensely important for designing consumer protection policies.

The simulation of the self-exploring user is the same as Figure 1. However, for RS-

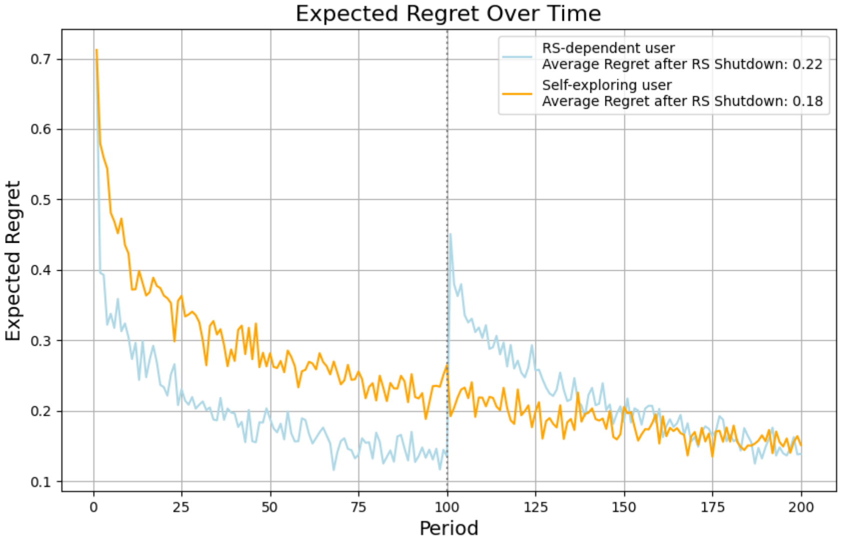

Figure 3: Expected regret over time for self-exploring user and RS-dependent user with RS shutdown at Period 100

dependent user i, she chooses and updates according to Algorithm 3 until period $\tau - 1$. From period τ, the RS is not present, so user i chooses and updates according to Algorithm 2, which is identical to the self-exploring user. Figure 3 examines our Remark 2 and reveals important insights. 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. As a result, the RS-dependent user's Shannon entropy at time τ is much higher than the self-exploring user. The RS-dependent user's post-shutdown regret is 122% (0.22/0.18) of the self-exploring user's regret.

4.3 Robustness Check and Extensions

We consider several robustness checks and extensions considering how our results would change if we change (1) the regret measure to consumer welfare, (2) from imputed rating to actual rating, (3) the RS shutdown period, (4) the self-exploring user's dimensionality of preference parameter d, (5) the self-exploring user's action set $\mathcal{A}$, (6) the self-exploring user's informativeness of prior beliefs relative to the RS, (7) the RS's rank used for matrix completion r, (8) to rational RS-dependent user with a search cost c. The figures and details are in Appendix B.3.

Our qualitative results remain unchanged under all these robustness checks: the RS-dependent user has lower regret before RS shutdown and higher regret after RS shutdown compared with the self-exploring user. When we switch from expected regret measurement to expected utility measurement, the RS-dependent user has higher expected utility when RS is available but lower utility after RS shutdown compared with the self-exploring user. Following Bresler et al. [2014], we only consider the top 200 users and the top 500 movies for the simulation, so we could use the real rating data not imputed rating. The resulting user-movie-rating matrix has 70% nonzero entries. The mean rating of this subsample is 3.73, which is slightly higher than the full sample (3.56). The data patterns are similar to our main results.

Our qualitative results still hold when RS shutdowns at periods 5, 10, 75, . . . , 195. When RS shutdowns at later periods, we observe that the RS-dependent user faces relatively higher regret post-shutdown compared with the self-exploring user. This illustrates that the problem of insufficient learning is more severe when the user depends on the system for a longer time. Our results persist when the self-exploring user's preference space dimension d equals $20, 40, \ldots, 120$. As d increases, the movie system becomes more complicated, leading to higher regrets due to more mistakes. By contrast, the RS-dependent user's expected regret before RS shutdown only increases slightly with respect to d, indicating that the RS could play a more significant role when the preference system is complicated.

In the main simulation, we set the action set $\mathcal{A}$ to consist of 500 random movies for the self-exploring user and the RS. Now we allow users to check fewer movies each time (20 or 50). Our primary results hold when the user's action set size (how many movies to check each time) is larger than 20. When the action set is too small, the RS performs similarly to the self-exploring user. In addition, users may have better prior information than RS since they have watched some movies before. We measure the information advantage of the user as she has already watched some movies before the first period (she starts to watch movies from period $-N$). We find that even if the self-exploring user has watched movies for 100 periods before the first period, her average regret before RS shutdown (from period 1 to period 100) is still higher than the RS-dependent user's regret. This shows that the RS's information advantage is significant and not easily overcome by the user's prior knowledge.

We argue that RS has the information advantage so that it can decompose the user's preference weights into a lower-dimensional representation. However, it is not always the case that a smaller r leads to lower regret. When r is very small, the RS-dependent user explores only minimally. Conversely, when r is very large, the RS-dependent user may explore excessively, resulting in high pre-RS-shutdown regret. We find that $r = 10$ performs better before RS shutdown compared with $r = 4, 20, 40$. Interestingly, the RS-dependent user's post-RS-shutdown regret decreases when r is large. This is probably because a large r leads to more exploration when the RS is available.

Finally, we discuss a scenario where the RS-dependent user chooses between exploring by herself or following the RS every period. When she decides to explore by herself, she faces the search cost c, and there is no search cost when she relies on the RS. We find that the higher the search cost, the lower the regret RS-dependent user has before RS shutdown. This occurs because, when the search cost is low, she may deviate from the RS's suggestion, leading to more mistakes. The result indicates that the RS benefits consumers more when the search cost is significant. After the RS shutdown, the expected regret of the RS-dependent user increases with the search cost c compared to the self-exploring user. This increase is due to the RS-dependent user's greater reliance on the RS when the search cost is high, leading to insufficient learning and, consequently, higher regret when the RS is no longer available.

5 Policy Implications

Our results in §4.2 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 have 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 a privacy regulation whereby the recommendation system cannot use personal user-level data. Our analysis of self-exploring users reflects the potential outcomes under privacy regulations that 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.

5.1 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. When RS is available, the RS-dependent user follows RS; otherwise, she chooses by herself, given her updated

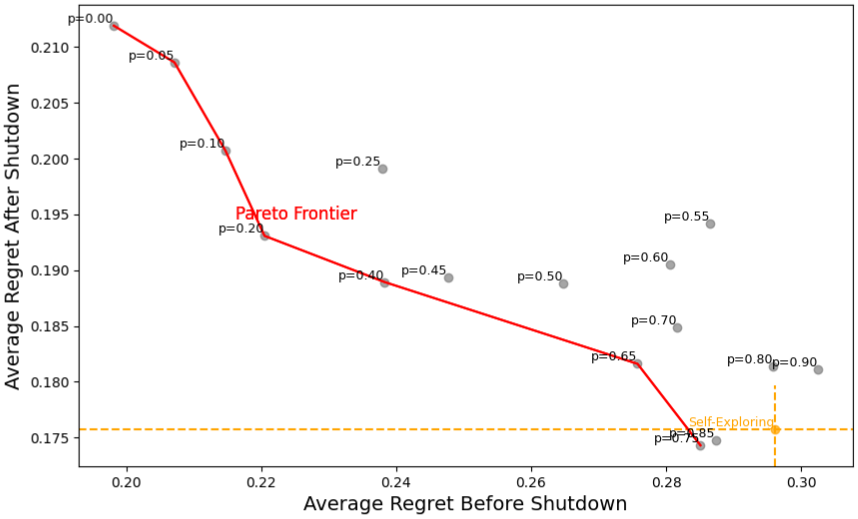

Figure 4: The performance of random availability policies in terms of regret before and after recommendation system shutdown.

belief. It is noteworthy that our extant RS embodies a policy where $p = 0$ indicates its full availability. Please see Appendix B.4 for the detailed algorithm.

We simulate the outcomes under each random availability policy and present the results in Figure 4. 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 the recommendation system.

5.2 Self-regulated Policies

Another class of policies we consider is the self-regulated RS. As shown in Figure 2, the 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 add to the Shannon entropy differently, so the RS could suggest movies that would reduce the Shannon entropy more. In other words, these movies could reduce uncertainty and aid consumer learning.

Similar to §4.2.2, in order to calculate user i's Shannon entropy at period t, we need to calculate the probability that movie j is the optimal movie given user i's belief $N(\mu_{i,t}, \Sigma_{i,t})$.

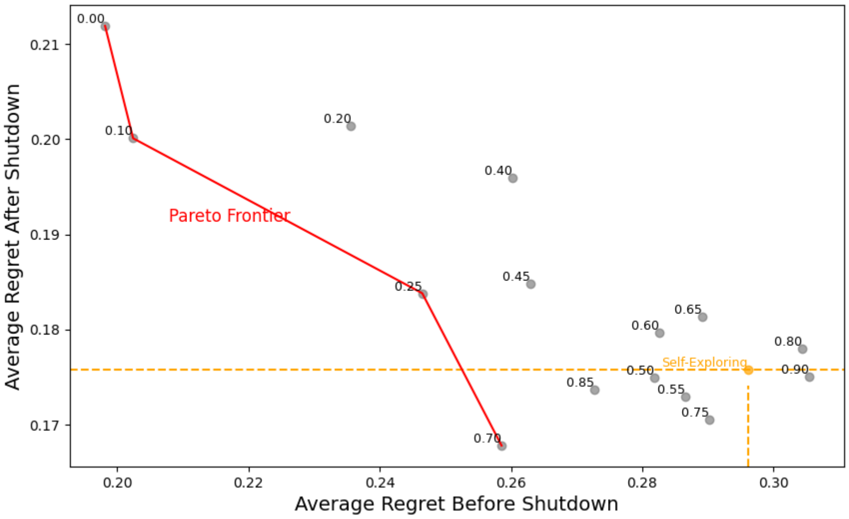

Figure 5: The performance of self-regulated policies in terms of regret before and after recommendation system shutdown (numbers in the graph represent values of λ).

Then, following the formula $H(A^*_{i,t}) = -\sum_{j\in\mathcal{A}} p_{i,j,t}\log_2(p_{i,j,t})$, we could calculate the change of Shannon entropy $\Delta H(A^*_{i,t,j}) = H(A^*_{i,t,j}) - H(A^*_{i,t-1})$ from choosing movie j at time t. Please see Appendix B.4 for the detailed algorithm on how to calculate $\Delta H(A^*_{i,t,j})$.

The regularization weight is represented by λ and λ is normalized from 0 to 1. A higher λ means a higher punishment for high entropy (high uncertainty), thereby resulting in more learning. Since the change of Shannon entropy $\Delta H(A^*_{i,j,t})$ is usually negative, by adding $-\lambda \times \Delta H(A^*_{i,j,t})$ in RS's selection function, we encourage the RS to pick the movies which could reduce the Shannon entropy more for users. Figure 5 shows the outcomes based on different λ values. Similar to Figure 4, we could also find policies that Pareto dominate the self-exploring user. The result provides us some ideas how we could improve RS to solve the consumer learning problem instead of an entire ban on the recommendation system.

6 Discussion and Conclusion

Personalized recommendation systems are now an integral part of the digital ecosystem. However, the increased dependence of users on these personalized algorithms 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 theoretically compare the expected regret for different types of consumers based on their usage of the personalized algorithm. Empirically, we demonstrate that consumers who rely more on personalized recommendations do not sufficiently learn their own preference parameters and make worse decisions in the absence of recommendation systems, compared to consumers who ignore personalized algorithms. Finally, we discuss a variety of consumer protection policies that help improve consumer learning and document the welfare implications of each.

References

D. A. Ackerberg. 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`.

G. Bresler, G. H. Chen, and D. Shah. A latent source model for online collaborative filtering. *Advances in neural information processing systems*, 27, 2014.

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.

V. Dani, S. M. Kakade, and T. Hayes. The price of bandit information for online optimization. *Advances in Neural Information Processing Systems*, 20, 2007.

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-Uribe 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.

P. Jeziorski and I. Segal. What makes them click: Empirical analysis of consumer demand for search advertising. *American Economic Journal: Microeconomics*, 7(3):24–53, 2015.

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.

A. Peleg, N. Pearl, and R. Meir. Metalearning linear bandits by prior update. In *International Conference on Artificial Intelligence and Statistics*, pages 2885–2926. PMLR, 2022.

O. Rafieian. Optimizing user engagement through adaptive ad sequencing. *Marketing Science*, 42(5):910–933, 2023.

O. Rafieian 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.

S. S. Tehrani and A. T. Ching. A heuristic approach to explore: The value of perfect information. *Management Science*, 2023.

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.

Appendices

A Part A: Proof

paragraphProof for the Bayes Updating Rule in Algorithm 1

Proof. At any given time t, we have an updated belief about θ_i which is normally distributed as:

$$\theta_i \sim \mathcal{N}(\mu_{i,t}, \Sigma_{i,t}) \tag{10}$$

New data comes in the form of $A_{i,t}$ (a $d \times 1$ vector of action attributes at time t) and $U_{i,t}$ (the utility observed from taking action $A_{i,t}$). Given the new observation $(A_{i,t}, U_{i,t})$, the likelihood function (probability of observing $U_{i,t}$ given $A_{i,t}$ and θ_i) is normal with mean $A_{i,t}^T\theta_i$ and variance σ_ϵ^2. The precision (inverse of the covariance matrix) of the prior distribution for θ_i is $\Sigma_{i,t}^{-1}$, and the precision of the new data is $\frac{1}{\sigma_\epsilon^2}A_{i,t}A_{i,t}^T$.

Therefore, the updated precision matrix (posterior precision) is the sum of the prior precision and the precision of the likelihood:

$$\Sigma_{i,t+1}^{-1} = \Sigma_{i,t}^{-1} + \frac{1}{\sigma_\epsilon^2}A_{i,t}A_{i,t}^T \tag{11}$$

The updated mean combines the prior mean and the new data, weighted by their respective precisions. The weight for the prior mean is its precision $\Sigma_{i,t}^{-1}$, and the weight for the new data $U_{i,t}$ is $\frac{1}{\sigma_\epsilon^2}$:

$$\mu_{i,t+1} = \Sigma_{i,t+1} \left(\Sigma_{i,t}^{-1} \mu_{i,t} + \frac{1}{\sigma_\epsilon^2} A_{i,t} U_{i,t} \right) \tag{12}$$

□

Proof for Remark 1 Before the set-up of the proof, we first need to utilize two results derived from Russo and Van Roy [2016] and Dani et al. [2007].

Lemma 1. *In the first τ periods with the absence of recommendation system shutdown, self-exploring user's expected regret R_{SE}^{τ} and RS-dependent user's expected regret R_{RS}^{τ} have the upper bounds as below:*

$$\mathbb{E}\left[Reg(\tau; \pi^{\mathrm{SE}}, t=0) \right] \leq \sqrt{\frac{1}{2} H(A_{i,SE}^*) d\tau} \tag{13}$$

$$\mathbb{E}\left[Reg(\tau; \pi^{\mathrm{RS}}, t=0) \right] \leq \sqrt{\frac{1}{2} H(A_{i,RS}^*) r\tau} \tag{14}$$

where $r < d$ as RS uses the low-rank techniques to reduce the dimensionality of item space from d to r.

Lemma 2. *In the first τ periods with the absence of recommendation system shutdown, self-exploring user's expected regret R_{SE}^{τ} and RS-dependent user's expected regret R_{RS}^{τ} have the lower bounds as below:*

$$\mathbb{E}\left[Reg(\tau; \pi^{\mathrm{SE}}, t=0) \right] \geq c_{SE} \sqrt{H(A_{i,SE}^*) d\tau} \tag{15}$$

$$\mathbb{E}\left[Reg(\tau; \pi^{\mathrm{RS}}, t=0) \right] \geq c_{RS} \sqrt{H(A_{i,RS}^*) r\tau} \tag{16}$$

where c_{SE}, c_{RS} are positive constants, independent of dimension (d, r) and periods τ.

Combined with Lemma 1, we could also see that $c_{SE}, c_{RS} < \sqrt{\frac{1}{2}}$.

In the first τ periods, both the self-exploring user and the RS are essentially solving the same dynamic programming problem, differing only in the dimensionality of the preference space. Therefore, we propose the following lemma concerning the entropy of the prior distribution of the optimal action.

Lemma 3. *With the same action set $\mathcal{A}$ and the same prior on the probability of each action $P(A^* = A)$, we have $H(A_{i,SE}^*) = H(A_{i,RS}^*) = H(A_i^*)$.*

Proof.

$$\begin{aligned}
H(A^*_{i,SE}) &= -\sum_{j=1}^{|\mathcal{A}|} P(A^*_{i,SE} = A_j) \log P(A^*_{i,SE} = A_j) \\
&= -\sum_{j=1}^{|\mathcal{A}|} P(A^*_{i,RS} = A_j) \log P(A^*_{i,RS} = A_j) \\
&= H(A^*_{i,RS})
\end{aligned}$$

In the empirical exercise, we start with the assumption that all actions are equally likely to be the optimal action, that is, $P(A^*_i = A_j) = \frac{1}{|\mathcal{A}|}$ for all i. Under this assumption, $H(A^*_{i,SE}) = H(A^*_{i,RS}) = H(A^*_i) = \log |\mathcal{A}|$. □

We denote the regret difference $\Delta R^{pre} = \mathbb{E}\left[\text{Reg}(\tau; \pi^{SE}, t = 0)\right] - \mathbb{E}\left[\text{Reg}(\tau; \pi^{RS}, t = 0)\right]$ as the regret reduction due to the RS system. Based on the lemmas presented earlier, we propose the following:

Proposition 1. *For any $\tau \in \mathbb{N}$, if $r < d$, and the ratio $\frac{r}{d}$ is relatively small, specifically $\frac{r}{d} \leq \min\left\{\frac{(c_{SE})^2}{2}, \frac{(c_{RS})^2}{2}\right\}$, then the pre-shutdown regret difference satisfies*

$$\Delta R^{pre} \geq \left(\sqrt{c^2_{SE} d} - \sqrt{\frac{r}{2}}\right) \sqrt{H(A^*)\tau} \geq 0.$$

Proof.

$$\begin{aligned}
\Delta R^{pre} &= \mathbb{E}\left[\text{Reg}(\tau; \pi^{SE}, t = 0)\right] - \mathbb{E}\left[\text{Reg}(\tau; \pi^{RS}, t = 0)\right] \\
&\geq \sqrt{c^2_{SE} H(A^*_i) d\tau} - R^{\tau}_{RS} && \text{by Lemma 2 and Lemma A} \\
&\geq \sqrt{c^2_{SE} H(A^*_i) d\tau} - \sqrt{\frac{1}{2} H(A^*_i) r\tau} && \text{by Lemma 1 and Lemma A} \\
&= \left(\sqrt{c^2_{SE} d} - \sqrt{\frac{r}{2}}\right) \sqrt{H(A^*_i)\tau} \\
&\geq 0 && \text{if } \frac{r}{d} \leq \frac{(c_{SE})^2}{2}.
\end{aligned}$$

□

This proposition shows that the regret reduction from RS is positive when $\frac{r}{d}$ is sufficiently small, indicating the welfare-improving effect of RS. Additionally, the formula for the lower bound of the regret difference is increasing in d and decreasing in r, aligning with our empirical findings.

Proof for Remark 2 We denote $H(A^*_{i,\tau,SE})$ and $H(A^*_{i,\tau,RS})$ as the entropy of self-exploring user's and RS-dependent user's posterior distribution at the shutdown period τ. Then following the existing lemmas, we could have the post-shutdown regret difference proposition as below:

Proposition 2. *For any $\tau \in \mathcal{N}$ and $\tau < T$, if $r < d$, and $\frac{r}{d}$ is relatively small ($\frac{r}{d} \leq \min\{\frac{(c_{SE})^2}{2}, \frac{(c_{RS})^2}{2}\}$). Then the post RS-shutdown regret difference,*

$$\Delta R^{post} \leq \left(\sqrt{\frac{1}{2}H(A^*_{i,\tau,SE})} - \sqrt{c^2_{RS}H(A^*_{i,\tau,RS})}\right)\sqrt{d(T-\tau)}$$

when $\frac{H(A^*_{i,\tau,RS})}{H(A^*_{i,\tau,SE})} \geq \frac{1}{2c^2_{RS}}$,

$$\Delta R^{post} \leq \left(\sqrt{\frac{1}{2}H(A^*_{i,\tau,SE})} - \sqrt{c^2_{RS}H(A^*_{i,\tau,RS})}\right)\sqrt{d(T-\tau)} \leq 0$$

Proof.

$$\begin{aligned}
\Delta R^{post} &= \mathbb{E}\left[\text{Reg}(T;\pi^{SE}, t=\tau)\right] - \mathbb{E}\left[\text{Reg}(T;\pi^{RS}, t=\tau)\right] \\
&\leq \sqrt{\frac{1}{2}H(A^*_{i,\tau,SE})d(T-\tau)} - \mathbb{E}\left[\text{Reg}(T;\pi^{RS}, t=\tau)\right] && \text{by Lemma 1 and Lemma A} \\
&\leq \sqrt{\frac{1}{2}H(A^*_{i,\tau,SE})d(T-\tau)} - c_{RS}\sqrt{H(A^*_{i,\tau,RS})d(T-\tau)} && \text{by Lemma 2 and Lemma A} \\
&= \left(\sqrt{\frac{1}{2}H(A^*_{i,\tau,SE})} - \sqrt{c^2_{RS}H(A^*_{i,\tau,RS})}\right)\sqrt{d(T-\tau)} \\
&\leq 0 && \text{if } \frac{H(A^*_{i,\tau,RS})}{H(A^*_{i,\tau,SE})} \geq \frac{1}{2c^2_{RS}}.
\end{aligned}$$

□

B Part B: Empirical Analysis Related Material

B.1 Simulation Exercise

Our goal in the simulation experiment is to simulation the two movie watching paths: $A_{i,self}$ for the self exploring user and $A_{i,RS}$ for the RS-dependent user. $A_{i,self}, A_{i,RS}$ are two vectors consist of T movies for T periods. To generate $A_{i,self}$, we need several inputs below:

- **User Preference Parameter Space Dimension** d: We set $d = 100$, corresponding to the 100 most frequently mentioned tags in the Movie Lens data, as shown in Table A1. We construct the action matrix $A_{[d\times J]}$ for all $J = 3080$ movies.
- **Data Splitting**: We split the rating data $Y_{[N\times J]}$ into three subsets: $Y^1_{N_1\times J}$, $Y^2_{N_2\times J}$, and $Y^3_{N_3\times J}$, where Y^3 is used for cold-start problem simulation. The subsets correspond to 40%, 43%, and 17% of the total users in the Movie Lens 1M dataset, with $N_1 = 2416$, $N_2 = 2624$, and $N_3 = 1000$, respectively.
- **Ground Truth** Θ: We obtain the ground truth user preferences as follows:
  1. Combine Y_1 and Y_2 to identify the best matrix decomposition model for imputing all ratings into $\tilde{Y}_{[N\times J]}$. We split Y_1 and Y_2 into an 80% training sample and a 20% test sample to evaluate performance across different r values (5, 10, ..., 100). Based on Figure 6, we select $r = 10$ for our main simulation.
  2. Impute the entire rating data $Y_{[N\times J]}$ into $\tilde{Y}_{[N\times J]}$. Using the action matrix $A_{[d\times J]}$, we calculate $\Theta_{[d\times N]^T} = \tilde{Y}_{[N\times J]} \times A^+$, where A^+ is the pseudo-inverse of $A_{[d\times J]}$, providing the least-squares solution to $\Theta_{[d\times N]^T} \times A_{[d\times J]} = \tilde{Y}_{[N\times J]}$.
- **RS's Factor Matrix** $F_{[d\times r]}$: Finally, we combine Y_2 and Y_3 for evaluation. For Y_2, we have true ratings observed, and for Y_3 users there is no data (cold-start). We decompose $\Theta_{[d\times N_2]}$ into the product of $F_{[d\times r]}$ and $\Gamma_{[r\times N_2]}$, with $d = 100$ and $r = 10$. Then RS updates the factor weight matrix $\Gamma_{[r\times N_2]}$ by solving the equation $F_{[d\times r]}\Gamma_{[r\times N_2]} \times A_{[d\times J]} = \tilde{Y}_{[N\times J]}$. RS combines $\Theta_{[d\times N_2]^T}$ and $A_{[d\times J]}$ to update the factor weight matrix $\Gamma_{[r\times N_2]}$.
- Finally, for all the new users in Y^3, we derive the ground-truth movie feature preference matrix $\Theta_{[d\times N_3]}$ and ground truth ratings $\tilde{Y}_{N_3\times J} = \Theta^T_{[d\times N_3]} \times A_{[d\times J]}$.
- **DGP for realized utility** : In the cold-start problem simulation, We assume the realized utility (rating) for user i watching movie j is $\Theta_{[d\times 1]^T} \times A_{[d\times 1]} + e$, where $e \sim N(0, \sigma^2)$. $Y_{N_3\times J}$ is split into an 80% training sample and a 20% test sample to

approximate σ using the test sample RMSE, using the exact matrix decomposition step as earlier. In this simulation, we get $\sigma = 0.86$.

Finally we could use these inputs and simulate $A_{i,self}, A_{i,RS}$ using the following procedure:

- *Step 1*: We pick one user i from Y_3 and select a random movie list consisting 500 movies to the user to choose every period. After a movie is chosen, it is removed from the movie list.
- *Step 2*:
  1. *Self-exploring user:* For self-exploring user i in the test data, the user starts with a prior about their own preference weights $\hat{\theta}_i^{(0)} \in \mathbb{R}^d$ and chooses a random movie in the beginning. In every time period t, the user updates the d-dimensional set of preference weights based on the realized utility from $\hat{\theta}_i^{(t-1)}$ to $\hat{\theta}_i^{(t)}$ following Algorithm 2.
  2. *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^2 to obtain a low-dimensional embedding of the movie features as in Equation (6). 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 $\hat{\gamma}_i^{(0)}$ and recommends a random movie in the first period. Like before, the user starts with a prior about their own preference weights $\hat{\theta}_i^{(0)} \in \mathbb{R}^d$. In every step t, the RS updates this r-dimensional vector of parameters from $\gamma_i^{(t-1)}$ to $\gamma_i^{(t)}$ using the procedure in Algorithm 3. User updates on the original d-dimensional vector space according to Algorithm 3.
- We simulate 200 periods and repeat the process until all users in Y_3 have been simulated. The expected regret and expected utility for all users in Y_3 are calculated as below:
  1. **Expected Utility**: user i's expected utility of watching movie $A_{i,t}$ at time t is

$$\mathbb{E}u_{i,t} = \theta_i^T \times A_{i,t}$$

  where the expectation is taken over the randomness in utilities.

2. **Expected Regret**: user i's expected regret at time t is calculated as:

$$\mathbb{E}\left[\mathrm{Reg}(T)\right] = \theta_i^T \times A_t^* - \theta_i^T \times A_{i,t}$$

   where $A_i^* \in \mathrm{argmax}_{a \in \mathcal{A}_t} \theta_i^T \times A]$ and $\mathcal{A}_t$ is the available movie list for user i at time t. The expectation is also taken over the randomness in utilities.

Below is the full list of most frequent 100 tags in the movie lens data.

Table A1: Top 100 Movie Lens Data Tags

Top 1-34	Top 35-68	Top 69-100
MovieId	betrayal	unlikely friendships
original	pg-13	passionate
mentor	cinematography	very interesting
great ending	redemption	dramatic
catastrophe	light	relationships
dialogue	intense	so bad it's funny
good	family	independent film
great	corruption	murder
chase	not funny	sexy
runaway	unusual plot structure	drinking
good soundtrack	twists & turns	childhood
storytelling	entirely dialogue	complex
vengeance	suprisingly clever	creativity
story	pornography	lone hero
weird	transformation	atmospheric
drama	cult film	based on book
greed	adapted from:book	first contact
great acting	happy ending	entertaining
imdb top 250	very funny	narrated
culture clash	death	friendship
brutality	life & death	obsession
fun movie	social commentary	based on a book
adaptation	stylized	loneliness
criterion	interesting	sexualized violence
life philosophy	enigmatic	oscar (best supporting actress)
suspense	fight scenes	very good
melancholic	harsh	gunfight
predictable	police investigation	stereotypes
visually appealing	revenge	underrated
talky	justice	secrets
great movie	quirky	nudity (full frontal - brief)
oscar (best directing)	excellent script	tagId
clever	feel-good	tag
destiny	gangsters	
fantasy world	violence	

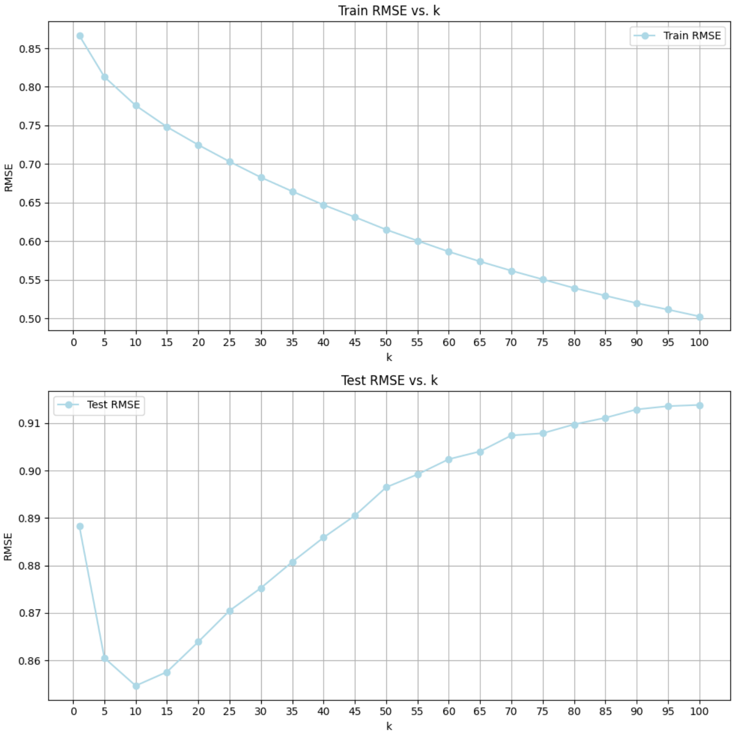

Figure 6: Training and Test Sample RMSE for Different Low Rank Representation r

B.2 Main Results

Entropy Calculation Since we have 500 movies in the action set (movie list), sampling to calculate $p_{i,A_j,t}$ (the probability of movie j being the optimal movie) takes a long time to compute. Therefore, in order to accelerate the simulation process, we revise the calculation by first selecting a random subset $\mathcal{S} \subset \mathcal{A}$, where $\mathcal{S}$ is of size s and we set $s = 50$. Then, following Algorithm 4, we calculate $p_{i,A_j,t}$ as the probability of movie j being the optimal movie in the subset $\mathcal{S}$. Finally, we calculate the entropy for the subset $\mathcal{S}_t$ at period t.

We consider several robustness checks/extensions considering how our results would change with respect to: (i) utility measurement instead of regret; (ii) the RS shutdown period; (iii) self-exploring user's preference parameter dimension d; (iv) self-exploring user's action set $\mathcal{A}$; (v) RS's low rank dimension r; (vi) self-exploring user has better prior information than RS; (vii) rational RS-dependent user with a search cost.

Other Learning Measurements In addition to entropy, we use the KL-divergence from the user's preference parameters' prior to measure learning. The prior belief on the preference vector $\theta_0 \sim N(\mu_0, \Sigma_0)$. $\mu_0 = 0$ and Σ_0 is an identity matrix. Suppose the posterior belief $\hat{\theta}$ at period t follows $N(\mu_{t+1}, \Sigma_{t+1})$. Then the KL-divergence between $\hat{\theta}$ and θ_0 is calculated as

Algorithm 4 Entropy Calculation for Action Selection with Subset Sampling

```
    Input: $\mu_{i,t}, \Sigma_{i,t}, \mathcal{A}, n, s$
    Output: $H(A^*_{i,t})$
Select a random subset $\mathcal{S} \subset \mathcal{A}$ of size $s$    ▷ Subset Selection
for $k = 1$ to $n$ do
    Initialize $count_{A_j} = 0$ for each $A_j \in \mathcal{S}$
    $\theta^{(k)}_{i,t} \sim N(\mu_{i,t}, \Sigma_{i,t})$    ▷ Distribution Sampling for Sample $k$
    $A_j \in \text{argmax}_{A \in \mathcal{S}} A^T \theta^{(k)}_{i,t}$    ▷ Action Selection for Sample $k$ from $\mathcal{S}$
    $count_{A_j} = count_{A_j} + 1$
end for
if $t = 1$ then
    $p_{i,A_j,t} = \frac{1}{|\mathcal{S}|}$ for each $j \in \mathcal{S}$    ▷ Uniform distribution for $t = 1$
else
    $p_{i,A_j,t} = \frac{count_{A_j}}{n}$ for each $j \in \mathcal{S}$    ▷ Frequency of selection as optimal action
end if
$H(A^*_{i,t}) = -\sum_{j \in \mathcal{S}} p_{i,A_j,t} \log_2(p_{i,A_j,t})$    ▷ Entropy calculation for subset $\mathcal{S}$
```

below:

$$KL(\mathcal{N}(\mu_{t+1}, \Sigma_{t+1}) \mid\mid \mathcal{N}(0, I)) = \frac{1}{2}\left(\text{tr}(\Sigma_{t+1}) + \mu_{t+1}^T \mu_{t+1} - k - \log(\det(\Sigma_{t+1}))\right)$$

where:

- $\text{tr}(\Sigma_{t+1})$ is the trace of the posterior covariance matrix Σ_{t+1},
- $\mu_{t+1}^T \mu_{t+1}$ is the dot product of the posterior mean vector μ_{t+1} with itself,
- k is the number of dimensions,
- $\log(\det(\Sigma_{t+1}))$ is the natural logarithm of the determinant of the posterior covariance matrix Σ_{t+1}.

Note that the determinant of the identity matrix is 1, and the logarithm of 1 is 0, which simplifies the last term.

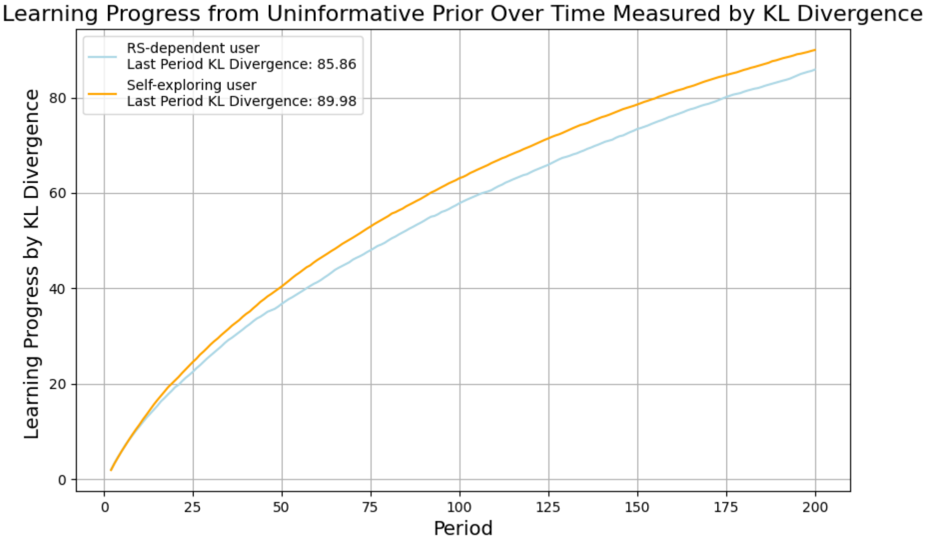

Figure 7: KL divergence from the uninformative prior over time for the self-exploring user and RS-dependent user without RS shutdown.

B.3 Main Results, Robustness Check and Extensions

Welfare Measurement Instead of Regret We plot Figure 3 using the expected utility measurement instead of the regret measurement in Figure 8. The pattern is the opposite of the regret pattern: Although utility is higher for the RS-dependent user when the recommendation system is available, this user has a lower utility after shutdown, which is due to insufficient learning. RS-dependent user's utility post shutdown is 99% (3.59/3.61) of the utility post shutdown compared to the self-exploring user. Though the difference in utility after RS shutdown is small, the difference in regret is huge as both self-exploring user and RS-dependent user are getting very close to the optimal movie.

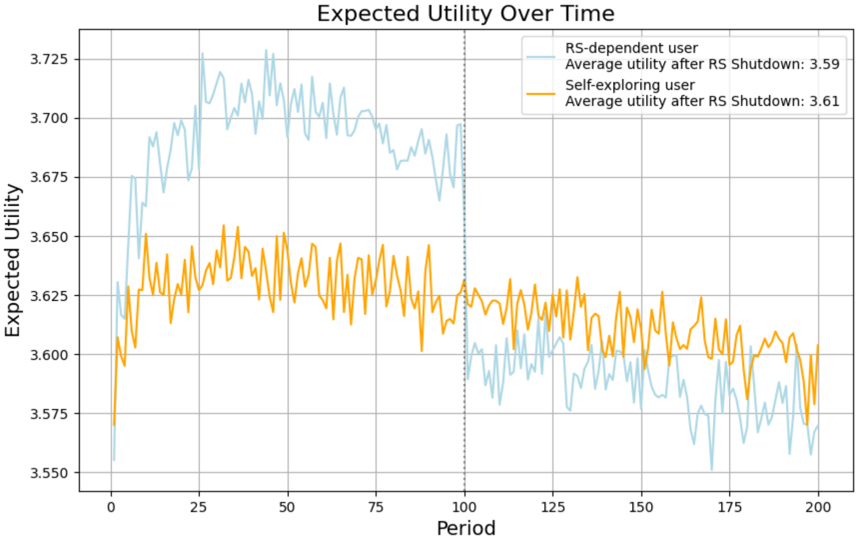

Figure 8: Expected utility over time for self-exploring user and RS-dependent user with RS Shutdown at Period 100

Real Rating not Imputed Rating Following Bresler et al. [2014], we only consider the top 200 users and the top 400 movies for the simulation, so we could use the real rating data not imputed rating. The resulting user-movie-rating matrix has 70% nonzero entries. The mean rating of this subsample is 3.73, which is slightly higher than the full sample (3.56). The data patterns are similar to our main results. We spilt the 200 users' sample into training sample (70%, 140 users) and test sample (30%, 60 users) and use the test sample for the cold-start problem simulation.

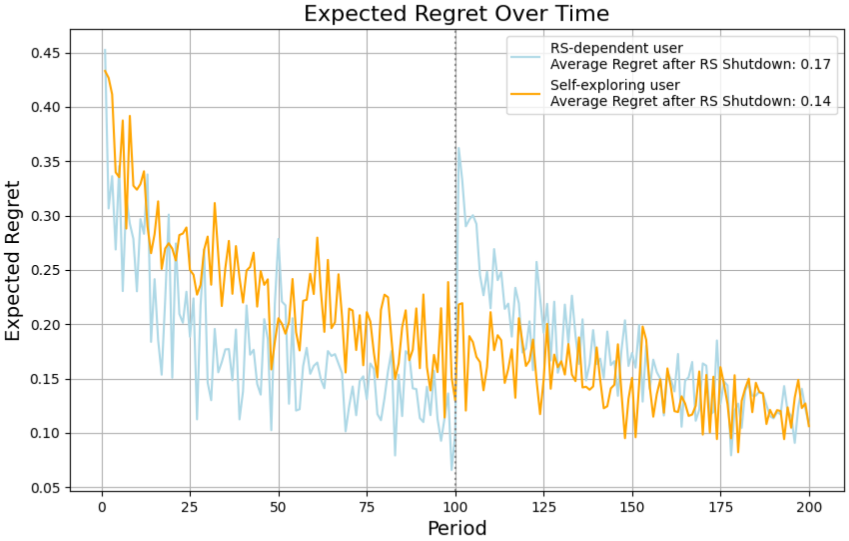

Figure 9: Expected regret over time for self-exploring user and RS-dependent user with RS Shutdown at Period 100

RS Shutdown Period As illustrated in our theoretical part, a RS-dependent user has lower regret before the RS shutdown but higher regret after, since they do not engage in sufficient learning. The later the shutdown event occurs, the higher the post-shutdown regret the RS-dependent user should face, as they rely on the system for a longer time. Figure 10 illustrates how the RS shutdown period (the timing of the shutdown event) affects the average expected regret of the RS-dependent user relative to that of a self-exploring user. In the left subplot, the RS-dependent user has lower regret compared with the self-exploring user across all RS shutdown periods (25, 50, 75, ..., 175). This pattern is consistent with that observed in Figure 1. In the right subplot, the RS-dependent user experiences higher regret relative to the self-exploring user. The data pattern illustrates that the problem of insufficient learning by the RS-dependent user is more severe when the user depends on the system for an extended period.

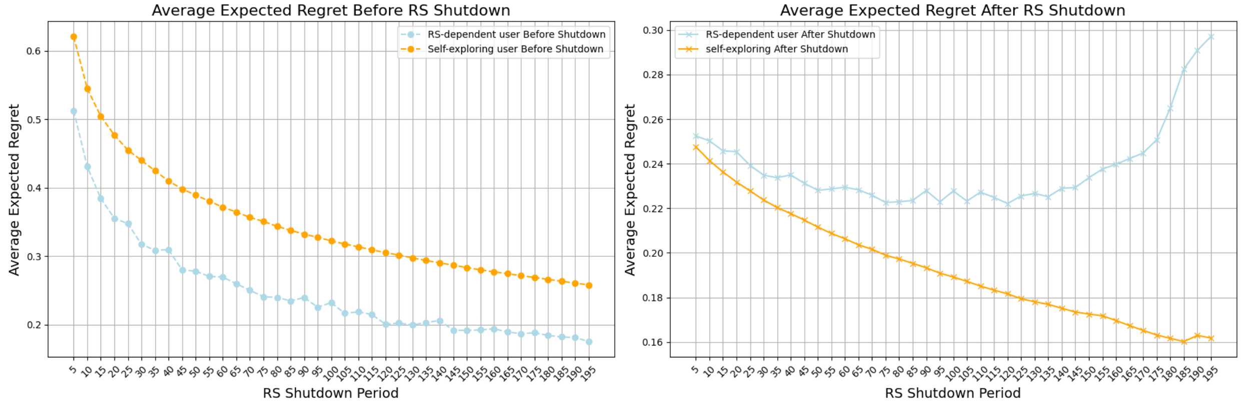

Figure 10: Expected regret for self-exploring user and RS-dependent user w.r.t RS shutdown period, 200 periods in total

Self-exploring User's Preference Parameter Dimension From the theory prediction, expected regret would increase with respect to d. When the movie system is getting more complicated, users may make more mistakes, thereby leading to higher regrets. Figure 14 shows how the preference parameter dimension d affects the self-exploring and RS-dependent users' regrets before and after RS shutdown. As d increases from 20 to 120, the RS-dependent user exhibits lower regret before RS shutdown but higher regret after. The self-exploring user's regret increases faster with respect to d compared with the RS-dependent user (Figure 14).[13]

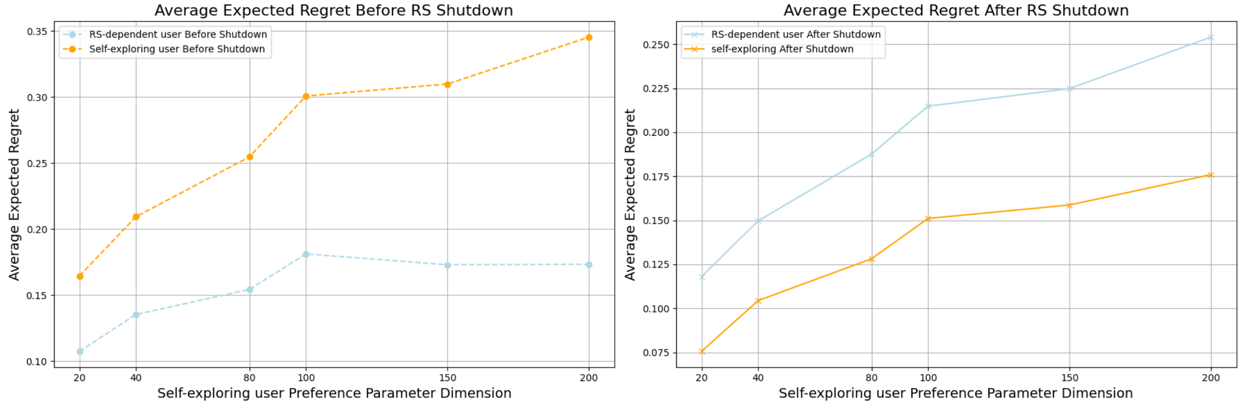

Figure 11: Expected regret for self-exploring user and RS-dependent user w.r.t self-exploring user's preference parameter dimension, 200 periods in total

[13] In the left subplot of Figure 14, we see that the RS-dependent user's regret also slightly increases with d. This is because, in our data simulation exercise, the ground truth lies at the dimension of d. When d increases, the system becomes more complicated and, therefore, the RS-dependent user also faces a slightly increasing regret path.

Self-exploring User's Action Set In the main simulation, we set the action set $\mathcal{A}$ to consist of 500 random movies for both the self-exploring user and the RS system. That is, a user is choosing a movie from a random subset of 500 movies, and the RS gives suggestions within the subset. However, in reality, a user may incur a large search cost and only check 20 or 50 movies. Therefore, we want to know how the size of the self-exploring user's action set affects our results. Specifically, we have the simulation exercise as below:

1. *Self-exploring user*: A self-exploring user picks a movie from an action set $\mathcal{A}$ with a size of 20, 50, ..., 1000 (movies). The action set $\mathcal{A}$ changes every period.
2. *RS*: The RS suggests a movie from all movies (3706 movies).
3. *RS-dependent user*: An RS-dependent user follows the RS's suggestion before the RS shutdown and picks a movie from the action set (same size as that of the self-exploring user) after the RS shutdown.

Figure 12 illustrates how expected regret changes with respect to the self-exploring user's action set. Our main result still holds: the RS-dependent user has lower regret before RS shutdown and higher regret after RS shutdown compared with the self-exploring user, despite the size of the self-exploring user's action set. The self-exploring user's regret increases when the action set sizes increase, but it becomes very stable when the action set is larger than 100 movies. Overall, the regret patterns do not change much with respect to the action set size. One possible explanation is that the action set is changed every period, so the randomness dominates the action set effect. It provides the implication that platforms may want to provide a variable subset of movies for the user to choose to encourage learning.

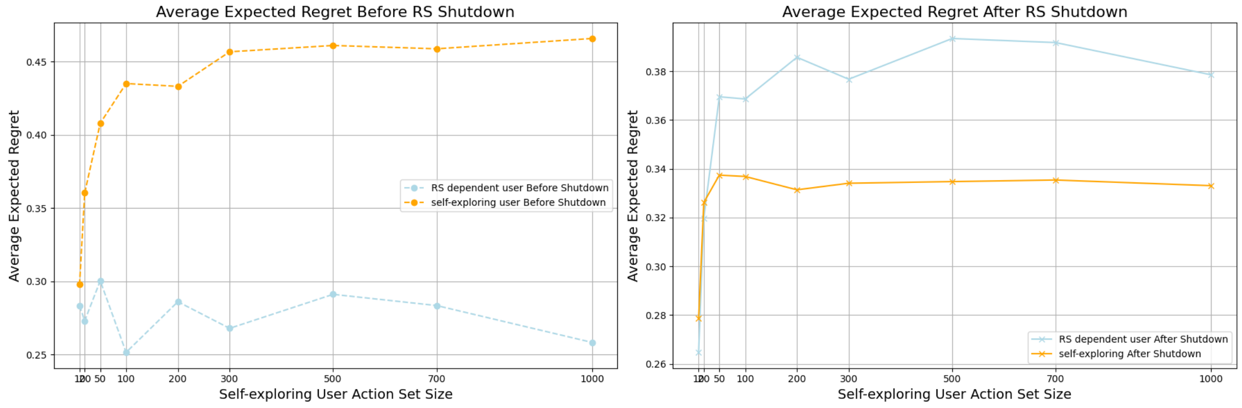

Figure 12: Expected regret for self-exploring user and RS-dependent user w.r.t self-exploring user's action set size, 200 periods in total

Self-exploring user has better prior information than RS We measure the information advantage of the self-exploring user as the periods she explored before the first period. We could understand the self-exploring user's better prior as she has already watched some movies in the past N periods. Figure 13 shows that the self-exploring user's before RS shutdown regret and after RS shutdown regret would go down if she has better prior information (more periods before the first period). Still, the RS's information advantage is large. Even if the self-exploring user has watched movies for 100 periods before the game, her average regret for the first 100 periods would be 0.27, still higher than the RS-dependent user's average regret 0.2. It shows that the RS's information advantage is huge and not easily overcome by the user's prior knowledge.

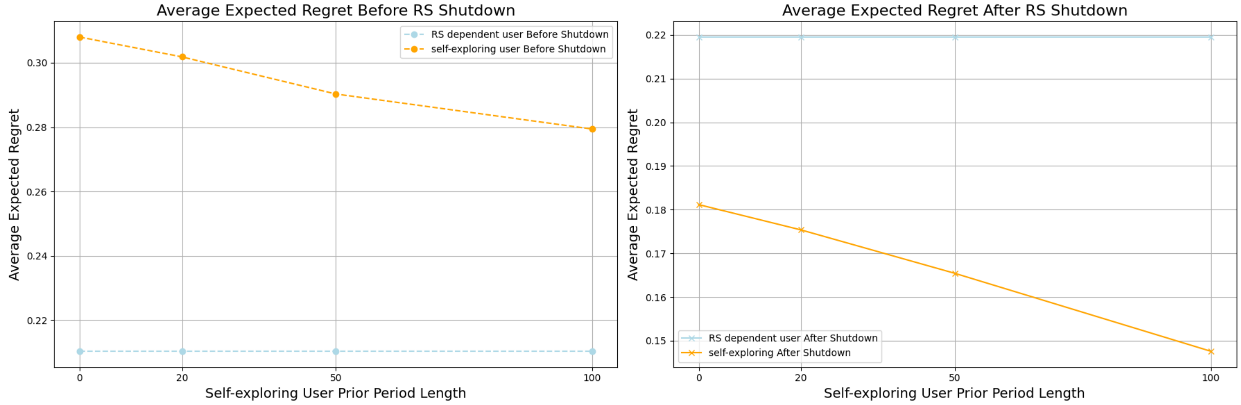

Figure 13: Expected regret for self-exploring user and RS-dependent user w.r.t self-exploring user's prior experience periods, 200 periods in total

RS's Low Rank Dimension As mentioned in Assumption 1, the RS has the information advantage so that it can decompose the user's preference weights $\Theta_{[d\times N]}$ into a lower-dimensional representation $\Gamma_{[r\times N]}$. However, it is not always the case that a smaller r leads to lower regret. When r is very small, the RS-dependent user would explore only minimally, which could lead to high pre-shutdown regret; the post-shutdown regret would also be high due to insufficient learning from the limited exploration. Conversely, when r is very large, the RS-dependent user may explore excessively, resulting in high pre-shutdown regret. However, the post-shutdown regret would be lower because of the more extensive learning from exploration.

Figure 14 illustrates how the RS-dependent user's expected regret changes with respect to r. In the left subplot, we see the RS-dependent user's before RS shutdown regret is the lowest when $r = 10$. The regret goes up when r is small ($r = 4$) or r is large ($r = 40$). Interestingly, the post RS-shutdown regret will go down when r is large. This is probably because a large r leads to more exploration when the RS is available. It gives us the potential policy implication that we could increase r a little higher than the optimal low rank representation to preserve enough learning for RS-dependent users.

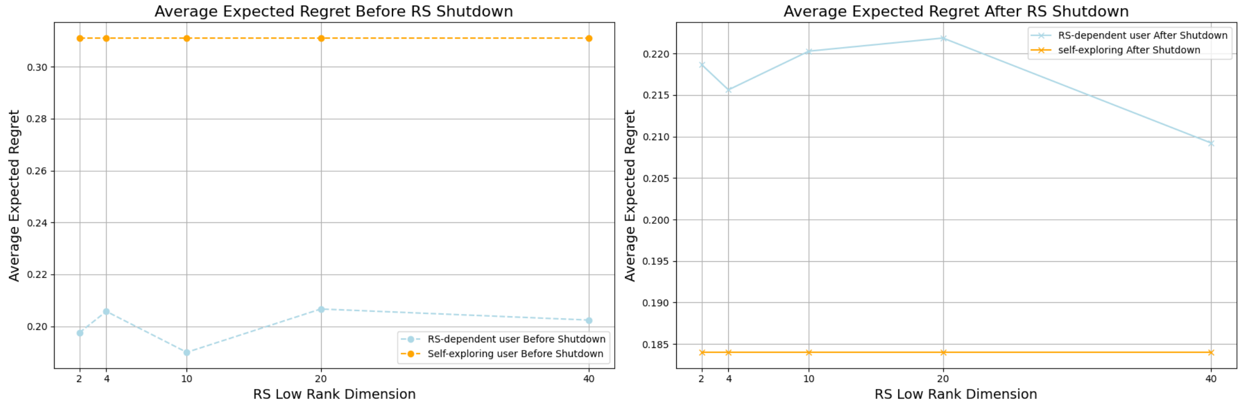

Figure 14: Expected regret for self-exploring user and RS-dependent user w.r.t RS's low rank dimension r, 200 periods in total

RS-dependent user has a search cost As discussed in § 3.3.2, if the RS-dependent user is rational and incurs a search cost when searching independently but no search cost when relying on the RS, then she faces a choice between searching on her own and following the RS every period. For the simulation, we have several assumptions as below:

1. Search cost is included in the welfare/regret and exists whenever the user needs to search by herself. The way we calculate user i's utility, expected utility, and expected regret from watching movie A_t defined as:

$$u_{i,j,t} = \theta_i^T \times A_{i,t} - c \times \mathbb{1}_{\{search\ by\ herself\}} + \epsilon_{it},$$
$$\mathbb{E}[u_{i,t}] = \theta_i^T \times A_{i,t} - c \times \mathbb{1}_{\{search\ by\ herself\}},$$
$$\mathbb{E}[\text{Reg}(i,t)] = \theta_i^T \times A_t^* - \theta_i^T \times A_{i,t} + c \times \mathbb{1}_{\{search\ by\ herself\}},$$

   where the search cost c reduces the expected utility and increases the expected regret when user needs to search by herself. The search cost term will not exist when the user follows the RS's suggestion.

2. *Self-exploring user*: everything as the main simulation but with the search cost c added to the expected regret.

3. *RS-dependent user*: The rational RS-dependent user i with a search cost c will choose movies and update her belief following Algorithm **??** below.

Algorithm 5 Choice and Learning for the Rational RS-Dependent Consumer with a Search Cost c

```
    Input: $\mu_{i,0}^{(\theta)}, \Sigma_{i,0}^{(\theta)}, \mu_{i,0}^{(\gamma)}, \Sigma_{i,0}^{(\gamma)}, \hat{F}, \mathcal{A}, \mathcal{T}$, c
    Output: $\mathcal{H}_{i,\mathcal{T}}, \{\mu_{i,t}^{(\theta)}, \Sigma_{i,t}^{(\theta)}\}_{t=1}^{\mathcal{T}}, \{\mu_{i,t}^{(\gamma)}, \Sigma_{i,t}^{(\gamma)}\}_{t=1}^{\mathcal{T}}$
for $t = 0 \to \mathcal{T}$ do
    $\hat{\gamma}_{i,t} \sim N(\mu_{i,t}^{(\gamma)}, \Sigma_{i,t}^{(\gamma)})$    ▷ RS: Distribution Sampling
    $A_{i,t}^{RS} \in \text{argmax}_{A \in \mathcal{A}} A^T \hat{F} \hat{\gamma}_{i,t}$    ▷ RS: Recommendation Selection
    $\hat{\theta}_{i,t} \sim N(\mu_{i,t}, \Sigma_{i,t})$    ▷ RS-dependent User: Distribution Sampling
    $A_{i,t}^{User} \in \text{argmax}_{A \in \mathcal{A}} A^T \hat{\theta}_{i,t}$    ▷ User: Action Selection
    $A_{i,t} \leftarrow \text{argmax} \left\{ (A_{i,t}^{\mathbf{User}})^T \hat{\theta}_{i,t} + c, (A_{i,t}^{\mathbf{RS}})^T \hat{\theta}_{i,t} \right\}$    ▷ User: Final Action
    $\Sigma_{i,t+1}^{(\theta)} \leftarrow \left( \left( \Sigma_{i,t}^{(\theta)} \right)^{-1} + \frac{1}{\sigma_\epsilon^2} A_{i,t} A_{i,t}^T \right)^{-1}$    ▷ Consumer: Belief Updating
    $\mu_{i,t+1}^{(\theta)} \leftarrow \Sigma_{i,t+1}^{(\theta)} \left( \left( \Sigma_{i,t}^{(\theta)} \right)^{-1} \mu_{i,t}^{(\theta)} + \frac{1}{\sigma_\epsilon^2} A_{i,t} U_{i,t} \right)$    ▷ Consumer: Belief Updating
    $\Sigma_{i,t+1}^{(\gamma)} \leftarrow \left( \left( \Sigma_{i,t}^{(\gamma)} \right)^{-1} + \frac{1}{\sigma_\epsilon^2} A_{i,t} A_{i,t}^T \right)^{-1}$    ▷ RS: Belief Updating
    $\mu_{i,t+1}^{(\gamma)} \leftarrow \Sigma_{i,t+1}^{(\gamma)} \left( \left( \Sigma_{i,t}^{(\gamma)} \right)^{-1} \mu_{i,t}^{(\gamma)} + \frac{1}{\sigma_\epsilon^2} A_{i,t} U_{i,t} \right)$    ▷ RS: Belief Updating
end for
```

Figure 15 shows the expected regrets of the self-exploring user and the RS-dependent user with respect to the search cost c. Before the RS shutdown, the RS-dependent user has relatively lower regret when the search cost is high compared to the self-exploring user. This occurs because, when the search cost is low, she may deviate from the RS's suggestion, leading to more mistakes. The result indicates that the RS benefits consumers more when the search cost is significant. After the RS shutdown, the expected regret of the RS-dependent user increases with the search cost c compared to the self-exploring user. This increase is due to the RS-dependent user's greater reliance on the RS when the search cost is high, leading to insufficient learning and, consequently, higher regret when the RS is no longer available.

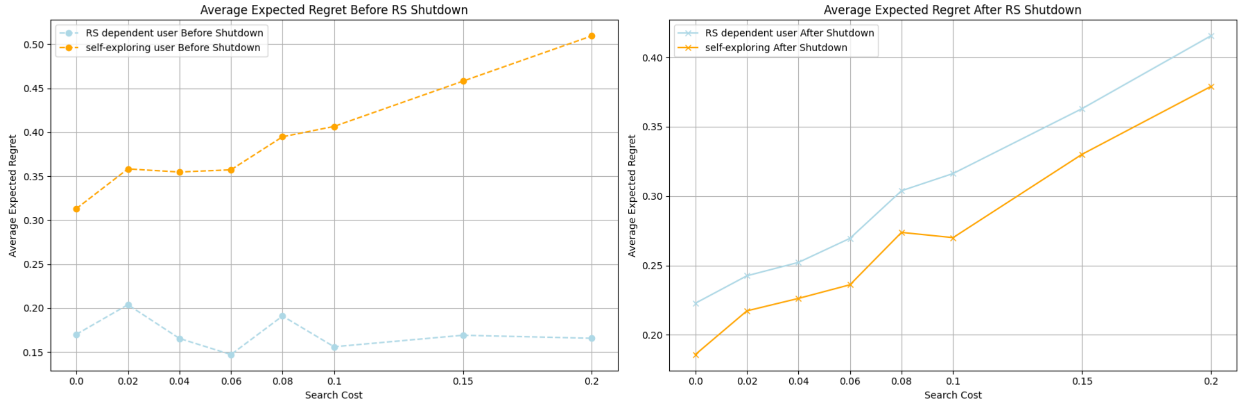

Figure 15: Expected regret for self-exploring user and RS-dependent user w.r.t RS's low rank dimension r, 200 periods in total

B.4 Policy Implications

Random Availability Policies The RS-dependent user chooses movies and updates her belief following Algorithm **??**.

Algorithm 6 Choice and Learning for the RS-Dependent Consumer with Stochastic RS Availability

```
Input: $\mu_{i,0}^{(\theta)}, \Sigma_{i,0}^{(\theta)}, \mu_{i,0}^{(\gamma)}, \Sigma_{i,0}^{(\gamma)}, \hat{F}, \mathcal{A}, \mathcal{T}, p$
Output: $\mathcal{H}_{i,\mathcal{T}}, \{\mu_{i,t}^{(\theta)}, \Sigma_{i,t}^{(\theta)}\}_{t=1}^{\mathcal{T}}, \{\mu_{i,t}^{(\gamma)}, \Sigma_{i,t}^{(\gamma)}\}_{t=1}^{\mathcal{T}}$
for $t = 0 \to \mathcal{T}$ do
    $\xi_{i,t} \sim \text{Bernoulli}(p)$                          ▷ Determine RS Availability
    if $\xi_{i,t} = 1$ then
        $\hat{\gamma}_{i,t} \sim N(\mu_{i,t}^{(\gamma)}, \Sigma_{i,t}^{(\gamma)})$            ▷ RS: Distribution Sampling
        $A_{i,t} \in \text{argmax}_{A \in \mathcal{A}} A^T \hat{F} \hat{\gamma}_{i,t}$        ▷ RS: Recommendation Selection
    else
        $\hat{\theta}_{i,t} \sim N(\mu_{i,t}, \Sigma_{i,t})$              ▷ User: Distribution Sampling
        $A_{i,t} \in \text{argmax}_{A \in \mathcal{A}} A^T \hat{\theta}_{i,t}$             ▷ User: Action Selection
    end if
    Refer to Algorithm 3 for belief updating steps.
end for
```

Self-regulated Policies The RS-dependent user chooses movies and updates her belief following Algorithm **??**.

The way we get the regulation term follows Algorithm 8:

Algorithm 7 Choice and Learning for the RS-Dependent Consumer under RS with Regulation

```
Input: $\mu^{(\theta)}_{i,0}, \Sigma^{(\theta)}_{i,0}, \mu^{(\gamma)}_{i,0}, \Sigma^{(\gamma)}_{i,0}, \hat{F}, \mathcal{A}, \mathcal{T}, \lambda\Delta_{(}H(A^*_{i,j,t}))$
Output: $\mathcal{H}_{i,\mathcal{T}}, \{\mu^{(\theta)}_{i,t}, \Sigma^{(\theta)}_{i,t}\}^{\mathcal{T}}_{t=1}, \{\mu^{(\gamma)}_{i,t}, \Sigma^{(\gamma)}_{i,t}\}^{\mathcal{T}}_{t=1}$
for $t = 0 \to \mathcal{T}$ do
    $\hat{\gamma}_{i,t} \sim N(\mu^{(\gamma)}_{i,t}, \Sigma^{(\gamma)}_{i,t})$    ▷ RS: Distribution Sampling
    $A^{RS}_{i,t} \in \text{argmax}_{A \in \mathcal{A}}\, A^T \hat{F} \hat{\gamma}_{i,t} - \lambda \times \Delta_{(}H(A^*_{i,j,t}))$    ▷ RS: Recommendation Selection after Regulation
    $A_{i,t} \leftarrow A^{RS}_{i,t}$    ▷ Consumer: Action Selection
    Refer to Algorithm 3 for belief updating steps.
end for
```

Algorithm 8 Change in Entropy Calculation Upon Movie Selection

```
Input: $\mu^{(\theta)}_{i,0}, \Sigma^{(\theta)}_{i,0}, \mathcal{A}, \mathcal{T}, n$
Output: $\Delta H(A^*_{i,j,t})$ for each period $t$ and movie $j$
for $t = 0 \to \mathcal{T} - 1$ do
    for each movie $j \in \mathcal{A}$ do
        Calculate $H(A^*_{i,t})$ using Algorithm 4 with $\mu^{(\theta)}_{i,t}$ and $\Sigma^{(\theta)}_{i,t}$
        Simulate the selection of movie $j$ at period $t$
        Update $\mu^{(\theta)}_{i,t+1}$ and $\Sigma^{(\theta)}_{i,t+1}$ based on the selection of movie $j$
        Calculate $H(A^*_{i,t+1,j})$ using updated beliefs $\mu^{(\theta)}_{i,t+1}$ and $\Sigma^{(\theta)}_{i,t+1}$
        $\Delta H(A^*_{i,j,t}) = H(A^*_{i,t+1,j}) - H(A^*_{i,t})$    ▷ Change in entropy from selecting movie $j$
    end for
    Record $\Delta H(A^*_{i,j,t})$ for each movie $j$
end for
```