ARTIFICIAL INTELLIGENCE, ALGORITHMIC RECOMMENDATIONS AND COMPETITION

EMILIO CALVANO
Università LUISS (Rome), Toulouse School of Economics and CEPR

GIACOMO CALZOLARI
European University Institute, Bologna University and CEPR

VINCENZO DENICOLÒ
Bologna University and CEPR

SERGIO PASTORELLO
Bologna University

Algorithmic recommendations increasingly influence consumer choices. We propose a method to identify and analyze their economic impact. Specifically, we borrow state-of-the-art recommender systems (RS) from the computer science literature and train them with synthetic data generated by a flexible model of consumer preferences and product differentiation. We demonstrate the usefulness of this framework by examining three debated issues. Firstly, we assess how algorithmic recommendations influence market concentration and consumer choice diversity. Secondly, we analyze their impact on equilibrium prices and consumer welfare, considering changes in consumer demand and hence accounting for firms' pricing incentives. Lastly, we investigate the potential for platforms to manipulate recommendations to prioritize profitability over product quality.

1. INTRODUCTION

Recommender systems (RSs) are AI algorithms that predict a user's potential interest in items they have not experienced. These predictions rely on users' feedback, such as ratings, clicks, or other kinds of user activity. This feedback is routinely gathered by digital platforms acting as intermediaries between consumers and suppliers of various products. The algorithm's predictions allow such platforms to provide personalized product recommendations that help consumers navigate the vast array of available products. [1]

The top-product signals generated by RSs enable consumers to initiate their search with products recommended specifically for them, creating *personalized prominence*. In this way, al-

Emilio Calvano: ecalvano@luiss.it
Giacomo Calzolari: giacomo.calzolari@eui.eu
Vincenzo Denicolò: vincenzo.denicolo@unibo.it
Sergio Pastorello: sergio.pastorello@unibo.it

We are grateful to seminars participants at Berkeley HAAS, Columbia, Harvard, University of Virginia, IMT Lucca, Rome Tor Vergata, EUI, Universitat Autònoma de Barcelona, CIVICA Data Science seminars, Aix-Marseille, Sciences Po Paris, Tel Aviv, Cattolica University Milan, Tsinghua University, Stockholm, the 2022-CLEEN online workshop, the CEPR IO Conference (Berlin), Toulouse School of Economics, the 2022-CCP Conference, the 14th Paris Conference on Digital Economics, and the ESEM/EEA 2023 conference in Barcelona for helpful comments. Within the last three years, Emilio Calvano has received research support from the European Research Council (AI-COMP Advanced grant), from University UCLouvain through a digital chair sponsored by Google and from the italian government (PRIN 2017).

[1] Recommender systems (RSs) have diverse applications beyond consumer product suggestions. Social media platforms use them to populate users' feeds, Spotify generates playlists, YouTube and Netflix recommend videos, financial institutions employ them for investor products (Robo-advisors), employment agencies match workers to employers, and more. In academia, they can help editors find suitable referees and suggest relevant articles to scholars.

gorithmic recommendations are already exerting a substantial influence on consumer demand (Jannach and Jugovac, 2019). With advancements in AI technology and the continuous accumulation of user data by platforms, the influence of such recommendations is expected to grow even further. In this paper, we propose a method to analyze the economic impact of algorithmic recommendations. We borrow state-of-the-art recommender systems (RS) from the computer science literature and train them with synthetic data generated by a flexible model of consumer preferences and product differentiation. This approach enables us to control the quality and quantity of information provided to the RS. Within this framework, we conduct a large number of "experiments"comparing scenarios where consumers receive personalized recommendations with a benchmark scenario where they rely solely on unassisted search.

We illustrate the possible applications of this framework by addressing three issues that have been widely debated among policymakers and scholars alike. First, we analyze the impact of algorithmic recommendations on market concentration and the diversity of consumer choices: do RSs help consumers discover niche products that would otherwise go unnoticed, or do they generate a rich-get-richer dynamic where a few popular items are disproportionately promoted by the algorithms, resulting in more highly concentrated markets?[2] Second, we analyze the effect of RSs on equilibrium prices and consumer welfare. By altering the way consumers conduct their search, algorithmic recommendations change consumer demand and hence firms' pricing incentives, even if firms cannot engage in price discrimination. At the same time, the recommendations may improve the matching between consumers and products and reduce costly search. The overall impact on consumer welfare is *a priori* uncertain. Third, we explore the possibility that platforms might manipulate their recommendations to promote more profitable products at the expense of genuinely superior ones.

Methodology. Before providing a preview of our results, it may be useful to describe our methodology in more detail. We aim to examine algorithms that closely resemble those commonly employed in real-world scenarios in terms of complexity and data availability. To this end, we follow an approach common in existing computer science and marketing literature, which involves deploying such algorithms in a computer-simulated marketplace and studying their behavior numerically. However, we depart from these literatures by embedding this analysis in an economically consistent model of preferences, product differentiation, and consumer search.

Specifically, we focus on latent factor collaborative-filtering RSs, a class of algorithms popular in computer science and presumably widely used in practice. We describe these algorithms in detail in Section 2. For now, suffice it to say that they aggregate and scrutinize feedback from numerous consumer-product pairs to discern patterns of similarity and dissimilarity among consumers and products.

The "collaborative" nature of the algorithms implies that if consumer preferences were entirely idiosyncratic, as often assumed in the search literature, any observed correlations in the data would be coincidental, rendering recommendations valueless if not misleading. Therefore, it becomes crucial to adopt a model of consumer preferences and product characteristics that incorporates systematic similarities and differences among consumers and products, which the algorithms can leverage to generate valuable predictions. At the same time, the analysis of the issues described above requires a model that displays important asymmetries, allowing for the co-existence of niche and mass products, as well as consumers with common and more eccentric preferences.

[2]Similar issues also arise in other settings. For example, it has been contended that the use of RSs may cause political polarization in social media and a loss of diversity in culture. See, for instance Abdollahpouri and Mansoury (2020), Abdollahpouri et al. (2021).

We develop a model of product differentiation that exhibits these properties in Section 3 and use it to generate an artificial marketplace. We then train the algorithms using synthetic data produced within this framework, replicating the quantity and quality of information contained in real-world datasets. Once trained in this manner, the algorithms can generate personalized recommendations, enabling us to numerically analyze how these recommendations affect consumer search and firm pricing. This approach allows us to assess the quality of the recommendations, study their biases, and analyze how they change as one modifies the level of information that the platform has access to.

We acknowledge that this approach may raise concerns about external validity. To address these concerns, we conduct an extensive robustness analysis. While there are limits to the number of variations we can explore, our methodology is flexible and enables the examination of settings beyond the initial scope. Additionally, it offers a practical test-bed environment for policymakers and platforms to assess the effects of various recommender systems in diverse economic contexts.

Results. We find, firstly, that RSs tend to favor mass products over niche products. The reason for this is that algorithmic recommendations tend to align with the preferences of the median consumer, creating what we call a “uniformity” effect. In other words, the algorithm estimates greater consumer similarity than actually exists and recommends products that align with the preferences of the median consumer too frequently. Furthermore, sometimes RSs tend to create their own “champions” without any clear objective basis for favoring one product over others. Overall, these effects lead to heightened market concentration. These biases disappear only when the amount and quality of data that the algorithm has access to become unrealistically high.

Secondly, we find that RSs prompt firms to raise prices, even if price discrimination is ruled out by assumption. In some cases, prices uniformly rise for all products; in others, a combination of price increases and decreases occurs and the overall increase in the average price is primarily driven by a composition effect. Still, the increase in prices always harms consumers. Compared to a scenario where algorithmic recommendations are not available, however, RSs also improve the matching between consumers and products and reduce the need for costly searches. As a result, their impact on consumer surplus is generally uncertain. Our analysis reveals that under reasonable parameter values, RSs tend to result in higher overall consumer surplus. Our methodology allows us to investigate various market scenarios, and show that a decline in consumer surplus occurs specifically under conditions of limited product availability, predominantly horizontal product differentiation, and elevated search costs.

Although consumers are generally better off with RSs compared to their absence, we also show that as algorithms gain access to more and better data, consumer surplus may eventually decrease. In other words, the relationship between information and consumer welfare may follow an inverted-U shaped curve. Initially, consumer surplus rises as the algorithm’s information level increases and consumers are better able to find products that meet their preferences. However, higher levels of information also lead to increased prices. Beyond a certain point, this negative effect prevails, and more information leads to a decrease in consumer surplus. This pattern arises regardless of the reason for the variation in information levels, whether it be a change in the quantity of data, a change in noise, or other factors.

Finally, we find that when platforms manipulate their recommendations, the prices for over-recommended products tend to decrease. This mitigates the negative impact of such practices on welfare and restricts their profitability, implying that the profit-maximizing rate of manipulation may be relatively small. The most significant impact of manipulation appears to be the decrease in profits experienced by competitors of the favored products. This observation suggests that the manipulation of recommendations may be more indicative of exclusionary practices rather than exploitative ones.

Related literature. We discuss more fully the relationships between each set of results and the relevant literatures in the subsequent sections, where we present the results in greater detail. For now, let us just mention that, to the best of our knowledge, only two papers in the economics literature specifically address the problem of statistical estimation faced by the algorithms when there is information only on a small fraction of user-item pairs: Lee and Wright (2021) and Castellini et al. (2023).[3] Our study differs from these papers in that it focuses specifically on the impact of RSs on product market competition and embeds the analysis in a fully-fledged framework of individual search.[4]

More broadly, this paper contributes to the recent literature on the implications of AI for industrial organization. Most of this literature has, so far, focused specifically on the issue of algorithmic collusion.[5] The impact of algorithmic recommendations on product market competition remains largely unexplored.

Structure of the paper. The rest of the paper is divided into two parts. In the first part, we present our analytical framework. Section 2 offers a self-contained introduction to the latent-factor, collaborative-filtering algorithms utilized. Section 3 presents our model of product differentiation. It is a model of the "address" variety that, for consistency, mirrors the latent-factor structure of the algorithms. The model encompasses different combinations of horizontal and vertical product differentiation, as well as both "niche" and "mass" products. Section 4 describes the data the algorithms use and how they are trained. Section 5 introduces the search theoretic framework that we use to model user behavior. Finally, Section 6 specifies the baseline parameterization of the model in preparation for the numerical analysis and describes a number of robustness checks that we have conducted.

The second part of the paper presents our substantive findings. Section 7 examines the impact of RSs on market concentration, while Section 8 analyzes the effect of RSs on equilibrium prices. In Section 9, we analyze the consequences of varying the amount and quality of information, demonstrating an inverted-U relationship between information and consumer welfare. Section 10 analyzes the case where the platform manipulates its recommendations. In the concluding section, we discuss potential extensions to our work and the policy implications of our findings. An online appendix includes details that are omitted from the main text. Extensions of narrower interest are included in supplementary material Calvano et al. (2023)[6]

[3] See also the empirical work of Lee and Musolff (2023), who empirically uncover price effects of algorithmic recommendations that align with the findings of our analysis.

[4] In contrast, Lee and Wright (2021) assess the information value of RS algorithms by comparing them to purely random choices, whereas Castellini et al. (2023) use complete information as their benchmark.

[5] Initially, the literature on algorithmic collusion adopted an experimental approach similar to that used in this paper: see, for instance, Calvano et al. (2020), Johnson et al. (2023), Asker et al. (2023), and Klein (2021). More recent work has added empirical evidence (Assad et al., forthcoming) and theoretical results (Possnig, 2023, Banchio and Mantegazza, 2023).

[6] This supplementary material is hosted on Harvard University's online repository servers and is accessible via search on the Harvard Dataverse (https://dataverse.harvard.edu/) or using its DOI link: https://doi.org/10.7910/DVN/KBDJOW

2. MODEL-BASED RECOMMENDER SYSTEMS

We focus on latent-factor, collaborative-filtering systems, a class of algorithms that includes the winner of the Netflix Prize.[7] According to (Aggarwal, 2016, p. 91), these algorithms are "considered to be state-of-the art in recommender systems," (see also Rokach et al., 2022) and as such, they are likely to be widely used in practice.

2.1. *The basic problem*

Consider a finite set of users $i = 1, 2, ..., I$, a finite set of items $j = 1, 2, ..., J$ and a sparse matrix $\tilde{\mathbf{R}}$ of size $I \times J$ which is commonly referred to as the 'rating' matrix in the literature (we adopt this terminology throughout). Existing entries $\tilde{r}_{ij}$ represent the rating of user i for item j when such rating is either reported or inferred from user behavior, and the entries for the unobserved user-item pairs are missing. (Economists naturally interpret these ratings as indicative of the utility user i derived from item j.) The objective of the algorithm is to estimate the full matrix of true ratings, denoted as $\mathbf{R}$.

Collaborative filters re-estimate the observed values and fill in the missing ones exploiting the correlation structure of observed ratings. The underlying assumption is that users whose observed ratings are similar are likely to have similar ratings for unobserved items, while items whose ratings are similar across observed users are likely to be rated similarly by unobserved users.

2.2. *Latent factors*

We focus on an important class of collaborative filtering algorithms, namely, latent-factor RSs. These algorithms explain the observed user-item ratings, and predict the unobserved ones, using hidden (latent) factors that represent inherent properties of items and users. This approach reduces the dimensionality of the rating matrix by representing ratings in terms of relatively few variables.

Analytically, the true rating of item j by user i is viewed as the inner product of a vector of user-specific parameters $\mathbf{t}_i$ and a vector of item-specific parameters $\mathbf{v}_j$:

$$r_{ij} = \sum_{h=1}^{H} t_{ih} v_{jh}, \tag{1}$$

where H is the number of latent factors. In matrix notation, $\mathbf{R} = \mathbf{T}\mathbf{V}'$, where $\mathbf{T}$ and $\mathbf{V}$ are the $(I \times H)$ and $(J \times H)$ matrices formed by the I vectors $\mathbf{t}_i$ and the J vectors $\mathbf{v}_j$, respectively.

The variable t_{ih} may be thought of as user i's proclivity for factor h, and the variable v_{jh} as product j's affinity to factor h. In applications, the H factors may have a semantic interpretation. For the algorithm, however, they need not have any specific meaning.

[7] In 2006, Netflix launched a one million dollar prize for the first RS that could improve the performance of Cinematch, their algorithm for predicting ratings, by at least 10%. The challenge provided participants with a rich data set. After three years, the prize was awarded in 2009 to BellKor's Pragmatic Chaos team for improving Netflix's algorithm by 10.06%. A new challenge in 2010 was discontinued due to concerns that the anonymity of the data provided had been breached. For more details, see Koren et al. (2009) .

2.3. *Estimation procedure*

The algorithm estimates the parameters $\hat{\mathbf{T}}$ and $\hat{\mathbf{V}}$ by minimizing some measure of the distance between the estimated ratings,

$$\hat{r}_{ij} := \sum_{h=1}^{H} \hat{t}_{ih}\hat{v}_{jh} \tag{2}$$

and the observed ones, $\tilde{r}_{ij}$. Using the Euclidean distance, the estimated parameters $\hat{\mathbf{T}}$ and $\hat{\mathbf{V}}$ solve:

$$\min_{\hat{\mathbf{T}},\hat{\mathbf{V}}} \sum_{(i,j)\in S} \left(\tilde{r}_{i,j} - \sum_{h=1}^{H} \hat{t}_{ih}\hat{v}_{jh}\right)^2, \tag{3}$$

where S denotes the set of all pairs (i,j) for which r_{ij} is (imperfectly) observed.[8] Using the completed matrix $\hat{\mathbf{R}} = \hat{\mathbf{T}}\hat{\mathbf{V}}'$, the algorithm can generate rankings of all items for all users, allowing it to provide personalized recommendations.

It is important to note that in collaborative-filtering RSs, attributes and tastes are estimated *jointly*, which marks a departure from much of the structural empirical work on demand estimation based on the Random Utility Model. Unlike in those estimations, where it is typically assumed that the choice attributes (i.e., the v_{jh}s) are observable while preferences (i.e., the t_{jh}s) are not, RSs observe neither product attributes nor consumer tastes and estimate both.[9]

In real-life applications, the density d of the matrix $\tilde{\mathbf{R}}$ of observed rating is quite low, often below 2%.[10] Consequently, estimates obtained from such limited data may exhibit small sample biases. In particular, the estimation procedure (3) may cause overfitting. A common approach for addressing this problem is to use regularization techniques. The idea is to penalize large values of the coefficients in the matrices $\hat{\mathbf{T}}$ and $\hat{\mathbf{V}}$ so as to decrease the variance of $\hat{\mathbf{R}}$. The actual minimization problem then becomes:

$$\min_{\hat{\mathbf{T}},\hat{\mathbf{V}}} \sum_{ij\in S} \left(\tilde{r}_{i,j} - \sum_{h=1}^{H} \hat{t}_{ih}\hat{v}_{jh}\right)^2 + \lambda_t \sum_{i=1}^{I}\sum_{h=1}^{H} \hat{t}_{ih}^2 + \lambda_v \sum_{j=1}^{J}\sum_{h=1}^{H} \hat{v}_{ih}^2, \tag{4}$$

where the non-negative parameters λ_t and λ_v are the regularization weights. The weights are chosen by a regularization procedure described in the online appendix.[11]

3. PRODUCTS AND PREFERENCES

In this section, we present a model of consumer preferences and product characteristics that mirrors the latent-factor structure employed by the algorithms. This ensures that the estimation can be based on a correctly specified model of the economic environment, and in cases

[8]Problem (3) assumes that the algorithm knows the true number of latent factors H, as we do in our baseline analysis. However, in a robustness check, we let the algorithm estimate the value of H by an internal cross-validation routine, as described in the online appendix. All results are nearly unchanged.

[9]The case where some attributes are observable by the RS can be handled by using hybrid RSs (also known as ensemble RSs). By symmetry, these algorithms can also deal with the case where consumer tastes are partly observable.

[10]The *density* of a sparse matrix is the fraction of non-empty cells, and the *sparsity* is the fraction of empty cells. Therefore, *density* $= 1-$ *sparsity*.

[11]The appendix also shows that our results are robust to the choice of these weights.

of mis-specification, it allows us to control its nature and extent. The model is rich enough to accommodate the coexistence of niche and mass products, as well as consumers with both common and more eccentric preferences.

3.1. *A latent-factor model of product differentiation*

We examine a monopolistic platform that serves as an intermediary between I buyers (indexed by $i = 1, 2, ..., I$) and the sellers of J products (indexed by $j = 1, 2, ..., J$). Consumer i's willingness to pay for product j, in monetary units, consists of a deterministic component $\bar{u}_{ij}$ that is specific to each product-consumer pair, along with an idiosyncratic shock ε_{ij}:

$$u_{ij} = \bar{u}_{ij} + \varepsilon_{ij}. \tag{5}$$

The shocks ε_{ij} are normally i.i.d. with zero mean and variance σ_ε^2. (They serve to smooth out the perceived demand functions, guaranteeing the existence of a price equilibrium in pure strategies.) The systematic component, on the other hand, is assumed to be:

$$\bar{u}_{ij} = \sum_{h=1}^{H} t_{ih} v_{jh}. \tag{6}$$

Note the analogy with the structure of the ratings r_{ij} that the algorithm tries to predict. From an economic viewpoint, H represents the number of characteristics of the products that are valued by consumers (akin to different dimensions of product quality), $v_{jh} \geq 0$ the level of the h-th characteristic in product j, and $t_{ih} \geq 0$ the value that consumer i attaches to it. It therefore appears that this latent-factor formulation is consistent with the Lancastrian approach, which posits that consumers derive "satisfaction from characteristics that [...] cannot be purchased directly, but are incorporated in goods" (Lancaster et al., 1974, p. 567).

We assume that the sum $\sum_{h=1}^{H} t_{ih}^2$ is constant across consumers, so all consumers have the same total willingness to pay for quality. The interpretation is that consumers may be heterogeneous in tastes but not in income. This assumption makes the model trivial if $H = 1$; therefore, we focus on the case $H \geq 2$. With no further loss of generality, we can normalize the total willingness to pay to one: $\sum_{h=1}^{H} t_{ih}^2 = 1$.

3.1.1. *Horizontal differentiation*

The case where product differentiation is purely horizontal is obtained when all goods have the same "total" quality $\sum_{h=1}^{H} v_{jh}^2$, which can then also be normalized: $\sum_{h=1}^{H} v_{jh}^2 = 1$. In this case, both products and consumers can be represented as points on the portion of the unit hyper-sphere that lies in the positive orthant.[12] Specifically, we assume that both the J products and the I consumers are equally spaced (see Figure 1).

For each possible consumer type (i.e., for each vector $\mathbf{t} = (t_1, ..., t_H)$), there is a different ideal product $\mathbf{v} = (v_1, ..., v_H)$, which under horizontal differentiation is $\mathbf{v} = \mathbf{t}$. When consumers purchase a product that differs from their ideal product, they incur a "transportation cost" akin to that considered in the standard Hotelling model. For example, when $H = 2$, using polar coordinates $\{t_{i1} = \cos\theta, t_{i2} = \sin\theta\}$ and $\{v_{j1} = \cos\upsilon, v_{j2} = \sin\upsilon\}$, it is easy to see that the transportation cost is $1 - \cos(\theta - \upsilon)$.[13]

[12] With two factors, the model becomes similar to Wolinsky (1983), with the twist that products and consumers are restricted to the non-negative portion of the unit circle.

[13] See the online appendix for more details and a comparison with a standard quadratic transportation cost function. Note that in this model the transportation cost is a convex function of the distance between products and consumers, and there is no explicit transportation cost parameter.

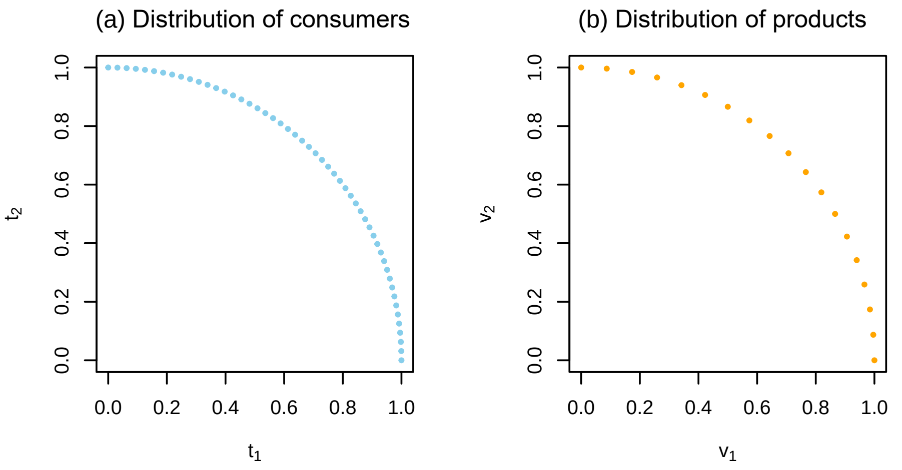


FIGURE 1.—The distribution of consumers and products in the two-factor model with purely horizontal differentiation.

The restriction to the non-negative orthant implies that products are not symmetric even if product differentiation is purely horizontal. To see this, consider for instance the case of three products. Clearly, the *central product*, located on the 45-degree line, enjoys higher demand compared to the two peripheral ones. Thus, it qualifies as a natural candidate for becoming a "mass" product. Conversely, the *peripheral products*, which are located on the x- and y-axes in Figure 1, represent the "niche" products. The presence of both mass and niche products enables us to study whether RSs lead to a "superstar effect" or a "long-tail effect." The issue will be taken up in section 7.

3.1.2. *Vertical differentiation*

The assumption that product differentiation is purely horizontal can be relaxed, by allowing for the possibility that the quality index $\sum_{h=1}^{H} v_{jh}^2$ may vary across products. This extension introduces vertical differentiation into the model.

For example, a model of pure vertical differentiation is obtained by assuming that products are located on the square rather than on the circle (Figure 2, right panel). In this case, all consumers agree on their favorite product, which lies on the 45-degree line.

Intermediate cases between purely horizontal and purely vertical differentiation can be obtained by varying the shape of the product locus from a circle to a square. This can be done parsimoniously, by means of a single parameter α that ranges from $\alpha = 0$ (circle, horizontal differentiation) to $\alpha = 1$ (square, vertical differentiation). To be precise, for each product $\mathbf{v}_j$ on the circle, define its radial projection on the square as $\mathbf{v}'_j = \frac{\mathbf{v}_j}{\sqrt{2}\max_h [v_{jh}]}$. By taking the convex combination $\alpha \mathbf{v}_j + (1-\alpha)\mathbf{v}'_j$ of the circle and the square, one can then generate all intermediate combinations of vertical and horizontal differentiation.

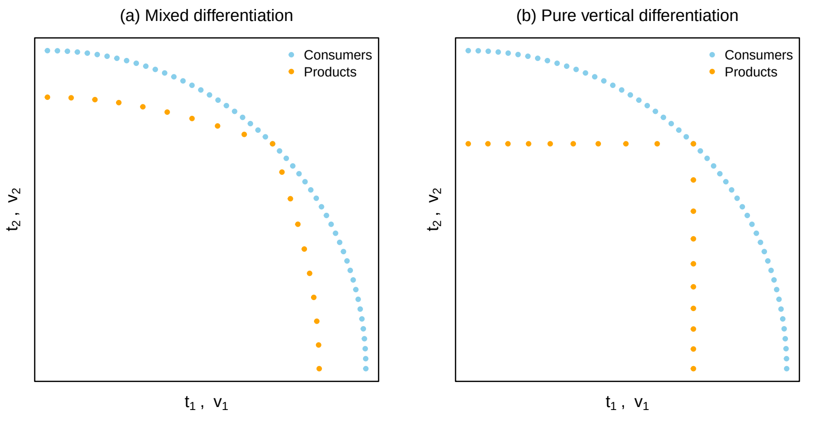


FIGURE 2.—The distribution of products under vertical, or mixed, differentiation.

As soon as $\alpha > 0$, a bunch of consumers (i.e., those close to the 45 -degree line) will prefer the central product. This is an additional reason why the central product stands out as a candidate mass product.[14]

4. DATA

After purchasing a product, consumers may explicitly or implicitly report a rating to the platform. These ratings are the data used by the algorithms to estimate the match values.

In the baseline scenario, we assume that the reported rating $\tilde{u}_{ij}$ is equal to the utility actually experienced, u_{ij}, plus a noise term:

$$\tilde{u}_{ij} = u_{ij} + \epsilon_{ij}. \tag{7}$$

The reporting noise ϵ_{ij} is normally i.i.d. with zero mean and variance σ_ϵ^2. The higher the variance σ_ϵ^2, the less informative the ratings reported by consumers.

Later on, we will consider extensions where the report is made on a Likert scale. It is important to note that ratings on a two-level Likert scale may be available to a platform even without explicit reporting by users. For instance, the platform could monitor whether a consumer has purchased an inspected product, or watched a movie until the end.

4.1. *Randomly generated data*

As noted, RSs typically have access to information about only a small fraction of all consumer-product pairs. What determines which ratings are observed and which are not?

We consider two alternative data generation processes, depending on whether the data have been generated based on recommendations provided by the algorithm itself in the past

[14]In principle, the flexibility of the model permits the consideration of scenarios where the highest-quality product does not necessarily coincide with the central product. Geometrically, this can be achieved by employing a rectangle instead of a square. Additionally, it is also possible to explore situations where peripheral products possess higher quality. However, these possibilities are not pursued here.

or not. In our baseline analysis, we assume that the algorithm itself does not play a role in generating the data it uses. Specifically, we assume that for each user, the platform observes ratings for a fixed number of products, drawn randomly and independently across users. The fraction of consumer-product pairs for which ratings are observed corresponds to the density d of the matrix $\tilde{\mathbf{R}}$ introduced in Section 2.

4.2. *Endogenous data*

Alternatively, we consider the case where the ratings observed by the platform have been created based on recommendations provided by the algorithm itself in the past. In this case, the matrix $\tilde{\mathbf{R}}$ is filled in gradually, in a number of successive steps indexed by $\tau = 0, 1, 2, \ldots$. Initially, $\tilde{\mathbf{R}}_\mathbf{0}$ is the empty matrix; that is, a matrix with all entries missing. At step $\tau = 1$, each consumer tries a randomly drawn product resulting in ratings $\tilde{u}_{ij}$ for the RS. The matrix $\tilde{\mathbf{R}}_\mathbf{1}$ therefore has a density of $1/J$.

At each subsequent step, the RS uses $\tilde{\mathbf{R}}_\tau$ to estimate $\hat{\mathbf{R}}_\tau$ according to the estimation procedure described in Section 2 and recommends to each consumer i the product with the highest rating:

$$j^*(i,\tau) = \arg\max_j \hat{r}_{ij\tau}. \tag{8}$$

This product is then tried by the consumer,[15] and the corresponding rating reported to the platform. Thus, $\tilde{\mathbf{R}}_{\tau+\mathbf{1}}$ is obtained by adding to $\tilde{\mathbf{R}}_\tau$ one and only one entry $\tilde{u}_{ij^*(i,\tau)}$ for each of the I rows (i.e. for each consumer). If $\tilde{u}_{ij^*(i,\tau)}$ is not missing in $\tilde{\mathbf{R}}_\tau$, its value is overwritten. All other elements of $\tilde{\mathbf{R}}_\tau$ are unchanged. Thus, $\tilde{\mathbf{R}}_{\tau+\mathbf{1}}$ is basically equal to $\tilde{\mathbf{R}}_\tau$ plus one new entry per consumer, which could be either a previous value overwritten or a brand new one.

As this process unfolds, the density of $\tilde{\mathbf{R}}_\tau$ increases. The algorithm's learning phase ends when the matrix $\tilde{\mathbf{R}}$ achieves a pre-specified density d.[16] At this juncture, we analyze the effects of the recommendations ultimately generated by the algorithm.

Data endogeneity creates a feedback loop extensively discussed in the computer science literature: the algorithm gathers better information on the user-item pairs it has selected in the past. This is often referred to as the "bias in the algorithms" (even though a more precise designation might be "bias in the data"). This bias emerges because, even if the final density matches that of random data, the user/item pairs observed by the algorithm are somewhat correlated, diminishing the quality of the available information. Consequently, the precision of estimates and the quality of recommendations decline.

5. SEARCH

We now integrate the above model into a search framework. This allows us to conceptualize recommendations as generating a form of personalized prominence, meaning that con-

[15]Note that during the learning phase, for simplicity, we abstract from the possibility that consumers may disregard the recommendation and engage in their own search. However, this possibility plays a crucial role in our analysis of the effects of recommendations once the learning phase has been completed, as we will see below.

[16]In practice, to achieve a target density level, it may be necessary to assume that the algorithm operates in "exploration mode" during a certain percentage of the periods; otherwise, the process might reach a steady state with a lower density. During the exploration mode, the algorithm recommends products randomly. By adjusting the probability of being in the exploration mode, we can generate data that lie between purely exogenous and endogenous. In our simulations, the case of endogenous data actually corresponds to a probability of being in exploration mode as low as possible, typically 10 percent.

sumers can begin their search with products specifically recommended for them, rather than from a randomly chosen one.

5.1. *Product markets*

Before proceeding, it is important to note that although the willingness to pay for any two of the J products is correlated to some extent, not all products traded on the platform necessarily serve as direct substitutes for one another. For example, the platform may offer both movies and books. Individuals who enjoy watching war movies may also like historical novels, while those who prefer wildlife documentaries may lean towards science books. The algorithm benefits from pooling all products together to leverage the correlation in consumers' evaluations. Nevertheless, books and movies belong to distinct markets.

To account for this, we partition the set of all J products into distinct subsets (markets) $\mathcal{M}$ and assume that each consumer buys at most one product from each market $\mathcal{M}$. We denote the number of products belonging to market $\mathcal{M}$ as $m_{\mathcal{M}}$.

5.2. *Prior information*

In general, consumer behavior in a market is shaped by the information individuals have regarding product availability, prices, and match values. Given that the specific role of RSs is to estimate the match values, we evaluate their impact by *holding other information constant*.

Different assumptions may be made regarding this other information. Following Wolinsky (1986) and Anderson and Renault (1999), in our analysis we assume that consumers are aware of product availability but do not possess knowledge about individual prices and match values;[17] they only know the distribution of product characteristics and equilibrium prices.[18] Acquiring information on individual prices and match values necessitates costly search.

Based on their prior information and knowledge of their own preferences, each consumer i calculates the probability distribution, denoted as $F_{i,\mathcal{M}}(s_i)$, for his expected surplus $s_{ij} = u_{ij} - p_j$ from the products $j \in \mathcal{M}$, where p_j denotes the price.

5.3. *Random search benchmark*

In the benchmark without recommendations, consumer i initially assigns an equal expected surplus to all products, which is the unconditional mean of s_i based on the probability distribution $F_{i,\mathcal{M}}(s_i)$. Following this, he has the option to inspect the products, incurring a unit search cost of c_s. Lacking specific knowledge about individual products, consumers must search randomly. When they sample a product, they observe its price and match value.

To ensure stationarity, we assume sampling with replacement and no recall.[19] Under this assumption, consumers will continue to search until they reach a cut-off level of surplus, de-

[17] While the assumption that consumers at the outset have no clue of product quality is reasonable when $\alpha = 0$, as α increases, it becomes progressively less tenable. Consequently, while we examine all conceivable combinations of horizontal and vertical product differentiation, our analysis particularly pertains to scenarios where vertical differentiation is limited.

[18] The alternative scenario, where consumers are aware of product prices but lack knowledge of match values, may also be realistic. However, in this case, consumers would engage in directed search. As discussed in the concluding section, this complicates the analysis, which justifies postponing further investigation of this case to future research. Yet another possibility is that consumers may not be fully aware of product availability. However, the analysis of Aridor et al. (2022) suggests that this factor may be of lesser relevance.

[19] Under the alternative assumption of sampling with replacement and perfect recall, since consumers know that the products are equally spaced, the probability distribution $F_{i,\mathcal{M}}(s_i)$ changes upon inspecting a product and observing

noted as $\hat{s}_{i,\mathcal{M}}$, which is the solution to:

$$\int_{x}^{\infty} (s_i - x)\, dF_{i,\mathcal{M}}(s_i) = c_s. \tag{9}$$

Intuitively, at the optimum, the expected benefit of one additional act of search must be equal to its cost.

5.4. *Personalized prominence*

Compared to this benchmark, we investigate a scenario where platforms utilize algorithms to estimate match values and provide personalized recommendations based on these estimates. As soon as the recommendations are just minimally informative, consumers will prioritize examining the products suggested by the RS first. This introduces a form of prominence, as described by Armstrong et al. (2009). Here, however, different products may be prominent for different consumers, so prominence is personalized.

Denote the product j with the highest estimated match value for consumer i in market $\mathcal{M}$ as $j^\star(i,\mathcal{M})$.[20] If the platform does not strategically manipulate the recommendations, it will suggest product $j^\star(i,\mathcal{M})$ to consumer i.[21] Upon receiving the recommendation, consumer i inspects the recommended product $j^\star(i,\mathcal{M})$, observes its price $p_{j^\star(i,\mathcal{M})}$, and obtains an assessment of the match value, $u_{ij^*(i,\mathcal{M})}$. At this point, he has two options: either purchase the recommended product or conduct further search, at the same unit cost c_s as in the benchmark. Any additional search conducted is random, similar to the case without RSs.[22] Thus, the consumer will choose to continue searching if the expected surplus $u_{ij^*(i,\mathcal{M})} - p_{j^\star(i,\mathcal{M})}$ falls below the cut-off $\hat{s}_{i,\mathcal{M}}$.

6. BASELINE PARAMETRIZATION AND ROBUSTNESS ANALYSIS

In this section, we specify the baseline parameterization of the model in preparation for the numerical analysis. Additionally, we provide descriptions of several robustness checks that have been conducted, with detailed results relegated to the online appendix and supplementary material.

6.1. *Baseline model*

We choose the baseline values of the parameters so that the model matches some key properties of the Netflix Challenge (see footnote 7 and Table I). In the dataset provided by

its location, implying that the cutoffs vary after each visit. Since the calculation of the cutoffs is by far the most time-consuming part of the numerical analysis, the property of stationarity is analytically convenient. One could dispense with the assumption of replacement without losing stationarity by assuming that products are distributed randomly instead of being equally spaced. In any case, the difference with the more common assumption of sampling with replacement tends to vanish as the number of products grows large and disappears with infinitely many products.

[20]We assume that recommendations are solely based on the match value to maintain consistency with the individual search benchmark, in which consumers are unaware of product prices. If recommendations were also based on product prices, the appropriate benchmark would be one of directed search. Otherwise, the algorithms would perform the double role of estimating match values and (implicitly) informing about product prices, creating ambiguity about which of these functions is responsible for the results.

[21]If the platform manipulates the recommendations, all that changes is that the recommended product may not be $j^\star(i,\mathcal{M})$.

[22]When the platform recommends a list of products, further search will not be random. This case is more complex and is left for future work.

Netflix to participants in the challenge, the ratio between the number of users and items was approximately 30, so we set $\frac{I}{J} = 30$. Likewise, the density d of the matrix of ratings $\tilde{\mathbf{R}}$ is set to the same value as in Netflix Challenge, i.e., 1.2%.[23]

To reduce the computational burden of the simulations, we scale down the number of products and consumers by a factor of approximately 20 compared to the Netflix problem. Thus, we set $J = 801$ and hence $I = 24,030$. (The number of products is odd so that their distribution can be symmetric around the central product.)

	Environment				Recommender System			
	Users I	Items J	Observations	Density	H	ℓ	H	ℓ
Netflix	500,000	17,000	100,000,000	1.2%	40	4.84	100	1.93
Baseline	24,030	801	231,265	1.2%	2	4.65	5	1.86

TABLE I: The baseline scenario and the Netflix Challenge

The winner of the Netflix Challenge used a number of latent factors H of either 40 or 100. (The trade-off is that more latent factors allow for more flexibility but increase the number of variables to be estimated with the same observations, reducing the precision of the estimates.) To calibrate the corresponding values for our baseline model, we match the ratio between the number of observations and the number of parameters to be estimated,

$$\ell = \frac{d}{H} \frac{J \times I}{(J + I)}. \tag{10}$$

This leads us to consider two possible values of H, namely $H = 2$ (which is our baseline choice and corresponds to $H = 40$ in the Netflix Challenge) and $H = 5$ (which is taken up among the extensions and corresponds to $H = 100$).[24]

For several other parameters of the model, the Netflix challenge or similar datasets provide no guidance. We proceed as follows. First, we consider a grid of values of the number of (equally spaced) products in the market, $m_{\mathcal{M}}$, and the degree of vertical differentiation, α, with $m_{\mathcal{M}}$ ranging from 7 to 31 with a step size of 6 (even though in the main text we report only the case $m_{\mathcal{M}} = 19$ for ease of exposition) and α ranging from 0 (pure horizontal differentiation) to 1 (pure vertical differentiation), with a step size of 0.25.[25]

Second, we explore different values of the unit search cost parameter c_s. To provide an intuition for their practical relevance, we relate c_s to the fraction of consumers who purchase the recommended product without further search. We let this fraction vary between 70% and 95% for the case where there are 19 products in a market. This translates into a range of values for c_s from 0.002 to 0.006.

[23] Computer scientists use several benchmark datasets to evaluate RSs. Some of these datasets have densities similar to Netflix's, such as the MovieLens 10M Dataset, which contains users' movie ratings collected by the GroupLens research project at the University of Minnesota and has a density of 1.34%. However, other benchmark datasets have lower densities. In the robustness analysis, we investigate the sensitivity of our findings to variations in density.

[24] One problem that arises in this extension is the absence of a general solution for evenly positioning points along the section of the unit hyper-sphere located in the positive orthant when $H > 3$. To overcome this difficulty, we have developed an iterative algorithm inspired by the k-means method to approximate the precise positioning of products. More details are provided in the supplementary material.

[25] In the online appendix we also report results with a step size of 0.1.

In the main text, we focus on the case $c_s = 0.004$, which corresponds to around 85% of consumers not searching further. Although this fraction is high, it serves to amplify the effects of RSs. Intuitively, as the search cost decreases, the influence of recommendations tends to diminish. In the extreme case where the search cost becomes negligible, consumers will choose to inspect all products before making a purchase, rendering the impact of recommendations negligible as well.

The online appendix gives more detailed results for the entire grid of values of α, $m_{\mathcal{M}}$, and c_s.

To proceed, remember that in our baseline analysis, we assume that consumers report the utility level $\tilde{u}_{ij} = \bar{u}_{ij} + \varepsilon_{ij} + \epsilon_{ij}$, inclusive of the normally distributed shocks. Both shocks have zero mean, so it remains to specify their variance. The standard deviation of the taste shocks σ_ε is set equal, for each consumer i, to 10% of the standard deviation of the distribution of $\bar{u}_{ij}$ across the J products. This value is about as low as it can be without prejudicing convergence of the iterative procedure for the calculation of equilibrium prices.[26] As for the reporting noise, in the baseline scenario we set $\sigma_\epsilon = 2\sigma_\varepsilon$.

6.2. *Robustness checks*

After analyzing the baseline scenario, we conducted several robustness checks.

Density. First, we allowed the density d of the observed ratings matrix $\tilde{\mathbf{R}}$, which in the baseline scenario is set to 1.2%, to vary from 0.6% to 2.4%, increasing in steps of 0.3%.

Number of consumers and products. Next, we varied the number of products and consumers. Specifically, we explored two changes to the baseline. First, we doubled and halved both I and J while keeping their ratio constant and equal to 30. Second, we reduced the ratio of I/J to 15, and then down to 3,[27] while changing the levels to ensure that the ratio of observations to the number of parameters to be estimated, ℓ, remains constant.

Reporting noise. In a further robustness check, we varied the standard deviation of the idiosyncratic shocks ϵ_{ij}, letting it range from 0 to 40% of the standard deviation of the expected utility across the J products, with a stepsize of 2.5%. (In the baseline, it is set to 20%.)

Likert scale. A different way to vary the information derived from the reported ratings is to assume that consumers provide a value on a Likert scale, instead of a utility level. For instance, consumers might rate the product they purchased by assigning it a certain number of stars, or the platform may infer implicit, coarse ratings from the observation of consumers' behavior. To explore this possibility, we partitioned the original range of the ratings $\tilde{u}_{ij}$ into ℓ intervals of the same size, effectively creating a Likert scale with ℓ levels. We assumed that the algorithm observes only the interval to which the rating belongs, instead of the exact value $\tilde{u}_{ij}$. We considered values of ℓ ranging from 2 to 10.

The results for the baseline scenario, as reported in the following sections, remain generally robust to all of these extensions. In the main text, we highlight the most significant

[26]When σ_ε is lower, the iterative procedure may become trapped in cycles, suggesting that equilibria may involve mixed strategies.

[27]This is motivated by the fact that the alternative dataset mentioned in footnote 23 typically have lower I/J ratios than the Netflix dataset.

variations, and we utilize the results of these checks in Section 9, where we study the effects of changing the level of information that the RSs have access to. A more comprehensive presentation of the robustness analysis is provided in the online appendix and supplementary material.

6.3. *Numerical simulations*

Since there are several sources of uncertainty,[28] for each set of parameters we run 100 simulations with different realizations of the uncertainty. The results are then averaged across sessions.

7. MARKET CONCENTRATION

In this section, we concentrate on subscription platforms like Netflix or Spotify, where consumers pay a fixed fee and can access any product without additional charges. Thus, the prices of individual products, p_j, are all zero. For this scenario, the primary question we address is the impact of RSs on market concentration.

We specifically revisit two opposing views of RSs that have emerged in the management and computer science literature: the "long-tail" and "superstar" views. As described by Bar-Isaac et al. (2012), the former refers to a scenario where they increase the popularity of niche products, while the latter refers to a scenario in which RSs increase the popularity of mass products. The long-tail view was first articulated by Anderson (2008), who argues that online markets exhibit a long-tail phenomenon due to the larger selection of products available to suppliers and consumers' easier access to niche products.[29] In contrast, Fleder and Hosanagar (2009), among others, have found that RSs may reinforce the popularity of already popular items, resulting in a decrease in diversity at the aggregate level, even though individual-level diversity may increase. (Note that these views are not mutually exclusive: in principle, when there are more than three products, both effects can coexist, at the expense of intermediate products.)

However, the existing literature in marketing and computer science often relies on *ad hoc* assumptions about consumer preferences, which are open to criticism. Our contribution to this debate lies in using a model of consumer preferences and product differentiation firmly rooted in economic theory. Furthermore, and perhaps even more importantly, we explicitly model consumer search, a factor typically neglected in those literatures.

To preview the results, we find that RSs tend to increase market concentration, with a noticeable superstar effect and no significant long-tail effects.

7.1. *Quality of recommendations*

To begin, it is crucial to determine if the RS successfully matches consumers with products. A natural metric for assessing this is the welfare obtained by consumers. Table II compares the average utility, net of search costs, obtained by the average consumer across the 100 simulations under RS and individual search, with the standard deviation shown in parenthesis.

As further benchmarks, the table also reports the net utility that the average consumer would obtain under complete information (this is less than 1 because not all consumers $\mathbf{t}$ can find a perfect match $\mathbf{v} = \mathbf{t}$, or because, when $\alpha > 0$, the quality of the ideal product may be less than 1) or when choices are entirely random.

[28] These include the utility shocks ε_{ij}, the reporting noise ϵ_{ij}, and the choice of the products included in $\tilde{\mathbf{R}}$.

[29] Brynjolfsson et al. (2011) claim that the long-tail phenomenon persists even when holding product availability constant.

The results show that the RS generates a significant utility gain, despite having limited information. Both the magnitude of the gain and its sources depend on the degree of vertical differentiation in the market. The gain is higher, the more vertically differentiated the products. As for the sources, the total gain can be decomposed into better matching and reduced search. With purely horizontal differentiation ($\alpha = 0$), it turns out that two-thirds of the gain comes from lower total search costs. However, when α is larger, a substantial portion of the gain is due to the RS's ability to identify the best match.

α	0	0.25	0.5	0.75	1
Full information	0.9957	0.9510	0.9199	0.9020	0.8963
Random choice	0.7969	0.7563	0.7158	0.6753	0.6347
*Un*assisted search	0.9694 (0.0000)	0.9225 (0.0000)	0.8844 (0.0000)	0.8569 (0.0000)	0.8411 (0.0000)
*R*ecommender *S*ystem	0.9834 (0.0001)	0.9396 (0.0001)	0.9102 (0.0001)	0.8944 (0.0001)	0.8911 (0.0001)
$\frac{RS-Un}{Un} \times 100$	1.44% (0.01%)	1.85% (0.01%)	2.92% (0.01%)	4.37% (0.01%)	5.95% (0.01%)

TABLE II: Consumer average utility, net of search costs (SD in parentheses).

7.2. *Tails*

Now we investigate whether RSs lead to an increase in the market share of niche products, resulting in a long-tail effect. To explore this question, we examine the average market shares of the most peripheral products, i.e., the ones lying on the x- and y-axis in Figure 1. As noted, these products are natural candidates for being niche products in our framework. The results are presented in Table III.

α	0	0.25	0.5	0.75	1
*Un*assisted search	0.0325 (0.0000)	0.0263 (0.0000)	0.0189 (0.0000)	0.0103 (0.0000)	0.0029 (0.0000)
*R*ecommender *S*ystem	0.0258 (0.0012)	0.0216 (0.0010)	0.0158 (0.0007)	0.0091 (0.0004)	0.0018 (0.0000)
$\frac{RS-Un}{Un} \times 100$	-20.66% (3.82%)	-18.16% (3.79%)	-16.39% (3.79%)	-11.63% (3.41%)	-38.11% (1.73%)

TABLE III: Market share of niche products (i.e., the two most peripheral products in Figure 1).

Contrary to the long-tail hypothesis, the market shares of niche products are consistently lower with RSs than in the benchmark. (In fact, we observe a decrease in the market shares not only of the two most extreme products, but also of the nearby ones.) The decline ranges from one-tenth to more than a third. Similar results are found for any number of products and any level of the search cost, as well as in all the extensions listed in Section 6. Therefore, we conclude that our findings do not support the notion that RSs encourage the diffusion of niche products.

7.3. *Superstars*

The alternative view of RSs in marketing and computer science is that they tend to produce superstars. In order to determine whether there is any evidence of a superstar effect, we perform the same analysis on the central product, which is located on the 45-degree line in Figure 1.

α	0	0.25	0.5	0.75	1
*Un*assisted search	0.0565 (0.0000)	0.1291 (0.0000)	0.2302 (0.0000)	0.3650 (0.0000)	0.5312 (0.0000)
*R*ecommender *S*ystem	0.0682 (0.0074)	0.3091 (0.0096)	0.5539 (0.0065)	0.7545 (0.0047)	0.9153 (0.0034)
$\frac{RS-Un}{Un} \times 100$	20.61 (13.10)	139.41 (7.40)	140.67 (2.82)	106.73 (1.29)	72.30 (0.64)

TABLE IV: Market share of the central product.

Remember that the central product stands out due to two reasons: its central position, which places it closest to the median consumer, and its higher total quality when $\alpha > 0$. As shown in Table IV, the market share of the central product increases significantly. Again, this result remains true varying the number of products and the level of the search cost, as well as in all the extensions mentioned in Section 6.

The rise in the market share of the central product is smaller when product differentiation is purely horizontal ($\alpha = 0$),[30] but in fact, the RS has a strong tendency to create superstars even in this case. This tendency is not fully apparent in Table IV because with $m_{\mathcal{M}} = 19$ products there are many products that are "central," and the ones selected by the RS as superstars vary randomly from session to session.

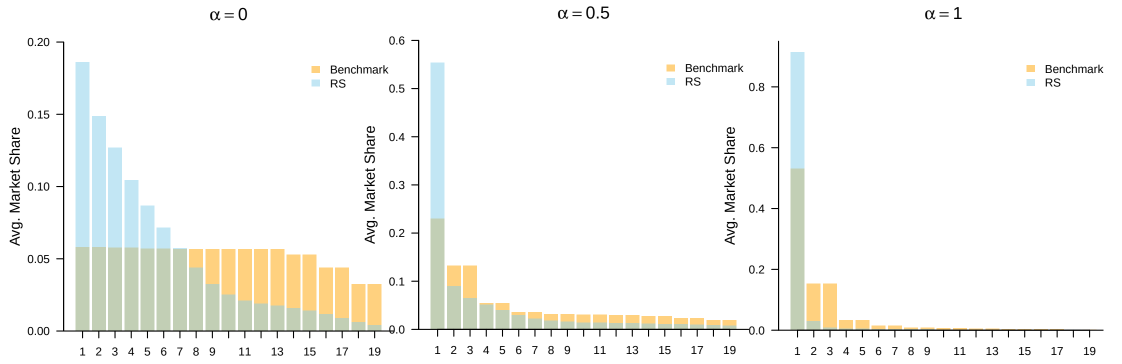
To control for this effect, we ranked products based on their market share and calculated the average share of the most popular product, the second most popular, and so forth. The resulting distribution is illustrated in Figure 3, for both the RS scenario and the benchmark of unassisted search. It appears that the RS creates a substantial superstar effect even when $\alpha = 0$, with the most popular product earning a market share more than three times larger than in the benchmark.

7.4. *The degree of market concentration*

The combination of superstar effect and reverse long-tail effect results in a significant increase in market concentration.

We assess concentration by ranking products and calculating the average share of the most popular product, the second most popular, and so forth, across 100 independent sessions. This analysis is conducted for both the recommendation system (RS) scenario and the benchmark of unassisted search. The resulting distribution is illustrated in Figure 3. The RS significantly amplifies market concentration by creating winners and losers compared to the unassisted search benchmark. This effect is particularly pronounced in cases of horizontal differentiation, where large firms become three times as large as expected while small firms shrink seemingly without a valid cause. To further quantify this phenomenon, we use the Herfindahl-Hirschman Index

[30]The rise is also less pronounced when product differentiation is predominantly vertical. This is because, in such cases, consumers can readily identify the superior product to a significant degree even without relying on the RS.



—Horizontal diff. —Mixed diff. —Vertical diff.

FIGURE 3.—Avg Market share by product rank

(HHI), as shown in Table V.[31] The increase in the HHI is substantial for all types of product differentiation, but is particularly significant for intermediate values of α.

For large values of α, the increase in market concentration may be seen as a benign outcome, as it indicates that the algorithm is able to effectively identify and promote the superior product to consumers. However, for low values of α, it is more concerning. In this case, the RS appears to create its own "champions" without any clear objective basis for favoring one product over others.

α	0	0.25	0.5	0.75	1
Unassisted search	538.5979 (0.0000)	650.1194 (0.0000)	1,057.3305 (0.0000)	1,902.0176 (0.0000)	3,323.1593 (0.0000)
Recommender System	1,078.0114 (11.4180)	1,593.3218 (37.1896)	3,329.5506 (57.4467)	5,788.5495 (61.6039)	8,410.1934 (56.8680)
$\frac{RS-Un}{Un} \times 100$	100.15% (2.12%)	145.08% (5.72%)	214.90% (5.43%)	204.34% (3.24%)	153.08% (1.71%)

TABLE V: The Herfindahl-Hirschman Index of market concentration.

7.5. *Estimation biases and the "uniformity effect"*

What, then, is the source of the superstar effect and the increase in market concentration? Given that the RS adopts a correctly specified model of consumer preferences, these effects must be caused by estimation biases due to small sample size. To verify this conjecture, we examine how the estimates $\hat{\mathbf{T}}$ and $\hat{\mathbf{V}}$ compare with the true coefficients $\mathbf{T}$ and $\mathbf{V}$. Given that the increase in market concentration is particularly concerning in the case of purely horizontal product differentiation ($\alpha = 0$), we focus our analysis on this scenario.

[31]We follow the common practice in industrial organization of normalizing the HHI so that it ranges between 0 and 10,000. We have also considered other indices of market concentration, such as the Gini index or the fraction of products that carry a positive market share, with similar results.

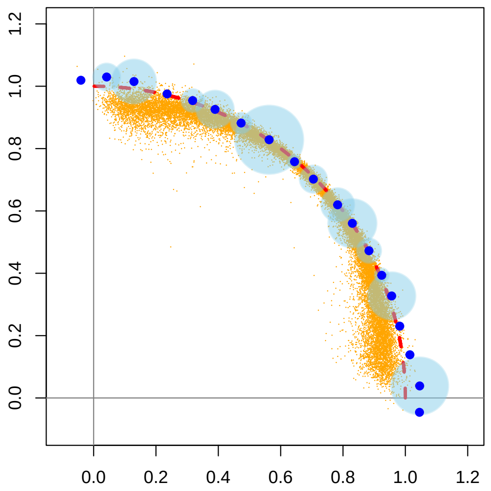


FIGURE 4.—The estimated location of consumers (orange dots) and products (blue dots), plotted against their true location (the red dashed curve), for a single session with $\alpha = 0$. The size of the disk around the blue dots represents the product's market share.

To gain some preliminary insight, we start by randomly selecting one session. For this session, Figure 4 depicts the algorithm's estimates of both consumers' and products' latent vectors, $\{(\hat{t}_{i1}, \hat{t}_{i2})\}_{i=1}^{I}$ and $\{(\hat{v}_{j1}, \hat{v}_{j2})\}_{j=1}^{J}$. The red dashed curve represents the "true" consumers and products, as in Figure 1, while the orange dots represent the "virtual" consumers $\hat{t}_i$ estimated by the algorithm, and the blue dots the "virtual" products $\hat{v}_j$. The size of the disk around each blue dot indicates how frequently that product is recommended.

Note that the virtual consumers $(\hat{t}_{i1}, \hat{t}_{i2})$ are fully described by the ratio $\frac{\hat{t}_{i2}}{\hat{t}_{i1}}$, with the level of the $\hat{t}$s being irrelevant. This means that equi-proportional changes in t_1 and t_2 will not affect consumers' choices as long as the market is covered. On the other hand, for the virtual products $(\hat{v}_{j1}, \hat{v}_{j2})$, both the ratio $\frac{\hat{v}_{j2}}{\hat{v}_{j1}}$ and the level of the $\hat{v}$s matter. The former represents the product's estimated "type," the latter the estimated "quality."

In principle, therefore, the algorithm can make three types of errors: (*i*) the estimated consumer tastes may differ from the true ones, resulting in $\frac{\hat{t}_{i2}}{\hat{t}_{i1}} \neq \frac{t_{i2}}{t_{i1}}$; (*ii*) the estimated product types may differ from the true ones, resulting in $\frac{\hat{v}_{j2}}{\hat{v}_{j1}} \neq \frac{v_{j2}}{v_{j1}}$; and (*iii*) the estimated product qualities may differ from the true ones, resulting in $\sum_{h=1}^{2} \hat{v}_{jh}^2 \neq \sum_{h=1}^{2} v_{jh}^2 (= 1)$.

The session shown in Figure 4 illustrates all three types of estimation biases. Firstly, the orange dots are clustered together towards the 45-degree line, indicating that the algorithm tends to overestimate consumer uniformity. We shall refer to this tendency as the uniformity effect. Secondly, the blue dots are scattered away from the centre, suggesting that the algorithm tends to overestimate the heterogeneity among products, although to a lesser extent than the first bias. Thirdly, some products are located above the red curve, especially those on the periphery, while others lie on or below it. This suggests that the quality of peripheral products tends to be overestimated.

The biases highlighted in Figure 4 are not unique to that particular session. Figure 5 presents a more general analysis that considers all 100 sessions. The left panel shows the difference between the estimated and true ratios $\frac{\hat{t}_{i2}}{\hat{t}_{i1}} - \frac{t_{i2}}{t_{i1}}$ for the I consumers, ranked from the x- to the y-axis. The estimated ratios are less dispersed than the true ratios with positive differences for consumers whose true ratio is small and negative differences for those with a high true ratio. This reflects the uniformity effect. The central panel displays the same analysis for the J products, plotting the differences $\frac{\hat{v}_{j2}}{\hat{v}_{j1}} - \frac{v_{j2}}{v_{j1}}$. Here, the estimated product types are more dispersed than the true types, although the bias is smaller than that observed for the consumer ratios. Finally, the right panel shows the distribution of $\sum_{h=1}^{2} \hat{v}_{jh}^2 - 1$ for the J products, representing the algorithm's tendency to overestimate products' overall quality. The figure shows a bias against central products.

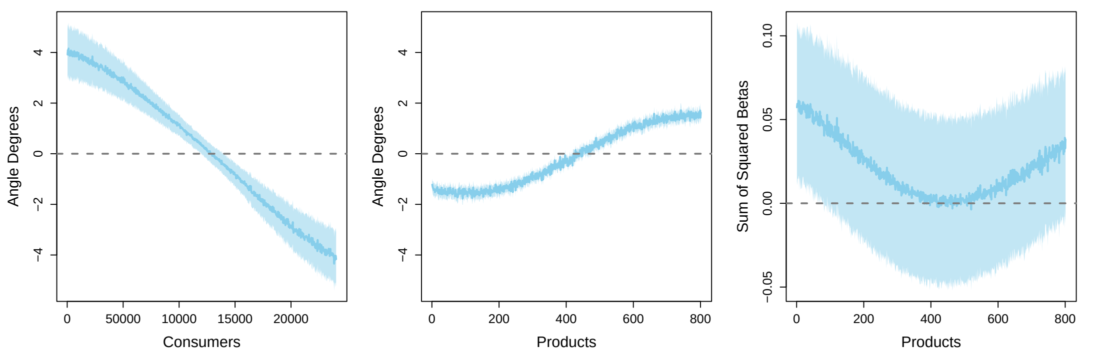


FIGURE 5.—Estimation biases with $\alpha = 0$: the consumer uniformity effect (left), the product heterogenization effect (center), and the quality bias against central products (right). The first two effects are measured by the difference between the angles, $\arctan \frac{\hat{t}_{i2}}{\hat{t}_{i1}} - \arctan \frac{t_{i2}}{t_{i1}}$ and $\arctan \frac{\hat{v}_{j2}}{\hat{v}_{j1}} - \arctan \frac{v_{j2}}{v_{j1}}$, respectively. The third effect is measured as the distance between the estimated quality $\sum_{h=1}^{2} \hat{v}_{jh}^2$ and the actual quality, 1.

The quality bias that favors peripheral products could have produced a long-tail effect. However, this bias is outweighed by the uniformity effect. That is, the algorithm may think that the peripheral products are better than they actually are, but it also assumes that only a few consumers like such "extreme" products, so it recommends them infrequently. This explains the reverse long-tail effect. On the other hand, the superstar effect arises, when $\alpha = 0$, because the algorithm estimates certain products to be of higher quality than their nearby counterparts, even when all products are of the same overall quality. The products with the highest estimated qualities become the superstars that command the largest market share.

Naturally, these biases tend to disappear as the density d of the matrix of observed ratings $\tilde{\mathbf{R}}$ is large. However, in the online appendix we show that for realistic values of d, the biases remain.

8. EQUILIBRIUM PRICES

We now shift our focus to platforms like Amazon, where consumers are charged separately for each product they purchase. We assume that sellers autonomously set prices for their products, while the platform collects transaction fees. We are interested in ascertaining the impact of algorithmic recommendations on equilibrium prices and consumer welfare.

Our analysis in this section is related to the extensive body of research in the field of industrial organization that has explored the impact of pre-search information on the market equilibrium. This literature remains largely unsettled. On one hand, de Corniere (2016) and Zhong (2023) demonstrate that when pre-search information directs consumers to a subset of products with the highest match values, but without ordering the products in this subset, having more information (i.e., a narrower, better-selected subset) tends to result in lower prices, unless the information becomes so precise that consumers cease searching altogether. On the other hand, Anderson and Renault (2000) assume that a fraction of consumers have full knowledge of their match values and demonstrate, for the case of two products, that as this fraction increases, prices rise. Zhou (2022), allowing for more general types of pre-search information, presents more nuanced results. Specifically, he shows that prices decrease if the additional information does not prompt consumers to search more, but this result can be reversed if this condition does not hold.

In light of these contradictory findings, it is not clear *a priori* what the price effect of algorithmic recommendations may be. Additionally, there are two important differences between our setup and those considered in the existing literature. First, the pre-search information in our framework consists of top-product signals that create personalized prominence, a case that, to the best of our knowledge, has not been considered thus far.[32] Furthermore, in the papers mentioned above, preferences are idiosyncratic, whereas in our setting, as explained earlier, we have included systematic differences and similarities among consumers and products.

It turns out that algorithmic recommendations generally lead firms to increase prices, even if consumers search less. After presenting our results, we briefly discuss how they relate to previous findings in the literature.

8.1. *Price equilibrium*

We assume that each of the $m_{\mathcal{M}}$ products traded in market $\mathcal{M}$ is supplied by a separate firm and that marginal production costs are zero. Firms compete in prices and correctly anticipate the demand functions.[33] It is important to remember that recommendations are personalized but we assume that sellers do not engage in price discrimination.

Given the large number of products and consumers, and given that we cannot exploit symmetry, we calculate the Bertrand-Nash equilibrium prices $\mathbf{p}^*$ numerically. The calculation is accomplished by iteratively solving the system of first-order conditions corresponding to the maximization of each firm's profit with respect to its own price.[34] The iteration starts with random prices $\mathbf{p}_0^*$. At each successive step $\tau = 1, 2, ...$, starting from the vector of candidate

[32] The case where the platform recommends the product with the best estimated match value could be regarded as a limiting case of the information transmission mechanism analyzed in de Corniere (2016) and Zhong (2023), where the subset of products presented to each consumer contains just one product. In fact, however, both of those papers focus on the case where the subset contains infinitely many products.

[33] The derivation of the demand functions is presented in greater detail in the online appendix. It is important to note that each firm is supposed to know perfectly which product is recommended to each consumer. This is a natural starting point for the analysis, but in future work, it would be interesting to consider the case where firms have access only to the probability that each product may be recommended to each consumer, and calculate the demand function accordingly.

[34] To solve the system of first-order conditions, we use subroutine DNQSOL in the Caltech's MATH77 Fortran library, exercising special care in checking the second-order conditions.

equilibrium prices $\mathbf{p}^*_{\tau-1}$, we calculate the cut-offs $\hat{s}_{i,\mathcal{M},\tau}$,[35] and then from these the system of individual demand functions. For each firm j, we then calculate the new $p^*_{j,\tau}$ by maximizing firm j's profit while holding the other prices in $\mathbf{p}^*_{\tau-1}$ constant.[36] In this way, we obtain the new candidate vector $\mathbf{p}^*_\tau$. The procedure is iterated to convergence.

8.2. *The impact of RSs on equilibrium prices*

Table VI presents the equilibrium prices, which are equal to the equilibrium price-cost margins under our cost normalization. The reported values are obtained by taking the average across products, with each product's weight equal to its equilibrium market share.

α	0	0.25	0.5	0.75	1
*Un*assisted search	0.0383 (0.0000)	0.0356 (0.0000)	0.0322 (0.0000)	0.0283 (0.0000)	0.0227 (0.0000)
*R*ecommender *S*ystem	0.0441 (0.0002)	0.0402 (0.0001)	0.0351 (0.0000)	0.0303 (0.0000)	0.0243 (0.0000)
$\frac{RS-Un}{Un} \times 100$	15.10% (0.41%)	13.01% (0.28%)	9.06% (0.15%)	7.39% (0.08%)	7.13% (0.03%)

TABLE VI: Average equilibrium prices.

Algorithmic recommendations lead to an increase in prices across all values of α. The magnitude of the increase varies with α, with the largest increase (around 15%) occurring when product differentiation is purely horizontal, and the smallest (around 7%) when it is purely vertical. The online appendix confirms that varying the number of products and the level of the search cost does not modify the result. The outcome remains valid even when the algorithm is trained on data that it has previously contributed to generating.

Figure 6 provides a more detailed view of the changes in prices by showing the price changes for each individual product. Under horizontal differentiation, the price increase is relatively uniform across all products. In contrast, under pure vertical differentiation, the price of the central product slightly decreases, while the prices of other products increase. However, the decrease in the price of the central product, which carries the highest price, is offset by an increase in its market share, resulting in an increase in the weighted average price.

8.3. *Consumer surplus*

The increase in prices obviously has a negative impact on consumer welfare. However, RSs also have positive effects. First, they improve the matching between consumers and products. Second, with RSs, consumers can afford to search less extensively than in the benchmark, as the recommended product is more likely to surpass the cut-off level of surplus than a randomly

[35]Calculating the individually optimal cut-offs $\hat{s}_{i,\mathcal{M}}$ for our 24,030 consumers at each step of the iteration is a computationally intensive task, accounting for over 95% of the simulation time. To expedite the process, we assume that equilibrium prices follow a normal distribution with the same mean and variance as the actual equilibrium prices. We verified the validity of this assumption by also using the true distribution of equilibrium prices in some cases, but found the differences to be negligible.

[36]The assumption that consumer search decisions depend on their expectations of prices and not on the actual prices is sometimes called *passive beliefs*.

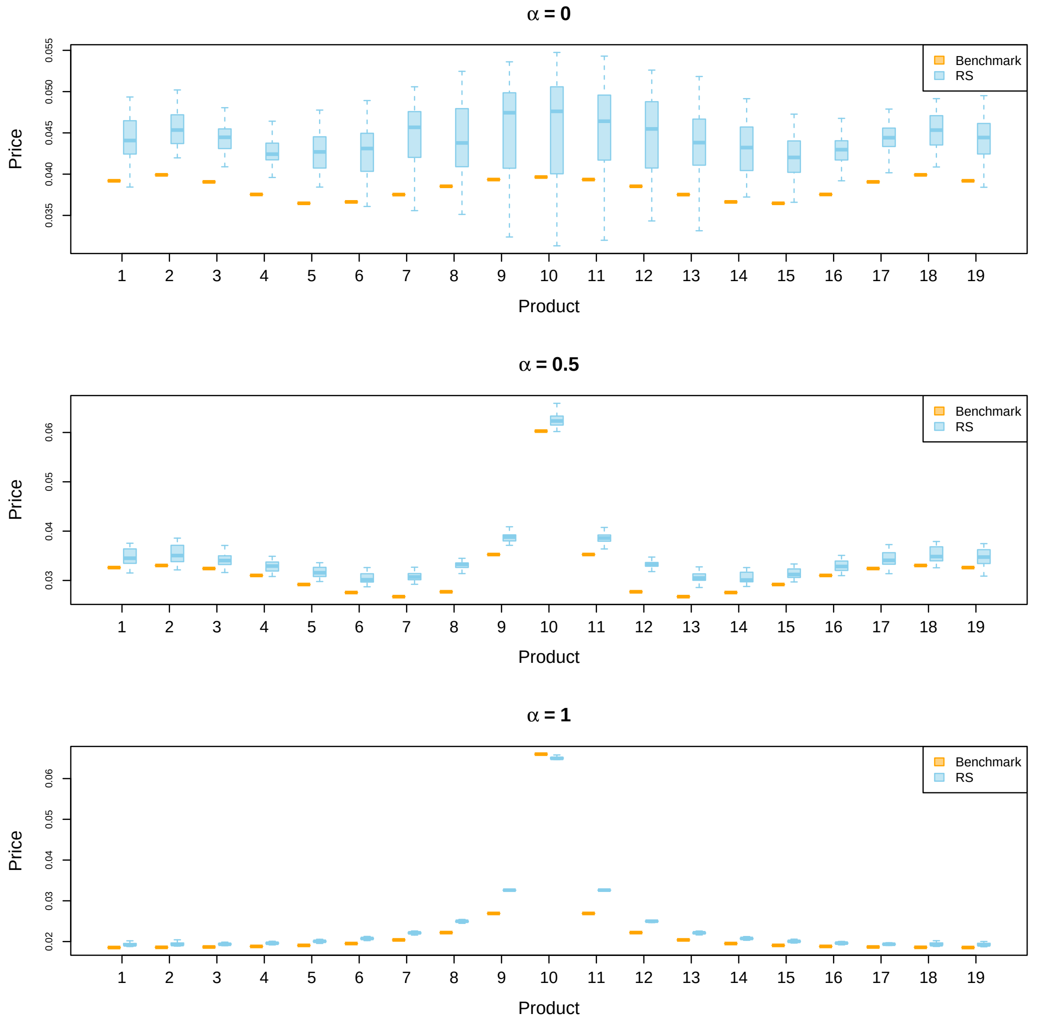


FIGURE 6.—The prices of the different products, ranked according to their location in the space of products: products 1 and 19 are peripheral, product 10 is central. The figure shows the median, inter-quartile range (IQR), and the highest and lowest value excluding outliers (obtained by subtracting 1.5 times the IQR range from the first quartile (Q1), and adding 1.5 times the IQR to the third quartile (Q3)).

chosen one. It is therefore important to ascertain whether these positive effects ultimately outweigh the negative impact of the price rise.

Table VII reports the change in consumer surplus, taking into account all of these effects. Overall, consumer surplus increases. Naturally, it does not increase as much as in Table II, due to the higher prices.

The three effects on consumer surplus resulting from the introduction of RSs, namely, higher prices, reduced search, and better matching, are of the same order of magnitude. To illustrate, let us consider for instance the case where $\alpha = 0.5$. In this case, the better matching between consumers and products alone increases consumer surplus by around 1.5%. The price increase has almost the same impact but of the opposite sign, which would leave the consumer surplus almost unchanged (-0.09%) if these two effects were taken together. Thus, the overall increase of 1.3% in consumer surplus nearly coincides with the reduction in total search costs.

The online appendix shows that the increase in consumer surplus is lower when the search cost is higher and the number of products is lower. RSs can actually decrease consumer surplus when three conditions cumulatively hold: the search cost is high, the number of products is small, and product differentiation is mainly horizontal. Taken together, however, these conditions seem rather implausible; for example, in this scenario, the fraction of consumers who follow the platform's recommendation is more than 95%. For more reasonable parameter values, the overall impact of RSs on consumer surplus is positive.

α	0	0.25	0.5	0.75	1
*Un*assisted search	0.9313 (0.0000)	0.8855 (0.0000)	0.8457 (0.0000)	0.8130 (0.0000)	0.7886 (0.0000)
*R*ecommender *S*ystem	0.9371 (0.0001)	0.8930 (0.0001)	0.8581 (0.0000)	0.8329 (0.0000)	0.8202 (0.0000)
$\frac{RS-Un}{Un} \times 100$	0.62% (0.01%)	0.85% (0.01%)	1.47% (0.00%)	2.45% (0.00%)	4.00% (0.00%)

TABLE VII: Consumer surplus, net of search costs.

8.4. *Profits*

Since the market is always fully covered, the increase in prices generally results in greater profits for firms. However, the impact of RSs on firms' profits is not uniform and can lead to both winners and losers, as discussed in detail in the online appendix.

8.5. *Demand shifts*

What drives the price changes that we have uncovered? To gain insight, we now examine the shifts in demand generated by algorithmic recommendations. Figure 7 illustrates the overall impact of the RS on demand for both central and peripheral products across various combinations of horizontal and vertical differentiation. It appears that the way demand shifts is intricate and varies between mass and niche products, as well as with the degree of vertical differentiation, making it challenging to discern a clear pattern.

Upon closer examination, however, one can identify some major changes in the demand curves. To begin, let us focus on the first panel in the top row of the figure, which illustrates the demand for the central product under purely horizontal differentiation. We note, firstly, a counterclockwise rotation in the upper segment of the demand curve, specifically for prices higher than the equilibrium ones. This rotation occurs as a result of a more homogeneous customer base, which is due to the personalized nature of prominence. (If prominence were non-personalized, the rotation would be clockwise, reflecting a more diverse customer pool due to one product being recommended to all consumers.)

Secondly, for lower prices, we see an upward vertical shift in the demand curve. This shift occurs because RSs guide consumers toward their preferred products, increasing their willingness to pay compared to when they search individually.

Thirdly, for these lower prices, we observe a clockwise rotation of the demand curve. This rotation reflects the difference in demand elasticity between consumers who randomly

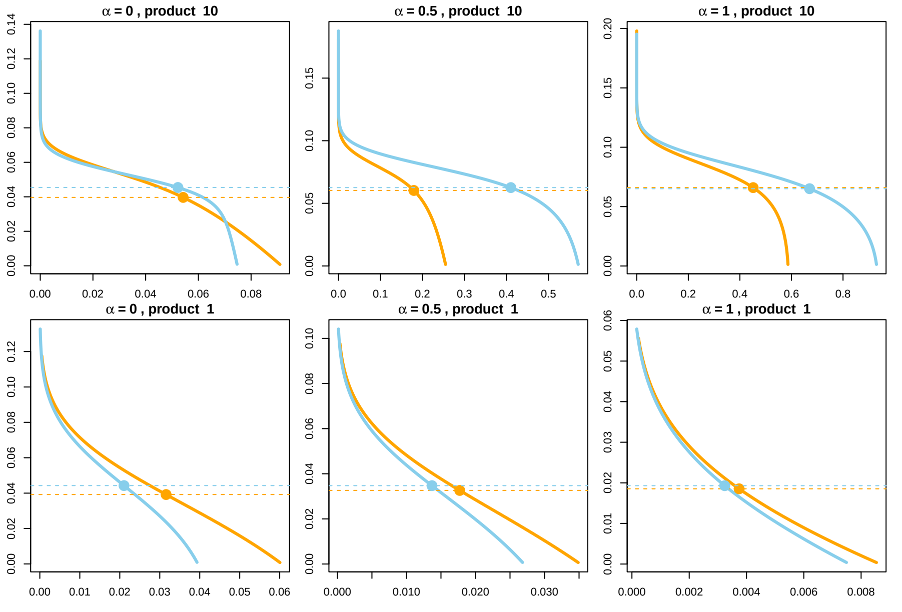


FIGURE 7.—The demand functions for the central product (product 10) and a peripheral product (product 1), for different combinations of horizontal and vertical differentiation in the benchmark (orange) and with the RS (light blue). The individual demands are calculated holding the prices of all other products constant at the equilibrium level in the benchmark case. The dots represent the equilibrium points.

encounter a product and those directed to their preferred products by the algorithm. When consumers can rely on algorithmic recommendations, random visits are less frequent, leading to a less elastic demand curve. Note that the last two effects dominate precisely within the price range where the equilibrium price lies.

The shifts described above are also detectable in the other panels of the figure, but they are somewhat masked by significant horizontal shifts. These shifts are positive for the central product when $\alpha > 0$, and negative for the peripheral ones. They correspond to the changes in volumes observed at zero prices and, like them, reflect the uniformity effect discussed in Section 7.4.

Some of the shifts described above lead to price decreases, while others result in price increases. Overall, it appears that the predominant effect is the clockwise rotation in the vicinity of the equilibrium price. This corresponds to the reduction in demand elasticity highlighted by Anderson and Renault (2000). In other words, with personalized recommendations, consumers are more likely to find a suitable match during their initial visit, reducing their incentive to search further. This decrease in search activity diminishes competition among firms, prompting them to increase prices. However, in the case of vertically differentiated products, the increase in volume for the central product leads to a counteracting decrease in elasticity, which may result in a lower equilibrium price. In this case, the rise in the average price is primarily driven by a composition effect.

9. INFORMATION AND WELFARE

Now, we conduct a comparative statics exercise by varying the quantity and quality of information available to the algorithms. We accomplish this by leveraging the robustness analysis presented in Section 6, as most of the variants considered alter the information accessible to the algorithms. For instance, the algorithms have access to more information as d, the density of the matrix of observed ratings, increases, and to more reliable information as σ_ϵ^2, the reporting noise, decreases, or when the reports are made on a finer Likert scale. Less obviously, changing the number of products and consumers also affects the available information, as it alters the ratio between the number of observations and parameters to be estimated. Lastly, we have already discussed that the quality of information degrades as one moves from the case of randomly generated data to that of endogenous data.

As it turns out, the effects of all of these changes depend more on the level of information than on the specific factors causing the change. This enables us to pool all extensions together to better elucidate the resulting pattern. The pooling capitalizes on the fact that the quantity and quality of information available to the algorithms translates into the precision of the estimates they produce and, consequently, the recommendations they make. Therefore, we consider all extensions as varying the precision of algorithmic recommendations, which we proxy by the average net utility that consumers would receive at zero prices.

Let us first examine how information affects equilibrium prices. In Figure 8, we can see the average price plotted against the level of information. The figure indicates that having more and better information leads to higher prices. Figure 8 confirms that the specific way in which the change in information is generated is less important than the overall level of information available. The same general pattern holds for profits, which generally increase with the level of information.

Turning to consumer welfare, more and better information has two opposing effects. On the one hand, it improves the matching between consumers and products, and this also results in a reduction in total search costs. On the other hand, we have just seen that equilibrium prices increase. When product differentiation is purely horizontal, the overall effect of information on consumer surplus exhibits a non-monotonic pattern, specifically an inverted-U shape as illustrated in Figure 9.[37]

The inverted-U relationship between information and consumer surplus suggests that there might be a social benefit to imposing limits on the amount of personal information that platforms can access. If the parametrization in our baseline scenario is realistic, we might already be on the downward slope of the curve. Therefore, some degree of privacy could be beneficial not only for its own sake but also for its positive effect on the intensity of product market competition.

However, firms have a vested interest in obtaining more information, as their profits monotonically increase with the level of information. This highlights the opposing interests of firms and consumers regarding the amount of personal information that should be available to the algorithms.

The trade-off we have just identified — when consumers have access to more information, the quality of the match between consumers and products improves, but firms also gain increased market power and set higher prices — also arises in different frameworks, such as those presented in Armstrong and Zhou (2022) and Jullien and Pavan (2019). However, these papers consider frameworks without consumer search, and the mechanism creating the trade-off differs significantly from ours. Zhou (2022) also presents examples where a trade-off similar to ours can arise, but, as discussed above, this occurs only if pre-search information prompts consumers to search more, not less.

[37]In the supplementary material, we show that the inverted-U pattern persists even when α is positive but not excessively large, indicating a predominantly horizontal product differentiation. In contrast, the pattern disappears when vertical differentiation predominates.

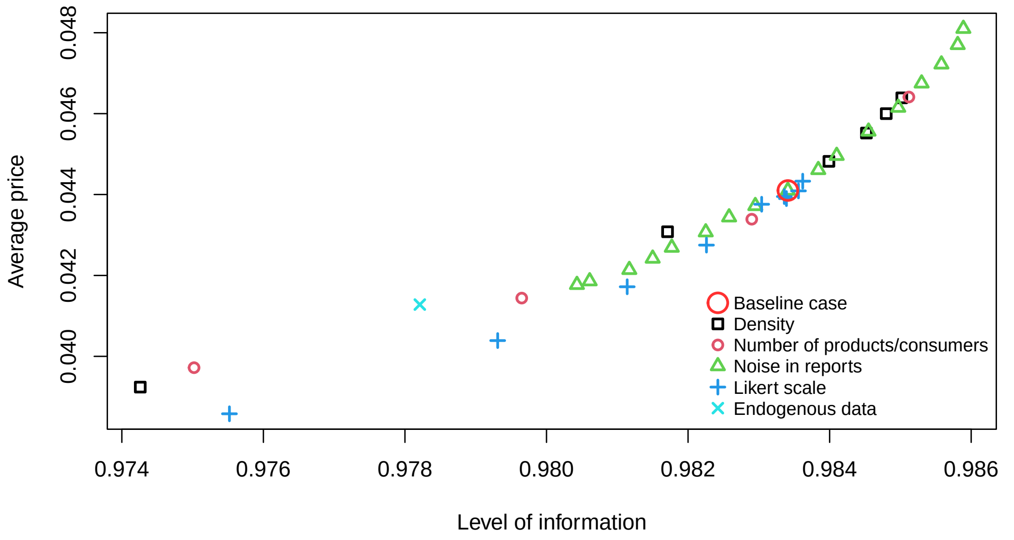


FIGURE 8.—The effect of information on equilibrium prices ($\alpha = 0$.)

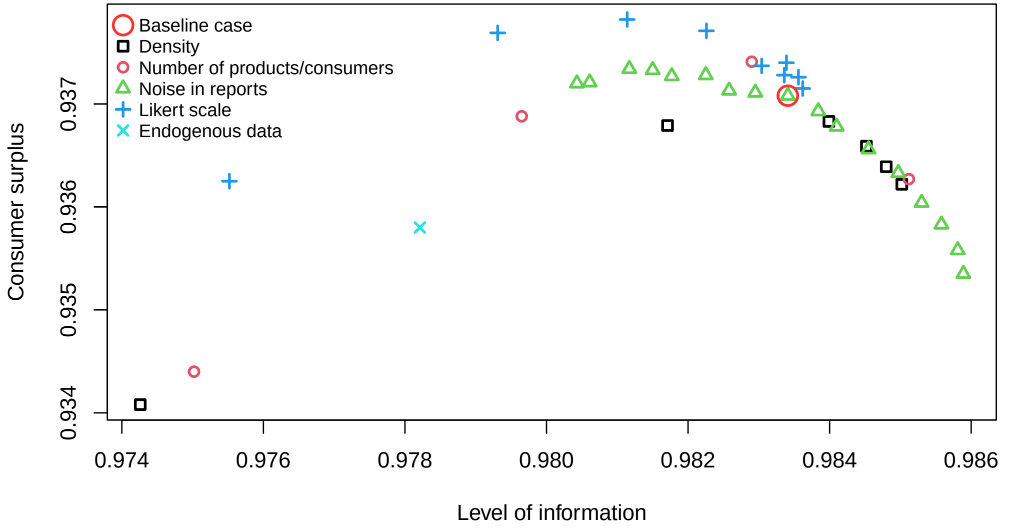


FIGURE 9.—The effect of information on consumer surplus.($\alpha = 0$.)

10. SELF PREFERENCING

We now consider the scenario where the platform suggests a product other than the one with the highest estimated match value. This situation may occur when the platform promotes specific products that yield higher earnings, possibly because they are supplied by a subsidiary of the platform rather than an independent supplier.

The possibility of such self-preferencing behavior has sparked various antitrust litigations and a growing body of economic literature,[38] which remains largely unsettled. The framework developed in this paper offers a fresh perspective on this ongoing debate.

The specific role of algorithmic estimates when the platform engages in self-preferencing is twofold: first, for genuine recommendations when the platform does not manipulate, and second, to determine when self-preferencing should occur. Specifically, we assume that, for a given rate of self-preferencing, the platform promotes its favored product to consumers whose estimated ideal product is closest to it. This strategy minimizes potential costs incurred by the platform when manipulating recommendations.[39]

We treat both the rate of self-preferencing, which represents the frequency with which the platform recommends its favored product to consumers for whom it is not already the best estimated match, and the platform's favored product as parameters in our analysis. Regarding the latter, we examine two scenarios: one where the platform's favored product is the central one (product 10),[40] and when it is one of the peripheral products (say product 1).

10.1. *Intensity of competition*

In all cases, self-preferencing intensifies competition and reduces equilibrium prices.

Figure 10 presents the change in equilibrium prices, relative to the case of genuine recommendations, for the case where the favored product is the central one and the rate of self-preferencing is 50%. The online appendix shows that the same qualitative pattern holds more generally. It also shows that the largest decrease in price occurs for the favored product and its closest competitors. This can be explained by the fact that without self-preferencing, the pool of consumers inspecting the favored product is relatively homogeneous. When instead the platform manipulates the recommendations, this pool becomes both larger and more heterogeneous. As a result, the demand for the favored product increases, but also becomes more elastic, leading the supplier to reduce the price.[41]

Conversely, the demand for competing products decreases as fewer consumers are directed toward them. As a result, competitors decrease their prices to remain competitive. These firms therefore suffer both from lower volumes and lower prices, so their profits decrease. These effects are largest for the products that are closest to the favored one, as the consumers who are misdirected are those whose ideal product is relatively similar to it. For more distant products, the effect is smaller.

[38] Following the seminal contribution of Hagiu and Jullien (2011), more recent papers include Calvano and Jullien (2018), de Cornière and Taylor (2019), Teh and Wright (2022), Bourreau and Gaudin (2022), Peitz and Sobolev (2022), and Bar-Isaac and Shelegia (2023).

[39] When recommendations are misleading, consumers will revert to individual search more frequently. In such cases, consumers may end up purchasing from an alternative marketing channel, potentially resulting in revenue loss for the platform. Additionally, the platform risks losing users if their surplus falls below a certain level.

[40] Note that when α is large, most consumers are already directed towards the central product, regardless of self-preferencing. As a result, in this case the impact of self-preferencing is almost imperceptible.

[41] This is similar to the effect of uniform prominence in Armstrong et al. (2009) .

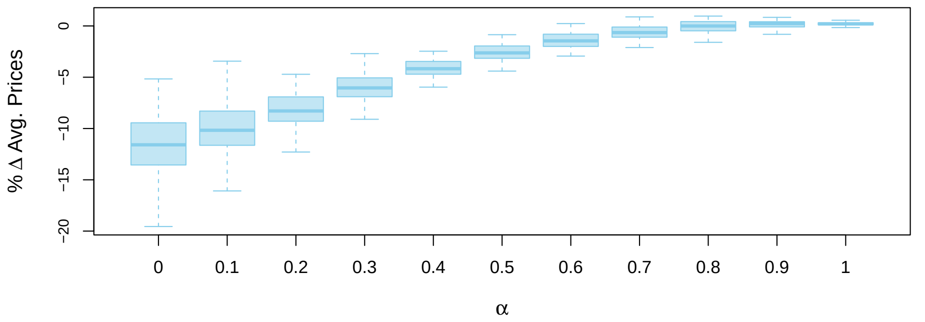


FIGURE 10.—The distribution across sessions of the impact of self-preferencing on the average price. The platform's favorite product is the central one, and the rate of self-preferencing is 50%.

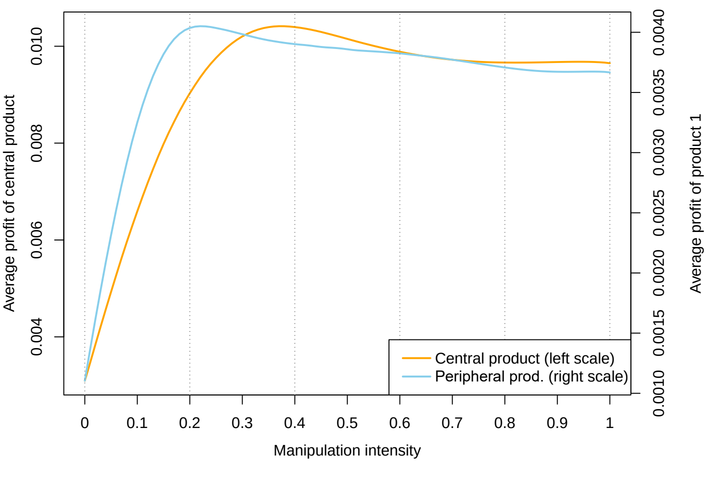


FIGURE 11.—The profits of the favored products (central in light blue, peripheral in orange) as a function of the rate of self-preferencing, with $\alpha = 0$.

10.2. *Profitability*

Now suppose that the platform adjusts the rate of self-preferencing. Neglecting the potential costs associated with self-preferencing mentioned in footnote 39, we examine the impact on the profits of the favored products. To investigate this, we vary the rate of self-preferencing from 0% to 100% and calculate the corresponding profits. The results are depicted in Figure 11, which indicates that the profit of the favored product reaches its peak at relatively low rates of self-preferencing, roughly between 20% to 30%.

From the figure, it becomes apparent that there is a trade-off between price and volume: increasing self-preferencing leads to higher demand for the favored product, but it also intensifies competition, resulting in lower prices. The observed decline in profit as the rate of self-preferencing increases is driven precisely by the competition-enhancing effect highlighted above.

10.3. *Consumer surplus*

Consumer surplus decreases when recommendations are manipulated relative to the case where they are sincere. Due to the price reduction, however, the decrease is relatively small. The analysis presented in the online appendix suggests that, given a certain market structure, the impact of self-preferencing on consumer surplus is likely limited. However, a more concerning issue could be the decrease in competitors' profitability, which may lead to exits or deter entry in the long run. As a result, policymakers may view self-preferencing as more of an exclusionary abuse rather than an exploitative one.

11. CONCLUSIONS

There are growing concerns about the potential anti-competitive effects of algorithmic recommendations on competition in product markets. This paper proposes a methodology to address these concerns. One concern is that RSs could favor a subset of firms for no good reason resulting in increased market concentration. Another concern is that RSs could enable sellers to more effectively segment the market and charge higher prices. Additionally, there are worries about platforms manipulating recommendations for their own benefit.

Our analysis confirms that these concerns are valid. However, RSs also have pro-competitive effects, such as enhancing the match between products and consumers, and reducing the need for costly search. Based on our quantitative assessment, the pro-competitive effects of RSs are at least as significant as the anti-competitive effects, and in many cases, greater.

Nevertheless, we also find that increasing the amount of information available to RSs may have a negative impact on consumers. Therefore, imposing limits on the platforms' access to personal information may be socially desirable. In addition to safeguarding privacy, such limitations could promote competition among sellers, leading to higher consumer surplus.

That being said, it is crucial to emphasize the need for caution when drawing policy conclusions at this stage, as our analysis overlooks several factors that could impact the policy implications. However, our methodology is flexible and can be adjusted to incorporate different algorithms, preferences, and assumptions about the search process so as to deliver more robust policy recommendations. To conclude the paper, we briefly discuss some potential extensions that could be explored in future work.

Directed search. In this paper, we have assumed that consumers know the equilibrium price distribution but not individual product prices. This approach is widely used in search theory. However, in online markets, consumers can easily obtain information on product prices. Therefore, a realistic assumption might be that consumers initially know the prices of individual products but not how well each product fits their personal preferences.

The analysis of this case necessitates two adjustments. Firstly, in the individual search benchmark, we must account for the possibility of ordered search, wherein the sequence of searching is determined by the products' prices. Secondly, for the sake of consistency, the platform should recommend the product with the highest estimated net surplus ($\hat{r}_{ij} - p_j$) rather than the highest estimated utility ($\hat{r}_{ij}$). Both of these modifications result in lower prices, and they must be implemented simultaneously to avoid spurious comparisons. However, the introduction of directed search complicates the calculation of the cut-offs $\hat{s}_{i,\mathcal{M}}$, which is already the most time-consuming part of the numerical analysis.

Multiple recommendations. When a platform recommends several products instead of just one, and presents them in a particular order, it complicates the analysis of consumer behavior with RSs (while the individual search benchmark does not change). In this scenario, consumers may have a stronger incentive to continue searching after inspecting the first product because further sampling would not be random. This increased search effort is likely to be pro-competitive and reinforces our conclusion that RSs are likely to increase consumer surplus.

Entry and exit. In more extended models, one could analyze the impact of RSs on other economic decisions, such as entry and exit, product quality, and R&D investment. Although these other aspects are beyond the scope of this study, our analytical framework can be extended to systematically analyze them in future research.

Multiple platforms. Often two or more platforms compete to attract consumers. In this case, the quality of the algorithm's estimates may represent a factor of competitive advantage or disadvantage.

Regulation. Our framework can be used to assess the effect of various forms of regulations. For example, it may be interesting to consider a policy that limits the level of personalization in recommendations. Although this could potentially decrease the match-value of the recommendation system, it could also help to mitigate the price increases we have observed.

Endogenous tastes. Finally, an intriguing and challenging extension to our analysis would be to consider the possibility that RSs may influence individual preferences. This possibility arises because tastes are endogenous, a fact that has been recognized at least since Knight (1923) but has been rarely studied analytically. The act of inspecting a product, such as starting to watch a movie, may not only reveal the match value but also shape consumer preferences. If this is the case, it raises concerns about the potential for platforms to manipulate individual tastes to their advantage.

REFERENCES

ABDOLLAHPOURI, HIMAN AND MASOUD MANSOURY (2020): "Multi-Sided Exposure Bias in Recommendation," . [2]

ABDOLLAHPOURI, HIMAN, MASOUD MANSOURY, ROBIN BURKE, BAMSHAD MOBASHER, AND EDWARD MALTHOUSE (2021): "User-Centered Evaluation of Popularity Bias in Recommender Systems," in *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, 119–129. [2]

AGGARWAL, CHARU C. (2016): *Recommender Systems: The Textbook*, Springer, 1st ed. 2016 edition ed. [5]

ANDERSON, CHRIS (2008): *The Long Tail: Why the Future of Business Is Selling Less of More*, Hachette Books, revised edition ed. [15]

ANDERSON, SIMON P AND REGIS RENAULT (1999): "Pricing, product diversity, and search costs: A Bertrand-Chamberlin-Diamond model," *The RAND Journal of Economics*, 719–735. [11]

ANDERSON, SIMON P AND RÉGIS RENAULT (2000): "Consumer information and firm pricing: negative externalities from improved information," *International economic review*, 41 (3), 721–742. [21, 25]

ARIDOR, GUY, DUARTE GONÇALVES, DANIEL KLUVER, RUOYAN KONG, AND JOSEPH KONSTAN (2022): "The economics of recommender systems: Evidence from a field experiment on movielens," *arXiv preprint arXiv:2211.14219*. [11]

ARMSTRONG, MARK, JOHN VICKERS, AND JIDONG ZHOU (2009): "Prominence and consumer search," *The RAND Journal of Economics*, 40 (2), 209–233. [12, 28]

ARMSTRONG, MARK AND JIDONG ZHOU (2022): "Consumer information and the limits to competition," *American Economic Review*, 112 (2), 534–577. [26]

ASKER, JOHN, CHAIM FERSHTMAN, AND ARIEL PAKES (2023): "The impact of artificial intelligence design on pricing," *Journal of Economics & Management Strategy*. [4]

ASSAD, S, R CLARK, D ERSHOV, AND L XU (forthcoming): "Algorithmic pricing and competition: Empirical evidence from the German retail gasoline market," *Journal of Political Economy*. [4]

BANCHIO, MARTINO AND GIACOMO MANTEGAZZA (2023): "ARTIFICIAL INTELLIGENCE AND SPONTANEOUS COLLUSION," *mimeo*. [4]

BAR-ISAAC, HESKI, GUILLERMO CARUANA, AND VICENTE CUÑAT (2012): "Search, design, and market structure," *American Economic Review*, 102 (2), 1140–1160. [15]

BAR-ISAAC, HESKI AND SANDRO SHELEGIA (2023): "Search, showrooming, and retailer variety," *Marketing Science*, 42 (2), 251–270. [28]

BOURREAU, MARC AND GERMAIN GAUDIN (2022): "Streaming Platform and Strategic Recommendation Bias," 31 (1), 25–47. [28]

BRYNJOLFSSON, ERIK, YU (JEFFREY) HU, AND DUNCAN SIMESTER (2011): "Goodbye Pareto Principle, Hello Long Tail: The Effect of Search Costs on the Concentration of Product Sales," 57 (8), 1373–1386. [15]

CALVANO, EMILIO, GIACOMO CALZOLARI, VINCENZO DENICOLO, AND SERGIO PASTORELLO (2020): "Artificial intelligence, algorithmic pricing, and collusion," *American Economic Review*, 110 (10), 3267–3297. [4]

——— (2023): "Calvano Calzolari Denicolò Pastorello Supplementary appendix for "Artificial Intelligence, Algorithmic Recommendations and Competition"," *Harvard Dataverse. DOI: 10.7910/DVN/KBDJOW*. [4]

CALVANO, EMILIO AND BRUNO JULLIEN (2018): "Can We Trust the Algorithms That Recommend Products Online? A Theory of Biased Advice," . [28]

CASTELLINI, JACOPO, AMELIA FLETCHER, PETER L ORMOSI, AND RAHUL SAVANI (2023): "Recommender Systems and Competition on Subscription-Based Platforms," SSRN working paper. [4]

DE CORNIERE, ALEXANDRE (2016): "Search advertising," *American Economic Journal: Microeconomics*, 8 (3), 156–188. [21]

DE CORNIÈRE, ALEXANDRE AND GREG TAYLOR (2019): "A Model of Biased Intermediation," 50 (4), 854–882. [28]

FLEDER, DANIEL AND KARTIK HOSANAGAR (2009): "Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity," *Management science*, 55 (5), 697–712. [15]

HAGIU, ANDREI AND BRUNO JULLIEN (2011): "Why Do Intermediaries Divert Search?" 42 (2), 337–362. [28]

JANNACH, DIETMAR AND MICHAEL JUGOVAC (2019): "Measuring the Business Value of Recommender Systems," 10 (4), 1–23. [2]

JOHNSON, JUSTIN P, ANDREW RHODES, AND MATTHIJS WILDENBEEST (2023): "Platform design when sellers use pricing algorithms," *Econometrica*, 91 (5), 1841–1879. [4]

JULLIEN, BRUNO AND ALESSANDRO PAVAN (2019): "Information management and pricing in platform markets," *The Review of Economic Studies*, 86 (4), 1666–1703. [26]

KLEIN, TIMO (2021): "Autonomous algorithmic collusion: Q-learning under sequential pricing," *The RAND Journal of Economics*, 52 (3), 538–558. [4]

KNIGHT, FRANK HYNEMAN (1923): *The ethics of competition*, Transaction Publishers. [31]

KOREN, YEHUDA, ROBERT BELL, AND CHRIS VOLINSKY (2009): "Matrix Factorization Techniques for Recommender Systems," 42 (8), 30–37. [5]

LANCASTER, KELVIN ET AL. (1974): *Socially optimal product differentiation*, UC Berkeley. [7]

LEE, GUNHAENG AND JULIAN WRIGHT (2021): "Recommender Systems and the Value of User Data," . [4]

LEE, KWOK HAO AND LEON MUSOLFF (2023): "Entry into two-sided markets shaped by platform-guided search," *Working Paper*. [4]

PEITZ, MARTIN AND ANTON SOBOLEV (2022): "Inflated recommendations," . [28]

POSSNIG, CLEMENS (2023): "Reinforcement learning and collusion," *University of British Columbia*. [4]

ROKACH, LIOR, BRACHA SHAPIRA, AND FRANCESCO RICCI (2022): *Recommender systems handbook*, Springer. [5]

TEH, TAT-HOW AND JULIAN WRIGHT (2022): "Intermediation and Steering: Competition in Prices and Commissions," 14 (2), 281–321. [28]

WOLINSKY, ASHER (1983): "Retail trade concentration due to consumers' imperfect information," *The Bell Journal of Economics*, 275–282. [7]

——— (1986): "True monopolistic competition as a result of imperfect information," *The Quarterly Journal of Economics*, 101 (3), 493–511. [11]

ZHONG, ZEMIN (2023): "Platform search design: The roles of precision and price," *Marketing Science*, 42 (2), 293–313. [21]

ZHOU, JIDONG (2022): "Improved Information in Search Markets," *mimeo*. [21, 26]