Ranking Algorithms and Equilibrium Prices

Yufeng Huang — University of Rochester
Malika Korganbekova — University of Chicago
Aliya Korganbekova — Harvard University

December 5, 2025

CLICK HERE FOR MOST RECENT VERSION

Abstract

Ranking algorithms commonly reward products with strong past sales performance, creating a tension where high rankings may confer market power and raise prices, or may accelerate algorithm learning and intensify dynamic price competition as sellers invest in future ranks. We study an experiment on a major U.S. platform that randomly boosted rankings for a subset of products. Treated products reduce wholesale prices by 2.3%, with price cuts concentrated among younger products where the algorithm updates fastest—suggesting that the investment incentive creates intense competition even at high ranks. We estimate a dynamic pricing model and show it reproduces, out of sample, the observed treatment effect on prices. We demonstrate that platforms can exert substantial influence on the degree of competition through algorithm design: adjusting a single parameter shifts average prices by up to 8%.

Keywords: ranking algorithms, dynamic pricing, competition, platform design

1 Introduction

Algorithms increasingly mediate market outcomes on digital platforms, determining which products consumers see and how competition unfolds. Policymakers and researchers have grown concerned that these algorithms may distort competition, entrench incumbent sellers, and enable market power. Yet, empirical evidence on how these algorithms actually affect equilibrium market outcomes remains limited.

On e-commerce platforms, established sellers and products dominate top ranking positions while new entrants struggle to gain visibility. A natural concern arises: ranking algorithms contribute to market concentration by favoring products with strong historical sales performance, creating barriers to entry, and potentially granting incumbents market power.[1] In this paper, we present experimental evidence showing that a common-in-industry ranking algorithm in fact intensifies competitive pressure even among highly-ranked sellers. We then study counterfactual ranking algorithms' impact on market equilibrium, showing that the platform can actively shape market competition through its ranking algorithm design.

Empirical research on the equilibrium impact of ranking algorithms faces two obstacles. First, algorithms are typically black boxes with unknown or intractable structure, making it difficult for researchers to characterize how rankings depend on algorithm inputs (such as past impressions and sales quantities). Second, even when the algorithm's structure is known, identification of its effect is difficult. Algorithm inputs and outputs are endogenous equilibrium objects, so the algorithm's causal effect is confounded with underlying demand and supply conditions.[2] Empirical analysis requires both observable algorithm structure and exogenous variation in its inputs and outputs.

We study the impact of ranking algorithm on market equilibrium on a major U.S. e-commerce platform, focusing on one consumer durable goods category. Crucially, we observe the algorithm's structure: it ranks products by their posterior mean conversion rate—the expected probability that an impression leads to a purchase—updated through Bayesian learning as impression and sales quantity data accumulate. The algorithm incorporates a forgetting factor that places higher weight on more recent sales performance data. We observe the full history of algorithm's input and output (prior and posterior mean conversion rate and

[1] Even platform managers themselves share this concern. For example, Amazon's design of ranking algorithm acknowledge the need to deliberately give new products more exposures to give them an opportunity to climb up the ranking. See Han et al. (2022).

[2] Also, experiments that completely changes the algorithm (when the market has to re-equilibrate) are prohibitively costly for the platform.

variance), and we leverage experimental variation in the algorithm's prior, thereby allowing us to measure the algorithm's causal effect on consumer and firm decisions.

We first present experimental evidence on the algorithm's causal effect on demand and supply. The platform induces a large, permanent increase in the algorithm's prior for a random subset of products. We verify that this intervention affects treated products without shifting the broader market equilibrium. We find that, on the demand side, treated products experience substantial increases in ranking position and impressions, leading to higher sales quantities. On the supply side, sellers respond by reducing wholesale prices by 2.3%. This finding implies that ranking improvement does not confer market power. Instead, competition among top-ranked products appears more intense than among lower-ranked products.

We then present evidence that pins down the core mechanism: competition is more intense at higher rankings because the higher inflow of impressions amplifies the rate of algorithm updates, leading to a stronger responsiveness of future rankings to current conversion outcomes, and thus creating a stronger downward pricing pressure from the incentive to invest in future rankings. Consistent with this mechanism, price reductions concentrate among younger products—those for which the algorithm has accumulated limited information and thus responds more strongly to realized sales. We rule out two competing explanations—demand is not more elastic at higher rankings, and price responses are similar across sellers of varying experience and scale—thus showing that changes in consumer segmentation or sellers' lack of experience do not appear to drive the experiment's findings.

The intense competition at the top arises purely from the structure of the ranking algorithm, which gives the platform a key lever to shape market competition, even though the platform does not set prices directly. How much can the platform influence equilibrium prices through its ranking algorithm?

To answer this question, we develop and estimate a dynamic equilibrium model of seller pricing under ranking algorithms. Consumer impressions arrive at a rate that increases with ranking, and they make rational purchase decisions. The Bayesian ranking algorithm learns each product's conversion rate from past impressions and conversion outcomes, and ranks all products in descending order of the posterior mean. Forward-looking sellers maximize discounted sum of profits, recognizing that current prices affect future rankings through the ranking algorithm. We estimate demand and ranking algorithm parameters using observed impressions, sales quantities, and algorithm posteriors. Importantly, we estimate sellers' marginal costs using pre-treatment data, and demonstrate that the estimated model can reproduce the observed treatment effect on prices "out of sample," lending further credibility that the algorithm's structure indeed shapes equilibrium competition and explains the experimental results.

We use the model to quantify the platform's influence over market outcomes. Specifically, we keep the algorithm's structure constant and vary only its forgetting rate. The forgetting rate captures the persistence of historical realized outcomes on future rankings. A low forgetting rate implies that top ranking positions are persistent and valuable, but also that reaching such positions requires sustained price cuts as investments. A high forgetting rate implies rankings reset quickly, reducing the return to investment, but it also means that a high ranking position is less valuable. We simulate equilibrium outcomes under a full range of forgetting rates and find that a moderate forgetting rate—one that implies a half-life of information of 23 days—generates the strongest price competition on the platform, lowering average stationary-equilibrium prices by 8%. Extremely high or low forgetting rates yield prices close to the static benchmark, where the investment channel is absent. Therefore, through ranking algorithm design alone, the platform can exert significant influence on market average prices without setting them directly.

Our findings carry implications for both platforms and policymakers. Algorithm design substantially affects equilibrium prices and competitive intensity. Platforms can shape competition through their technical design, and one key lever is the rate at which historical information decays. For policymakers, rather than demanding full algorithmic transparency—often infeasible or uninformative—regulators could require platforms to report summary measures that capture competitive effects, such as the persistence of historical data in influencing current ranks. Our findings offer new insights about ranking algorithm's structure and implications, which are completely absent from current regulatory frameworks.

1.1 Related literature

Our paper contributes to the growing literature on how platform design shapes market structure, with particular emphasis on search frictions, ranking algorithms, and seller incentives. Broadly, we connect three strands of work: (i) empirical and theoretical research on platform search and visibility design; (ii) studies of dynamic seller incentives and platform-mediated signaling in markets with cold-start problems; and (iii) the classic literature on search frictions and price competition.

Platform search design and ranking algorithms. A large literature studies how platforms' ranking and search design affect consumer search and market outcomes. The closest paper to our work is Dinerstein et al. (2018), who study a large-scale redesign of eBay's search interface and show how platform search design shapes both consumer search patterns and price competition.[3] We differ in mechanism: whereas their platform allocates visibility based on prices and product characteristics in a static way, we study a Bayesian ranking algorithm that dynamically updates based on conversion outcomes and generates forward-looking pricing incentives through its learning structure.

Our work also builds on empirical studies of ranking rules and visibility allocation, including Ursu (2018), who quantifies the causal effect of ranking positions on online consumer search and purchases in a sequential search framework, and Choi and Mela (2019), who model how product rankings and sponsored listings shape both consumer search and sellers' advertising decisions on an online marketplace. In addition, Jin et al. (2023) analyze how a sales-based ranking algorithm on a major e-commerce platform creates incentives for merchants to engage in "brushing" (fake orders) to improve rank, and show that the design of the ranking rule—such as whether it depends on cumulative or recent sales—can have non-monotone effects on consumer welfare. These papers highlight that ranking rules can meaningfully affect search, visibility, and welfare. Our contribution is to show, using experimental and structural evidence, that a Bayesian learning-based ranking algorithm can itself intensify price competition among top-ranked sellers by linking current pricing decisions to future visibility.

Dynamic incentives, learning, and platform-mediated signaling. Our results also relate to the literature on platform-provided signals and market design in environments with cold-start frictions. Hui et al. (2022) study a certification policy on a large online marketplace and show that platform-issued badges alter both entry and the distribution of seller quality by mapping past performance into a persistent, platform-mediated signal. Our mechanism is similar in spirit: the ranking algorithm implicitly acts as a performance-based credential that converts good past outcomes into future traffic. It differs in two key respects, however. First, in our setting consumers do not update beliefs about unobserved quality from rankings; rankings matter only through exposure. Second, the ranking rule is explicitly Bayesian and dynamic, so sellers' current pricing choices affect the evolution of their future visibility via the

[3]They model platform design as a visibility rule that can increase effective price elasticities by giving greater prominence to low-price offers, and estimate the impact of eBay's redesign on browsing and transaction prices.

learning process itself.

More broadly, our paper contributes to research on dynamic seller incentives in markets with reputation or feedback systems. Cabral (2010) shows that on eBay, negative feedback has large and persistent effects on sellers' sales and exit behavior, highlighting the dynamic consequences of reputation accumulation. Klein et al. (2016) exploit a major change in eBay's feedback system to distinguish adverse selection from moral hazard and show that greater transparency in feedback can improve market performance by influencing sellers' behavior. Tadelis (2016) surveys this literature and emphasizes how reputation and feedback mechanisms provide incentives for high effort by linking future demand to past performance. These papers focus on how reputation affects demand directly, through buyers' beliefs. In contrast, we study how past conversion outcomes feed into an algorithmic ranking state, and how this state in turn shapes sellers' dynamic pricing incentives even when consumers do not interpret rankings as quality signals.

Finally, our framework connects to the broader literature on information markets and the design of signals and scoring rules. Bergemann and Bonatti (2022) survey models in which intermediaries optimally design information structures—such as ratings and scores—to influence market participants' behavior and equilibrium outcomes. Our contribution is complementary: rather than characterizing an optimal signal, we take a specific Bayesian ranking rule used by a large e-commerce platform and show empirically and structurally how its mechanical learning dynamics generate strong incentives for sellers to invest in future rankings through current prices.

Search frictions and price competition. Finally, our work contributes to the classic literature on search frictions and pricing, beginning with Stigler (1961), who highlights how costly information acquisition can rationalize persistent price dispersion. Subsequent models of consumer search and oligopolistic pricing (e.g., Varian, 1980; Burdett and Judd, 1983; Stahl, 1989) formalize how search costs and heterogeneous information can generate market power and equilibrium price dispersion even in homogeneous-product markets.

On the empirical side, Hortaçsu and Syverson (2004) use the S&P 500 index fund market to quantify the role of search costs and non-price differentiation in sustaining fees above the frictionless benchmark, while De los Santos et al. (2012) use detailed web browsing and purchase data to test and reject standard sequential-search models in favor of fixed-sample search. Ellison and Ellison (2009) show how firms in an online environment use obfuscation strategies and loss-leader pricing to interact with price search engines, documenting that search frictions and prominence can meaningfully affect price elasticities and equilibrium prices. Our contribution is to show that modern platforms' use of Bayesian ranking algorithms creates a *dynamic* competitive force that is absent from static search models: even holding search frictions fixed, the algorithm's updating rule ties current pricing decisions to future visibility, thereby altering sellers' incentives and equilibrium price competition.

Methodologically, our dynamic model of seller behavior is related to the empirical IO literature on dynamic oligopoly and investment, particularly work that estimates dynamic games of industry competition such as Aguirregabiria and Ho (2012) and the broader framework surveyed in Aguirregabiria et al. (2021). A key distinction is that in our setting the state variable that governs future payoffs is generated by the platform's Bayesian ranking algorithm, rather than by firms' physical capital, installed base, or network size.

Summary. Taken together, our paper is among the first to empirically identify a dynamic pricing mechanism induced by a real-world ranking algorithm. Whereas prior work emphasizes how rankings shape consumer search or how platform signals shape credibility, we show that the algorithmic learning rule itself can intensify competition—even among top-ranked sellers—and that platforms can meaningfully shift market equilibrium through technical design parameters such as the forgetting rate.

2 Context, Data, and Facts

2.1 Platform Setting

We study a U.S. e-commerce platform, focusing on one consumer durable goods category with 106,000 products in total. Consumers typically make one-time rather than repeat purchases. They navigate the platform through ranked product listings, entering either through search queries on specific keywords or by browsing category and subcategory pages. Figure 1 shows an example of a ranked product list from Google Shopping as an illustration, where products ranked at the top of the page are more likely to be seen and considered by consumers.

Once presented with the ranked list, consumers can click on products to view details, add items to their shopping cart, and complete purchases. Products shown on a consumer's screen as they browse (when browsing has reached that product) are recorded as impressions. The

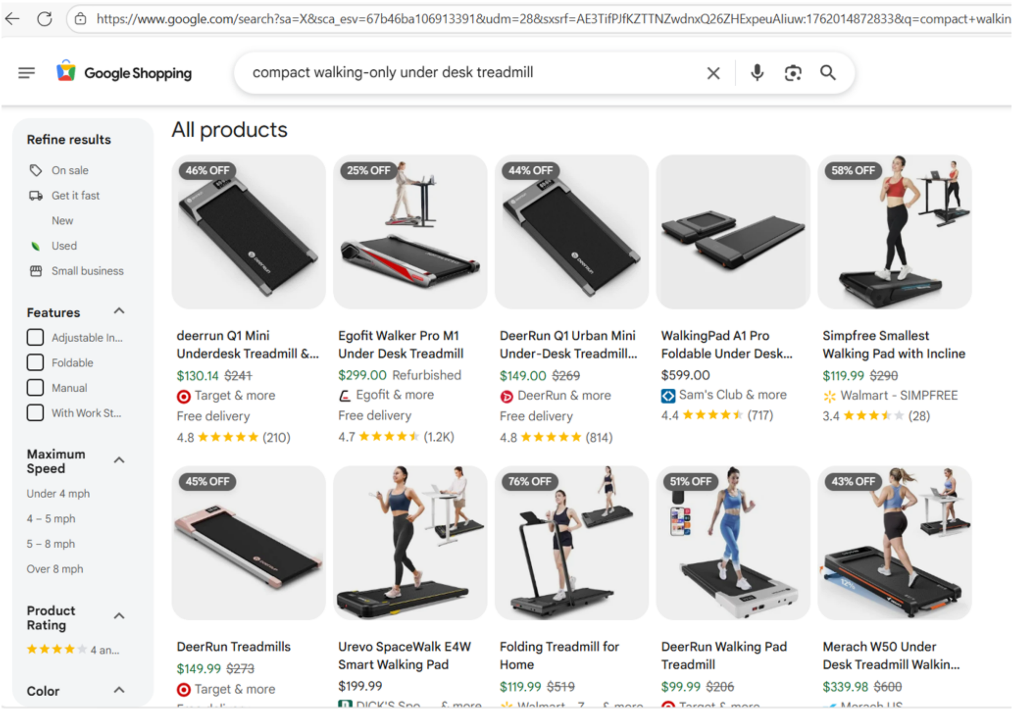


Figure 1: An Illustration of Platform ranking interface (Google Shopping)

ratio between the number of purchases and impressions is the conversion rate. In other words, the conversion rate is the purchase probability of an average impression.

The sellers on this platform are experienced third-party suppliers. Sellers set wholesale prices, and the platform adds a percentage markup, which is determined by a clear rule tied to demand elasticity. Unless otherwise stated, we will refer to the wholesale prices sellers set as "prices." There is no personalized ranking and no sponsored search advertising in the focal category during our sample period.

2.2 Ranking Algorithm

The platform's ranking algorithm ranks products in a search query by their posterior mean conversion rate. For products that fall within a consumer's search query (or in the browsed category), the platform displays them approximately in descending order of this posterior.[4]

The algorithm updates the posterior through Bayesian learning. It combines a prior belief about the product's conversion rate with observed data on impressions and sales quantities. The algorithm starts with an optimistic prior—the prior conversion rate is set substantially

[4]In reality, the platform's actual algorithm uses Thompson sampling from the posterior distribution of conversion rates and ranks products by these samples. So in reality, ranking does still vary across consumer sessions, even those who search the same set of products. However, to maintain model tractability, we assume that the platform's Bayesian updating is closed form and that products are ranked in descending order of the posterior mean. Our key insight remains true: the algorithm updates based solely on realized quantities and impressions and not on prices and other information.

higher than the average realized conversion rate. This optimism means that new products typically begin with high rankings that decline over time as actual impression and sales quantity data accumulate. Importantly, the algorithm does not use price information when forming the posterior. The algorithm also places higher weight on recent data when updating beliefs, so recent data on sales performance has greater influence on current rankings than older data.

Sellers understand the algorithm's structure through the platform's communication and their own experience. This knowledge of how the algorithm works—that it rewards higher conversion rates, starts optimistic, and weights recent performance more heavily—explains their strategic responses to the experiment we discuss later.

2.3 Data

Our data covers the period from November 2021 to February 2023. The main sample consists of 23,500 products in the focal category launched during this period, sold by 1,831 sellers.

Our primary dataset is a panel at the product-date level, aggregating all consumer activity for each product on each day. For each product on each day, we observe the entire consumer purchase funnel: the number of impressions, clicks, add-to-cart events, and purchases. For tractability, we focus on the overall conversion from impression to purchase, abstracting from intermediate stages. As defined earlier, we compute the conversion rate as the ratio between purchases and impressions.

We also observe the algorithm's state variables, including the posterior mean conversion rate and the posterior variance. The posterior variance captures the algorithm's uncertainty about a product's true conversion rate, which will be important when we test mechanisms later. It is key to our empirical exercise to know the algorithm's Bayesian structure and observe both the algorithm's inputs and outputs—that is, how the algorithm updates from the previous day's posterior to the current posterior using the realized impressions and purchases.

We observe complete price information for each product-date. This includes the wholesale price set by the seller, the retail price faced by consumers, and the platform's markup. We mask the scale of prices and markups for confidentiality reasons. But we do use their true values in our model estimation. The data also contains product and seller characteristics.

Table 1: Summary Statistics

Variable	Mean	Median	P25	P75	N
Panel A: Product-Date Level					
Impressions	183.346	32	11	94	3,607,156
Clicks	8.638	0	0	2	3,607,156
Add to cart	1.514	0	0	0	3,607,156
Orders	0.015	0	0	0	3,588,589
Conversion rate (%)	0.003	0	0	0	3,447,023
Panel B: Seller Level					
Products per seller (in-sample)	17.309	6	2	17	1,831
Products per seller (pre-sample)	104.926	45	16	110	1,426
Tenure (days, pre-sample)	1033.638	894	651	1134	1,426

Notes: Panel A presents statistics at the product-date level. Conversion rate is defined as (orders / impressions) × 100. Panel B presents statistics at the seller level for 1,831 sellers. Products per seller (in-sample) only counts products in-sample products, which does not overlap with products per seller (pre-sample).

2.4 Stylized Facts

Table 1 reveals two key patterns. First, sellers are large and experienced. On average, they carry 105 products and have 1,033 days (about 2.8 years) of tenure before the sample period. Among the focal set of 23,500 products launched during the sample period, the average seller carries 17 (median is 2). Sellers are multi-product firms who are familiar with business on the platform, including how the algorithm works.[5]

Second, the market environment is characterized by low activity per product, with a thin conversion funnel. On average, products receive 183 impressions per day, with an overall conversion rate of only 0.003%, or approximately one unit sold per 33,000 impressions. Given these modest impressions and low conversion rates, products strive to reach the top of rankings where they can capture most consumer consideration.

Indeed, Figure 2 shows that sales revenue concentrates heavily in top-ranked products. Products ranked 1-50 capture the vast majority of revenue, with a sharp drop after rank 50. This sharp drop coincides with the typical first-page boundary, suggesting that while consumers do browse and consider most products on the first page, they do not often go beyond it.

Given the typical consideration set size, new products start at a disadvantage. For products

[5] Sellers do not observe exact details of the algorithm parameters and other technical details of the algorithm. However, they do understand the algorithm at a high level.

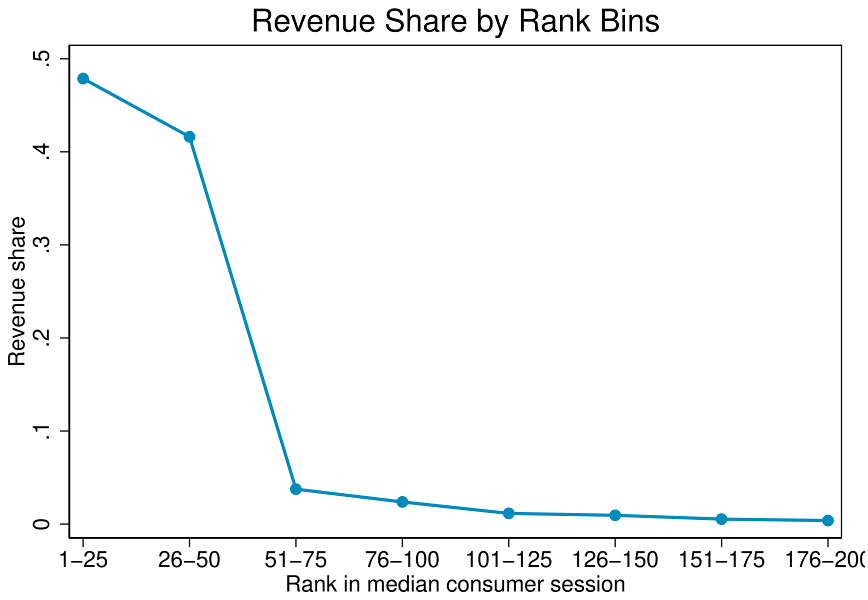


Figure 2: Revenue share by rank bins

Notes: This figure shows the share of total revenue captured by products in different rank bins. Rankings are based on the median consumer session rank.

in their first 14 days since launch, the average rank is 328, and the median is 113. These positions place new products outside the high-traffic area, requiring them to work their way up to page one before they can generate meaningful sales.

However, new products already start with an optimistic prior from the algorithm, so the vast majority of their rankings do not move up, and instead, gradually move down. Figure 3 shows the evolution of the algorithm's posterior conversion rate for products during their first 40 weeks on the platform. The platform's ranking algorithm uses a substantially optimistic prior for new products—the first week's posterior conversion rate is 0.06%, significantly higher than the average true conversion rate of 0.003%. As such, realized conversion rates are typically much lower than the prior, so the vast majority of products see declining posteriors and rankings over time. A small subset of products do maintain or improve their ranking—for example, the 90th percentile of the posterior distribution goes up over time.

Together, these figures highlight two important facts. First, maintaining and improving ranking is critical for sellers. Second, achieving high ranking is difficult—most products experience declining posteriors over time as their actual conversion rates fall short of the algorithm's optimistic prior. Given the algorithm's structure—rewarding higher conversion rates and weighting recent performance more heavily—sellers may use strategic pricing to influence (or invest in) their future rankings. Understanding this investment incentive through the algorithm's structure motivates the model and the subsequent empirical analysis.

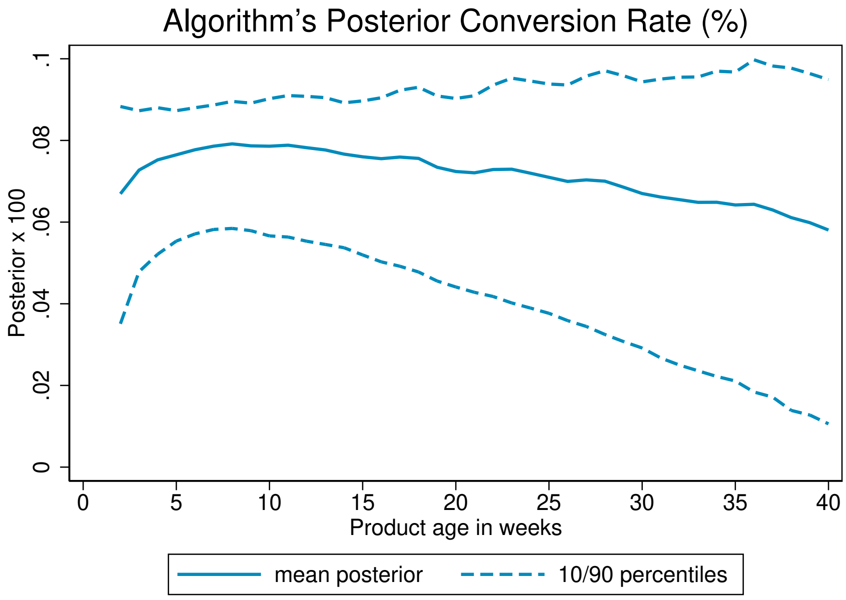


Figure 3: Evolution of algorithm's posterior conversion rate for new products

Notes: This figure shows the mean, 10th percentile, and 90th percentile of the algorithm's posterior conversion rate for new products over their first 40 weeks on the platform. The lines represent point estimates from regressions of posterior conversion rates on the number of weeks since product launch (in dummies), controlling for product and year-month fixed effects.

3 Model

We now present a model to conceptualize how ranking algorithms affect equilibrium prices through sellers' dynamic investment incentives. The model consists of three components—consumer demand, the ranking algorithm, and seller pricing decisions—that jointly determine market outcomes. This framework generates testable predictions about how sellers respond to changes in their ranking position, which we will test in the subsequent section. We will later parameterize and estimate this model and use it to quantify counterfactual platform policies.

3.1 Consumer Demand

Consumers arrive at product pages through the platform's ranking system following a Poisson process. Focus on one product, but suppress its subscript for expositional simplicity. The arrival rate of impressions $\lambda(r_t)$ depends on the product's rank r_t on day t. Each impression represents a consumer who reaches the focal product on the product page, and $s_t(p_t, r_t)$ denotes their average purchase probability—or conversion rate. The time subscript on $s_t(\cdot)$ captures the possibility that the residual demand of one product might depend on that day's market conditions (e.g., prices and availability of other products), something we will assume away in the stationary equilibrium. The product's expected demand thus equals

$$\mathbb{E}[q_t] = \lambda(r_t) \times s_t(p_t, r_t), \tag{1}$$

where ranking affects quantity sold through both the number of impressions and the composition of consumers reached at different ranking positions.

It is obvious that higher-ranked products (a higher r_t) should receive more impressions. However, it is less obvious how the conversion rate changes with rank r_t. This is because the composition of consumers reached at different ranking positions can vary systematically, but the way it varies could depend on consumer search behavior. For example, consumers who decide not to scroll down might be less price-sensitive or have higher opportunity costs of time, implying that their choice probability $s_t(p_t, r_t)$ may be less elastic to price. However, it is equally plausible that consumers who persist in scrolling to lower ranks have more specific preferences or niche needs, leading to less elastic demand at lower rank positions. The net effect on demand elasticity across ranks is therefore an empirical question.

3.2 Ranking Algorithm

The platform updates its belief about products' true conversion rate and ranks products in descending order of their posterior means. For tractability, assume the algorithm holds the prior belief that the true conversion rate (a product-specific constant $s \in [0, 1]$) follows a Beta distribution $s \sim Beta(Q_0, N_0 - Q_0)$. It updates the posterior using Binomial signals. On date t, it observes a series of n_t impressions, among which q_t choose to purchase, and $n_t - q_t$ do not. One can derive that the posterior belief also follows a Beta distribution, with posterior mean equal to

$$\hat{s}_t = \frac{Q_0 + \sum_{\tau=1}^{t} q_\tau}{N_0 + \sum_{\tau=1}^{t} n_\tau}. \tag{2}$$

This posterior mean has a natural interpretation, that the platform's perceived conversion rate is the fraction of historical impressions that lead to a purchase, adjusted for the prior N_0 "hypothetical" impressions that lead to Q_0 "hypothetical" purchases.

Having formed the posterior mean, the platform ranks products in descending order. Therefore, we model a product's rank as the empirical cumulative distribution function (CDF) of all products' posterior conversion rates, or $r_t = F_t(\hat{s}_{t-1})$.

In reality, the algorithm incorporates forgetting, placing higher weight on recent data. As we elaborate in Section 5, we model this as a geometric forgetting factor, where the cumulative impressions and sales are discounted by a factor $\rho < 1$ each period.

3.3 Seller Pricing in Stationary Equilibrium

In principle, all products' demand, algorithm belief evolution, and pricing incentives jointly determine their equilibrium pricing strategies. However, given the thousands of sellers and tens of thousands of products, each product's price has little impact on own and competitor behavior and outcomes. As such, we model individual product's residual demand and stationary equilibrium pricing strategy given consumers' and other products' equilibrium actions.

In the stationary equilibrium, we can drop the time subscript in F, assuming stationary equilibrium distribution of posterior mean conversion rate as fixed, so that each product's change in posterior mean simply moves its percentile in the fixed distribution. This assumption also motivates us to drop the time subscript in the residual demand function, leading to a time-invariant residual demand function $s(p_t, r_t)$.

Given the algorithm's structure, current ranking, $r_t = F(\hat{s}_{t-1})$, evolves following a Markov process conditional on current price p_{t-1}, posterior mean $\hat{s}_{t-1}$, and the cumulative number of impressions $N_{t-1} := N_0 + \sum_{\tau=1}^{t-1} n_\tau$. This Markovian state transition thus gives rise to a Markov optimal pricing policy function on the states $(\hat{s}_{t-1}, N_{t-1})$.

Specifically, forward-looking sellers maximize the discounted sum of current and future profits, recognizing how current prices affect future ranking through the structure of the algorithm. Sellers' optimal pricing policy function is thus captured by the Bellman equation:

$$V(\hat{s}_{t-1}, N_{t-1}) = \max_{p_t} \{\pi(p_t, \hat{s}_{t-1}) + \delta \mathbb{E}[V(\hat{s}_t, N_t) | p_t, \hat{s}_{t-1}, N_{t-1}]\} \tag{3}$$

where $\pi(p_t, \hat{s}_{t-1}) = \lambda(r_t) \times s(p_t, r_t) \times (p_t - c)$ is the current flow profit. The state transition follows Poisson-Bernoulli distribution (Poisson arrival rate of impressions, and each impression converts into sales quantity with probability $s(p_t, r_t)$) and, as discussed above, is Markov.

This structure creates a dynamic investment incentive. Today's price p_t affects today's sales quantity q_t. Realized quantity updates the posterior mean $\hat{s}_t$ according to the algorithm's Bayesian learning rule, which determines tomorrow's ranking r_{t+1} and thus future demand. Lower prices today generate higher sales quantity, improving the posterior and future ranking, which increases the continuation value $V(\hat{s}_t, N_t)$. Therefore, the ranking algorithm's structure creates a tradeoff between current markups and future ranking position.

We are primarily interested in characterizing the stationary equilibrium where each seller faces a stable residual demand and stable ranking function (mapping between algorithm posterior mean and ranking). This stationary equilibrium is reached when:

1. Impressions arrive following the Poisson arrival process in $\lambda(r_t)$.
2. Sales quantities are realized following Bernoulli draws for each impression, with the purchase probability $s(p_t, r_t)$.
3. Algorithm belief $(\hat{s}_t, N_t)$ evolves given the impression and demand arrival process and given the stationary equilibrium price distribution.
4. Every product follows the same pricing strategy defined in equation (3), up to product type differences, resulting in a Markov pricing policy function $p(\hat{s}_t, N_t)$.

3.4 Testable Implications

The model generates testable predictions about how equilibrium prices respond to their ranking state and the algorithm's learning speed.

First, the response of price to the posterior mean, $\partial p/\partial \hat{s}_{t-1}$, captures how pricing changes as products move through the ranking distribution. The sign of this derivative is theoretically ambiguous. Higher $\hat{s}$ (implying better ranking) generates faster impression flow, accelerating the rate at which the algorithm learns from pricing decisions. This could strengthen investment incentives, leading to lower prices at higher rankings. However, prices' response to $\hat{s}$ and ranking also depends critically on how demand shifts and rotates with ranking—specifically, the shape of $s(p_t, r_t)$ and how the elasticity of demand varies across ranks. Therefore, how prices respond to the algorithm's posterior mean is ultimately an empirical question.

Second, the cross-derivative, $\partial^2 p/\partial \hat{s}_{t-1} \partial N_{t-1}$, captures how the price-ranking relationship varies with the algorithm's updating speed. The cumulative impressions N affects only the algorithm's updating speed but not the current flow profits. So, it would not affect prices if sellers are myopic. In addition, N affects $\partial p/\partial \hat{s}_{t-1}$ in particular ways. When N is low, the algorithm updates quickly—whether or not each impression results in a sale has a larger impact on the posterior—creating stronger investment incentives to lower prices at low N. As N increases, the algorithm posterior becomes more "settled" and, absent of a forgetting factor, optimal prices should approach static optimum. We therefore predict that the investment motive (if it exists) should be strongest when N is low.

Testing these predictions faces a fundamental identification challenge: prices, rankings (through $\hat{s}$), impressions, and quantities are all endogenously determined in equilibrium.

We cannot identify these derivatives from observational data alone. Our solution is to experimentally increase $\hat{s}$ for a small and randomly selected set of products, creating exogenous variation in the ranking state. Moreover, we leverage quasi-random variation in the timing of treatment, which means products receive the ranking boost at different values of N. This variation allows us to identify both $\partial p/\partial \hat{s}$ and the cross-derivative with N.

The next section presents the experimental design in detail. We then return to test these specific predictions and address alternative explanations for any observed price responses to ranking changes.

4 Experiment and Mechanism

4.1 Experimental Design

We leverage an experiment run by the platform to test our model's predictions about how ranking affects pricing decisions. The platform induced a large and permanent increase in the algorithm's prior for a small and randomly selected set of products. This intervention mechanically increases the posterior mean conversion rate $\hat{s}_t$ for the treated products, creating exogenous variation in ranking position that allows us to identify the causal effect of ranking on pricing decisions.

The platform randomly assigns treatment stratified by five product groups, where the groups are predetermined as a function of product characteristics that ex-ante predict conversion rates. Groups 1-3, corresponding to products with lower predicted conversion rates, have a 10% treatment probability. Groups 4-5, containing products with higher predicted conversion rates, are eventually all treated. The treatment assignment is staggered, in that about 2,000 products are treated per quarter, representing about 2% of the 106,000 total products in the category. Over the sample period from November 2021 to February 2023, approximately 11,000 products have been treated. Products launched before the sample period are never treated.

The treatment assignment is random and stratified by groups, and we confirm in Appendix Table A1 that the treatment assignment (whether a product is treated) is uncorrelated with pre-experiment product and seller characteristics. On the other hand, whereas treatment timing conditional on assignment is not explicitly randomized, we know anecdotally that the timing is not strategically decided. Indeed, we show in Appendix Table A1 that the timing conditional on treatment is uncorrelated with the same set of product and seller characteristics.

One might worry about spillover effects that would violate the stable unit treatment value assumption (SUTVA). These could manifest as either direct effects where increasing one product's ranking decreases other products' rankings, or equilibrium shifts where treatment-induced price changes affect the market equilibrium. However, spillover effects might not be of primary concern for two reasons. First, ever-treated products remain a small share of the total number of products in the category. Second, because consumers' searched product sets differ substantially, there is no clearly defined neighboring products to cause local spillover effects among product clusters. Indeed, we show in Appendix Table B2 that treatment has no effect on untreated products' rank numbers, even those that are closest in rank before treatment. Additionally, we find no differential treatment effects as more products receive treatment over time, indicating the absence of gradual equilibrium shift.

4.2 Main Effects

Effect on Posterior and Rank. The treatment—increase in algorithm prior—produces a significant impact on the posterior conversion rate as intended. Figure 4 presents the evolution of the posterior mean conversion rate 8 weeks before to 40 weeks after treatment, controlling for the same set of fixed effects as we discuss below. The algorithm's posterior conversion rate increases by 0.07 percentage points immediately upon treatment. This effect persists throughout the post-treatment period but gradually declines as the algorithm incorporates new sales quantity data that reflect actual conversion outcomes.

This posterior change translates to substantial improvements in rank. As shown in Table 2, the treatment improves rank by approximately 167 positions, moving treated products significantly higher in search results.

Effect on Impressions and Purchases. We formally estimate treatment effects using a linear two-way fixed effects specification,

$$Y_{jt} = \beta \cdot \text{Treated}_{jt} + \alpha_j + \delta_{gt} + \epsilon_{jt} \tag{4}$$

where we control for product and group-date fixed effects, α_j and δ_{gt}. The group-date fixed effects are important because assignment is stratified at the group level. We leverage staggered treatment timing between products for identification, though we can also estimate this without product fixed effects, which additionally leverages treatment-control comparisons in groups

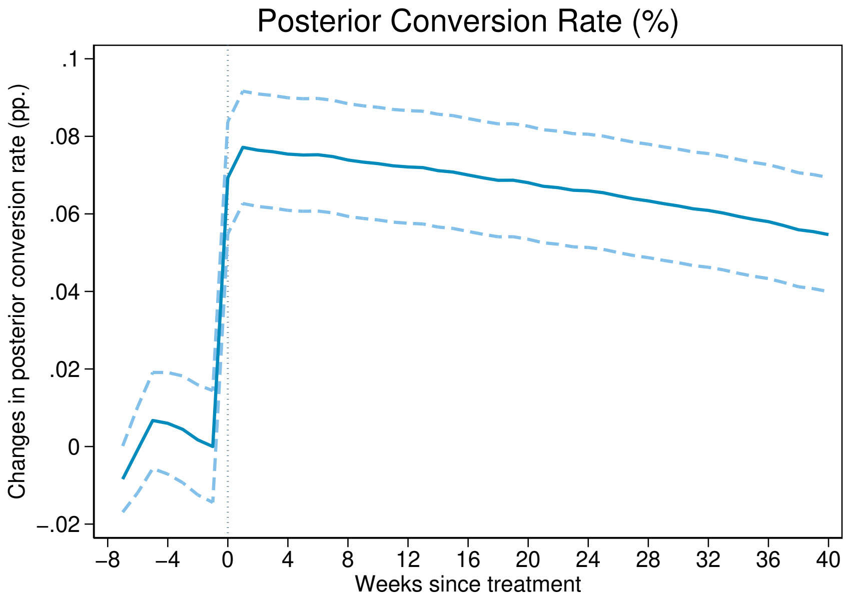


Figure 4: Evolution of Posterior Conversion Rate Around Treatment

Notes: This figure shows the event study estimates of changes in the algorithm's posterior conversion rate (in percentage points) relative to the week before treatment. The solid line represents point estimates and dashed lines show 95% confidence intervals. Week 0 denotes the first week of treatment.

1-3.

The ranking improvement generates substantial changes in consumer-side outcomes. Table 2 shows that the daily impressions increase by 158, representing a 130% increase over the pre-treatment baseline of 122.[6] Daily purchases rise by 0.010 units, a 91% increase over the baseline of 0.011. Comparing the ratio between average purchases and average impressions pre- and post-treatment, there is an indication that conversion rate is lower post-treatment, despite the increase in sales quantity driven by the higher impressions. This indication suggests that the much higher impressions might bring consumers with lower residual demand for the focal product—an important part of the core mechanism that we will examine more formally.

Effect on Wholesale Prices. Table 2 also shows that the rank changes lead to lower wholesale prices despite expanded demand from more impressions. The treatment reduces wholesale prices by 2.3%. This finding echoes our first testable implication in Section 3, confirming that $\partial p/\partial \hat{s} < 0$. One potential explanation, in line with our intuition in Section 3, is that higher traffic due to a higher rank might accelerate algorithm learning, hence intensify the incentive to lower prices to invest in future ranking. Below, we discuss this mechanism and rule out several competing explanations.

[6] The constant term is the average of inferred product and group-date fixed effects, so it can be interpreted as the average of the dependent variable absent of treatment.

Table 2: Main Treatment Effects

	Rank Number	Impressions	Purchases	Log Masked Wholesale Price
Treated	-166.702	158.089	0.010	-0.023
	(5.941)	(13.416)	(0.001)	(0.002)
Constant	276.033	121.778	0.011	–
	(2.360)	(1.567)	(0.000)	–
Observations	3,460,167	3,606,921	3,588,370	3,606,790
Product FE	Yes	Yes	Yes	Yes
Group-Date FE	Yes	Yes	Yes	Yes

Notes: This table reports the effects of the ranking treatment on various outcomes. The dependent variables are rank number (lower is better), daily impressions, daily purchases, and log masked wholesale price. Standard errors clustered at the product level are reported in parentheses. All specifications include product and group-date fixed effects.

We also find that the wholesale price response is immediate and persistent throughout the post-treatment period. Figure 5 shows that the treatment effect is stable over weeks since treatment. We also show that the treatment effect is stable over calendar quarters in Appendix Figure C1. These findings support the earlier discussion that, because the set of treated products is small, treatment does not induce gradual equilibrium shifts.

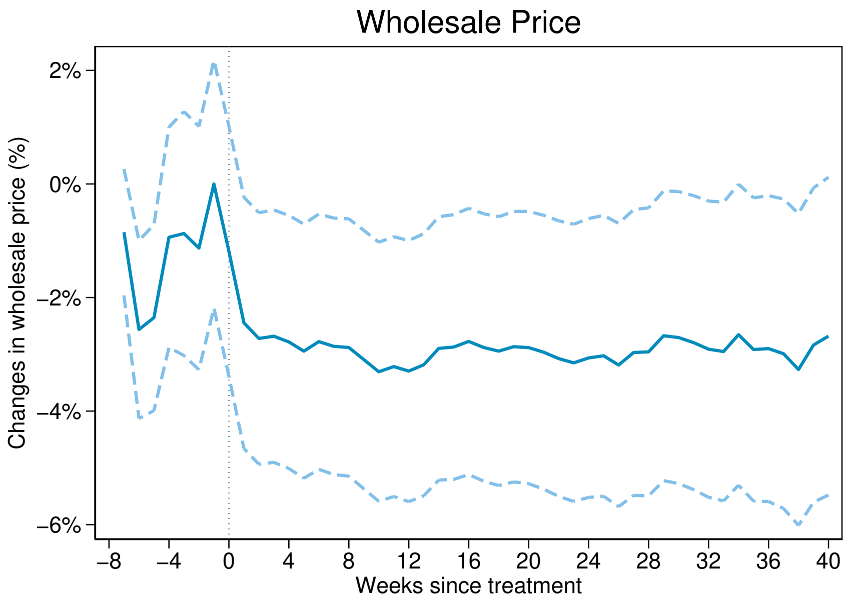


Figure 5: Wholesale Price Response to Treatment
Notes: This figure shows the event study estimates of changes in log wholesale prices (in percent) relative to the week before treatment. The solid line represents point estimates and dashed lines show 95% confidence intervals. Week 0 denotes the first week of treatment.

4.3 Primary Mechanism: Investment for Future Ranking

The primary mechanism driving sellers' price responses to the experiment stems from the algorithm's Bayesian updating structure. When sellers reduce prices, they generate higher conversion outcomes for a given level of impressions. The algorithm observes these higher conversions and increases its posterior belief about the product's conversion rate, translating into improved future rankings. This structure creates an investment incentive: sellers sacrifice current markups to build "reputation" in the eye of the algorithm, generating higher future profits through better ranking.

As we formalized in Section 3, an increase in the posterior mean $\hat{s}$ directly improves ranking, leading to a higher volume of impressions per day, and thus more algorithm updates per day. This faster updating intensifies sellers' investment incentive to reduce prices. The observed 2.3% price decrease in response to the exogenous ranking boost is consistent with this mechanism—sellers recognize that maintaining high rankings requires continuous investment through lower prices.

A related mechanism reinforces this investment incentive. Products at improved ranks face lower actual conversion rates. Consumers who stop browsing at top positions may have generic tastes and lower match value to the focal product compared to consumers who scroll deeper.[7] Additionally, neighboring high-ranked products may be inherently more attractive and hence lower the focal product's residual demand. These lower conversion rates create downward pressure on the algorithm's posterior, causing rankings to deteriorate more quickly at the same volume of impressions. The combination of faster updating (more impressions) and negative updating (lower conversion rates) strengthens sellers' incentive to cut prices to preserve their rankings.

The model in Section 3 generates a direct testable prediction: the strength of the algorithm's posterior (captured by cumulative impressions N) should moderate how prices respond to ranking changes. When N is high, new information barely moves the posterior, so an increase in $\hat{s}$ should not trigger much price cuts. When N is low, the algorithm updates quickly, so an increase in $\hat{s}$ should generate significant price reductions. While our main experimental variation is on $\hat{s}$, the staggered treatment timing provides quasi-random variation in N—products receive treatment at different ages (weeks since product launch) when the algorithm's belief varies in strength—thus allowing us to examine how the treatment effect varies by the

[7] When posterior conversion rate is lower, products are ranked lower on average and are more likely to face consumers with specific tastes, who browse subcategories or search by niche keywords.

strength of algorithm beliefs.

Figure 6 presents treatment effects by age-at-treatment quartiles. Products in the youngest two quartiles reduce prices by 3%-3.5%, whereas those in the oldest quartile reduce prices by only 1.5%. This near-monotonic decline of treatment effect with product age supports our mechanism—products treated when younger face an algorithm that updates more quickly due to its uncertainty (lower N) and thus have stronger investment incentives.

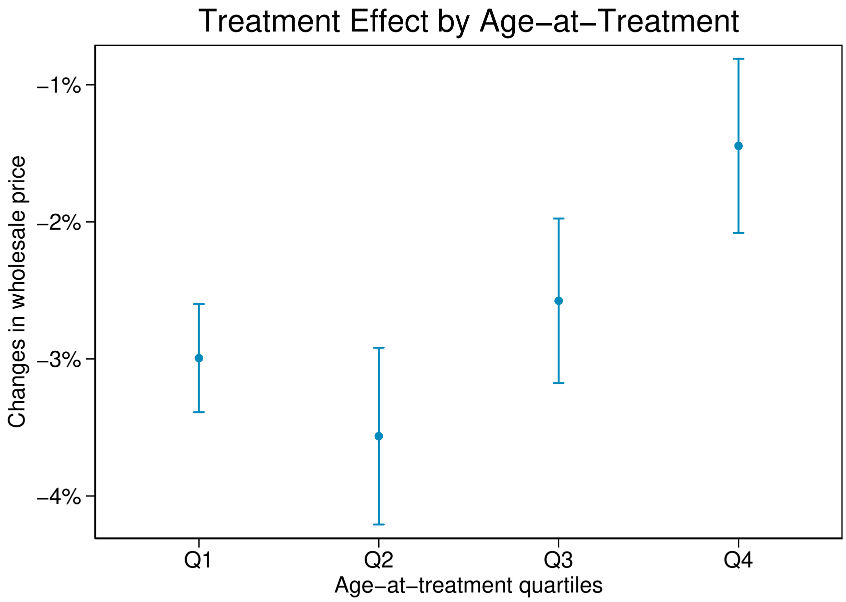


Figure 6: Treatment Effect on Wholesale Price by Product Age at Treatment
Notes: This figure shows the treatment effect on log wholesale prices by quartiles of product age at treatment assignment. Age is measured in weeks since product launch. Error bars represent 95% confidence intervals with standard errors clustered at the product level.

We also directly estimate how treatment effects vary with the algorithm's posterior variance at treatment assignment. Table 3 interacts the treatment dummy with the algorithm's posterior standard deviation at treatment and the age at treatment. The interaction with age-at-treatment shows an effect consistent with Figure 6. The interaction with algorithm uncertainty shows that a one percentage point increase in posterior standard deviation corresponds to approximately three percentage point larger price cuts. In fact, extrapolating the linear interaction would imply that products treated when the algorithm has no uncertainty (thus no updates) would not have lowered their prices at all. Although one might be concerned that posterior variance reflects unobserved product characteristics, these patterns still provide suggestive evidence supporting the investment mechanism.

Overall, the evidence shows that price responses to ranking improvements reflect forward-looking behavior that anticipate future ranking updates. Products cut prices more aggressively

Table 3: Heterogeneous Treatment Effects by Algorithm Uncertainty

	(1) × Algorithm Uncertainty at Treatment	(2) × Product Age at Treatment
Treated	-0.004	-0.026
	(0.005)	(0.002)
× Posterior Std. × 100	-2.985	
	(0.684)	
× Age (weeks)		0.001
		(0.000)
Observations	1,300,309	1,468,394

Notes: Posterior standard deviation measured at the last observation before treatment. All specifications include product and group-date fixed effects. Standard errors clustered at the product level in parentheses.

when the algorithm learns faster, indicating that investment incentives are one plausible explanation.

4.4 Alternative Mechanisms

We now examine three alternative explanations for the observed price reductions that do not involve algorithm dynamics. First, demand could become more elastic at higher rankings, leading to lower static-optimal prices. Second, the platform's markup formula might adjust to demand changes, with an increased markup pushing wholesale prices down. Third, inexperienced sellers, who are learning about the algorithm, might respond in unexpected ways. We show that none of these mechanisms can explain our main findings.

Demand Elasticity Changes. Higher ranking could change demand elasticity by altering the composition of consumers who reach the product. Products at high-ranking positions face intense local competition because neighboring products, which earned their positions organically, likely have higher quality or lower prices. This environment could make consumer residual demand more elastic to price as ranking improves. Conversely, consumers who stop browsing at top positions might have higher opportunity costs of time and care less about price, implying a less elastic demand as ranking improves. Although the net effect of ranking on price elasticity is theoretically ambiguous, a more price-elastic demand would imply lower prices absent the investment mechanism.

We estimate how demand elasticity varies with ranking using the Poisson pseudo-maximum likelihood (PPML) approach of Chen and Roth (2024), which handles the prevalence of zero quantities in our daily product-level data while preserving the elasticity interpretation of parameters. Specifically, we estimate:

$$\mathbb{E}[q_{jt}|p_{jt}, \mu_j, \delta_t] = \exp\left(\alpha_r \log(p_{jt}) \cdot \mathbb{I}_{r_{jt} \in r} + \mu_j + \delta_{gt}\right) \tag{5}$$

where q_{jt} is realized sales quantity, α_r is the price elasticity in rank bin r, and μ_j and δ_{gt} are product and group-time fixed effects.[8]

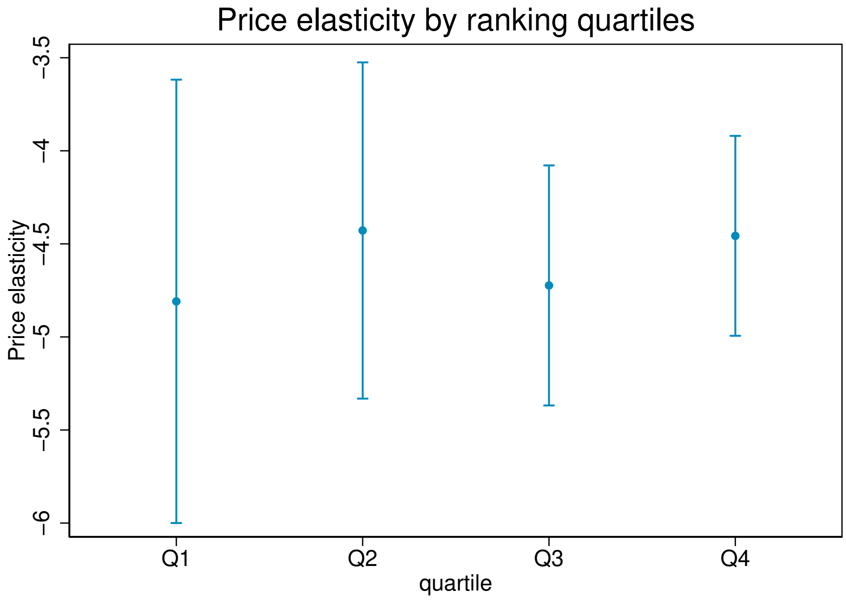


Figure 7: Estimated Price Elasticity by Rank Quartile
Notes: This figure shows the estimated price elasticities from equation (5) by rank-quartile bins. Error bars represent 95% confidence intervals with standard errors clustered at the product level.

Figure 7 shows that elasticity remains stable at approximately -4.5 across all ranking quartiles. These estimates rule out the possibility that improved rankings make demand substantially more elastic. The slight variations across quartiles are not statistically significant and, if anything, suggest marginally less elastic demand at higher ranks—directionally supporting the conjecture that consumers who stop searching early are less price-sensitive. Therefore, the observed wholesale price reductions cannot be explained by static-optimal prices' response to a more elastic residual demand.

To further validate this finding, we examine how the experimental variation in ranking affects the platform's own elasticity estimates. The platform runs separate price experiments to forecast demand elasticity using proprietary models. Table 4 shows that the platform-estimated elasticity remains constant at around -4.66 (column 1), confirming our estimation

[8] Because impressions enter multiplicatively in the demand function and are themselves not elastic to prices, we can estimate elasticity directly from the quantity-price relationship without explicitly modeling impressions.

results using observational data.

Table 4: Treatment Effects on Platform-Estimated Elasticity and Log Retail Markup

	(1) Platform-Estimated Elasticity	(2) Log(Retail price/ Wholesale price)
Treated	-0.003	0.027
	(0.006)	(0.003)
Constant	-4.662	–
	(0.001)	
Observations	3,606,921	3,588,245

Notes: All specifications include product and group-date fixed effects. Standard errors clustered at the product level in parentheses.

Platform Markup Changes. The suggestive evidence that demand might be slightly less elastic at higher ranks implies that platform markups, which depend on demand elasticity, could increase with ranking. In the vertical channel, wholesale and retail markups are strategic substitutes—when the platform increases its percentage markup, sellers optimally reduce wholesale prices in response.[9] This mechanism could potentially explain the observed wholesale price reductions without invoking dynamic investment incentives.

Column (2) of Table 4 directly tests this possibility by examining the log ratio of retail to wholesale prices. The log ratio increases by 0.027 with treatment, indicating that retail markups rise by 2.7% relative to the pre-treatment level. Whereas we cannot disclose the retail markup, we examine this mechanism more thoroughly using our structural model. There, we demonstrate that platform markup adjustments account for only a small fraction of the observed wholesale price response, with the investment mechanism explaining the majority of the effect.

Inexperienced Sellers. A final alternative explanation is that inexperienced sellers might drive the observed price reductions through various learning mechanisms. Sellers could infer their product's quality or the algorithm's behavior from the observed ranking change, or use the improved traffic to invest in lower future costs through learning by doing. Not all of these mechanisms have a clear directional prediction about the wholesale price change. But if these

[9]This intuition comes from optimal pricing with an exogenously given tax rate. Suppose final price is $p = (1+\tau)\cdot w$ where w is wholesale price and τ is a fixed percentage retail markup. Suppose seller earns profit $(w-c)\cdot q(p)$. Then, standard first-order conditions imply the optimal static wholesale price is $w^* = c - \frac{q(p)}{(1+\tau)q'(p)}$. Hence, an exogenous increase in retail markup (holding demand fixed) could lead to a reduction in wholesale price.

mechanisms were primary drivers, we would expect the price response to vary systematically with seller experience and scale.

Table 5: Treatment Effects by Seller Experience and Scale

	(1) × Tenure (years)	(2) × Annual Orders (000)	(3) × Cum. Orders (000)
Treated	-0.034	-0.024	-0.024
	(0.004)	(0.002)	(0.002)
Interaction	0.004	0.000	0.000
	(0.001)	(0.000)	(0.000)
Observations	3,072,804	2,638,660	2,638,660

Notes: Standard errors clustered at product level in parentheses. All specifications include product and group-date fixed effects.

Table 5 tests this prediction by examining how treatment effects vary with different measures of seller experience and scale—specifically, tenure (years on the platform), annual order volume, and cumulative historical order volume. The results show remarkably uniform price responses across sellers regardless of their tenure and scale. The interaction terms are economically small and explain little of the baseline treatment effect. For example, an additional year of tenure reduces the treatment effect by only 0.4 percentage points, while doubling annual orders has essentially no impact on the price response. Therefore, the behavior of inexperienced sellers does not explain the main effect.

5 Structural Estimates

5.1 Estimation Approach

We estimate the model in three steps: first consumer demand, then algorithm parameters, and finally marginal costs. This sequence allows us to use the estimated demand when recovering algorithm parameters, and both when recovering costs. We employ a method of moments approach, matching model predictions to experimental and empirical moments. Standard errors are computed via the delta method on moment gradients.[10]

Consumer demand. For product j on date t, consumer arrivals follow a Poisson process with rate $\lambda_{jt}(\hat{s}_{jt}) = \exp(\lambda_{0g} + \lambda_1 \cdot F(\hat{s}_{jt}))$, where product j belongs to group $g = 1, ..., 12$,[11]

[10]Currently, the standard errors are stage-specific.

[11]We originally identified 17 product clusters based on cost distributions, but computational constraints in solving the dynamic program with fine state space discretization limited us to 12 groups. These account for significant portion of products and significant portion of paltform revenue.

and $F(\cdot)$ transforms the posterior to its percentile rank in the distribution. Conditional on arrival, purchase probability follows a binary logit model:

$$s_g(p_{jt}, \hat{s}_{jt-1}) = \frac{\exp(u_{jt})}{1 + \exp(u_{jt})} \tag{6}$$

with mean utility:

$$u_{jt} = \beta_{0g} + \beta_1 \cdot F(\hat{s}_{jt-1}) + (\alpha_{0g} + \alpha_1 \cdot F(\hat{s}_{jt-1})) \cdot \log((1 + \Delta_{jt})p_{jt}). \tag{7}$$

The platform applies a known markup $\Delta_{jt} = \delta_0 + \delta_1 \cdot F(\hat{s}_{jt-1})$ to the wholesale price p_{jt}.

We estimate the platform markup rule as a function of ranking (i.e., parameters δ_0 and δ_1) and keep these parameters fixed. Given markup and ranking $F(\cdot)$, we then estimate main demand parameters $(\lambda_{0g}, \lambda_1, \beta_{0g}, \beta_1, \alpha_{0g}, \alpha_1)$ jointly across the 12 product groups. Each group has its own baseline parameters (λ_{0g}, β_{0g}, α_{0g}), while the ranking effects (λ_1, β_1, α_1) are common across groups. We match three sets of moments: impressions by ranking position, conversion rates by ranking position, and price elasticities by ranking position. We use the platform-estimated demand elasticities and how they change with exogenous variation in ranking.

Ranking algorithm. The platform's algorithm updates its posterior belief about each product's conversion rate using a Beta-Bernoulli model with geometric forgetting. Let q_{jt} and n_{jt} denote the sales and impressions for product j in period t. The posterior mean conversion rate evolves as:

$$\hat{s}_{jt} = \frac{Q_0 \rho^t + \sum_{\tau \le t} \rho^{t-\tau} q_{j\tau}}{N_0 \rho^t + \sum_{\tau \le t} \rho^{t-\tau} n_{j\tau}} \tag{8}$$

where N_0 is the initial prior strength and $\rho \in [0, 1]$ is the forgetting factor. The prior mean Q_0/N_0 is calibrated to match the observed average prior conversion rate in the data.

We parameterize ranking as the empirical cumulative distribution function of the previous day's posterior mean, $F(\hat{s}_{jt-1})$. We parameterize its log odds ratio, $\log(F(\hat{s}_{jt-1})) - \log(1 - F(\hat{s}_{jt-1}))$, as a third-order polynomial on $\hat{s}_{jt-1}$ and estimate it using the mapping between observed posterior and their in-sample percentile ranks. We interpret this mapping as the ranking in the current stationary equilibrium. In counterfactual experiments, we repeatedly solve for $F(\cdot)$ as we re-rank products under alternative ranking algorithms—so rankings $F(\cdot)$ arise endogenously.

To identify the algorithm parameters, we exploit experimental variation in how quickly the algorithm updates. The experiment randomly boosted products' posteriors, creating exogenous variation in the state variable $\hat{s}_{jt}$. We match four moments capturing how the posterior changes conditional on sales realizations: the average change when sales occur versus when they do not, for both new products (with limited history) and mature products (with extensive history). These moments pin down both the speed of updating (N_0) and the rate of forgetting (ρ).

Pricing and marginal costs. Sellers solve a dynamic pricing problem with state space $(\hat{s}_{jt-1}, N_{jt})$, where $\hat{s}_{jt-1}$ is the previous period's posterior mean, and N_{jt} captures cumulative impressions (affecting the speed of algorithm updating). We discretize the state space with $\hat{s}_{jt-1} \in [0, 0.3]$ in steps of 0.01 and $N_{jt} \in [0, 30]$ in steps of 2. The price grid is tailored to each group's observed price range.

The seller's value function satisfies the Bellman equation:

$$V_g(\hat{s}_{jt-1}, N_{jt}) = \max_{p_{jt}} \{\pi_g(p_{jt}, \hat{s}_{jt-1}) + \delta \mathbb{E}[V_j(\hat{s}_{jt}, N_{jt+1}) | p_{jt}, \hat{s}_{jt-1}, N_{jt}]\} \tag{9}$$

where profit is $\pi_g(p_{jt}, \hat{s}_{jt-1}) = (p_{jt} - c_g) \cdot \lambda_g(\hat{s}_{jt-1}) \cdot s_g(p_{jt}, \hat{s}_{jt-1})$ and the expectation is over tomorrow's state given today's price choice. State transitions follow the Bayesian updating rule, where the probability of a sale depends on the chosen price. We solve the dynamic program via value function iteration with daily discount factor $\delta = 0.999$.

To recover marginal costs, we match each group's observed pre-treatment wholesale prices to the model's predicted optimal prices. The first-order conditions of the dynamic optimization problem identify the marginal cost parameter c_g for each group. We reserve the post-treatment period for validation purposes, which we explain later.

Discussions. In this model, we treat two sets parameters as given but recognize they may be equilibrium objects. First, we assume the residual demand function $s_g(\cdot)$ as given. One may imagine the residual demand can change in the counterfactual equilibrium as other products change their prices. We plan to implement a future version that accounts for how other products' prices influence the residual demand function. Second, we hold platform markup fixed throughout the counterfactuals. We find limited markup changes with ranking in our experiment, suggesting that downstream markup changes in the counterfactual market design is unlikely a first-order issue.

5.2 Parameter Estimates

Demand estimates. The demand estimates reveal systematic patterns across product groups. Table 6 presents the results. Baseline log arrival rates (λ_{0g}) decrease monotonically with product cost, falling from -1.29 for group 1 to -2.94 for group 12 products, indicating substantially lower baseline traffic for the expensive, niche products. In contrast, utility intercepts (β_{0g}) increase from 22.3 for group 1 to 28.5 for group 12, suggesting that the latter groups are attractive for some consumers. Remarkably, price sensitivity (α_{0g}) remains stable across all groups at approximately -4.69, implying similar price elasticities across groups.

Table 6: Common Parameter Estimates

Parameter	Estimate	(Std. Err.)
log(arrival) $\times$ ranking (λ_1)	3.800	(0.036)
util. incpt. $\times$ ranking (β_1)	-2.336	(3.653)
log(price) $\times$ ranking (α_1)	0.015	(0.018)
init. prior strength (N_0)	1.977	(2.318)
forgetting factor (ρ)	0.994	(0.000)

Table 7: Group-Specific Demand Parameters and Marginal Costs

Group	λ_0		β_0		α_0		Marginal Cost	
	Estimate	(Std. Err.)	Estimate	(Std. Err.)	Estimate	(Std. Err.)	Estimate	(Std. Err.)
1	-1.286	(0.006)	22.343	(0.665)	-4.620	(0.002)	1.000	(0.006)
2	-1.935	(0.007)	24.357	(0.375)	-4.687	(0.005)	1.261	(0.005)
3	-2.157	(0.008)	24.906	(0.377)	-4.690	(0.005)	1.421	(0.003)
4	-2.718	(0.006)	24.941	(0.313)	-4.680	(0.001)	1.519	(0.005)
5	-2.836	(0.009)	25.831	(0.344)	-4.690	(0.002)	1.792	(0.016)
6	-2.769	(0.005)	25.646	(0.303)	-4.686	(0.003)	1.794	(0.006)
7	-2.274	(0.012)	25.380	(0.473)	-4.700	(0.003)	1.519	(0.006)
8	-2.830	(0.005)	26.252	(0.278)	-4.688	(0.002)	2.000	(0.006)
9	-3.104	(0.005)	27.239	(0.247)	-4.690	(0.002)	2.478	(0.007)
10	-2.994	(0.012)	28.414	(0.276)	-4.698	(0.002)	3.090	(0.014)
11	-3.072	(0.009)	28.104	(0.216)	-4.689	(0.004)	2.977	(0.009)
12	-2.937	(0.012)	28.452	(0.238)	-4.698	(0.000)	3.116	(0.014)

Ranking significantly affects the arrival of impressions. Table 7 shows that the log arrival rate increases strongly with rank percentiles, with a coefficient of $\lambda_1 = 3.800$, implying that products moving from the bottom to the top of rankings receive approximately 45 times more impressions (see Figure 8). However, ranking lowers the conversion rate conditional on impressions, with $\beta_1 = -2.336$, although statistically insignificant (SE = 3.653). This implies that products ranked on top would receive only 1/8 of the conversion rate compared to if it were ranked at the bottom. As discussed earlier, the combination of more impressions and lower conversion rates at top positions are key drivers of the price dynamics.

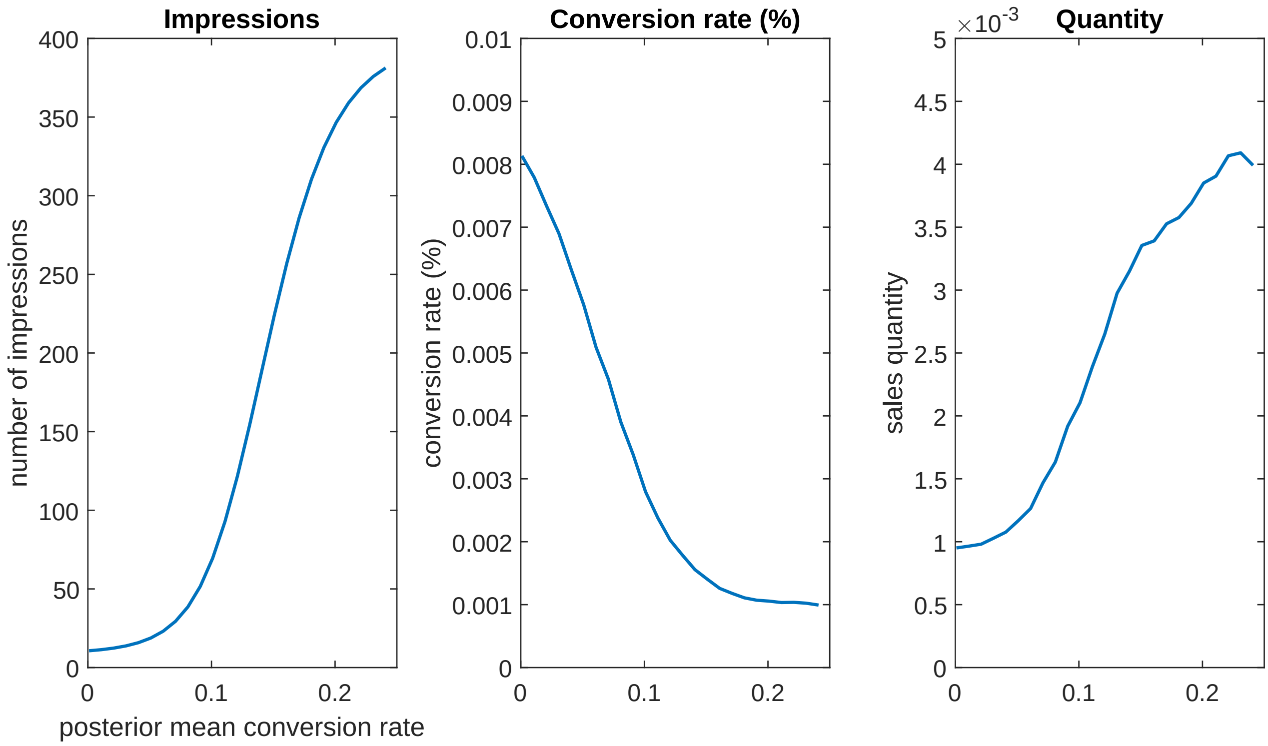


Figure 8: Decomposition of demand into arrival rate and conversion probability by ranking position

We also estimate cross-price elasticities to assess substitution patterns. In a specification similar to Equation (5) but including both own and competing products' prices, we find cross-price elasticities of 0.31, about 8% in magnitude relative to own-price elasticity (see Appendix Table C3). This estimate confirms limited substitution between products and supports our modeling strategy of treating residual demand independently. But in our counterfactual simulations, we do account for equilibrium price changes' effect on residual demand through the cross-price elasticities.

Algorithm parameters. The algorithm parameters capture how quickly the platform learns about product quality. The forgetting factor is precisely estimated at $\hat{\rho} = 0.994$, implying a half-life of information of 117 days. That is, although the platform does discount older information, it still places substantial weight on historical performance that are a few months old.

This slow decay creates asymmetric competitive dynamics. Products that achieve high rankings benefit from persistent advantages: even with poor performance, it would take months to fully lose a strong position. Conversely, low-ranked products face a steep climb that may take months of consistently strong sales to overcome.

Figure 9 illustrates these dynamics with two example products, which are randomly drawn among products with more than two transactions in the sample. The blue line shows a product that starts with low posterior beliefs but achieves several sales early on, jumping its posterior from near zero to 0.10%. However, on days when there are impressions but no sale, the algorithm does update negatively, so the high posterior essentially vanishes within a few months. The orange line shows a trajectory with a similar nature—the improved posterior from the initial two transactions takes several months to almost vanish, but jumps up towards the end of the sample due to another transaction.

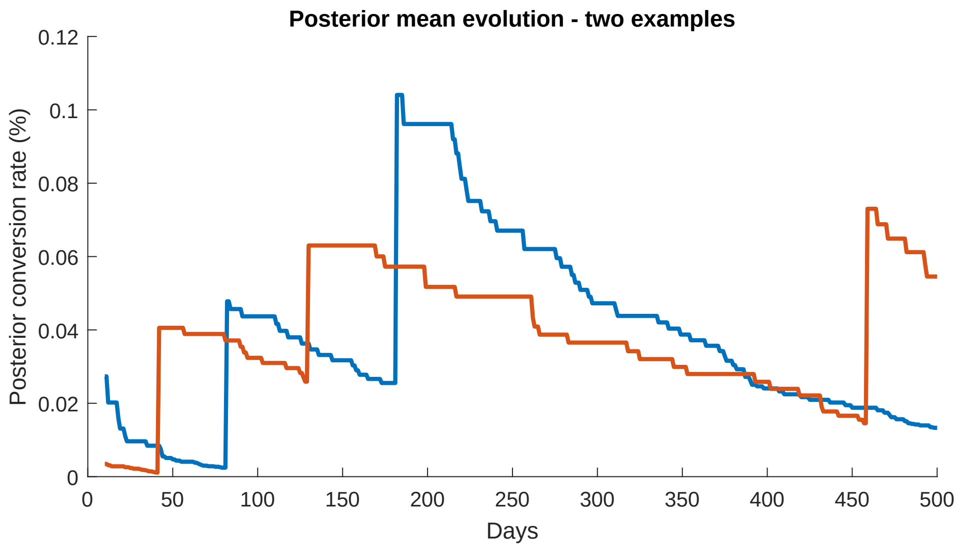


Figure 9: Evolution of posterior beliefs for two example products over time

Cost parameters. Marginal costs, normalized relative to group 1, exhibit substantial heterogeneity across product clusters. Lower group numbers generally correspond to lower cost products.

The model fits pre-experiment wholesale prices well, as shown in Figure 10. Across all groups, model-predicted prices closely match observed prices in the estimation data. This fit is almost by construction because costs are identified from matching this moment.

Model validation. To validate that our model captures the mechanism driving the experimental findings, we conduct an out-of-sample test. We simulate the model's response to the same ranking boost implemented in the experiment—an increase in the posterior of approximately 0.07 percentage points.

Given the treatment on algorithm posteriors, we trace out the pricing policy function at the new state. The model predicts an average wholesale price decrease of 2.3%, closely matching the observed experimental effect of 2.7%. Figure 11 shows this comparison, including

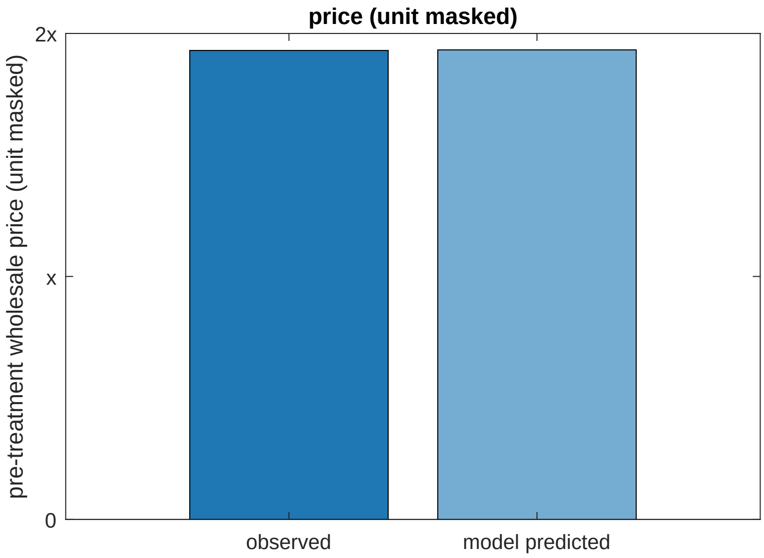


Figure 10: Model fit: Predicted versus observed pre-treatment wholesale prices (masked units)

a placebo simulation where rankings are fixed (eliminating the investment motive), which produces virtually no price change.[12]

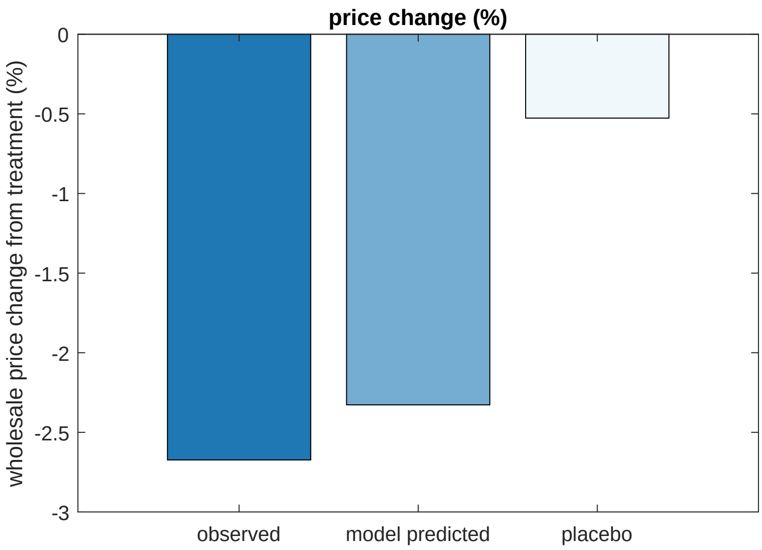


Figure 11: Model validation: Predicted versus observed wholesale price responses to experimental ranking boost

The model also captures heterogeneity in price responses across product groups. Groups 1–7, products with the lowest cost show the largest price decrease. In contrast, groups 8–12 show virtually no price decrease. The close match between model predictions and experimental outcomes, achieved without using the post-treatment data in estimation, validates our proposed mechanism of dynamic price competition through ranking algorithms.

[12]Given the lack of elasticity change, and given the fixed ranking shutting down all dynamics, the only plausible mechanism through which wholesale prices would decrease is through the increase in retail markup, Δ.

6 Counterfactual Algorithm Designs

The platform can influence the intensity of price competition through the design of the ranking algorithm. We focus on the forgetting factor ρ, which governs how the algorithm weighs recent versus historical performance. We ask: within this class of algorithms, what range of equilibrium prices can the platform induce?

The parameter ρ controls the persistence of ranking. A high ρ makes rankings persistent: top positions are difficult to reach but valuable once attained. A low ρ makes rankings volatile: top positions are easy to reach but difficult to maintain. Sharing an intuition with the dynamic pricing equilibrium in presence of network effects (Farrell and Klemperer, 2007), a moderate ρ may lead to the highest incentive to compete for (or maintain) top positions. We therefore focus on varying the forgetting factor, keeping everything else fixed at estimated values.

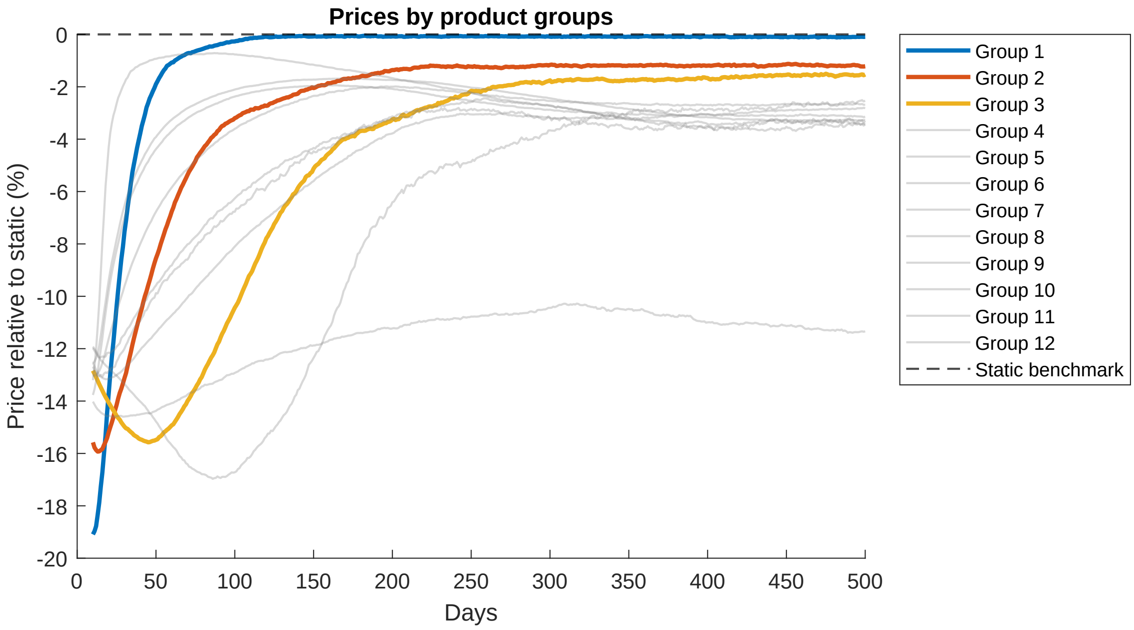
For each value of ρ, we solve for a new stationary equilibrium by iterating between two steps. First, given the empirical ranking function $r_{jt} = F(\hat{s}_{jt-1})$, we solve for sellers' optimal pricing policy function $p(\hat{s}_{jt-1}, N_{jt-1})$. Second, given the pricing policy function, we forward-simulate the daily realizations of ranking, prices, and consumer demand, starting with the empirical state distribution and for a total duration of 500 days. We then update the ranking function F to reflect how the new distribution of states map into rankings. This two-step procedure is iterated until both the pricing policy function and the ranking function converge.

We benchmark outcomes against the static optimum: prices that maximize single-period profits, ignoring future rankings. Deviations from this benchmark measure the strength of the investment channel through the ranking algorithm.

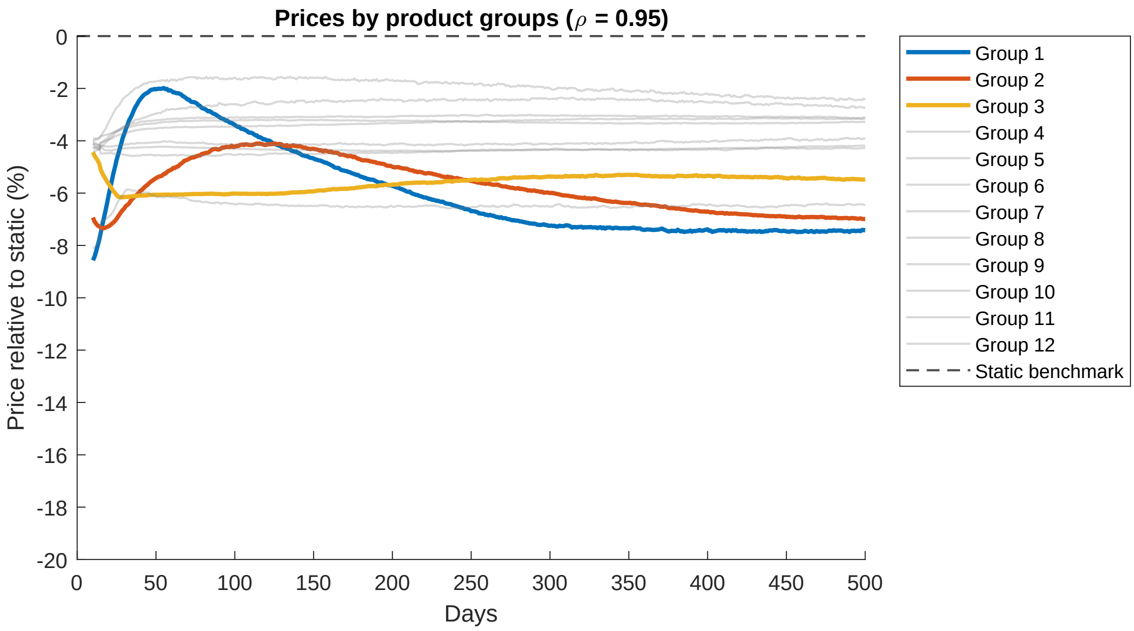
Baseline results. Figure 12 Panel (a) shows price trajectories under the estimated algorithm ($\hat{\rho} = 0.9941$, half-life of 117 days). In the short run, prices are 12–19% below the static benchmark. The algorithm is uncertain about new products, so a successful conversion has a large effect on the posterior. The return to price cuts is therefore highest early in a product's life.

In the long run, prices converge toward but do not reach the static benchmark. Most product groups price 2–4% below static. Group 1 (lowest cost, highest conversion rate) prices near static—these products maintain rankings without price cuts.

Figure 13 Panel (a) summarizes long-run outcomes. Average price is 3.0% below static, seller profit is 2.8% below static, and consumer surplus is 10.5% above static.



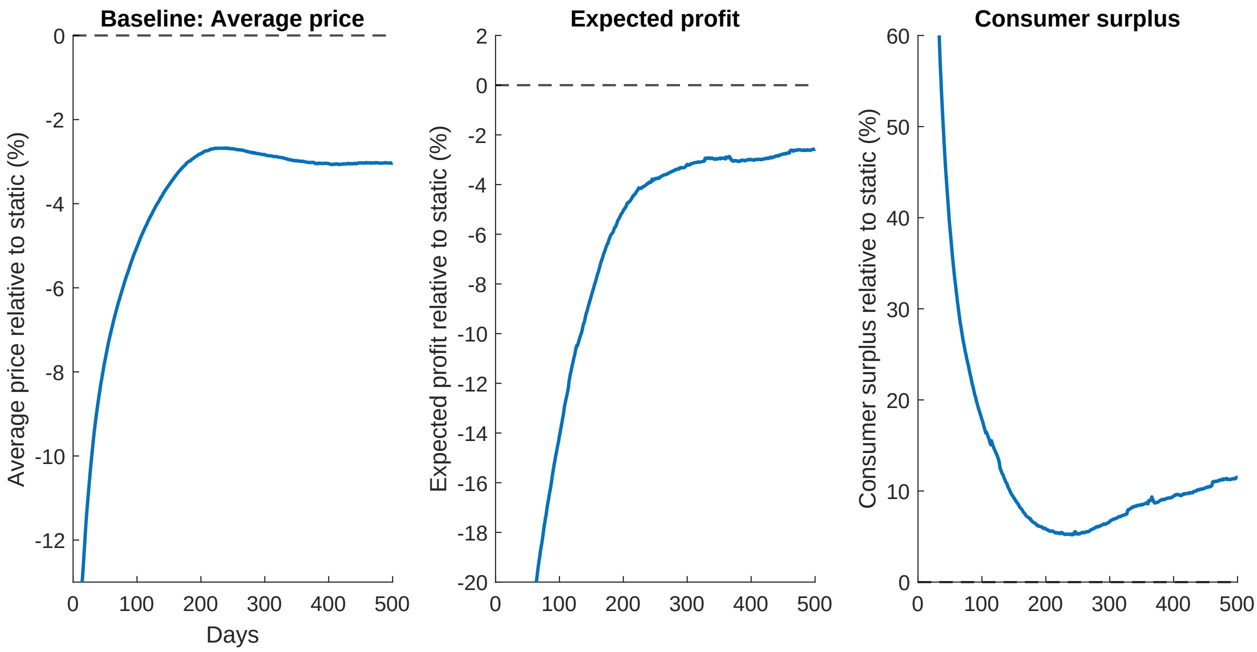
(a) Baseline algorithm ($\hat{\rho} = 0.9941$)



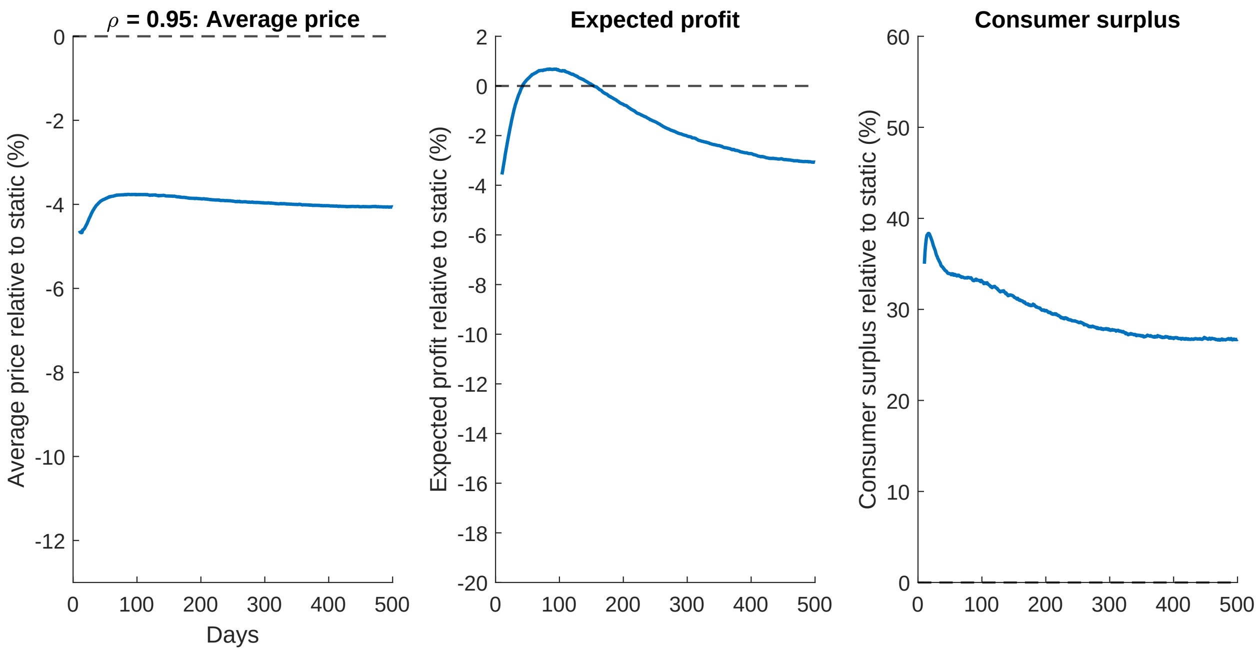
(b) Faster Forgetting ($\rho = 0.95$)

Figure 12: Price trajectories by product group

Notes. Prices expressed as percentage deviations from the static benchmark. Each line represents a product group.



(a) Baseline algorithm ($\hat{\rho} = 0.9941$)



(b) Faster forgetting ($\rho = 0.95$)

Figure 13: Market outcomes over time

Notes. All outcomes expressed as percentage deviations from the static benchmark.

6.1 Faster Forgetting

Figure 12 Panel (b) shows results for $\rho = 0.95$ (half-life of 13 days). Prices are lower than baseline throughout.

The key difference is that top-performing products now compete intensely. Under the baseline, Groups 1–3 priced near static in the long run. Under $\rho = 0.95$, these groups price 5–8% below static—the largest reductions of any groups. Faster forgetting prevents even high-conversion products from resting on their rankings.

Grid search over ρ. Figure 15 reports long-run outcomes across a grid of ρ values. The price-minimizing value is $\rho \approx 0.9686$ (half-life of 23 days), which yields prices 8% below static. At this ρ, consumer surplus is 32% above static, and seller profit is 16% below static.

As discussed before, the relationship is non-monotonic. At high ρ, top ranking positions are secure, lowering sellers' incentive to keep prices low to maintain or invest in top rankings. At low ρ, top ranking positions are too insecure, so sellers lack the incentive to invest in them.

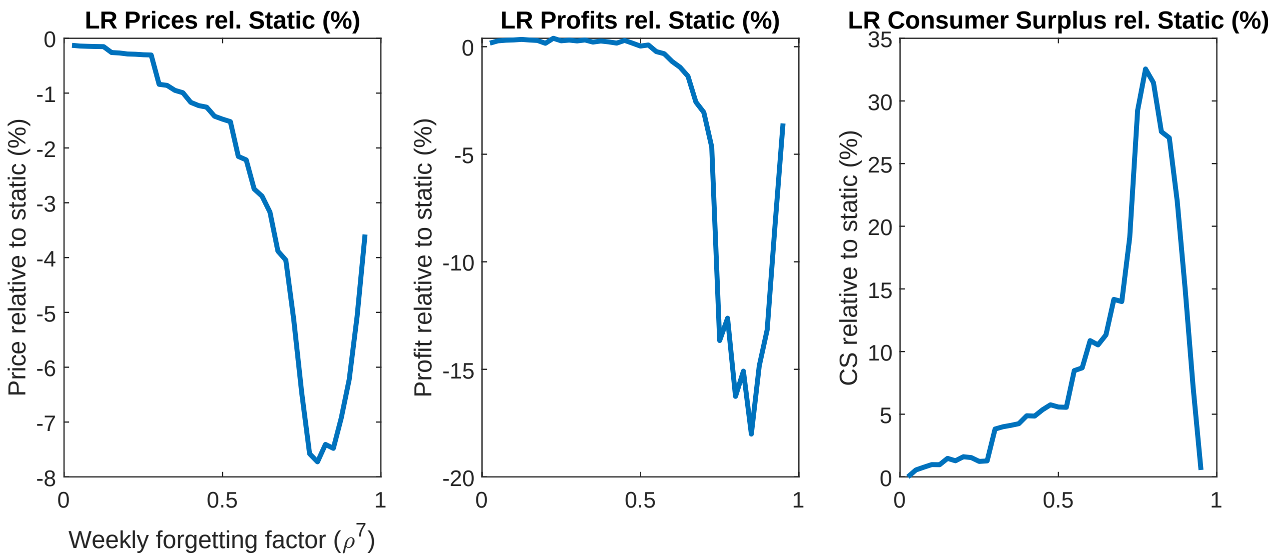


Figure 14: Long-run market outcomes as a function of ρ

Notes. All outcomes expressed as percentage deviations from the static benchmark.

Discussion. By adjusting a single parameter, the platform can shift average equilibrium prices by approximately 8 percentage points. The platform does not set prices, yet it can significantly influence the degree of price competition through algorithm design.

Whether the platform wants to minimize prices depends on its objective. Lower wholesale prices benefit consumers and the platform. So, from purely a static objective, the platform may want to choose $\rho \approx 0.97$ to induce the most intense wholesale price competition. However, this

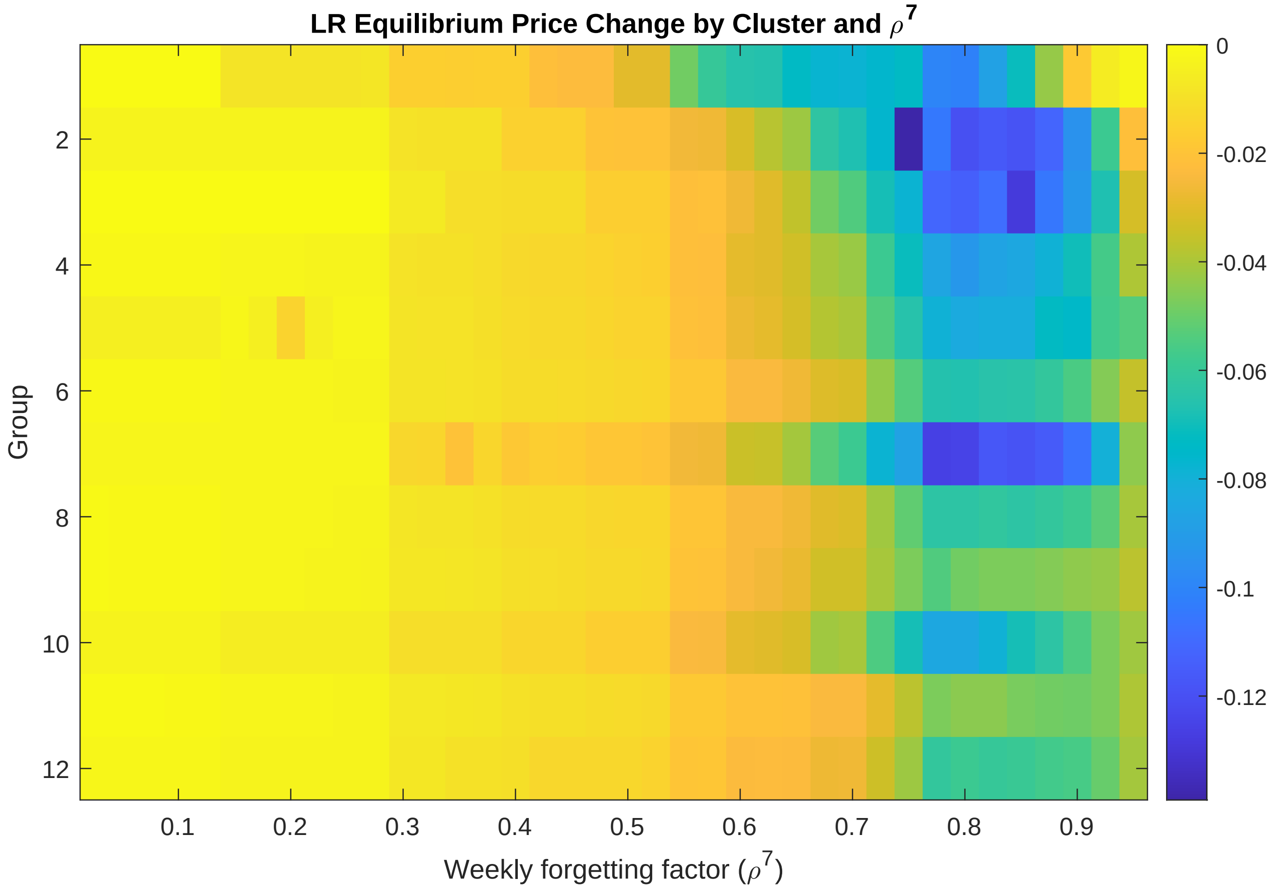


Figure 15: Long-run price changes by ρ and product group

Notes. All outcomes expressed as percentage deviations from the static benchmark.

intense competition may drive out products that cannot maintain high rankings—such as niche products that may not cater to everyone but are vital to maintain a healthy product variety. Or, in a category with platform's private-label products, its incentive to shape equilibrium competition may also vary. These additional considerations go beyond our paper.

For policy, our analysis suggests that one crucial parameter regulators may try to understand is the forgetting factor—or equivalently, how persistent is historical data's impact on current ranking. This parameter is interpretable, verifiable, and acts as a crucial statistic for understanding the algorithm's impact on competition intensity. Requiring platforms to report such parameters would be less burdensome than full disclosure of the entire algorithm.

7 Conclusion

This paper shows that a common Bayesian ranking algorithm used by a large e-commerce platform can *intensify* rather than relax price competition. In a field experiment that permanently raises the algorithm's prior for a random set of products, we show that treated items move up in the rankings, receive substantially more impressions and sales, and yet experience a 2.3% reduction in wholesale prices. Higher visibility does not confer market power; instead, competition is strongest at the top of the ranking.

We interpret these effects through a dynamic mechanism: because the ranking algorithm updates posterior conversion rates based on realized sales, current prices are an investment in future ranking. This investment motive is strongest where impressions, and hence learning, are most responsive to outcomes. Consistent with this mechanism, price cuts are concentrated among younger products for which the algorithm is more uncertain, and cannot be explained by changes in demand elasticity or seller experience.

To quantify the platform's power over competition, we estimate a dynamic pricing model that embeds the observed Bayesian ranking rule and use it to vary the algorithm's forgetting rate. These counterfactuals show that ranking design alone can meaningfully shift equilibrium prices: a moderate forgetting rate—implying a half-life of information of about 23 days—induces the strongest investment incentives and lowers average stationary prices by roughly 8%, whereas very low or very high forgetting rates yield prices close to a static benchmark. Thus, even without setting prices directly, platforms can exert substantial control over competitive intensity through simple, interpretable features of their ranking algorithms.

Our findings speak to policy debates on competition in digital markets. They show that ranking algorithms can be powerful, non-price instruments for governing competitive intensity and consumer prices, even when platforms do not directly set prices. For regulators, this suggests shifting focus from full algorithmic transparency—which is often infeasible and not especially informative—to simpler, behaviorally relevant objects such as the persistence of historical performance (e.g., the half-life of data in rankings) and the sensitivity of rankings to recent outcomes. Disclosure or oversight of these summary statistics could provide a more tractable way to monitor and discipline the competitive effects of algorithm design, while still allowing platforms flexibility in implementation.

References

Aguirregabiria, Victor, Allan Collard-Wexler, and Stephen P. Ryan, "Dynamic Games in Empirical Industrial Organization," in Kate Ho, Ali Hortaçsu, and Alessandro Lizzeri, eds., *Handbook of Industrial Organization*, Vol. 4, Elsevier, 2021, pp. 225–343.

_ **and Chun-Yu Ho**, "A Dynamic Oligopoly Game of the US Airline Industry: Estimation and Policy Experiments," *Journal of Econometrics*, 2012, *168* (1), 156–173.

Bergemann, Dirk and Alessandro Bonatti, "Markets for Information: An Introduction," in "Handbook of Industrial Organization," Elsevier, 2022.

Burdett, Kenneth and Kenneth L. Judd, "Equilibrium Price Dispersion," *Econometrica*, 1983, *51* (4), 955–969.

Cabral, Luís, "Small Sellers and Reputation in Online Marketplaces," *RAND Journal of Economics*, 2010, *41* (2), 208–231.

Chen, Jiafeng and Jonathan Roth, "Logs with zeros? Some problems and solutions," *The Quarterly Journal of Economics*, 2024, *139* (2), 891–936.

Choi, Hana and Carl F Mela, "Monetizing online marketplaces," *Marketing Science*, 2019, *38* (6), 948–972.

Dinerstein, Michael, Liran Einav, Jonathan Levin, and Neel Sundaresan, "Consumer price search and platform design in internet commerce," *American Economic Review*, 2018, *108* (7), 1820–59.

Ellison, Glenn and Sara Fisher Ellison, "Search, Obfuscation, and Price Elasticities on the Internet," *Econometrica*, 2009, *77* (2), 427–452.

Farrell, Joseph and Paul Klemperer, "Coordination and lock-in: Competition with switching costs and network effects," *Handbook of industrial organization*, 2007, *3*, 1967–2072.

Han, Cuize, Pablo Castells, Parth Gupta, Xu Xu, and Vamsi Salaka, "Addressing cold start in product search via empirical bayes," in "Proceedings of the 31st ACM International Conference on Information & Knowledge Management" 2022, pp. 3141–3151.

Hortaçsu, Ali and Chad Syverson, "Product differentiation, search costs, and competition in the mutual fund industry: A case study of S&P 500 index funds," *The Quarterly journal of economics*, 2004, *119* (2), 403–456.

Hui, Xiang, Ginger Zhe Jin, and Meng Liu, "Designing quality certificates: Insights from eBay," Technical Report, National Bureau of Economic Research 2022.

Jin, Chen, Luyi Yang, and Kartik Hosanagar, "To Brush or Not to Brush: Product Rankings, Consumer Search, and Fake Orders," *Information Systems Research*, 2023, *34* (2), 532–552.

Klein, Tobias J, Christian Lambertz, and Konrad O Stahl, "Market transparency, adverse selection, and moral hazard," *Journal of political economy*, 2016, *124* (6), 1677–1713.

los Santos, Babur De, Ali Hortaçsu, and Matthijs R. Wildenbeest, "Testing Models of Consumer Search Using Data on Web Browsing and Purchasing Behavior," *American Economic Review*, 2012, *102* (6), 2955–2980.

Stahl, Dale O., "Oligopolistic Pricing with Sequential Consumer Search," *American Economic Review*, 1989, *79* (4), 700–712.

Stigler, George J, "The economics of information," *Journal of political economy*, 1961, *69* (3), 213–225.

Tadelis, Steven, "Reputation and Feedback Systems in Online Platform Markets," *Annual Review of Economics*, 2016, *8*, 321–340.

Ursu, Raluca M, "The power of rankings: Quantifying the effect of rankings on online consumer search and purchase decisions," *Marketing Science*, 2018, *37* (4), 530–552.

Varian, Hal R., "A Model of Sales," *American Economic Review*, 1980, *70* (4), 651–659.

A Randomization checks

Table A1: Randomization checks by treatment timing

Outcome Variable	Mean of Never-Treated Group		Treated Group Difference		Treatment Timing Effect	
	Never Treated	Ever Treated	Difference	t-stat	Coefficient	t-stat
Number of Products	389.23	353.75	-35.47	-3.65	0.05	0.45
Products Sold	76.22	70.08	-6.14	-1.31	0.02	0.30
Revenue	2.44	2.58	0.13	0.51	0.00	0.64
Total Items Sold	3216.29	3477.22	260.93	0.62	1.00	0.17
Average Price	1131.94	1157.46	25.52	1.85	-0.09	-0.57
Transaction Days	274.42	270.43	-3.99	-0.33	-0.08	-0.55
Seller Tenure (Days)	1031.31	1042.26	10.95	0.74	0.74	4.25
Incident Rate	0.12	0.12	0.00	0.57	0.00	0.49
Cancellation Rate	0.09	0.09	0.00	0.48	-0.00	-0.10
Average Rating	4.22	4.22	-0.00	-0.36	-0.00	-0.56

Notes. This table shows randomization checks between never treated and ever treated groups, as well as the treatment timing effects.

B Spillover effects

Table B2: Spillover effects

	(1)	(2)
Neighbor product treated	-0.569	-0.541
	(12.713)	(12.719)
Weeks since neighbor's treatment		-0.967
		(3.465)
Constant	279.903	296.717
	(11.380)	(60.712)
Product FE	✓	✓
Group-Date FE	✓	✓
N	54,796	54,796

Standard errors clustered at product level

Notes. This table shows the impact of treatment on untreated neighboring products. The dependent variable is the rank number in the median consumer session (lower means higher rank).

C Additional results

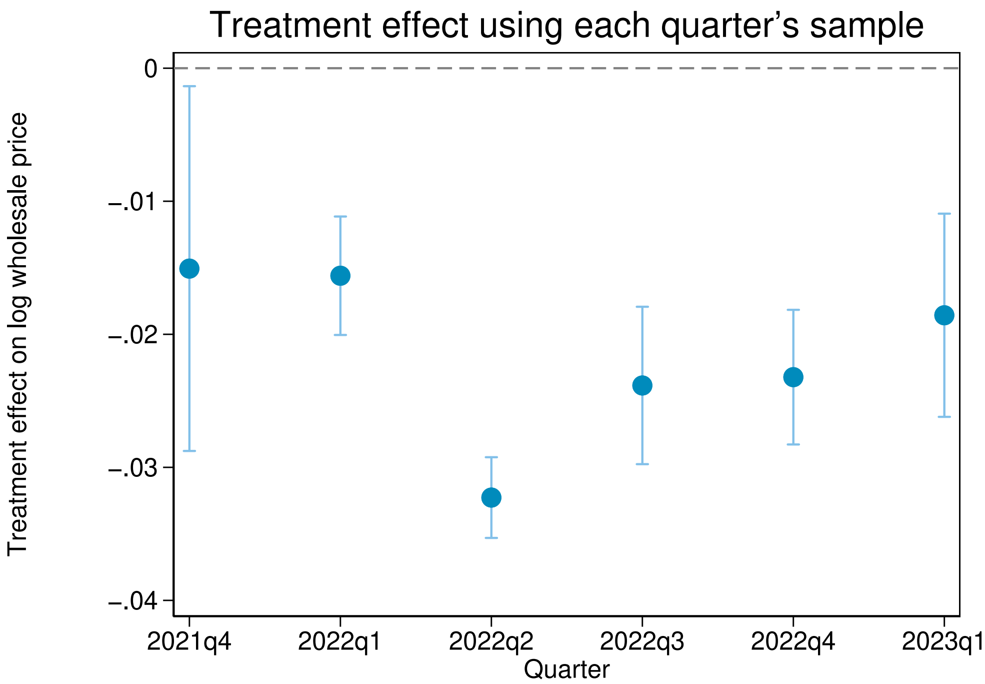


Figure C1: Quarterly treatment effects

Notes. This figure shows the treatment effects on wholesale prices across each quarter.

Table C3: Cross-price elasticities

	Baseline	Group-month FE	Cross
Log Price	-4.205***	-3.831***	-3.833***
	(0.216)	(0.302)	(0.303)
log_avg_price_others			0.310
			(0.329)
Observations	768309	780232	780232

Standard errors in parentheses
* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Notes. This table shows the price elasticities estimated by including the own and competing products' prices.