Personalization, Pricing, and Experimentation on Digital Platforms

by

Yu Song

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Business Administration)
in The University of Michigan
2026

Doctoral Committee:

Assistant Professor Jessica Fong, Co-Chair
Professor Puneet Manchanda, Co-Chair
Associate Professor Zach Y. Brown
Professor Jun Li

Yu Song
yyusong@umich.edu
ORCID iD: 0000-0002-1233-6340

© Yu Song 2026

ACKNOWLEDGEMENTS

I am indebted to many people who have contributed to this dissertation and to my growth as a researcher.

My deepest gratitude goes to my advisors, Jessica Fong and Puneet Manchanda. Jessica, thank you for your unwavering support and mentorship. Your sharp attention to detail, your willingness to work through all challenges alongside me, and your genuine investment in my development as a researcher have shaped not only this dissertation but also the way I approach problems. I am fortunate to have had you as an advisor.

Puneet, thank you for your invaluable guidance and for always pushing me to see the bigger picture. Your encouragement to think about the broader implications of my work and your steady confidence in my potential have been instrumental throughout the doctoral program. Beyond research, you taught me the importance of being kind, patient, and generous—lessons that I will carry with me well beyond academia.

I also thank my dissertation committee members, Jun Li and Zach Brown. Your thoughtful questions and constructive suggestions during the defense and beyond have made this work substantially stronger. I am grateful for your willingness to engage deeply with both essays.

I owe a special debt to Mercari for generously providing access to the data that made this dissertation possible. I also thank Ajay Daptardar, Michael Manzon, Vamshi Teja Racha, and Takuma Yamaguchi for their collaboration and willingness to support this research.

I thank the faculty of the Marketing Department at Ross for creating an intellectually stimulating environment. I am especially grateful to Anocha Aribarg, Fred Feinberg, Yesim Orhun, and S. Sriram for their advice and encouragement at various stages of my doctoral studies.

My fellow doctoral students have been a constant source of inspiration, camaraderie, and moral support. I thank Gwen Ahn, Hayoung Cheon, Jangwon Choi, Bruno Castelo Branco, Varad Deolankar, Tim Doring, Yuqing Gu, Rex Hsieh, Yuwen Hu, Aishwarya Joshi, Jamie Lee, Yiqi Li, John Loretizo, Prashant Rajaram, Yeonjin Sung, Candice Wang, Zetao Wang, Celeste Yi, and Fan Zhou for the countless conversations, feedback sessions, and moments of levity that sustained me through the program. I also thank my friends Yimeng Du and Yao Zhen, whose messages and phone calls provided comfort, laughter, and perspective when I needed them most.

Finally, I am profoundly grateful to my family for their unconditional love and support. This dissertation is dedicated to them.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS ii

LIST OF FIGURES v

LIST OF TABLES vii

LIST OF APPENDICES ix

ABSTRACT x

CHAPTER

1 Optimizing Multi-Stage Personalization in the Customer Journey 1

1.1 Introduction 1
1.2 Literature Review 5
1.3 Empirical Setting and Data 6
1.3.1 Empirical Setting 6
1.3.2 Data Description 7
1.4 Personalization Field Experiments 7
1.4.1 Personalizing Discovery Stage 8
1.4.2 Personalizing Query Stage 12
1.4.3 Experimental Results Summary and Structural Model Motivation 16
1.5 Structural Search Model 17
1.5.1 Model Setup 17
1.5.2 Model Specification 18
1.5.3 Optimal Customer Behavior 23
1.5.4 Estimation and Identification 24
1.6 Estimation Results and Counterfactual Simulations 28
1.6.1 Estimation Results 28
1.6.2 Counterfactual Simulations 28
1.7 Conclusion 32

2 How Effective Is Suggested Pricing?: Experimental Evidence from an E-Commerce Platform 35

2.1 Introduction 35
2.2 Literature Review 38

2.3 Institutional Background 40
2.3.1 Empirical Setting 40
2.3.2 Preexperiment Descriptive Statistics 43
2.4 Field Experiment 46
2.4.1 Experimental Design 46
2.4.2 Data and Summary Statistics 49
2.5 Experiment Results 52
2.5.1 Treatment Effects on Listing Price 52
2.5.2 Treatment Effects on Selling Outcomes 57
2.5.3 Seller-Level Outcomes 60
2.6 Testing for Interference 62
2.7 Follow-Up Experiment: Getting Closer to the Full Equilibrium 68
2.8 Conclusion 70

APPENDICES 73

BIBLIOGRAPHY 137

LIST OF FIGURES

FIGURE

1.1 Heterogeneous Effects of Personalizing Discovery Stage 11
1.2 Price Distribution of Products at Query Stage by Customer Price Preference 14
1.3 Distribution of Clicked Positions 14
1.4 Structural Search Model Overview 19
1.5 Total Revenue of Multi-Stage Personalization 31
1.6 Revenue of Personalized Personalization Based on Preference Consistency 32

2.1 Randomization Check 51
2.2 The Likelihood of Sale and Revenue in Seven Days by Seller Type 52
2.3 Treatment Effects on Listing Price 53
2.4 Ratio of Listing Prices to Max Suggested Prices 56
2.5 Treatment Effects on Selling Outcomes 59
2.6 Treatment Effects of Suggested Price Levels by Saturation Group 64
2.7 Treatment Effect of No Suggested Price by Saturation Group: New Sellers 65
2.8 Heterogeneity in Treatment Effects on Revenue by Other Items' Experimental Conditions 67
2.9 Treatment Effects in the Second Experiment 69
2.10 Exp 2: Heterogeneous Treatment Effects by New and Experienced Sellers 69
2.11 Exp 2: Heterogeneous Treatment Effects by Quartile 70

A.1 Manipulation Checks 74
A.2 Comparisons of Users in Control Condition Pre- and Post-Experiment 75
A.3 Treatment Effects of Personalized Discovery Stage by Day 76
A.4 Comparisons of Users in Status Quo Condition Pre- and Post-Experiment 77
A.5 Treatment Effects of Personalized Query Stage by Day 77
A.6 Price and Item Condition Across Stages 79
A.7 Shannon Entropy and HHI 80
A.8 The Impact of Personalizing Discovery Stage on Search Query Specificity 81
A.9 Correlation between Discovery Products and Query by Condition 82
A.10 The Impact of Personalized Query Stage on Latency 84
A.11 Silhouette Score of Clusters 89
A.12 Beliefs about Hidden Attributes 90
A.13 Neural Network Training Illustration 91
A.14 An Illustration of Platform Layout 100
A.15 Heterogeneous Effects of Personalizing Query Stage 101
A.16 Clicks and Time by Item Position 101

B.1 Model Simulations: Beliefs about the Platform's Objective . . . 109
B.2 Listing Price as a Function of Suggested Prices under Seller Competition . . . 110
B.3 Assigned and Realized Treatment Saturation . . . 112
B.4 Realized Deviation in Suggested Price across Experimental Conditions . . . 112
B.5 Treatment Effect on Whether Seller Clicked on Any Other Items . . . 116
B.6 Treatment Effects on Listing Price: No Items in the Same Category Clicked Before Listing Creation . . . 116
B.7 Treatment Effects on Search the Same Category before Listing . . . 117
B.8 Treatment Effects on Listing Price on Sellers Who Do Not Search the Same Category . 117
B.9 Heterogeneity in the Treatment Effect on Whether Sold in Seven Days by Other Items' Experimental Conditions . . . 119
B.10 Treatment Effect on Listing Price by Week . . . 120
B.11 Treatment Effect on Whether Sold in Seven Days by Week . . . 121
B.12 Treatment Effect on Seven-Day Revenue by Week . . . 121
B.13 Example Listing Creation Page . . . 131
B.14 Distribution of Selling Time (Preexperiment Data) . . . 131
B.15 Treatment Effects on Whether Listing Creation Is Completed . . . 132
B.16 Randomization Check of Whether New Seller . . . 132
B.17 Relationship between Time since Listed and Fraction of Listings Sold . . . 133
B.18 Treatment Effects on Shipping Fees . . . 133
B.19 Treatment Effects on Listing Price Adjustments . . . 134
B.20 Treatment Effects on Item-Level Likelihood of Sale and Revenue in 14 Days and 28 Days . . . 135
B.21 Treatment Effects on Whether the Item is Sold through Bargaining . . . 135
B.22 Treatment Effects on Revenue (Deducting the Seller-Paid Shipping Fees) . . . 136

LIST OF TABLES

TABLE

1.1 Treatment Effects of Personalizing Discovery Stage 10
1.2 Treatment Effects of Personalizing Query Stage 15
1.3 Estimation Results of Structural Search Model 29

2.1 Correlation between Price Suggestion and Listing Price and Likelihood of Sale 44
2.2 New and Experienced Seller Comparison 44
2.3 Evidence for Pricing Uncertainty by New Sellers and Used Items 45
2.4 Treatment Groups and Probability Assignment 49
2.5 Sample Size by Experimental Condition 49
2.6 Summary Statistics of Prices 51
2.8 Treatment Effects on Selecting the Maximum Suggested Price 57
2.9 Treatment Effects on Whether Listing Price Ends in 0 58
2.10 Seller-Level Treatment Effects 61
2.11 Listing Prices by Treatment Saturation: Control Sellers 63

A.1 Randomization Check 73
A.2 Relationship between Discovery Stage and Query Stage 78
A.3 Personalizing Discovery Stage on Query and Total Revenue by Stratification 82
A.4 Effects of Search Query Latency 85
A.5 Treatment Effects of More Advanced Personalized Discovery Stage 86
A.6 The Effects of Personalized Query Ranking (Alternative Model) 87
A.7 Monte Carlo Simulation Results 94
A.8 Model Fit 95
A.9 User-Level Treatment Effects of Personalization at Discovery Stage 97
A.10 User-Level Treatment Effects of Personalization at Query Stage 98
A.11 Treatment Effects of Personalized Query Stage on Discovery-Stage Outcomes and Total Outcomes 98
A.12 Effects of Refinement on Customer Outcomes 99

B.1 Suggested Price Reproduction 102
B.2 Predicted Revenue under Different Pricing Systems 115
B.3 Experimental Condition Co-Occurrence Matrix 118
B.4 Correlation between Price Suggestion and Likelihood of Sale in 14 and 28 Days 122
B.5 Distribution of Suggested Prices for Listings in Top Categories 122
B.6 Treatment Effects on Whether Listing Price is Within Suggested Price Range 123
B.7 Treatment Effects on Listing Price and Selling Outcomes 124

B.8 Heterogeneous Treatment Effects on Listing Price 125
B.9 Heterogeneous Effects on Listing Price: Other Measures of Pricing Uncertainty 126
B.10 Treatment Effects on Listing Price When Listing Price is Not Equal to Max Suggested Price 126
B.11 Linear Treatment Effects on Selling Outcomes 127
B.12 Seller-Level Treatment Effects: New Sellers 128
B.13 Treatment Effect of Suggested Price Level by Saturation Group 129
B.14 Treatment Effect of No Suggested Price by Treatment Saturation Group: New Sellers . 130

LIST OF APPENDICES

APPENDIX

A Appendix to Chapter 1 . . . . . **73**

B Appendix to Chapter 2 . . . . . **102**

ABSTRACT

This dissertation investigates the role of platform information design such as personalization and suggested pricing on consumers and suppliers in two essays. The first essay studies how personalization at different stages of the customer journey affects consumer decisions and platform outcomes. Firms increasingly apply personalization to influence product discovery and engagement along the customer journey. However, effective multi-stage personalization is challenging because customers' later-stage decisions depend on earlier experiences, creating cross-stage spillovers. I study personalization at two stages on an e-commerce platform: an early, firm-driven discovery stage and a later, customer-initiated query stage. Using two field experiments, I find that personalizing the discovery stage increases immediate engagement but reduces engagement in the query stage, producing no net gain. When the query stage is also personalized, total revenue rises without cannibalizing discovery-stage revenue. Personalization has stronger effects for customers with more consistent preferences. To identify the revenue-maximizing policy across stages and customer types, I build and estimate a structural search model that uses experimental variation for identification and a neural-network estimator for computational tractability. The results show that personalizing only at the query stage maximizes total revenue. Discovery-stage personalization shapes beliefs about the value of querying and can trigger early exit when matches are weak, whereas query-stage personalization conditions on query-revealed signals to deliver more relevant results. Moreover, targeting personalization to customers with more consistent preferences further increases revenue. These findings offer firms guidance on optimizing personalization across stages and customers.

The second essay examines how platform-suggested prices influence sellers' pricing decisions and selling outcomes. In collaboration with Mercari, a peer-to-peer e-commerce platform, I conduct a field experiment that varies whether a seller receives a suggested price, and if so, the suggested price itself. I find that a 30% change in suggested prices leads to a 5% change in listing prices in the same direction. Suggested prices are more influential when pricing is more challenging, such as for new sellers and used items. Lower suggested prices improve both the likelihood of sale and the resulting seller revenue. I show, in a subsequent experiment, that these results are likely to generalize to the full equilibrium. The findings imply that platform-suggested pricing is an effective compromise that guides seller pricing while allowing them to incorporate their private information.

CHAPTER 1

Optimizing Multi-Stage Personalization in the Customer Journey

1.1 Introduction

As digital markets mature, personalization has become a central tool influencing how customers discover and engage with products. Firms that excel at personalization generate 40% more revenue than their peers.[1] Platforms such as Amazon and Spotify attribute 35% of transactions and a large share of a 1,000% increase in revenue, respectively, to personalized recommendation systems (MacKenzie et al., 2013; Abraham and Edelman, 2024). Given its potential, firms such as Netflix and eBay have begun to apply personalization to multiple stages of the customer journey (Ostuni et al., 2023).[2] In many digital settings, customers begin their journey by discovering recommendations on a landing page, and then decide whether to initiate search queries to retrieve additional products relevant to their queries. These two stages, which I refer to as the discovery stage and the query stage, are prevalent across various platforms, including Amazon and eBay in e-commerce, TikTok and YouTube in video sharing, and Uber Eats and DoorDash in food delivery.

Many firms manage each stage of the customer journey in isolation, seeking to maximize returns within a single stage. Organizational design and measurement systems can reinforce this siloed approach by prioritizing stage-level metrics over journey-level outcomes.[3] However, this approach can overlook spillovers across stages and lead to suboptimal outcomes. The discovery stage is particularly salient because it sets the trajectory for the remainder of the journey. On the one hand, personalizing the discovery stage may help customers find relevant products quickly,

[1] McKinsey & Company (2022), "The Value of Getting Personalization Right—or Wrong—is Multiplying," https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/the-value-of-getting-personalization-right-or-wrong-is-multiplying, accessed March 2026.

[2] Modern Retail (2023), "eBay Steps Up Personalization Efforts to Improve Search, Ads and Experience," https://www.modernretail.co/retailers/ebay-steps-up-personalization-efforts-to-improve-search-ads-and-experience/, accessed March 2026.

[3] McKinsey & Company (2016), "From touchpoints to journeys: Seeing the world as customers do," https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/from-touchpoints-to-journeys-seeing-the-world-as-customers-do?, accessed March 2026.

freeing attention and time for broader exploration later in the journey (Song, 2021). On the other hand, if discovery surfaces satisfactory options too early, it may weaken incentives to engage in deeper search (Fong, 2017; Chen et al., 2025), potentially lowering total revenue (Yuan et al., 2025). Moreover, when discovery fails to produce a good match (for example, because customer preferences shift across sessions), customers may interpret the discovery results as evidence that the platform does not offer relevant products and exit without purchasing or continuing the journey.

In this paper, I study how personalization affects customer search and whether stage-by-stage optimization is suboptimal when cross-stage spillovers are present. Specifically, I aim to answer two research questions. First, how do the effects of personalization vary across different stages of the customer journey? I focus on its effects on customer behavior (e.g., clicks and purchases) and platform revenue. Second, how do these effects differ across customers? If personalization is effective for some customers but generic (e.g., popularity-based) experiences work better for others, firms may be able to leverage this heterogeneity to "personalize personalization"—that is, to target personalization efforts toward those most likely to benefit while leaving others with a more generic experience.

To study these questions, I collaborate with Mercari, Inc. (Mercari), an e-commerce platform that enables individuals to buy and sell items across a wide range of categories. Founded in Japan, it has become Japan's largest community-powered marketplace and expanded its operations to the United States (US) in 2014. This paper focuses on Mercari US. I first analyze a large-scale field experiment that only personalizes the discovery stage. I find that the personalized discovery stage increases user engagement (clicks, orders, and revenue) within that stage but reduces the likelihood that customers proceed to the subsequent query stage. These findings indicate that personalizing discovery can generate negative spillovers on downstream search. When the personalized discovery stage offers a good match, customers purchase immediately and thus do not proceed to the query stage. When it offers a poor match, customers may interpret the weak match as evidence that the platform lacks relevant products, infer a low value of querying, and exit early without purchasing.

I then design and implement a subsequent field experiment that personalizes the query stage.[4] I develop two versions of a ranking algorithm: a non-personalized version that uses only item-level features (e.g., brand) and contextual features (e.g., inventory of similar items at the time of the query), and a personalized version that additionally incorporates user-level features (e.g., past purchases in the last 30 days). To ensure that outcome differences are driven by personalization rather than by differences in algorithms, I apply the same ranking algorithm, LambdaMART, to both conditions. LambdaMART operates by using many small decision trees to compare item pairs

[4]All users in this experiment receive a personalized discovery stage. A full factorial design would allow me to identify the optimal way to personalize across stages, but it is often difficult to implement in practice. The company faces operational constraints, especially when the two stages are managed by different teams.

and learn how to order the query results. Deploying LambdaMART-based ranking models in a field experiment on Mercari, I find that users exposed to personalized product rankings at the query stage click on more items and make more purchases, with no spillover effects on the discovery stage. As a result, total revenue across both stages increases.

Across both experiments, personalization is more effective for customers with more consistent preferences (who tend to click similar items over time). By contrast, discovery-stage personalization reduces revenue for customers with more dynamic preferences (who click across a broader set of products). Together, these results suggest that firms may "personalize personalization" to increase total revenue. For example, they could intensify personalization for customers with more consistent preferences while offering a more generic experience for others.

While my two experiments offer valuable insights, they cannot evaluate all combinations of multi-stage personalization (e.g., comparing personalization only at the query stage versus personalizing at both stages), nor can they assess alternative strategies such as personalized personalization. To fill these gaps, I develop a structural search model building on Weitzman's (1979) optimal sequential search framework. The model captures customers' inspection (i.e., clicking) and purchase decisions as they transition from an initial discovery stage to a subsequent query stage. In the discovery stage, customers observe limited product information. They can choose whether to inspect an item by clicking to reveal additional product information or to submit a query to view a new set of products in the query stage; both actions incur costs. Customers use the products they see in the discovery stage to form beliefs about the value of querying, and they place more weight on this signal when the discovery set aligns with their purchase intent. If the customer conducts a query, they switch from the discovery stage to the query stage. At the query stage, the customer can similarly incur a cost to inspect an item. The customer can terminate the process at any point, either by selecting the outside option or by purchasing one of the inspected products.

I exploit experimental variation in customers' propensity to transition to the query stage to estimate the structural search model. I apply the neural network estimator (NNE) developed by Wei and Jiang (2025), which trains a neural network to learn the mapping from observed data moments to the model's structural parameters. This approach avoids direct integration over unobservables and overcomes computational challenges.

Using the estimated model, I conduct counterfactual simulations to evaluate a range of multi-stage personalization strategies. In the first counterfactual simulation, I consider a utility-based ranking framework (Ursu, 2018a). Using the same framework at both stages allows me to isolate the role of personalization from differences in the underlying ranking algorithms across stages. The results show that although personalization generally increases platform revenue, its effectiveness depends on where it is applied. Applying personalization to more stages does not necessarily improve total outcomes. In fact, personalizing only the query stage maximizes total revenue:

query-only personalization generates 2.9% higher revenue than personalizing both stages and 12% higher revenue than personalizing only the discovery stage. Discovery-stage personalization surfaces items that are more aligned with a customer's purchase intent, which makes customers more likely to treat the discovery experience as informative when forming beliefs about the value of querying. However, because the discovery set is limited and may reflect preferences imperfectly, a weak preference fit can lead customers to update downward their expected gains from querying, causing some to exit early rather than continue the journey. By contrast, a popularity-based (non-personalized) discovery stage matches intent less precisely, so customers place less weight on it when forming beliefs, which attenuates these negative continuation effects. Furthermore, the query stage reveals richer information about the customer's current intent and preferences (e.g., brand). This information allows the ranking to condition on that information and deliver more relevant results with a higher likelihood of satisfying demand. In the second counterfactual simulation, I evaluate "personalizing personalization" based on customers' preference consistency. I find that applying personalization only to customers with consistent preferences increases platform revenue by 14.7% relative to no personalization and by 3.3% relative to applying it to all customers.

This paper offers several managerial implications. First, as firms increasingly coordinate personalization across multiple stages of the customer journey, they should account for both within-stage effects and cross-stage spillovers. Given evidence of negative spillovers from the discovery stage to the query stage, myopic personalization that optimizes each stage in isolation can be suboptimal. Managers should therefore evaluate personalization decisions jointly across stages, weighing incremental benefits against implementation costs when deciding whether, and at which stages, to personalize. A holistic approach can also help preserve product diversity. For example, keeping the discovery stage less personalized can expose customers to a broader assortment, which may mitigate concerns about "filter bubbles," where highly tailored content limits exposure to new or diverse products (Pariser, 2011). Second, firms can tailor personalization intensity at the individual level. A simple starting point is to reduce or shut down personalization for some customers. For instance, platforms might provide more tailored recommendations to customers with consistent preferences while offering a broader, more generic experience to customers whose preferences are more dynamic. Finally, although I study a single platform, the challenge I document is broadly relevant: stage-level metrics can obscure cross-stage trade-offs. The empirical approach in this paper provides a practical template for identifying whether such spillovers arise in other settings and for quantifying the net value of personalization across the full journey. In this sense, the paper offers a foundation for designing experiments and decision rules that optimize personalization end to end rather than locally within individual stages. In practice, firms can apply the same logic by designing experiments that measure both direct effects and downstream outcomes. This enables them to detect spillovers, quantify net effects on engagement and revenue, and choose personalization

policies that optimize performance at the journey level.

The remainder of the paper is structured as follows. Section 2.2 situates my work within the existing literature. Section 1.3 describes the empirical setting and data. Section 1.4 details two field experiments and shows the experimental results. Section 1.5 presents the structural search model and outlines the identification and estimation procedures. Section 1.6 reports the search model estimation results and counterfactual simulations. Section 2.8 concludes with implications and limitations.

1.2 Literature Review

This paper relates closely to the literature on personalized recommendations and personalized search ranking. Personalized recommendations at the discovery stage have been shown to increase consumer engagement (e.g., Hosanagar et al., 2014) and purchase propensity (e.g., Li et al., 2022b). However, they may also reduce consumption diversity and diminish long-term user engagement (Holtz et al., 2020; Chen et al., 2024). Personalized search ranking has been found to improve platform outcomes and consumer welfare, even when platforms take revenue into account (Donnelly et al., 2024). While prior work focuses on a single stage of the customer journey, this paper is, to the best of my knowledge, the first to examine the combined effects of personalization across multiple stages.

My paper also relates to empirical work on the interplay between firm recommendations and consumer search. Song (2021) documents a positive spillover effect: personalized recommendations improve search efficiency in the essential product category and stimulate broader exploration in other categories. Wan et al. (2024) find that recommendations help consumers identify higher-value products, through lower prices or better preference alignment. In contrast, Yuan et al. (2025) show that when product discovery relevance declines, consumers compensate by increasing their search activity. Fang et al. (2026) demonstrate that the relationship between recommender and non-recommender searches depends on the number of information cues provided. My paper adds to the literature by examining multi-stage personalization using a consumer search model.

This paper contributes to the growing literature on the customer journey. Prior research has employed a wide range of approaches to modeling the customer journey, including probabilistic machine learning techniques (Padilla et al., 2024), Poisson point process models (Goić et al., 2022), and transformer-based architectures (Lu and Kannan, 2026). My approach is most closely related to work that employs structural search models to analyze consumer decision-making. Existing research has examined how consumers progress from awareness to consideration and choice (Honka et al., 2017; Greminger, 2022), explored how they navigate product discovery pathways (Zhang et al., 2023), and investigated post-search behaviors such as revisits (Dang et al., 2025) and retargeting

responses (Jiang et al., 2021). I develop a sequential search model to understand the impact of multi-stage personalization on consumer decisions.

Furthermore, my work adds to the literature on the empirical search model (Honka et al., 2019; Honka et al., 2024). There is growing empirical search literature built on the tractable solution offered by Weitzman (1979), such as modeling how consumers use refinement (Chen and Yao, 2017; De los Santos and Koulayev, 2017; Gu and Wang, 2022; Wang and Liu, 2026), incorporating consumer learning (Hodgson and Lewis, 2025; Korganbekova and Zuber, 2023), considering browsing behaviors (Greminger, 2022; Choi and Mela, 2019; Zhang et al., 2023), modeling which product attributes customers search over (Compiani et al., 2024), and so on. Search models typically assume that the decision to conduct a search is exogenous, but this assumption may not hold when prior experience with landing-page recommendations influences that decision. I contribute to the literature by endogenizing consumers' decisions to conduct search queries. Modeling these decisions is important when firms seek to coordinate consumers' active search (initiating a query) with passive search (exposure to firm-generated recommendations).

1.3 Empirical Setting and Data

1.3.1 Empirical Setting

The empirical setting of this paper is Mercari US. As of 2024, it has 5 million monthly active users, 0.6 million sellers, 11 million new listings per month, and over 100 million app downloads.[5] Users can purchase new or used items from a wide range of categories (e.g., women's fashion, men's fashion, office supplies, home goods, etc.). Mercari emphasizes ease of use through its app-based platform, positioning itself as "a C2C marketplace app that allows anyone with a smartphone to easily sell items they no longer need."[6] This paper focuses on the Mercari app, including the field experiments and data.

In Figure A.14 in the web appendix, I illustrate the platform layout with examples of Mercari's discovery page, query results page, and product page. When users enter the app, they first see product recommendations on the discovery page, which I refer to as the discovery stage. This stage is pivotal because it receives 100% of user traffic. On average, customers spend over one minute in the discovery stage before entering the query stage. This suggests that customers engage with both stages of the journey, making each an important component.[7] At a high level, the platform constructs

[5]Sources: https://about.mercari.com/en/press/news/articles/20230201_mercari10th/; https://thesmallbusinessblog.net/mercari-users/; https://www.mercari.com/about/. All accessed March 2026.

[6]Source: https://careers.mercari.com/en/services/, accessed March 2026.

[7]Users also see "similar items" at the bottom of the product detail page after clicking on a product. Due to scope, I do not study this recommendation surface and leave it for future research.

discovery-stage discoveries based on popularity, trendiness, and, when personalization is applied, the user's historical activity. Section 1.4.1 provides a detailed description of the discovery-stage algorithm used in this study.

Users can conduct a query; when they do, the platform returns a ranked list of results relevant to the query.[8] The order from the discovery stage to the query stage is mechanically fixed. Mercari employs a three-stage ranking process to generate query results, following a standard industry pipeline documented in the academic literature (e.g., Zhan et al., 2024; Chen et al., 2024). The process consists of: (1) a retrieval stage, where the platform retrieves a broad set of candidate products based on the query, typically ranging from 400 to 10,000 items depending on relevant product availability; (2) a ranking stage, where items are primarily sorted by relevance and recency, with more relevant or recent items appearing higher in the list; and (3) a re-ranking stage, where the top 100 items from the ranking stage are re-ranked using a machine learning model. Section 1.4.2 provides further details on how products are re-ranked in this study.

1.3.2 Data Description

For each user session, I observe the full sequence of consumer actions in both the discovery and query stages, along with their timestamps. These actions include (1) conducting a query, (2) inspecting a product, as measured by the user clicking on the product, (3) making a purchase, and (4) exiting the platform. In both stages, I observe the set of displayed products and their positions.

I observe each user's historical activity on the platform, including product clicks, purchases, and search queries. For each product, I record attributes such as brand, category, physical condition, item age, number of photos, price, and seller attributes. I also track downstream selling outcomes, including whether the product is sold and, if so, the sale date, transaction price, and buyer.

1.4 Personalization Field Experiments

In this section, I present two field experiments. The first experiment (Section 1.4.1) varies whether the discovery stage is personalized, while the second experiment (Section 1.4.2) varies whether the query stage is personalized. Section 1.4.3 summarizes the experimental results and motivates the structural search model.

As users may self-select into future sessions based on their first session's experience, I focus on each user's first session during the experiment to mitigate selection bias, following the approach of Fong et al. (2026) and Fong (2024). User-level outcomes across all sessions are similar and are

[8] Users may also conduct non-query-based searches by choosing product categories. In such cases, item rankings follow the platform's default algorithm and are unaffected by my experiment. As the majority of searches on the platform are query-based, this paper focuses exclusively on query-based search.

reported in Tables A.9 and A.10 in the web appendix. Both experiments pass the randomization check (see Web Appendix A.1.1) and manipulation check (see Web Appendix A.1.2). Finally, I rule out the potential stable unit treatment value assumption (SUTVA) violations using several checks in Web Appendix A.1.3.

1.4.1 Personalizing Discovery Stage

The personalized discovery stage experiment was conducted for two weeks in Fall 2022. The experiment was implemented on the Mercari mobile app; users accessing the platform via the web continued to receive non-personalized discovery stage. My analyses focus on user activity on the Mercari app. In this experiment, 50% of users were assigned to the control condition that received a non-personalized discovery stage, and 50% of users were assigned to the treatment condition that received a personalized discovery stage. In the non-personalized condition, the platform recommended products based on product popularity and trendiness. Product popularity was measured by the cumulative historical revenue of its category-brand pair, and trendiness was defined as the maximum week-over-week revenue increase for that same pair.[9] In the personalized condition, the platform used a neural-network architecture. Specifically, it constructed a latent preference profile for each user based on their recent activity, such as search queries and item clicks. Products the user interacted with were simultaneously embedded into the same latent space using item attributes such as category and title. User and item representations were trained jointly so that items similar to those a user had previously engaged with were positioned closer to their profile.

For new or inactive users (i.e., those without activity in the past 6 months), the platform applied a cold-start (non-personalized, popularity-based) algorithm to generate product discoveries. Thus, the estimated effects should be interpreted as intent-to-treat.

1.4.1.1 Summary Statistics

During the experimental period, 3,050,146 customers viewed the discovery-stage recommendations. 19.5% of customers clicked on at least one recommended item at the discovery page; among them, the median number of clicks was 1, and the mean was 2.2. 0.9% of customers made a purchase from the recommended products.[10] Customers spent an average of 72 seconds in the discovery stage before initiating a query. This pattern suggests that the discovery stage is a meaningful part of

[9] As products on the platform are single-unit, popularity and trendiness are measured at the category–brand-pair level.

[10] A product purchase is attributed to the current session if the customer eventually completes the transaction for that product. This practice is consistent with prior work (e.g., Donnelly et al., 2024; Korganbekova and Zuber, 2023). I adopt this definition of purchase throughout the paper.

the customer journey, because many users engage with the discovery page rather than immediately submitting a query. 56.6% of customers initiated a search query. Conditional on viewing query results, customers clicked on a median and mean of 3 and 5.6 items, and 2.8% of them made a purchase from the query results.[11]

1.4.1.2 Treatment Effects of Personalizing Discovery Stage

Because users were randomly assigned to either the personalized or non-personalized discovery-stage condition, differences in outcomes between the two conditions identify the causal effect of personalizing the discovery stage. I examine the number of clicks, orders, and revenue in both the discovery and query stages, as well as whether the user initiates a query.

Table 1.1 reports the treatment effects. Personalizing products at the discovery stage increases discovery-stage engagement: users click on 12.1% more items, place 8.4% more orders, and generate 8.2% more revenue compared to the non-personalized condition. However, there is a reduction in engagement in the subsequent query stage. Users exposed to personalized product discoveries are 4.2% less likely to conduct a search query, click on 4.7% fewer items, place 3.5% fewer orders, and generate 4.1% less revenue from the query stage. I further establish the substitution effect between product recommendations at the discovery stage and consumer search by leveraging experimental variation as an instrumental variable in Web Appendix A.2.1.

Since the query stage presents more relevant products, has a higher purchase rate and contributes a larger share of total revenue, even a modest percentage drop in its revenue offsets the revenue gains from personalizing the discovery stage. As a result, total orders and total revenue (combining both stages) do not differ significantly between experimental conditions.

I next examine heterogeneity in the effects of personalizing the discovery stage. Because preferences can evolve over time (Yoon and Simonson, 2008), the extent to which past behavior predicts preferences in the current session likely varies across users, which in turn affects personalization performance. I test whether the benefit of personalized discovery differs for customers with relatively consistent versus dynamic preferences. As a proxy for preference consistency, I use clicking behavior in the 30 days prior to the experiment and compute the average number of unique brands and price levels clicked in each category. Higher values indicate more diverse clicking and thus more dynamic preferences. I classify customers as dynamic if this measure is above the median.[12]

As shown in Figure 1.1, discovery-stage personalization is more effective for customers with consistent preferences, whereas the non-personalized discovery stage performs better for customers who tend to click a broader range of products. This heterogeneity suggests that firms could tailor personalization to users' preference consistency. For example, firms could personalize the discovery

[11] I only consider the top 100 items in the query results, as items ranked lower than 100 are rarely clicked.

[12] The results are robust to alternative definitions of the measure and to different cutoff thresholds.

Table 1.1: Treatment Effects of Personalizing Discovery Stage

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Discovery Clicks	0.404	0.453	12.135	29.018	< 0.001
Discovery Orders	0.009	0.010	8.380	6.629	< 0.001
Discovery Revenue	0.379	0.411	8.222	4.341	< 0.001
I(Initiate a Query)	0.578	0.554	-4.160	-42.344	< 0.001
Query Clicks	3.243	3.090	-4.717	-22.865	< 0.001
Query Orders	0.018	0.017	-3.517	-3.733	< 0.001
Query Revenue	0.576	0.552	-4.055	-2.361	0.018
Total Clicks	3.647	3.543	-2.850	-15.123	< 0.001
Total Orders	0.027	0.027	0.621	0.813	0.416
Total Revenue	0.955	0.963	0.821	0.638	0.524

Notes. This table shows the treatment effects of personalizing the discovery stage. 1,528,471 (50%) users are in the personalized condition. I report the mean of outcomes in both conditions. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

stage only for customers with consistent preferences while keeping broader exposure for customers with more dynamic preferences. Because this "personalized personalization" approach is not tested in the experiment, I evaluate its incremental value using counterfactual simulations.

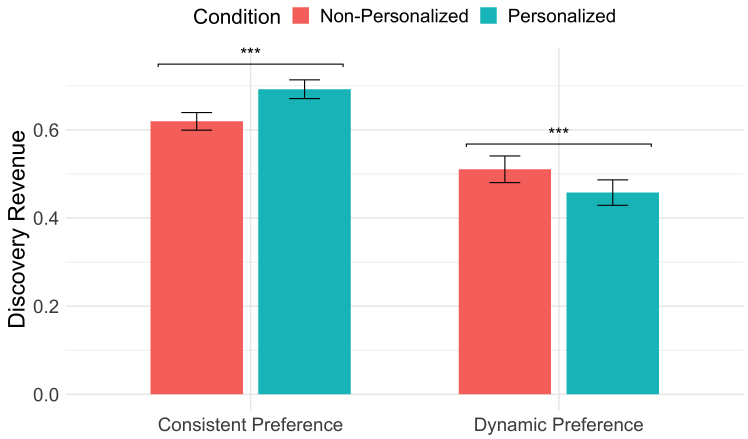
1.4.1.3 Additional Evidence and Discussion

Discovery versus Query Stages Personalization can work differently in the discovery and query stages because the firm's information set and the customer's decision context differ across the two. In the discovery stage, the firm mainly relies on a user's historical activity to curate an assortment of items it expects the user will like, and users are passively exposed to these recommendations. In the query stage, the firm observes not only past behavior but also a contemporaneous signal of intent and preference through the submitted query, so personalization operates within a narrower, query-relevant set of items. Additionally, users actively initiate the query.

To characterize these stages, I summarize descriptive evidence on the products shown at each stage along three dimensions: price, item condition, and product diversity. As shown in Web Appendix A.2.2.1, discovery-stage items are more likely to be in new condition and have slightly higher average prices. The discovery stage also exposes users to a broader set of products than the query stage (see Web Appendix A.2.2.2 for details).

Composition of Customers Customers on Mercari may arrive with a fairly well-defined purchase intent (demand fulfillment) or may browse to discover what they like (demand formation). Although I do not directly observe a user's underlying intent, I present suggestive evidence that behavior in

Figure 1.1: Heterogeneous Effects of Personalizing Discovery Stage



Notes. This figure plots the revenue at the discovery stage for customers in the personalized and non-personalized conditions, separately for those with consistent and dynamic preferences. The sample is restricted to customers who clicked more than one item in the 30 days prior to the experiment. Error bars represent 95% confidence intervals.

my setting is largely demand fulfillment. First, applying the framework of Yuan et al. (2025), I find that personalized discovery-stage recommendations lead users to submit less specific queries (Web Appendix A.2.3.1). Following Yuan et al. (2025), this pattern is consistent with users arriving with an existing demand and using discovery as a substitute for query refinement.

Second, I test whether discovery-stage exposure shapes subsequent query behavior by measuring the semantic alignment between the session's query and the products shown in the discovery stage (Web Appendix A.2.3.2). If users mainly come to be inspired by broad, non-personalized discoveries, alignment should be higher in the non-personalized condition. Instead, I find that alignment is higher under personalization (Web Appendix A.2.3.2).

Third, internal interviews at Mercari indicate that many users treat the platform as a destination for targeted shopping rather than purely open-ended browsing. Taken together, these patterns suggest that customers in my setting often arrive with specific intent, which helps rationalize the observed substitution between discovery and query behavior.[13]

Why Discovery-Stage Personalization May Not Improve Total Outcomes The discovery stage at the beginning of the journey may shift users' beliefs about the continuation value of querying. After observing the discovery list, users may update their expectations about how likely the platform is to surface attractive alternatives. For instance, if users view a personalized discovery list as informative about what the platform can find for them, then a close but unsatisfactory match may

[13]Future work could measure customer composition more directly, for example, by pairing experiments with survey-based intent elicitation or by leveraging richer session-level signals of goal-directed versus exploratory browsing.

lower the perceived gains from submitting a query, reducing query initiation and increasing early exit. To probe this possibility, I stratify users by whether they make a purchase at the discovery stage and compare outcomes between treatment and control within each stratum (see Web Appendix A.2.4). Among users who purchase at the discovery stage, those in the personalized condition are 4.2% less likely to initiate a query, with no significant change in total revenue, consistent with substitution between discovery and query stages. More importantly, among users who do not purchase at the discovery stage, those in the personalized condition are also 4.1% less likely to initiate a query and spend less overall across both stages.[14] These results suggest that discovery-stage personalization can harm a subset of users, potentially by discouraging further query after an unsatisfactory initial set of recommendations.

1.4.2 Personalizing Query Stage

Recall that personalization occurs in the re-ranking stage, where the top 100 products identified by the ranking stage are re-ordered. I first describe the ranking model, LambdaMART, used in this re-ranking stage. LambdaMART (Burges, 2010) is a widely used learning-to-rank algorithm that optimizes item ordering by minimizing a ranking loss. It combines gradient-boosted decision trees with a pairwise loss. LambdaMART has been widely used in practice (e.g., Microsoft's Bing search engine) and has shown strong empirical performance (Yoganarasimhan, 2020; Kaye, 2024). In my experiments, I use LambdaMART for both the personalized and non-personalized rankings. The two models are trained on the same set of product and contextual features, and the personalized model additionally includes user-specific features. Item features include product attributes (e.g., brand) and past engagement measures (e.g., item likes in the past 30 days). Contextual features capture market conditions at the time the query is submitted (e.g., real-time inventory levels). User features include recent activity over the past 30, 60, and 365 days, inferred preferences for brand, price, and category, past purchases, and other user-level signals (e.g., ratings and tenure). Further details on ranking models are in Web Appendix A.3. Since I hold the ranking algorithm fixed and vary only the inclusion of user features, any differences between the personalized and non-personalized conditions can be attributed to personalization features rather than to the algorithm itself.

The personalized query stage field experiment lasted for 12 days in Fall 2024. I randomly assigned 10% of users to the personalized query condition and another 10% to the non-personalized query condition. The remaining 70% of users received the status quo non-personalized ranking

[14]There are several reasons why the platform may fail to deliver a good match at the discovery stage. For instance, one possibility is that the personalization algorithm is imperfect. However, I find similar patterns in a separate field experiment that uses a more advanced personalization approach, suggesting that algorithm improvements alone may not eliminate this channel. Another possibility is that users' preferences vary across sessions (see Section 1.4.1.2), which reduces the predictive value of historical activity at the moment a user arrives.

and 10% of users received the personalized version of the status quo ranking (i.e., the same model augmented with individual features).[15]

1.4.2.1 Summary Statistics

I record user activity on the Mercari app during the experiment.[16] The experiment included 360,679 individuals, of whom 187,818 (52.1%) conducted a search query. Among those who entered the query stage, users clicked 4.2 items on average (with a median of 3), and 4.5% made a purchase from the items in the query results.

To validate that the treatment condition indeed produces personalized rankings at the query stage, I examine the top 12 items displayed on the initial screen of the query results. Using price preference as an illustrative example, Figure 1.2 plots the price distribution of these items separately for customers inferred to prefer high versus low prices based on past purchases.[17] The figure shows that, under personalization, users with a high-price preference are more likely to see higher-priced items, while users with a low-price preference are shown lower-priced options. At the same time, the average item price is similar across the two experimental conditions. In addition to price preferences, Web Appendix A.1.2 shows that the personalized condition also increases the share of top-ranked items that match users' inferred brand preferences.

Figure 1.3 presents the distribution of clicks across item positions by experimental condition. Items ranked higher in the list receive more clicks, and this pattern is more pronounced under the personalized query-stage condition. This suggests that personalization effectively elevates preferred products to more prominent positions.

1.4.2.2 Treatment Effects of Personalizing Query Stage

I present the treatment effect of personalizing the query stage.[18] I examine how personalizing the query stage influences customer engagement and purchases by measuring item clicks, orders, and revenue. I find in Table 1.2 that customers in the personalized condition click on 3% more items, make 9.2% more purchases, and generate 10.6% higher revenue. Table A.11 in the web appendix

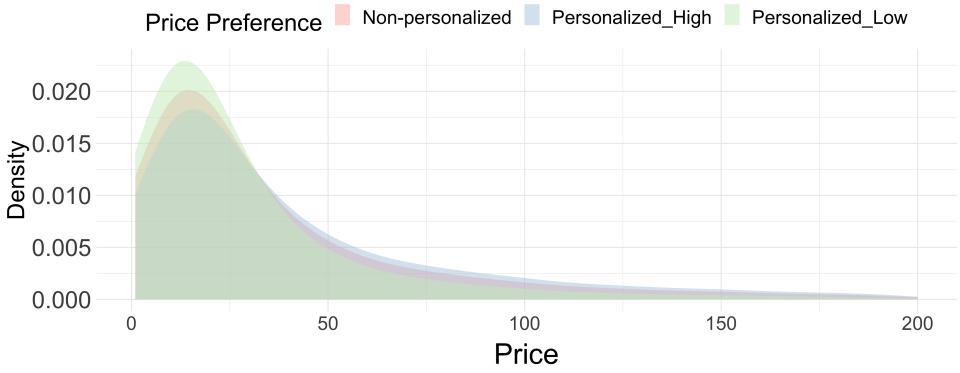
[15] I use LambdaMART with and without user features as my experimental conditions, because the only distinction between these two conditions is the inclusion of user-level inputs. While the remaining conditions also compare personalized and non-personalized rankings, they employ different training approaches, making their results less straightforward to interpret. I use these remaining 80% of users to examine the robustness of personalization effects under an alternative ranking algorithm in Web Appendix A.5.

[16] Users were assigned to treatment conditions at the user level rather than the device level. Thus, a user assigned to the personalized condition received personalized rankings on both the app and the web (including mobile web).

[17] For example, I classify a user as preferring high-priced products if the maximum price among their past purchases exceeds $50. I choose this threshold based on historical data.

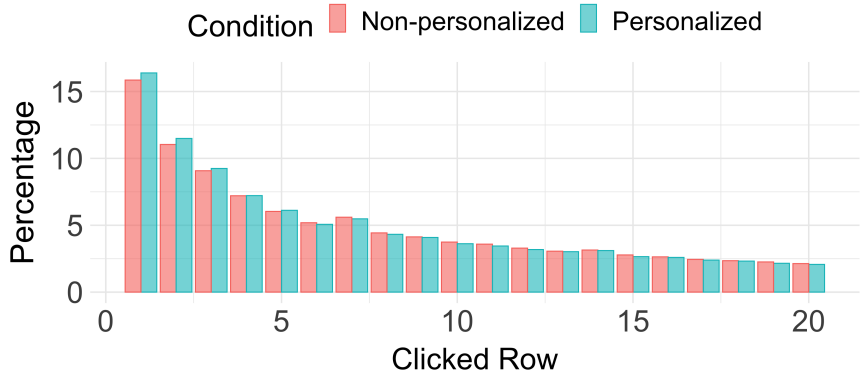
[18] Users were randomly assigned to an experimental condition upon arriving at the platform. By including users who did and did not submit a search query, I estimate the overall impact of later-stage personalization under an intent-to-treat framework.

Figure 1.2: Price Distribution of Products at Query Stage by Customer Price Preference



Notes. This figure shows the price distribution of products displayed on the initial screen at the query stage (top 12 items), segmented by experimental condition ("personalized" vs. "non-personalized") and customers' price preferences inferred from past purchase histories ("high" vs. "low").

Figure 1.3: Distribution of Clicked Positions



Notes. This figure plots the distribution of positions of clicked items across experimental conditions, focusing on the top 20 rows of the query results. Each row contains three items. The x-axis represents the row number, and the y-axis represents the percentage of total clicks.

shows no changes in discovery-stage outcomes, while aggregate outcomes across the two stages are higher in the personalized condition.

Table 1.2: Treatment Effects of Personalizing Query Stage

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Clicks	2.240	2.307	2.995	4.867	< 0.001
Orders	0.025	0.028	9.165	3.882	< 0.001
Revenue	0.838	0.927	10.622	2.870	0.004

Notes. This table shows the treatment effects of the personalizing query stage on query-stage outcomes. There were 360,679 customers in the experimental conditions, and 180,438 (50%) customers in the personalized condition. I report the mean of outcomes in both personalized and non-personalized conditions. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

However, personalization can also impose costs on customers. Because it requires additional computation, it can increase search latency. Search latency is a critical concern for online platforms because even small delays can degrade user experience and reduce key performance indicators.[19] Reflecting this concern, many companies have prioritized latency reduction, including Netflix (Ostuni et al., 2023), Airbnb[20] and Booking.com.[21] In Web Appendix A.4, I present experimental evidence that average search latency is higher in the personalized condition (274 milliseconds) than in the non-personalized condition (259 milliseconds). Moreover, a 1% increase in search latency reduces customer clicks, orders, and revenue by 4.6%, 0.4%, and 1.2%, respectively.

I also examine how the effectiveness of the personalized query stage varies with customers' preference consistency. Using the same procedure outlined in Section 1.4.1.2, I present the results in Figure A.15 in the web appendix. The findings indicate that personalized query rankings have a stronger positive effect for customers with more consistent preferences (i.e., those who tended to click on similar items prior to the experiment). In contrast, there is no significant effect for customers with more dynamic preferences. Given the potential latency costs associated with personalization, the non-personalized query stage might work better for these customers.

[19]For example, a Deloitte study reports that a 0.1-second improvement in site speed increases retail conversions by 8.4% and average cart value by 9.2%. Source: Deloitte Consulting LLP (2021), "Milliseconds Make Millions: The Economic Impact of Site Speed," https://www.deloitte.com/ie/en/services/consulting/research/milliseconds-make-millions.html , accessed March 2026.

[20]Airbnb Engineering (2023), "Embedding-Based Retrieval for Airbnb Search," https://medium.com/airbnb-engineering/embedding-based-retrieval-for-airbnb-search-aabebfc85839, accessed March 2026.

[21]Booking.com Development (2022), "The Engineering Behind Booking.com's Ranking Platform: A System Overview," https://medium.com/booking-com-development/the-engineering-behind-booking-coms-ranking-platform-a-system-overview-2fb222003ca6, accessed March 2026.

1.4.3 Experimental Results Summary and Structural Model Motivation

To summarize, the experimental evidence shows that the personalized discovery stage boosts revenue in that stage but crowds out revenue in the subsequent query stage, leaving total revenue unchanged. When the query stage, in addition to the discovery stage, becomes personalized, customers make more purchases and increase their total spending.

While these experiments shed light on multi-stage personalization, they have limitations. First, they do not on their own establish which stage(s) should be personalized to maximize revenue. The experiments do not cover all relevant combinations of multi-stage personalization. In particular, they do not test a setting in which only the query stage is personalized, so its performance relative to personalizing both stages depends on the size of cross-stage spillovers. In addition, because the two experiments were run in different time periods, it is hard to compare them directly without concern about time trends or other platform-wide changes. The algorithms in the two stages also differ in two field experiments. Although it is standard industry practice to use distinct algorithms at each stage,[22] this variation makes it difficult to disentangle the effect of personalization from the effect of the underlying algorithms.

Second, one might worry that the effects I document are specific to the particular algorithms used in these experiments. To assess robustness, I run additional field experiments that test alternative personalization approaches. I summarize the main takeaways here, with details in Web Appendix A.5. First, because of substitution between stages, improving discovery-stage personalization does not translate into higher total outcomes. Second, when I hold the ranking algorithm fixed, adding individual-level features generally increases platform revenue. Even with these additional experiments and analyses, the set of experiments on Mercari cannot cover all relevant algorithmic designs. For example, some platforms (such as Amazon) blend individual preferences with aggregate popularity signals (e.g., "Best Seller" lists). Moreover, no experiment evaluates the value of "personalized personalization."

These limitations motivate a structural model that enables counterfactual simulations. The model allows me to: (i) hold the underlying ranking models fixed while varying only the stage at which personalization is applied; (ii) compare alternative stage-specific personalization regimes, including regimes not directly tested in the field; and (iii) evaluate alternative algorithmic designs, including hybrids that combine individual-level and aggregate signals. More broadly, the structural model complements these experiments by serving two purposes. First, it provides a mechanism-based framework that clarifies why personalization can generate cross-stage spillovers, which helps interpret the experimental results and organize the underlying economic forces. Second, it functions

[22] For example, Netflix employs Large Language Models (LLMs) for personalized discoveries and neural networks to rank query results. See Netflix Tech Blog at https://netflixtechblog.com/foundation-model-for-personalized-recommendation-1a0bd8e02d39, accessed March 2026.

as an offline decision-support tool that allows the platform to evaluate counterfactual policies that would be costly to test exhaustively in the field.[23]

Importantly, the experiments remain essential even with the structural model. The first experiment provides motivating evidence that adding personalization to more stages does not necessarily improve total outcomes, while the second experiment provides exogenous variation that supports identification of the model.

1.5 Structural Search Model

I begin by laying out the structural search model and its key components: the utility function, the information set and beliefs, and the costs. I then formalize the decision rules that govern customer behavior. Finally, I describe the estimation approach and discuss how the parameters are identified.

1.5.1 Model Setup

An overview of the structural search model is presented in Figure 1.4. The model consists of two stages: the discovery stage and the query stage. In each stage, customers engage in sequential search. I assume that the customer's decision to visit the platform is exogenous (e.g., it is independent of whether the discovery stage is personalized).[24] Additionally, customers are assumed to have a preexisting unit demand and stable preferences that remain fixed throughout the platform visit in each session (e.g., viewing or clicking products does not change their preferences).[25,26]

In the discovery stage, the customer is presented with a list of recommended products. They can costlessly observe the visible (i.e., pre-inspection) attributes of these products. In my setting, I consider price, brand, and category as visible attributes.[27] Upon viewing the discovery list, the customer faces three options: (1) inspect an item from the discovery list to access its product page, (2) conduct a search query to switch from the discovery stage to the query stage, or (3) leave the platform by opting for the outside option (e.g., no purchase or purchase on competing platforms). If the customer inspects an item, they fully uncover its hidden attributes and (post-inspection) match

[23]The structural model is not intended for production deployment. Instead, it is intended to inform high-level design choices about which stage and for whom personalization should be applied.

[24]As personalization is now widespread and many users even expect platforms to provide personalized recommendations at the discovery (landing) page, I assume that whether the discovery stage is personalized does not affect overall traffic to the platform.

[25]As shown in Section 1.4.1.2, customers visit the platform to fulfill an existing demand, thereby supporting my model assumption.

[26]In my estimation sample, I focus on customers whose intent is to purchase a product in the "Video Games and Consoles" category. Details on how I construct the estimation sample are provided in Section 1.5.4.1.

[27]As shown in Figure A.14 in the web appendix, customers observe each product's price and thumbnail image. I assume that customers infer the product's category and brand from the thumbnail, which typically displays a brand logo.

value. Hidden attributes in my setting include the number of photos, the item's physical condition, item age, and the seller's past sales. The inspected item enters the customer's consideration set. After inspecting an item, a customer can either terminate the process by purchasing an item in their consideration set or choosing the outside option, or continue by inspecting another item or conducting a search query. Customers can purchase only from their consideration set because, on Mercari, an item can be purchased only from its product page (i.e., after inspection).

If the customer conducts a search query, they exit the discovery stage and enter the query stage. In the query stage, they are presented with a new list of products relevant to the query, and can costlessly view the visible attributes of these products. The set of visible and hidden attributes is consistent across both stages. After viewing the query results, the customer faces two options: (1) inspect an item from the query result list, or (2) leave the platform by choosing the outside option.[28] Similarly, by inspecting an item in the query stage, the customer learns its hidden attributes and (post-inspection) match value. These inspected items expand the customer's consideration set.

I adopt the standard assumptions of a sequential search model. Specifically, I assume that it is costless for customers to navigate from a product page back to the discovery list or the query results. In addition, within each session, customers have perfect recall: they remember the utility from previously inspected products (or can costlessly revisit them). I do not model refinement actions such as sorting or filtering because, consistent with Korganbekova and Zuber (2023), I find no experimental evidence that sorting or filtering significantly affects purchase behavior (see Table A.12 in the web appendix).

1.5.2 Model Specification

1.5.2.1 Utility

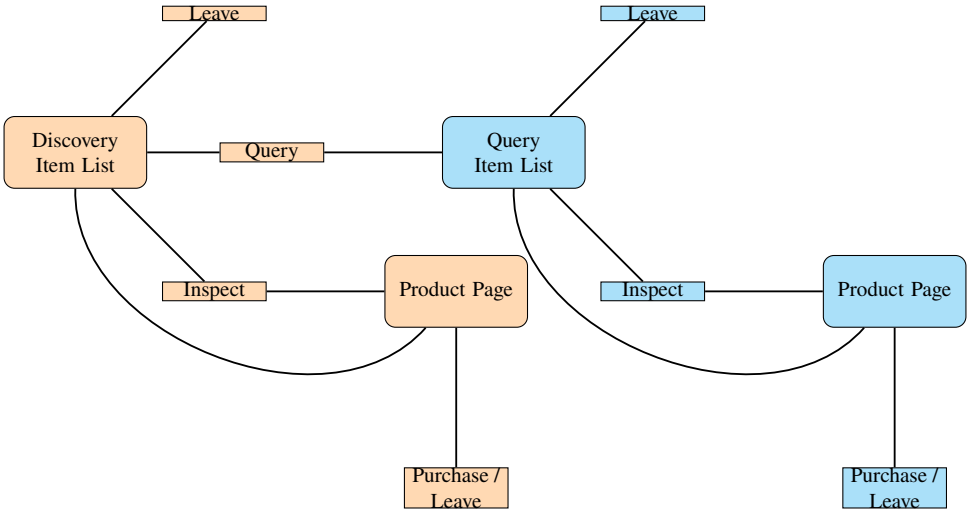
Customer i derives the following utility from purchasing product j:

$$u_{ij} = X_j'\alpha_i + Z_j'\beta_i + \varepsilon_{ij}^{pre} + \varepsilon_{ij}^{post}. \tag{1.1}$$

X_j denotes visible attributes (i.e., attributes revealed before inspection), and Z_j denotes hidden attributes (i.e., attributes revealed after inspection). The parameters α_i and β_i capture customer i's preferences for visible and hidden attributes respectively. Customer i's product-specific pre-

[28] In practice, customers may also return to the discovery stage or submit another search query. In this case, the model could be extended to allow for two additional actions in the query stage: returning to the discovery stage to inspect previously uninspected items, or submitting another query. Both actions mean that customers will enter a new sequential search problem, in which they update cost structures and carry over their current highest realized utility. In this paper, I abstract away from these behaviors to preserve model tractability. In my estimation, I keep only the first query if customers submit multiple queries within a session (16.5% of sessions) and only the first discovery-stage activity if the user returns to the discovery stage after entering the query stage (3.5% of sessions).

Figure 1.4: Structural Search Model Overview



Notes. Consumers begin the process at the discovery stage, where they are shown a list of discovery items. If they choose to conduct a query to enter the subsequent query stage, they will be presented with a list of query results.

inspection idiosyncratic taste shock ε_{ij}^{pre} is known to the customer before inspection, and post-inspection taste shock ε_{ij}^{post} is known to the customer after inspection. Both taste shocks are unknown to the researcher. Examples of ε_{ij}^{pre} include the i's perceived attractiveness of product j's thumbnail image, and examples of ε_{ij}^{post} include the match between the customer i and the seller of product j. The utility from the outside option u_{i0} is normalized to be ε_{i0}^{post}.

I assume that the idiosyncratic taste shocks (ε_{ij}^{pre} and ε_{i0}^{post}) follow normal distributions, consistent with the commonly adopted assumptions in Weitzman-type search models (e.g., Kim et al., 2010; Chen and Yao, 2017). The standard deviation of the pre-inspection taste shock is set to one as a scale normalization (i.e., $\varepsilon_{ij}^{pre} \sim \mathcal{N}(0,1)$). I assume that the post-inspection taste shock is independently and identically distributed across customers and products, i.e., $\varepsilon_{ij}^{post} \sim \mathcal{N}(0, \sigma^2_{\varepsilon^{post}})$, where its variance $\sigma^2_{\varepsilon^{post}}$ is estimated empirically.[29]

I allow for customer heterogeneity in price sensitivity and hidden attributes through random coefficients, such that $\alpha_i^{price} \sim \mathcal{N}(\alpha^{price}, \Omega_{\alpha^{price}})$ and $\beta_i \sim \mathcal{N}(\beta, \Omega_\beta)$, where α^{price} and $\Omega_{\alpha^{price}}$ represent the mean and variance of price sensitivity, and β and Ω_β denote the mean and variance of preferences for hidden attributes.

[29] See Ursu et al. (2024) for detailed discussions of why fixing the variance of the pre-inspection shock does not lead to a loss of generality, but why both taste shock standard deviations cannot be fixed simultaneously if one aims to monetize the inspection cost.

1.5.2.2 Information Set and Beliefs

I use ${A_i}^D$ and ${A_i}^Q$ to denote the list of products seen by customer i in the discovery stage D and query stage Q, respectively. Customer i's initial information set consists of item attributes X_j, ε_{ij}^{pre} and position pos_{ij} of ${A_i}^D$, their preferences $\{\alpha_i, \beta_i\}$, the search query q_i, and the value of the outside option u_{i0}.

Conducting a Query While still in the discovery stage, customer i cannot observe the query-stage results that would appear after submitting a query. Instead, the customer forms beliefs about the value of conducting query q, denoted by u_i^q. If the discovery-stage products do not match the customer's purchase intent, these beliefs are based only on the expected payoff from the query-stage environment. If the discovery-stage products match the customer's purchase intent, the customer additionally incorporates information revealed in the discovery stage when forming beliefs.

I first describe the case in which the discovery-stage products do not match the customer's purchase intent. Consistent with prior work (Koulayev, 2014; De los Santos and Koulayev, 2017; Gu and Wang, 2022), I summarize the value of entering the query stage by the maximum utility among the products the customer expects to *inspect* after submitting query q. This is because when conducting a query, the customer cares only about the product with the highest utility in the query results. Furthermore, focusing on inspected products is important because personalization changes the ordering of the same set of query-relevant items, and inspection is position-dependent (see Figure 1.3).[30] Formally, for a query-result list $\tilde{A}^Q$, let $\bar{\tilde{A}}^Q$ denote the set of inspected products. I define the query-stage signal as $u_{iq}^Q \equiv \max\{u_{ij} : j \in \bar{\tilde{A}}^Q\}$.

To operationalize this belief, I estimate the distribution of query-stage lists using the empirical distribution. For each query q, I construct $H(\tilde{A}^Q \mid q, treatment, segment)$, where *treatment* indicates the personalization treatment and *segment* (only for treated users) corresponds to the behavioral inputs used by the personalized ranking algorithm (e.g., user price sensitivity, brand preferences, and activity). If customer i is in the non-personalized condition, then H is formed using all observed query-result lists (including positions) shown to users who are in the non-personalized condition and submit query q. If customer i is in the personalized condition, I further restrict H to query-result lists shown to users who are in the personalized condition, submit query q and belong to the same segment as i. Conditioning on segment captures systematic differences in what users expect to see under personalization. For example, price-sensitive users may expect to see lower-priced items ranked near the top (Figure 1.2). Additional details are provided in Web Appendix A.6.1.

[30] Personalized and non-personalized query-stage rankings contain the same set of products in the top 100 positions for a given query but differ in their ordering. In the absence of inspection costs, the customer would expect to reach the same maximum utility with and without personalization.

Because computing the distribution of $\max\{u_{ij} : j \in \tilde{\bar{A}}^Q\}$ is computationally costly, I approximate it via bootstrap simulation (see Koulayev, 2014 and Ghose et al., 2019). For each user i, I repeatedly (i) draw a query-result list $\tilde{A}^Q$ from H given $(q, treatment, segment)$, (ii) simulate which products the user inspects, yielding $\tilde{\bar{A}}^Q$, and (iii) compute $\max_{j \in \tilde{\bar{A}}^Q} u_{ij}$. Across bootstrap draws, I obtain the mean and variance and summarize the query-stage signal as

$$u_{iq}^Q \sim \mathcal{N}\big(\mu_{iq}^Q, (\sigma_{iq}^Q)^2\big), \qquad \mu_{iq}^Q = \mathbb{E}\left[\max_{j \in \tilde{\bar{A}}^Q} u_{ij}\right], \qquad (\sigma_{iq}^Q)^2 = \mathrm{Var}\left(\max_{j \in \tilde{\bar{A}}^Q} u_{ij}\right). \tag{1.2}$$

Next, when the discovery-stage products match the customer's intent, I treat the discovery stage as an additional signal about the latent payoff from querying. I capture the discovery-stage information relevant for beliefs by constructing an analogous signal from the products the customer inspects in the discovery stage. Let $\tilde{\bar{A}}^D$ denote the set of inspected products in the discovery stage. I define

$$u_{iq}^D \equiv \max\{u_{ij} : j \in \tilde{\bar{A}}^D\}, \qquad u_{iq}^D \sim \mathcal{N}\big(\mu_{iq}^D, (\sigma_{iq}^D)^2\big). \tag{1.3}$$

When the customer's intent does not match the discovery-stage content, the customer relies only on the expected query-stage payoff, so $u_i^q = u_{iq}^Q$. When the customer's intent matches the discovery-stage content, both u_{iq}^Q and u_{iq}^D are noisy but informative signals about u_i^q. Let their precisions be $\tau_{iq}^Q = 1/(\sigma_{iq}^Q)^2$ and $\tau_{iq}^D = 1/(\sigma_{iq}^D)^2$. Assuming conditional independence, the customer combines the two signals using Bayes' rule. The resulting posterior belief is

$$u_i^q \sim \mathcal{N}\big(\mu_i^q, (\sigma_i^q)^2\big), \qquad \mu_i^q = \frac{\tau_{iq}^Q \mu_{iq}^Q + \tau_{iq}^D \mu_{iq}^D}{\tau_{iq}^Q + \tau_{iq}^D}, \qquad (\sigma_i^q)^2 = \frac{1}{\tau_{iq}^Q + \tau_{iq}^D}. \tag{1.4}$$

In this case, discovery- and query-stage information jointly shape the customer's beliefs about the value of conducting a query.

Inspecting a Product Before inspecting product j, customer i observes j's visible attributes X_j, its rank position pos_{ij}, and its stage s_{ij}. These observables can affect the customer's beliefs about the product's utility in two ways. First, visible attributes may be informative about hidden attributes. For example, a higher price may be correlated with better physical condition. Ignoring this correlation can bias estimates of preferences. Second, both rank position (Fong et al., 2024; Kaye, 2024) and stage can shift beliefs. For instance, conditional on the same visible attributes, items shown in the discovery stage tend to be associated with different seller characteristics than items shown in the query stage (see Figure A.12 in the Web Appendix).[31]

[31] One plausible reason is that the platform showcases products from different sets.

Putting these channels together, I represent customer i's pre-inspection beliefs about product j's utility by a conditional distribution G_i and write $u_{ij} \mid (X_j, s_{ij}, pos_{ij}) \sim G_i$. I assume that G_i is normal, with mean and variance that depend on (i) the item's visible attributes X_j, (ii) its stage s_{ij}, and (iii) its rank position pos_{ij}. To discipline beliefs about hidden attributes, I model the relationship between hidden attributes Z_j and visible attributes X_j using the multivariate linear regression $Z_j = \Phi_0 + \Phi_1 X_j + \boldsymbol{\eta}_j$. In addition, to capture stage- and position-specific information, I estimate the empirical distribution of product attributes within each (s, pos) cell by pooling items observed at that stage and position across sessions.[32] Details are presented in Web Appendix A.6.2.

Under these assumptions, the conditional mean of G_i can then be written as $X_j'\alpha_i + \varepsilon_{ij}^{pre} + (\hat{\boldsymbol{\Phi}}_{\mathbf{0}} + \hat{\Phi}_1 X_j)'\beta_i$. The conditional variance of G_i is $\beta_i^\top \hat{\Sigma}_\eta \beta_i + \sigma^2_{\phi_{ij}}$, where $\hat{\Sigma}_\eta$ is the estimated variance-covariance matrix of the regression residuals $\boldsymbol{\eta}_j$, and $\sigma^2_{\phi_{ij}}$ is the customer's belief about the variance of the post-inspection idiosyncratic taste shock ε_{ij}^{post}. I assume $\sigma^2_{\phi_{ij}}$ equals its true variance $\sigma^2_{\varepsilon_{ij}^{post}}$.

1.5.2.3 Costs

Cost of Conducting a Query I model the cost of conducting a query as:

$$c_i^{query} = exp(\tau_0 + \tau_1 Personalized_i), \tag{1.5}$$

where $Personalized_i$ is an indicator that equals 1 if customer i is in the personalized query-stage ranking condition, and 0 otherwise. I use this term to capture any potential costs borne by customers as a result of personalization. For example, as presented in Section 1.4.2.2, personalization increases search latency, which results in longer query result loading times and subsequently reduces product inspection decisions.[33] Similar to Gu and Wang (2022), who assume that customers form rational expectations about loading time costs, I assume that customers form rational expectations about the costs introduced by personalization. The baseline cost of conducting a search query is captured by τ_0, which can also be interpreted as the cost incurred when the query-stage ranking is not personalized. The exponential functional form is standard in the literature (e.g., Kim et al., 2010; Ghose et al., 2012; Chen and Yao, 2017) and ensures that costs are positive.

[32] In principle, one could allow these beliefs to vary by personalization condition or customer segment. In my setting, allowing the attribute distributions to vary by personalization condition does not improve model fit. To preserve precision, I impose a common prediction rule across personalization conditions. While the prediction rule for hidden attributes is common across customers, the implied distribution of utility remains heterogeneous through customer-specific preference coefficients.

[33] One might expect discovery-stage personalization to increase latency and impose additional costs as well. However, I do not find evidence of this in my setting: discovery-stage latency does not differ statistically between the personalized and non-personalized conditions. A plausible explanation is that query-stage ranking requires real-time computation based on the user's query, whereas discovery-stage recommendations can be largely precomputed and served from the backend.

Inspection Cost Since the layout of product lists in the discovery and query stages is identical (see Figure A.14 in the web appendix), I adopt the same inspection cost structure in both stages. Inspection costs are specified as:

$$c_{ijs}^{inspect} = exp(\delta_{0s} + \delta_1 pos_{ijs}), \quad s \in \{D, Q\}, \tag{1.6}$$

where pos_{ijs} is the position of product j in customer i's list at stage s. As individuals typically read and process product information sequentially from top left to bottom right (Rayner, 1998), items in lower positions can be viewed as incurring higher time or cognitive costs.[34] This interpretation is consistent with Figure A.16 in the Web Appendix, which shows that items in lower positions receive fewer clicks and are inspected later in the session. The parameter δ_{0s} is the baseline inspection cost at stage s. I allow this baseline cost to differ between the discovery and query stages, as, for instance, inspection in the query stage may be subject to tighter time constraints given its later point in the customer journey.

1.5.3 Optimal Customer Behavior

The model builds on the classic cost-benefit framework of Weitzman (1979), in which individuals evaluate each action by weighing its expected gains against the costs. Optimal decision rules are characterized by reservation utilities, which define the threshold at which an individual is indifferent between continuing the process and stopping.

The reservation utility of a product j equates the marginal gain from inspecting product j with the marginal cost of doing so. Intuitively, it represents the utility level a product must offer for the customer to be just indifferent between inspecting that product and stopping. For each uninspected product j, the customer computes its reservation utility z_{ij} as:

$$c_{ijs}^{inspect} = \int_{z_{ij}}^{\infty} (u_{ij} - z_{ij})\, dG_i(u_{ij} \mid X_j, s_{ij}, pos_{ijs}) \tag{1.7}$$

Similarly, the reservation utility for conducting a search query q equates its marginal gain with its expected marginal cost. The customer computes the reservation utility z_i^q as:

$$c_i^{query} = \int_{z_i^q}^{\infty} (u_i^q - z_i^q)\, dF_i(u_i^q). \tag{1.8}$$

Optimal behavior is determined by the relative values of reservation utilities. Under the assump-

[34] Jameei Osgouei et al. (2026) indicate that, when presented with a matrix, two reading patterns yield qualitatively similar results: (1) reading each column from top to bottom before moving to the next column, and (2) reading each row from left to right before proceeding to the next row.

tion that customers' beliefs follow a normal distribution, I use the solution derived by Kim et al. (2010) and implement a look-up table to recover reservation utilities (see detailed implementation discussion in Ursu et al. (2024)). Let $R(n)$ denotes the index of the selection option with the n-th largest reservation utility $z_{R(n)}$.[35] Let $O = \{A, q\}$ denotes the set of available options, which includes both the set of uninspected products A and the unvisited query q. Let $\bar{A}$ be the set of products that have been inspected across both stages. The customer's current highest realized utility, denoted by u^*, is defined as the maximum utility among the inspected products and the outside option: $u^* = \max\left\{\max_{j\in\bar{A}} u_j,\ u_0\right\}$. I outline the optimal rules as follows.

Selection Rule: The customer ranks all available options (including uninspected products and the query if in the discovery stage) in decreasing order of reservation utilities, and then sequentially inspects each option in that order.

Stopping Rule: The customer stops the process if the current highest realized utility exceeds the maximum reservation utility of all remaining available options (i.e., uninspected products and the unvisited query if in the discovery stage). Thus, stopping at step k implies: $u^* \geq \max_{n=k+1}^{O} z_{R(n)}$.

Choice Rule: The customer purchases the product with the highest realized utility among the inspected products and the outside option. Therefore, purchasing product j implies: $u_j = u^*$.

1.5.4 Estimation and Identification

1.5.4.1 Estimation Sample Construction

In the structural search model estimation, I focus on consumer sessions that reveal purchase intent for items in the "Video Games and Consoles" category. I choose this category because its products are relatively standardized, meaning that, for example, product attributes like photos provide limited incremental information. In addition, it is one of Mercari's highest-volume categories in terms of queries, inspections, and sales. I define a consumer session as exhibiting purchase intent in the "Video Games and Consoles" category if it meets any of the following criteria: (i) the customer submits a relevant search query; (ii) if no query is submitted, the customer purchases a product from that category in the discovery stage; or (iii) if neither a query is submitted nor a purchase is made, the customer inspects only items from that category. In cases (ii) and (iii) where no query is conducted, I proxy for the customer's intended query using their most recent query in the "Video Games and Consoles" category.[36]

Because discovery-stage products can come from multiple categories depending on a customer's prior activity and many items are never inspected, it is difficult to identify category-specific co-

[35] For notational simplicity, I drop the customer-specific subscript i in what follows.

[36] Note that for a product in the "Video Games and Consoles" category to appear in the discovery stage, the customer must have previously submitted a related query.

efficients in the utility function. Thus, instead of including individual product categories, I use a dummy variable to indicate whether product j belongs to the same category as customer i's purchase intent (i.e., "Video Games and Consoles" category in my model estimation).

1.5.4.2 Model Estimation Approach

The parameters to be estimated Θ include the preference parameters $\{\alpha, \beta, \Omega_{\alpha^{price}}, \Omega_{\beta}\}$, cost parameters $\{\delta, \tau\}$, and the post-inspection taste shock standard deviation $\sigma_{\varepsilon^{post}}$. I estimate the model using data from the query-stage personalization experiment. The random assignment in the experiment introduces exogenous variation in customers' propensity to conduct a query. The dataset contains, for each customer, the full list of products shown at both stages (with their attributes and positions), the submitted query, the inspected and purchased items (including the outside option), and the order of inspections. I also observe each customer's personalization condition and segment.

I estimate the model using NNE proposed by Wei and Jiang (2025). The core idea behind NNE is to train a deep learning model to learn the mapping from data to the underlying structural parameters. In many structural econometric models, including the sequential search model, data moments are sufficient for pinning down parameters. This approach offers several advantages over traditional methods such as the simulated method of moments (SMM) and maximum likelihood estimation (MLE). First, NNE eliminates the need to evaluate high-dimensional integrals over unobservables, which are often required in SMM and MLE. Instead, the neural network learns the mapping from data to parameters. Second, NNE is robust to redundant or noisy moments, as the learning algorithm automatically prioritizes the most informative features for parameter identification. Third, this approach offers computational efficiency: once trained, the estimator can generate point estimates and statistical uncertainty measures with minimal additional computation. Finally, NNE is relatively robust to the choice of neural network architecture and hyperparameters.

I estimate my model parameters using NNE as follows. First, given the products and their positions in my actual estimation sample, I simulate consumer decisions over a grid of candidate parameter values. The simulated decisions include whether to inspect a product, conduct a query, and purchase a product. Second, for each parameter vector, I summarize simulated outcomes into a set of data moments (e.g., average purchase rate), computed at the product, session, and individual levels. Third, I train a neural network using the simulated data moments and their associated parameter values as the training set. These training steps learn an inverse mapping from moments to parameters: given a set of observed data moments, what parameter values are most likely to have generated them? Finally, I compute the same set of moments from the actual estimation sample and feed them into the neural network. The network then outputs the estimated model parameters and covariance matrix. Further details on NNE are provided in Web Appendix A.7.1.

1.5.4.3 Identification

This section presents the informal identification of parameters. I discuss the identification of position effects δ_1, preference parameters $\{\alpha_i, \beta_i\}$, price parameter α^{price}, cost parameters $\{\tau_0, \tau_1, \delta_0\}$, and standard deviation of post-inspection taste shock $\sigma_{\varepsilon^{post}}$. I perform a Monte Carlo simulation to verify parameter identification and find that the estimation procedure precisely recovers all parameters (see Web Appendix A.7.2).

Position Effects δ_1 At the discovery stage, products are selected and ranked based on customers' past histories. Although the proprietary discovery-stage algorithm is not directly observed, internal discussions with Mercari reveal its key input variables, such as historical search queries and item clicks. Similar to De los Santos and Koulayev (2017), I use these inputs to capture the systematic component of discovery products. Concretely, I estimate a predictive model of position using these algorithm inputs and obtain the fitted value. I then include this value as a control function in the utility specification and treat the resulting residual variation as exogenous to individual preferences.[37]

At the query stage, as detailed in Section 1.3.1, when a customer conducts a query, the platform first retrieves a set of items relevant to that query, then orders them by relevance, and finally re-orders the top 100 items using either a personalized or a non-personalized LambdaMART model. This procedure implies that the candidate set is primarily determined by query relevance, while the final ordering reflects the re-ranking model. I leverage full knowledge of the deployed LambdaMART specification and its input features to compute each item's ranking score in each session. These ranking scores determine an item's relative position in the list, and I include them as a control for position.

At both stages, I observe customers' repeated inspection decisions across positions, and these decisions are informative about the customer's sensitivity to position.

Preference Parameters α_i, β_i As discussed above, product attributes are not endogenous in the query stage. However, they may be endogenous in the discovery stage, since the algorithm selects and ranks products based on customers' past behaviors. I correct for this endogeneity using a control-function approach that leverages the discovery-stage algorithm's input variables, following the same approach used above.

I identify customer preferences based on patterns in purchase and inspection. As I observe purchases directly, the identification of the preference parameters is as in a conditional choice model.

[37] Since the recommendation algorithms at the discovery stage were designed to encourage serendipitous exploration during the period from which my estimation sample is drawn, the residual variation can be interpreted as the system's exploratory component, which is plausibly exogenous to individual preferences.

Similar to how purchase decisions among a set of products reveal mean preference coefficients in a traditional discrete choice model, purchase decisions among inspected products inform the mean preference coefficients in my model. Given the low purchase incidence (3.1% of sessions), I further incorporate customers' inspection decisions. The selection of which product to inspect helps to pin down the preference parameters (e.g., Bronnenberg et al., 2016; Chen and Yao, 2017; Honka et al., 2017). For example, a tendency to inspect more low-priced items indicates larger price sensitivity.

The heterogeneity of the preference coefficients across customers can be recovered since I observe a sequence of decisions for each customer, including product inspection, query submission, and purchase. Repeated observations for each customer are informative for customer-specific preference coefficients, while the variation in these coefficients across individuals enables me to identify preference heterogeneity.

Price Parameter α^{price} Estimating the price parameter is prone to endogeneity, as prices may correlate with unobserved product-level taste shocks. To address this, I follow a standard approach using instruments constructed from the product attributes of competing products (see Gandhi and Houde (2019)). I define a "market" in my setting as the set of products that belong to the same category-brand cluster and are listed in the same week.

Cost Parameters δ_0, τ_0, τ_1 Conditional on the consideration set (i.e., a set of inspected items), the purchase decision depends only on consumer preferences and not on inspection costs. Therefore, I can separate inspection costs from preference parameters by using conditional purchase probabilities. The joint variation in inspection and purchase enables the identification of the distribution of consumer preferences. Conditional on those preferences, observed inspection decisions reveal the distribution of inspection costs. I identify the baseline inspection costs δ_0 using the number of product inspections in each stage.

To identify the effect of personalization on customers' cost of conducting a query τ_1, I exploit variation in customers' experimental assignment (i.e., whether they are in the personalized or non-personalized condition). The baseline cost of conducting a search query τ_0 is inferred from the frequency with which customers conduct a query. In my estimation sample, 84.7% of sessions involve a query while 15.3% do not.

Post-Inspection Taste Shock Standard Deviation $\sigma_{\varepsilon^{post}}$ The variation in the number of items a customer inspects pins down the ratio of the inspection cost to the post-inspection taste shock standard deviation. These two parameters can be separately identified by the parametric form. As the product position shifts inspection costs but not post-inspection utility, this inspection cost shifter allows me to identify the standard deviation of post-inspection taste shock (Yavorsky et al.,

2021).

1.6 Estimation Results and Counterfactual Simulations

1.6.1 Estimation Results

Table 1.3 presents the estimation results of the sequential search model. The comparisons between observed and estimated data patterns in Web Appendix A.7.3 suggest that my model fits the data well. To express the coefficients in monetary terms, I normalize each estimate by dividing it by the absolute value of the price coefficient. I find that customers derive higher utility from items that are lower priced, have fewer photos, are newly listed (i.e., smaller item age), are in better physical condition, and are sold by more experienced sellers. The estimation results also reveal heterogeneity in these preferences. The positive coefficient of "Video Games and Consoles" category is not surprising as customers prefer products from categories that align with their purchase intent. Relative to unbranded products, customers are willing to pay premia for four major brands: Sony ($0.54), Microsoft ($0.65), Nintendo ($0.34), and Pokémon ($0.47).

In terms of costs, I estimate that the baseline inspection cost is $0.24 in the discovery stage and $0.17 in the query stage. The baseline cost of conducting a search query is $0.16. To quantify position effects, I compute the dollar equivalent of the change in inspection cost from a one-unit increase in position. The estimated position effect is $0.06 in the discovery stage and $0.04 in the query stage. For the initiation screen of 12 products, these inspection costs correspond to $0.72 for the discovery list and $0.48 for the query list. Personalization introduces frictions in transitioning to the query stage, with an estimated cost equivalent to about $0.16.

I have related the estimated search costs to comparable estimates in the existing literature. Prior studies have documented position costs of approximately $1.92 per position (Ursu 2018) and clicking costs of approximately $0.20 (Korganbekova and Zuber 2023). My estimates are lower in magnitude than those reported by Ursu (2018) and Korganbekova and Zuber (2023), which is consistent with the lower average product prices on Mercari relative to the hotel and furniture markets studied in those papers. However, the estimates are comparable to those in Zhang et al. (2023). They examine consumer search costs in the women's sandals category of an e-commerce platform, a context with product prices closer to those in my setting.

1.6.2 Counterfactual Simulations

As discussed in Section 1.4.3, some questions cannot be fully answered by field experiments alone and instead require counterfactual simulations. The first counterfactual simulation explores

Table 1.3: Estimation Results of Structural Search Model

Variables	Mean	SE	Heterogeneity (SD)	SE	Dollarized Value ($)
Preferences					
Constant	-3.4608	0.8528			
Price	-0.2453	0.0485	0.1134	0.0437	
1(Video Game Category)	0.8369	0.2501			0.6548
Brand: Sony	0.6929	0.1554			0.5422
Brand: Microsoft	0.8365	0.1558			0.6545
Brand: Nintendo	0.4343	0.1319			0.3398
Brand: Pokemon	0.5985	0.2166			0.4683
Brand: Others	0.2365	0.0965			0.1851
N Photos	-0.4013	0.0851	0.0284	0.0179	-0.3140
Item Age	-0.7931	0.3176	0.0367	0.0124	-0.4284
Physical Condition	0.1283	0.0305	0.0284	0.0476	0.1004
Seller Past Sales	0.0912	0.6099	0.0770	0.0477	0.0747
Costs					
Inspection Base (Discovery)	-1.1786	0.1034			0.2408
Inspection Base (Query)	-1.5134	0.0771			0.1723
Position	0.2157	0.0705			0.0588 / 0.0415
Query Base	-1.5985	0.5616			0.1582
Personalization	0.8237	0.2762			0.1584

Notes. The estimation sample includes 8,138 customers and 17,884 sessions. The last column reports the monetary value of each variable, calculated by dividing the coefficient estimate by the absolute value of the logged price coefficient. For example, the implied dollarized value of the preference is given by $\frac{\hat{\alpha}}{\exp(\hat{\alpha}^{price})}$ and the baseline cost is calculated as $\frac{\exp(\hat{\delta})}{\exp(\hat{\alpha}^{price})}$. Item age and seller past sales are (log+1)-transformed. The omitted brand is unbranded.

how personalization at different stages of the customer journey affects total revenue. The second counterfactual simulation quantifies the value of personalized personalization.

1.6.2.1 Multi-Stage Personalization

To isolate stage-specific effects from algorithm effects, I rank products in both the discovery and query stages using the same utility-based scoring rule. The ranking score is a convex combination of (i) predicted utility under an individual customer's preferences and (ii) predicted utility under average population preferences (see Web Appendix A.8 for details). I vary the weight on (i) versus (ii) to capture different personalization intensities: popularity-based ranking (no personalization), full personalization, and intermediate policies that blend individual-level and population-level signals.[38] Using these designs, I evaluate how personalization at each stage affects total revenue by comparing policies that apply personalization (or partial personalization) at neither stage, at exactly one stage (discovery or query), or at both stages.

The results, presented in Figure 1.5, highlight the importance of selecting the appropriate stage for personalization. Personalizing only the early discovery stage yields a modest increase (4.4%) in total revenue relative to no personalization at either stage. Extending personalization to both stages does not maximize revenue. Instead, personalizing only the query stage delivers the largest uplift, generating 2.9% more revenue than personalizing both stages. To understand this pattern, I examine how discovery-stage personalization affects continuation to the query stage. Discovery-stage personalization improves intent matching (by 33%), making customers more likely to use their discovery-stage experience to update beliefs about the expected gains from querying. However, the discovery stage may not always provide a strong fit with the customer's current demand. Customers who use this weak match to infer the value of querying may update toward a lower expected payoff, reducing continuation value and increasing early exit.

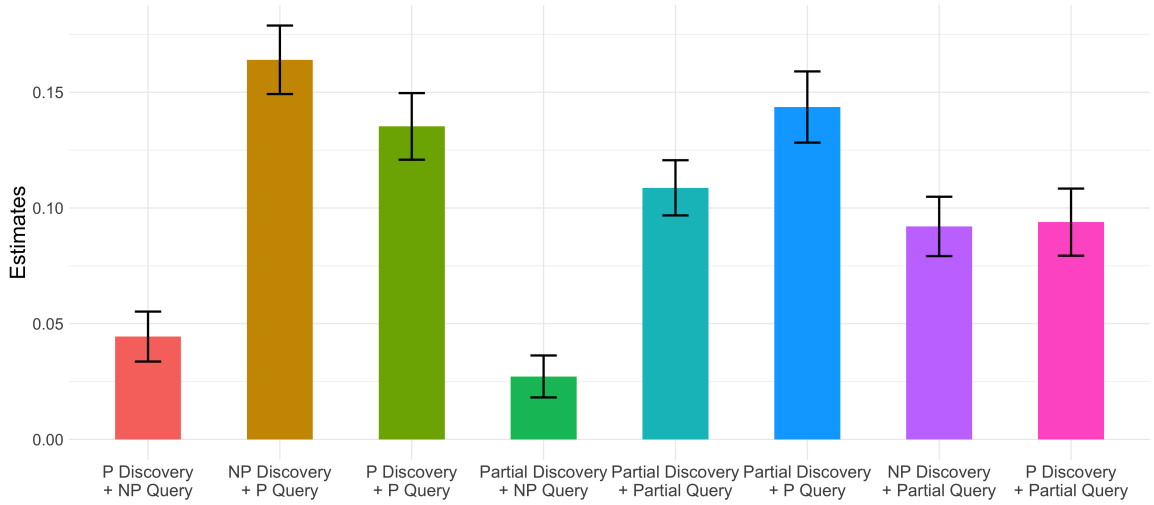
Partial personalization, which blends popularity-based ranking with individual utility, performs well despite slightly underperforming full personalization at the query stage or at both stages. This approach can be attractive when firms wish to preserve product variety for long-run performance (Chen et al., 2024), or when data and resource constraints limit the accuracy of individual-level preference inference.

1.6.2.2 Personalized Personalization

The experimental results in Section 1.4.1.2 show that personalization delivers larger gains for customers with consistent preferences, whereas the benefits attenuate and can even become negative

[38] I assume that τ_1, which captures how personalization affects the cost of querying, is invariant to the degree of personalization at the query stage.

Figure 1.5: Total Revenue of Multi-Stage Personalization

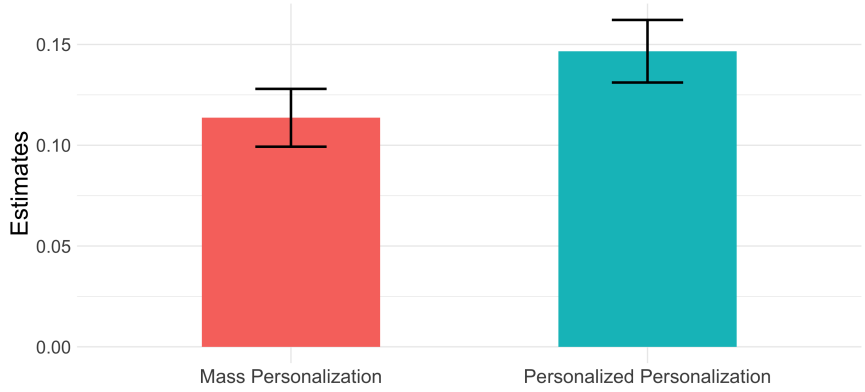


Notes. This figure reports estimates from the specification $\log(Revenue_i + 1) = \alpha_0 + \alpha_1 T_i + \epsilon_i$, where T_i denotes the personalization strategy. The omitted strategy is no personalization at either stage. The x-axis displays the personalization strategies. The y-axis reports the estimated percentage change in revenue relative to the omitted condition. Error bars represent 95% confidence intervals. "NP" denotes non-personalized and "P" denotes personalized.

for customers with more dynamic preferences. This heterogeneity suggests that a platform may improve performance by tailoring personalization according to preference consistency. To quantify the value of this strategy, I simulate an environment in which 20% of customers are randomly assigned to a "dynamic-preference" segment and 80% to a "consistent-preference" segment.[39] For customers in the dynamic-preference segment, I introduce a 10% random perturbation to their estimated preference parameters to mimic temporal shifts in preference, while customers in the consistent-preference segment experience no shocks. I then compare (i) universal personalization, which personalizes both stages for all customers, with (ii) personalized personalization, which personalizes both stages only for customers in the consistent-preference segment. As shown in Figure 1.6, personalized personalization increases total individual-level revenue by 3.3% relative to universal personalization and by 14.7% relative to applying no personalization at either stage. In practice, firms can infer preference consistency from historical behavior such as purchases. For example, a customer who repeatedly purchases low-priced or new-condition items is likely to benefit from personalization, whereas a customer whose choices frequently shift across price tiers or product attributes may be better served by a less personalized, popularity-based experience.

[39]This split best matches the empirical patterns; alternative splits such as 25%/75% yield similar results.

Figure 1.6: Revenue of Personalized Personalization Based on Preference Consistency



Notes: This figure reports estimates from the specification $\log(Revenue_i + 1) = \alpha_0 + \alpha_1 T_i + \epsilon_i$, where T_i denotes the personalization strategy. The omitted strategy is no personalization at either stage. The x-axis displays the personalization strategies. The y-axis reports the estimated percentage change in revenue relative to the omitted condition. Error bars represent 95% confidence intervals.

1.7 Conclusion

Personalization is central to the operations of many digital platforms. For instance, Spotify curates discovery feeds, Netflix tailors thumbnails, and Amazon customizes search query results. From discovery-page recommendations that capture initial engagement to query-stage rankings that showcase additional relevant products, personalization influences multiple stages of the customer journey. Although many firms make significant investments to optimize each touchpoint, this siloed approach could create tensions when gains from personalization in one stage offset benefits in another. Ignoring important spillovers between stages may lead to suboptimal outcomes. Designing effective multi-stage personalization thus requires a systematic understanding of both within-stage effects and cross-stage spillovers. This paper studies two stages where personalization is central—the firm-driven discovery stage and the customer-initiated query stage—and examines how outcomes change depending on which stage or stages are personalized.

Using two field experiments, I find that although personalizing the discovery stage can boost initial engagement, it can reduce user engagement in the subsequent query stage, ultimately leaving total revenue flat. When both stages are personalized, total revenue increases without cannibalizing discovery-stage revenue. These findings suggest a one-way negative spillover from the discovery stage to the query stage. I also find that the impact of personalization is more pronounced for customers with consistent preferences. In fact, personalizing the discovery stage can reduce revenue for those with more dynamic preferences.

To identify the revenue-maximizing multi-stage personalization strategy and quantify the value of personalized personalization, I develop and estimate a sequential search model. Counterfactual

simulations based on my model estimates show that personalizing only the query stage generates the highest total revenue, outperforming both no personalization and personalization at both stages. The model highlights that discovery-stage personalization can improve intent matching by surfacing items aligned with a customer's purchase intent, and customers use this early experience to form expectations about the benefits of continuing to the query stage. When discovery-stage personalization fails to deliver a strong preference fit, these expectations can decline, increasing early exit. Many of these customers would have found better matches had they proceeded to the query stage, where the submitted query reveals richer information about current preferences and rankings can condition on that signal to return better results. Finally, the model implies that personalization can be further improved by targeting it: offering personalization only to users with consistent preferences increases platform revenue by 3.3% relative to personalizing all customers.

Overall, this paper shows that expanding personalization to additional touchpoints does not necessarily improve aggregate performance, challenging the view that more personalization is always better. The findings caution against stage-by-stage optimization, which is common when product discovery and query are managed and evaluated separately. When cross-stage spillovers are present or the objective function is misspecified, optimizing each stage in isolation can be counterproductive; in some cases, turning off personalization at a stage improves overall outcomes. Moreover, when the platform cannot observe why a user abandoned a prior session or cannot reliably predict preference shifts, personalization based on stale signals can mis-rank products and reduce the perceived value of continuing the journey. Personalization should therefore be treated as a coordinated, multi-stage strategy embedded in the broader consumer decision process rather than as a sequence of independent interventions. A practical implication is that platforms should design objective functions and learning systems that evaluate and optimize personalization at the level of the full customer journey, instead of maximizing stage-level metrics. More broadly, this paper highlights these challenges and provides a framework for firms to decide whether, where, and how much to personalize in the customer journey.

Beyond coordinating personalization across stages, firms can improve performance by exploiting heterogeneity in responses to personalization. I illustrate this idea using preference consistency, but it is only a starting point. More broadly, firms can use behavioral histories to implement "personalized personalization" by directing personalization toward users with the highest expected gains and adjusting its intensity across users and stages.

Although this paper studies a single platform (Mercari), the setting shares key features with other e-commerce platforms such as eBay, Depop, Poshmark, and Etsy. To gauge how broadly these patterns may extend, I use cross-category variation in market environments as suggestive evidence. For instance, while many Mercari listings are single-unit, some categories have deeper inventories and many close substitutes, resembling multi-unit retail. In these larger-inventory,

more homogeneous categories, personalization effects are stronger, consistent with greater gains from improved matching when substitution opportunities are richer. In addition, Mercari primarily features pre-owned items whereas other platforms may offer a higher share of goods in new condition. I compare effects across categories that differ in the share of items shown in new (versus used) condition and find stronger effects in categories with a higher prevalence of new items.

This paper has several limitations. First, the field experiments last for less than two weeks, so the results primarily capture short-run effects of personalization. Longer-run impacts may differ if market participants adjust their behavior over time. For example, customers may learn how the platform's personalization works and adapt their behavior, and sellers may respond strategically by changing prices (Kaye, 2024) or product quality, which could in turn affect consumer welfare and long-term platform revenue. Second, I focus on two stages of the customer journey, but companies often personalize additional touchpoints such as push notifications, promotions, and checkout incentives. Studying these interventions jointly with discovery- and query-stage personalization is a natural extension. Relatedly, I use preference consistency to illustrate personalized personalization, but this is just one way to operationalize the idea. With richer data, platforms could vary personalization intensity across users, tailor it at the category level, or adapt it dynamically as preferences evolve. Third, my empirical setting is an e-commerce marketplace where users often arrive with relatively well-defined purchase intent. The implications may therefore differ in contexts with more exploratory or serendipitous consumption, such as social media platforms. Finally, the structural model is estimated using data from a single product category and all analyses are conducted on a single platform, so the generalizability of the findings should be interpreted with caution. Future research could examine multi-stage personalization over longer horizons, across platforms with different market environments and customer compositions, and under more advanced algorithms.

CHAPTER 2

How Effective Is Suggested Pricing?: Experimental Evidence from an E-Commerce Platform

2.1 Introduction

Setting prices is a complex issue that requires an understanding of market competition and consumer demand. In traditional retail settings, experienced professionals typically handle pricing decisions. However, many modern peer-to-peer (P2P) platforms face the challenge of not being able to directly set prices for products. Instead, they rely on sellers to determine prices, as seen on platforms like Airbnb and eBay. These sellers often lack the professional expertise of traditional pricing managers, leading to significant pricing frictions (Pan and Wang, 2021; Huang, 2022), which may consequently decrease platform revenue. To address this, some platforms, such as Uber, use centralized pricing, where the platform sets prices for sellers' products or services using a pricing algorithm. However, centralized pricing may not be ideal when sellers possess significant private information about their offerings (Gaineddenova, 2022; Filippas et al., 2023).

A middle-ground approach between centralized and decentralized (i.e., no pricing guidance) pricing is for the platform to provide suggested prices, allowing sellers the flexibility to either adopt these recommendations or set their own prices. Increasingly, online platforms like Steam, Airbnb, eBay, and Zillow are implementing price suggestion tools.[1] These platforms typically leverage their data on user behavior, market demand, and supply to generate price recommendations, which sellers can choose to follow. In this paper, we investigate the effectiveness of platform-suggested prices in influencing sellers' pricing decisions and downstream outcomes on a digital platform. Specifically, we ask the following research questions. First, whether and how do platform-suggested prices impact listing prices? Second, how do suggested prices affect whether items are sold, seller and platform revenue, and the number of listings by sellers?

How platform-suggested prices impact sellers' pricing decisions depends on how sellers use

[1] For example, Steam uses recommended regional pricing. See https://steamdb.info/blog/valve-price-matrix-2022-update/, accessed July 2025.

the suggested price. Suggested prices may have little effect if sellers view them as self-serving for the platform. Sellers aim to maximize their profits, while platforms seek to maximize overall (platform) revenue. Sellers also often possess private information (Dietvorst et al., 2015), leading to distrust and resistance toward platform suggestions (Jussupow et al., 2020). Conversely, listing prices can move in the *same* direction as suggested prices if sellers trust the platform's suggestions, particularly in situations with high seller uncertainty (i.e., pricing used goods), or use suggested prices as anchors. Listing prices could also move in the *opposite* direction to suggested prices if sellers undercut the suggested prices to gain a competitive edge over other sellers. Beyond pricing, suggested prices could deter sellers from the platform if they find suggested prices to be too low.

We address these questions by conducting a large-scale field experiment in collaboration with Mercari, Inc. (Mercari). Mercari is a digital marketplace targeted towards people who are looking to sell their used items.[2] Consistent with this, most sellers on Mercari are non-professionals.[3] Prior to our experiment, Mercari provided sellers with a suggested price range and an auto-filled price during the listing process. Our experiment randomly assigns sellers to one of eight experimental conditions. In the "No Suggested Price" (NoSP) condition, sellers do not see an auto-filled suggested price or range. In the "Usual Suggested Price" (UsualSP) condition, sellers see the auto-filled suggested price and range that they would have seen outside the experiment. Comparing these two conditions allows us to measure the treatment effect of showing a suggested price. The remaining six experimental conditions involve perturbations of the usual suggested price and range by predetermined amounts, ranging from −30% to +30%, which enables us to measure the causal effect of suggested price levels.

The two-sided nature of platforms complicates causal inference due to the potential violation of the stable unit treatment value assumption (SUTVA) (Imbens and Rubin, 2015), which occurs when one unit's treatment affects others' outcomes. In our setting, SUTVA is violated (i.e., there is "interference") if sellers observe others' prices, or if selling outcomes depend on prices of other items. Several methods exist to eliminate or quantify bias,[4] but our setting lacks sufficient non-overlapping markets (Holtz et al., 2024), and the treatment does not naturally extend to two-sided randomization (Li et al., 2022a). We instead address interference in our experiment using three methods. The first is via a random saturation design, where we randomly assign treatment shares to a cluster (product category), and then randomly assign sellers to an experimental condition based on their cluster's treatment shares.[5] Second, we leverage user search and click data to observe

[2]This is reflected in Mercari's messaging to its sellers on its website:"Say 'goodbye' to the things you're no longer using." https://www.mercari.com/about/, accessed July 2025.

[3]See https://www.mercari.com/us/help_center/article/407/, accessed July 2025.

[4]Examples include Holtz et al. (2024); Goli et al. (2023); Li et al. (2022a); Johari et al. (2022).

[5]See Hudgens and Halloran (2008); Tchetgen and VanderWeele (2012); Liu and Hudgens (2014); Offer-Westort and Dimmery (2021); Baird et al. (2018); Cai et al. (2022).

which items sellers view and which items enter buyers' consideration sets. Third, we measure treatment effects over time with the rationale that listings created earlier in the experiment were exposed to fewer other experimental listings and thus were less subject to interference (if any).

We find that the presence of a suggested price significantly reduces the listing prices of new sellers (i.e., those without listings prior to the experiment). Relative to receiving a suggested price, new sellers who do not receive a suggested price have 7.3% ($6.5) higher listing prices, while experienced sellers (i.e., those who have at least one preexperiment listing) who do not receive a suggested price have similar listing prices as those who receive the "usual suggested price." In addition, listing prices move in the same direction as suggested prices. On average, a 30% increase in the suggested price increases the listing price by 4.7%. We find that suggested prices are more influential for sellers with less experience (e.g., fewer past listings) and for used items, which are situations where pricing is likely more challenging. These results are consistent with sellers finding suggested prices to be informative, especially in circumstances where they are less certain about pricing. Additionally, sellers process this information in a manner consistent with the anchoring effect. A substantial proportion of sellers select the maximum suggested price regardless of its value, and those who are likely more uncertain about pricing are even more inclined to do so. We also find that the presence of suggested prices makes pricing more salient to sellers, as sellers who do not receive a suggested price are less likely to subsequently adjust their listing prices. These results together demonstrate that suggested prices play an informational role in influencing listing prices and are likely to generalize to settings where sellers are uncertain about pricing.

In addition, we find that lower suggested prices improve selling outcomes. On average, sellers who receive 30% lower suggested prices are 10% more likely to sell their items within seven days, compared to those who receive either the usual or no suggested price. Despite setting lower listing prices, sellers earn 4.2% higher revenue. We also find no impact on sellers' platform usage over the experimental period—neither the presence nor the amount of the suggested price affects the number of listings created.

A common criticism of field experiments is that their results can only be described as being representative of a "partial" equilibrium. In order to address this, we carried out a follow-up experiment, where we picked the treatment condition that maximized the revenue for sellers and the platform and rolled it out to almost the entire platform for a longer period. In our follow-up experiment, 80% of sellers were put into the -30% condition and 10% each into the UsualSP and NoSP conditions for three months (recall that our main experiment ran for 1.5 months). The results from this follow-up experiment get closer to the "full" equilibrium while also having a control group to compare against. We find very similar results. In fact, based on the results of this follow-up experiment, the platform switched all its suggested prices to the -30% condition.

The results from our study contribute to the literature on pricing on P2P marketplaces in the

following ways. First, we find that platform-suggested prices move listing prices *in the same direction* and can be used to increase revenue for the platform and for sellers. Second, at the same time, we do not find that suggested prices reduce seller usage of the platform (e.g., number of listings), a downside that has been observed when platforms use centralized pricing schemes to influence pricing (Gaineddenova, 2022; Filippas et al., 2023). These two findings imply that suggested prices are an effective tool to guide sellers' pricing while also allowing sellers the freedom to factor in private information in their pricing decisions. The outcome data for the "best" experimental treatment condition also indicate that suggested prices represent a benign platform intervention that has the potential to leave all parties (sellers, buyers, and the platform) better off. Finally, our findings suggest that the platform can use suggested prices as a lever to help new sellers increase their revenue rather than relying on them to do this via gaining pricing experience over time (for example, Huang et al., 2022). We discuss our contributions to the literature in more depth in Section 2.2.

The remainder of the paper is structured as follows. Section 2.2 situates our paper and its contributions to the existing literature. Section 2.3 presents the empirical setting and motivating evidence. Section 2.4 details the field experiment, including the experimental design and data. Section 2.5 reports the experimental findings. We test for interference bias in Section 2.6. Section 2.7 describes the follow-up experiment. We conclude in Section 2.8 with limitations and directions for future work.

2.2 Literature Review

This paper contributes to several interrelated bodies of literature. The first is the literature on pricing on online platforms. This stream of literature covers a variety of topics, including personalized pricing (Smith et al., 2023; Misra et al., 2019), pricing frictions (Huang, 2022; Pan and Wang, 2021; Williams, 2022; Aparicio and Simester, 2022), and selling mechanisms (Gaineddenova, 2022; Filippas et al., 2023; Wei and Lin, 2017). Our paper relates most closely to the latter two topics. Huang (2022) and Pan and Wang (2021) document the presence of pricing frictions on Airbnb. Huang (2022) demonstrates that platform-assisted pricing, in which the seller sets a price and the platform makes adjustments based on market conditions, can eliminate almost all price frictions and improve seller welfare over centralized pricing. Filippas et al. (2023) show that switching from decentralized to centralized pricing on a car rental platform increases seller attrition. In the context of a ride-hailing platform, Gaineddenova (2022) demonstrates that decentralizing pricing improves market efficiency over centralized pricing because centralized pricing does not account for drivers' private information. This literature highlights the benefits of decentralized pricing as opposed to centralized pricing. Foroughifar and Mehta (2023) study hosts' adoption of Airbnb's AI

pricing tool, Smart Pricing. They find that adoption is low primarily due to hosts' beliefs that their pricing can outperform Smart Pricing. The most closely related paper is Nakamura et al. (2024), which conducts a field experiment in Pakistan's used car market to study the effect of providing sellers with price information. They find that offering price recommendations influences sellers' pricing strategies, reduces advertising efforts, and generates positive spillover effects on page views. However, unlike our paper, they do not vary suggested price levels nor study effects on transactions. Our paper contributes to the literature by documenting the causal impact of not only the presence of but also the levels of platform-suggested prices on listing prices and subsequent outcomes.

The second is the literature on platform design and information disclosure. Several papers have studied how information impacts user behavior, such as information about competition on job search (Fradkin et al., 2023), number of potential matches and competition in online dating (Fong, 2024), certifications and licensing on eBay and home-services platforms (Farronato et al., 2020; Hui et al., 2023), and information about individual consumer preferences on sellers' targeted prices of video games (Wu et al., 2023). We add to this literature by focusing on how information in the form of suggested prices impacts sellers' behavior.

The final literature is on experimentation in two-sided platforms. The two-sided nature of platforms creates difficulties in designing and implementing large-scale field experiments, as well as in interpreting the results. Several studies have documented methods for reducing interference bias in online platform experimentation (e.g., Holtz and Aral, 2020; Blake and Coey, 2014). Holtz et al. (2024) use a cluster-randomized design, in which units with shared characteristics (e.g., geographic markets) are in the same treatment assignment, to reduce interference bias. Li et al. (2022a) build a simple, tractable market model to study bias and variance in platform experiments with interference. They show that the bias-optimal choice of experiment type depends on the relative amounts of supply and demand in the market, and the choice of treatment proportion induces a bias-variance tradeoff. Johari et al. (2022) propose a double randomization design where both sides of the market are randomized. Goli et al. (2023) correct interference bias in ranking algorithms through a post-experiment framework and measure the total average treatment effect using a deep learning model. Zhan et al. (2024) propose a "recommender choice model" that combines a structural choice model with neural networks to directly model the interference pathway in recommender system experimentation. In addition to using network data, our paper addresses interference by applying a random saturation experimental design, a method that has been primarily implemented in offline settings.[6] To our knowledge, this is the first application of such an experimental design in an online platform setting.

[6]Examples include studies in development economics Banerjee et al. (2021); Burke et al. (2019); Egger et al. (2022); Giné and Mansuri (2018) and voter behavior Sinclair et al. (2012); Giné and Mansuri (2018).

2.3 Institutional Background

2.3.1 Empirical Setting

The setting of this paper is Mercari, a P2P e-commerce platform. Founded in Japan in 2013, Mercari has since become the country's largest P2P online marketplace. It subsequently expanded its business to the United States (Mercari US) in 2014. As of 2023, Mercari US has an average of 5 million monthly active users, 600,000 sellers, 11 million listings per month, and 100 million app downloads.[7,8] Our collaboration, including the experiment, data, etc., is with Mercari US.

As noted earlier, Mercari is targeted towards non-professional users. This is evident from the messaging on their website, such as: "We all have things we don't use, never used, or have simply outgrown. But these treasured items still have value. Our team is always trying to find new ways to make exchanging items even easier."[9] Furthermore, the majority (60%) of Mercari listings consist of used/second-hand items, and setting accurate prices for these products can be more difficult compared to new products (Gu et al., 2023). To mitigate these pricing difficulties, Mercari US launched a price suggestion tool in 2019 to assist sellers with pricing decisions.[10]

The suggested prices are displayed to sellers during the listing creation process in the following way. The seller first provides item details, including title, description, category, brand, physical condition (new, like new, good, fair, poor), photos, size (if applicable), color and hashtags (the last two items are optional).[11] Based on this information, the platform computes a suggested price and a lower and upper bound of the suggested price. The suggested price is auto-filled as the listing price.[12] Sellers can change the auto-filled price. The platform labels the item as a "speedy sale" if the chosen listing price is less than the lower bound of the suggested price, "just right" if the listing price is within the suggested price range, and a "slow sale" if the listing price is higher than the upper bound of the suggested price. The platform also reports the seller's revenue given the chosen listing price.[13] Figure B.13 in the appendix shows an example of the listing creation process. The listing creation process requires the seller to choose a listing price, in the form of an integer, but

[7] https://about.mercari.com/en/press/news/articles/20230201_mercari10th/, accessed July 2025.

[8] See https://thesmallbusinessblog.net/mercari-users/, accessed July 2025.

[9] See https://www.mercari.com/about/ for details, accessed July 2025.

[10] Although Mercari Japan also offers a price suggestion tool, the suggested price in the US market is generated only based on information about US listings.

[11] Product categorization is highly granular. For example, *Skating Equipment* and *Hiking Equipment* are separate categories rather than being grouped under a broad label like "All Sports Equipment."

[12] The auto-filled suggested price is between the lower and upper bounds but is not always the midpoint of this interval. We do not observe the prices sellers intended to set before seeing the suggested price.

[13] Seller revenue is the listing price minus a 10% commission, a payment processing fee of 2.9% of the selling price plus $0.50, and any applicable shipping fees (if covered by the seller). Note that starting in April 2024, after our experiments concluded, Mercari shifted transaction fees to buyers, allowing sellers to list their items for free. However, this change did not impact the suggested pricing process.

the listing price can be changed at any time. The item is then sold for the listing price at the time of the transaction. Buyers can also bargain with the seller by making an offer for the item.

The suggested price and its associated range are generated by a neural network–based model that predicts selling prices using historical sales of similar products. It is trained on historical data, including the selling price and product attributes like name, description, image, brand, category, physical condition, and whether the seller or buyer pays the shipping fee.[14] Therefore, the algorithm's objective is to accurately predict selling prices of similar items using past demand. The model is typically retrained every few months to incorporate recent market environments. Note that it was not retrained during our experimental period. An item receives a suggested price if its product category supports this feature (among the 3,189 product categories on Mercari, 775 support suggested prices), if the suggested price exceeds \$5, and if the firm determines that there is sufficient training data to generate an accurate prediction.[15] Approximately 90% of items receive a suggested price.

How might suggested prices impact listing prices? We develop a stylized model of seller pricing to illustrate why sellers respond to suggested prices and how their beliefs about the pricing algorithm's objectives influence their responses. We describe the intuition here and report model details in Appendix B.2. In our model, sellers are uncertain about how quickly an item will sell at a given price. They need to decide how much weight to place on their own beliefs about time to sell versus the information provided by the platform. Suggested prices convey the platform's information about time to sell, which sellers may interpret differently depending on their beliefs about the platform's objective: (1) to provide accurate information about time to sell, (2) to maximize total platform revenue, or (3) to prioritize quick sales (i.e., heavily discounting future revenue). A seller's belief about the expected time to sell, given a suggested price, depends on their belief about the platform's objective. For example, a suggested price of \$10 implies very different selling times depending on the seller's beliefs. If the seller believes the platform is prioritizing quick sales, they will infer that \$10 is an underestimate of their profit-maximizing price. If they are more patient than the platform, they would pick a price above \$10, which could lead to a longer time to sell than the platform's estimate. The model also allows for heterogeneity in the weight placed on the platform's information. For example, sellers with more prior experience or private information may place less weight on the platform's information.

The model then generates predictions on how listing prices change with suggested prices under each objective. It shows that sellers who rely more on the platform's information, such as less

[14] We do not have full information on the algorithm and training data. We try to replicate the production function of the suggested price using observables in Appendix B.1. Overall, we find that observed item attributes can predict the suggested price fairly well.

[15] The categories that do not support price suggestions tend to have fewer sold listings, and thus have less training data to predict suggested prices and, based on preexperiment data, contribute to 11.6% of total revenue.

experienced sellers, respond more strongly to suggested prices. The extent of responses also varies with beliefs about the platform's objective. We assume that Mercari is less patient than the average seller, as the platform depends on transaction volume to sustain its operations. In contrast, most sellers on the platform are not professional retailers and do not rely on sales as their primary source of income.[16] Under the assumption that Mercari is less patient, sellers respond more strongly to suggested prices when they believe the platform provides accurate information than when they believe it aims to maximize revenue. Additionally, holding all else constant, the seller's responsiveness to suggested price becomes more similar under all three objectives as the average seller's marginal cost approaches zero. When marginal costs are near zero, the platform and seller objectives become aligned, as the platform's commission-based revenue is a constant proportion of the seller's revenue.

We also consider an extension allowing for seller competition, where the focal seller believes other sellers adopt the suggested price. In this setting, the seller can benefit from undercutting their competitors' prices. We find that if lower prices boost product visibility and the returns to visibility increase sufficiently with the suggested price, the demand gains from undercutting can outweigh the loss in per-unit profit, resulting in listing prices decreasing with suggested prices. As such, the direction of the empirical effect of suggested prices on listing prices is ambiguous.

In sum, the model illustrates the intuition behind why suggested prices can influence listing prices. It shows that sellers who are more uncertain about pricing (e.g., those who place greater weight on the platform's information relative to their own) respond more strongly to changes in suggested prices. Specifically, these sellers are more likely to increase their listing prices when the suggested price rises. While we cannot directly observe seller uncertainty, we proxy for it using seller experience and the item's physical condition in the empirical analyses. We can thus directly test this prediction. Additionally, the model highlights the role of sellers' beliefs in determining the magnitude of the response to suggested prices. Although seller beliefs are not directly observable, understanding how they influence pricing decisions can provide some generalization of our effects to other platforms. In particular, the model predicts that when sellers' marginal costs approach zero, their beliefs about the platform's objective become less relevant in determining how their listing prices respond to suggested prices. Finally, the direction of the effect of suggested prices on listing prices is an empirical question, as the model shows that the direction may not always be positive. In particular, when sellers interpret the suggested price as informative about competitors' pricing, undercutting can reverse the effect of higher suggested prices on listing prices.

[16] If the seller believes that the platform is more patient than the average seller, they respond more strongly to suggested prices under objective (1) than under objectives (2) or (3) only when the average seller's marginal cost is high. See Appendix B.2 for details.

2.3.2 Preexperiment Descriptive Statistics

In this section, we document three stylized facts about seller behavior on the platform using observational data collected prior to the experiment: (1) items that receive suggested prices have lower listing prices and are more likely to sell, (2) new sellers price significantly higher, and (3) listings by new sellers and for used items exhibit higher price variation and more price changes, suggesting higher pricing uncertainty. We focus on listings created between March 1–14, 2023, which includes 1.4 million listings by 309,868 sellers. 16.2% of these sellers are new, having posted their first listing during this period. Of these listings, 87.4% receive a suggested price. Among these, 42.3% are priced above the suggested price range, and 44.8% fall within it. 22.7% have prices equal to the maximum suggested price, while 3.5% and 2.5% have prices equal to the default and minimum suggested price, respectively.

First, we examine how receiving a price suggestion correlates with the listing price and the likelihood of a sale within seven days by estimating the following linear regression:[17]

$$y_{ij} = \alpha + \beta I(HasPriceSuggestion_{ij}) + Z'_{ij}\Gamma + \epsilon_{ij}. \tag{2.1}$$

The terms i and j denote a seller and a listing, respectively. The term $I(HasPriceSuggestion_{ij})$ is an indicator for whether listing j receives a suggested price. The matrix Z_{ij} contains the listing's attributes, including its item type, defined by the item's product category and brand, and physical condition ("New," "Like New," "Good," "Fair," or "Poor"). The dependent variables are the listing's price (logged), and whether the listing is sold within seven days. We focus on product categories that support suggested prices. Recall that even for items in categories that support suggested prices, items do not receive a suggested price if they (i) have suggested prices below $5, or (ii) lack sufficient training data. Table 2.1 displays the estimates of Equation (2.1). We find that items that receive suggested prices tend to have 5.2% lower listing prices (Column 1) and are 9.9%, or 1.2 percentage points, more likely to sell in seven days (Column 2).[18]

Second, to document the differences in listing prices and selling outcomes between new and experienced sellers, we estimate the following linear regression:

$$y_{ij} = \alpha + \beta I(NewSeller_{ij}) + Z'_{ij}\Gamma + \epsilon_{ij}, \tag{2.2}$$

where $I(NewSeller_{ij})$ is an indicator for whether listing j is seller i's first listing. We define a seller as "new" only for their first listing, hence the j subscript. The matrix Z_{ij} is the same as in

[17] Figure B.14 in the appendix plots the distribution of time until sold for all sold listings during the preexperimental period. Our results are robust to using 14 and 28 day windows, as reported in Table B.4 in the appendix.

[18] For whether the item sold within seven days, we use the mean likelihood of sale within seven days as the baseline, which is 0.119. Thus, the percentage change is $\frac{0.0118}{0.119} = 10.3\%$.

Table 2.1: Correlation between Price Suggestion and Listing Price and Likelihood of Sale

	log(Listing Price) (1)	I(Sold in 7 Days) (2)
I(Has Price Suggestion)	-0.0516	0.0118
	(0.0047)	(0.0013)
	[<0.001]	[<0.001]
Item Type × Physical Condition FE	Yes	Yes
Observations	1,407,287	1,407,287
R^2	0.5303	0.1560

Notes. This table reports the OLS estimates of Equation (2.1). The dependent variables are the item's listing price (logged) and an indicator for whether the item is sold within seven days. Standard errors in parentheses, clustered at the seller level. P-values in square brackets.

Equation (2.1), with the following exception. Because new sellers may list different types of items than experienced sellers, we also control for the item's suggested price p^s_{ij}. Thus, the data used to estimate Equation (2.2) include only items with suggested prices.

Table 2.2: New and Experienced Seller Comparison

	log(Listing Price)		I(Sold in 7 Days)	
	(1)	(2)	(3)	(4)
New Seller	0.3959	0.2622	0.0097	0.0002
	(0.0052)	(0.0047)	(0.0021)	(0.0021)
	[<0.001]	[<0.001]	[<0.001]	[0.9346]
Log(SP)		0.7698		0.0548
		(0.0022)		(0.0009)
		[<0.001]		[<0.001]
Item Type × Physical Condition FE	Yes	Yes	Yes	Yes
Observations	1,230,244	1,230,244	1,230,244	1,230,244
R^2	0.5251	0.6598	0.1587	0.1644

Notes. This table reports the OLS estimates of Equation (2.2). Standard errors in parentheses, clustered at the seller level. P-values in square brackets.

We report the estimates of Equation (2.2) in Table 2.2. Column (1) shows that new sellers price 40% higher than experienced sellers, but this difference shrinks to 26% after controlling for the item's suggested price (Column 2). This suggests that part of the price premium for new sellers comes from more valuable items. Column (3) shows that new sellers are more likely to sell their items, but after controlling for the item's suggested price (Column 4), the difference between new and experienced sellers is not statistically significant. This indicates that the boost in the probability of selling for new sellers is primarily driven by their more valuable items. These findings collectively

indicate a positive correlation between receiving a suggested price and achieving favorable listing outcomes, such as a higher probability of sale. However, as mentioned earlier, the assignment of a suggested price is not independent of these outcomes. For example, rarer items, which have less available training data, are less likely to receive a suggested price but might also have higher demand.

Table 2.3: Evidence for Pricing Uncertainty by New Sellers and Used Items

	SD(Price)		N Price Edits
	(1)	(2)	(3)
New Seller	12.35		
	(3.312)		
	[<0.001]		
New Item		-0.1660	-0.0152
		(0.0338)	(0.0019)
		[<0.001]	[<0.001]
Log(Seller's N Past Listings)			-0.0156
			(0.0005)
			[<0.001]
Category FE	Yes	Yes	Yes
New Item FE	Yes		
Standard Errors	Category	Seller	
Observations	2,669	616,842	616,842
R^2	0.6442	0.2353	0.2023

Notes. Column (1) reports the OLS regression estimates where each observation is at the category, item physical condition (new or used), and seller type (new or experienced) level. The dependent variable is the standard deviation of listing prices for all items with the same category, physical condition, and seller type. Columns (2)–(3) report the OLS regression estimates where each observation is at the listing level for all listings that have a suggested price. We add one to the number of past listings before taking the log. Standard errors in parentheses, clustered at the category level in Column (1), and at the item level in Columns (2)–(3). P-values in square brackets.

Finally, Table 2.3 provides evidence of greater pricing uncertainty by comparing price dispersion for less experienced sellers and for used items. This sample includes only items that do not have a suggested price to eliminate the effects driven by suggested prices. If the suggested prices influence listing prices, we would expect less price dispersion. In Column (1), items are grouped by product category, physical condition, and seller type, and the dependent variable is the standard deviation of listing prices in each group. The positive coefficient for New Seller indicates that items listed by new sellers exhibit higher price variance. In Columns (2) and (3), the unit of observation is the listing, and we examine within-listing price adjustments over time. The dependent variable in Column (2) is the within-listing standard deviation of listing prices.[19] The results show that new items tend to have lower price variance. In Column (3), the dependent variable is the number of

[19]Listings with no price changes have a standard deviation of zero.

times a seller edits a listing's price. Listings for new items, as well as listings posted by more experienced sellers, are associated with fewer price changes. These results demonstrate that seller experience and item physical condition are reasonable proxies for pricing uncertainty.

2.4 Field Experiment

2.4.1 Experimental Design

We design a field experiment aimed at measuring the causal effects of receiving a suggested price and of the suggested price level. We randomly assign sellers to one of eight experimental conditions. The seller's experimental condition impacts all listings that the seller creates for the duration of the experiment. In one condition, the seller does not receive a suggested price for any listings. In the other seven conditions, the minimum, maximum, and auto-filled suggested prices displayed to sellers are scaled to a percentage of their "usual" values—the suggested prices they would have seen under normal circumstances outside the experiment.[20] The percentages (one per condition) are $\{-30\%, -20\%, -10\%, 0\%, +10\%, +20\%, +30\%\}$. These conditions introduce random variation in suggested prices while ensuring that the suggested prices are close to the usual suggested price. We report the distribution of suggested prices of the top ten categories by listings using preexperiment data in Table B.5 in the appendix.

Experiments in networks are vulnerable to SUTVA violations, or interference. In our setting, interference can occur if sellers refer to the other sellers' (in other experimental conditions) listing prices before setting their own prices, thus biasing the treatment effects on listing prices. It can also arise from buyer-side behavior, as prices impact which products buyers include in their consideration set. Decreasing the price for one item can directly impact the probability of sale for similar items, thus biasing treatment effects on downstream outcomes.

Prior work addresses SUTVA violations by clustering on geographic markets (Holtz and Aral, 2020; Holtz et al., 2024) or controlling for network structure (Leung, 2020). In our setting, there are no separable geographic markets because items can be shipped nationwide. While one might consider clustering by geographic regions or countries in other settings, this approach is not applicable here since our analysis focuses only on Mercari US. Clustering by product category is another option. While this method is feasible in many e-commerce contexts, it may be underpowered for detecting differences across experimental conditions in our setting. With 775 categories and eight treatment arms, this approach would yield approximately 97 units per condition.

Given these constraints, we examine interference using three methods. First, we employ a random

[20]The "slow sale," "just right" and "speedy sale" labels described in Section 2.3.1 are analogously moved to the new range of suggested prices as well.

saturation design (e.g., Baird et al., 2018; Tchetgen and VanderWeele, 2012; Cai et al., 2022) that experimentally varies the proportion of sellers in each condition across product categories. Second, we leverage clickstream data to observe sellers' and buyers' consideration sets. Third, we analyze treatment effects over time, leveraging the idea that sellers who join earlier in the experiment are less exposed to potential interference. We first describe how interference can arise in our setting and the intuition behind the random saturation design before going into implementation details. We discuss the other methods in Section 2.6, as they do not affect the experimental design.

The listing price and revenue for a listing j can depend on the prices of other listings, as sellers can see similar listings, and buyers can compare alternatives. Therefore, item j's listing price p_j and revenue π_j can be expressed as:

$$p_j = f_p(X_j, p_j^s, p_{-j}) \tag{2.3}$$

$$\pi_j = f_\pi(X_j, p_j, p_{-j}), \tag{2.4}$$

where $-j$ denotes all similar listings, excluding j. X_j denotes the item's and seller's attributes, excluding listing price, and p_j^s is the suggested price. We assume that the suggested price does not impact X_j (i.e., p_j^s does not influence the types of items on Mercari).[21] Equations (2.3) and (2.4) illustrate the potential for interference: because p_{-j} depends on p_{-j}^s, and p_{-j}^s depends on the treatment assignments of other sellers, these assignments can indirectly affect both p_j and π_j. We refer to the effect of the seller's suggested price p_j^s on p_j and π_j as the "direct effect," and the effect of other sellers' prices on p_j and π_j as the "indirect effect."

Since sellers are unlikely to know the prices of *all* similar items on Mercari, we assume their listing prices are determined based on the price distribution of similar listings, rather than individual prices. Similarly, the seller's revenue depends on the price distribution, as the seller does not observe buyers' consideration sets. Thus, with the additional assumption that listing prices are normally distributed, we can express Equations (2.3) and (2.4) as:

$$p_j = f_p(X_j, p_j^s, \bar{p}_{-j}, \sigma_{-j}^p) \tag{2.5}$$

$$\pi_j = f_\pi(X_j, p_j, \bar{p}_{-j}, \sigma_{-j}^p), \tag{2.6}$$

where $\bar{p}_{-j}$ and σ_{-j}^p are the mean and variance of the distribution of p_{-j}. We assume that sellers have correct beliefs about $\bar{p}_{-j}$ and σ_{-j}^p. Interference remains a concern because the treatment assignments can impact the distribution of prices. For example, assigning more listings to the NoSP condition may increase price variance compared to the preexperimental period. This makes

[21] The item's price and revenue can also depend on several other attributes, like seasonality, other items' attributes, etc., but we omit them for simplicity as these attributes are not directly impacted by suggested prices.

it difficult to disentangle the direct effects of the treatment (e.g., the presence of p^s) from the indirect effects due to changes in the price distribution.

We address this problem using a two-stage group randomization framework—the random saturation design. In the first stage, we divide units (i.e., sellers) into groups and randomly assign each group a treatment saturation level—that is, the proportion of units within the group assigned to each treatment. In the second stage, we randomly assign treatments to individual units within each group according to that group's assigned saturation level. This design enables measurement of both the direct effects of treatment—through individual-level randomization—and the indirect or spillover effects—by leveraging variation in the proportion of treated units across groups.

Equations (2.5) and (2.6) show that varying the mean $\bar{p}_{-j}$ and variance σ^p_{-j} of other sellers' prices can influence the magnitude of the indirect effects. In order for there to be interference on p_j, sellers also need to be aware of such changes in the price distribution. Therefore, to implement this design, we vary $\bar{p}_{-j}$ and σ^p_{-j} by experimentally varying the proportion of sellers allocated to the NoSP condition, while keeping the proportions of sellers in the other conditions the same. We vary the NoSP proportion for several reasons. First, we expected that varying the NoSP proportion would impact mean prices because preexperimental results suggest that listings without suggested prices have higher listing prices (see Table 2.1). Second, we also expected that NoSP sellers would have higher variance in their listing prices because they have less information.[22] These shifts in price levels and dispersion drive the indirect effects shown in Equations (2.5) and (2.6). Third, we expected that sellers in the NoSP condition would be more likely to view other listings, and therefore would be more aware of changes in $\bar{p}_{-j}$ and σ^p_{-j} due to the experiment.[23] We expect the higher proportion of NoSP sellers, the greater the interference bias.

We allocate sellers to groups based on the product category of their first listing that they create or edit in the experiment. We assume that there is no interference across product categories (e.g., women's boots prices do not affect women's sneakers prices).[24] Each product category is randomly assigned to one of five "treatment saturation groups." The probability that each seller is assigned to each experimental condition depends on the product category of the first listing they create or edit during the experimental period. The following steps outline this process. First, the seller either edits an existing listing or creates a new listing for an item in category c. The category c is associated with a treatment saturation group, which dictates the likelihood that the seller is assigned to each of the eight experimental conditions. Table 2.4 reports the experimental condition allocations per

[22] This is empirically true in our experimental data: observed attributes explain less variation in listing price for NoSP sellers than sellers that receive a suggested price.

[23] Although we expected this ex ante to the experiment, we do not find statistically significant differences in the propensity to view other listings across treatment groups, as we will describe later.

[24] We refer readers to Baird et al. (2018) for a more in-depth discussion on the other necessary assumptions for the random saturation design.

treatment saturation group. For example, if category c is in treatment group G1, a seller has a 2% chance of being assigned to NoSP condition and 14% for each of the other seven conditions. The random saturation design allows us to detect interference bias by comparing treatment effects across saturation groups. Differences in outcomes by experimental condition across groups would indicate interference (Cai et al., 2022).

Table 2.4: Treatment Groups and Probability Assignment

Group	No SP	-30%	-20%	-10%	0%	+10%	+20%	+30%
G1	0.02	0.14	0.14	0.14	0.14	0.14	0.14	0.14
G2	0.09	0.13	0.13	0.13	0.13	0.13	0.13	0.13
G3	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125
G4	0.3	0.1	0.1	0.1	0.1	0.1	0.1	0.1
G5	0.65	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Notes. This table reports the probability that a seller is assigned to each of the eight experimental conditions for each treatment saturation group.

2.4.2 Data and Summary Statistics

Our field experiment ran for 6 weeks. We exclude listings priced above $500 (0.8% of all listings), and those missing the usual suggested price (0.2%) or the treatment saturation group assignment (0.86%). Details on data cleaning are in Appendix B.3. In our final dataset, 845,803 sellers initiated at least one listing, and 450,541 (53.3%) of them completed at least one listing. We do not find any treatment effects on whether the seller completes the listing (Figure B.15 in the appendix), so we focus on completed listings, where listing prices are observed. Among these sellers, 16% were new, meaning their first listing on Mercari was created during the experiment. For experienced sellers, the median and mean number of listings prior to the experiment are 57 and 404, respectively.

Table 2.5: Sample Size by Experimental Condition

Experimental Condition	N Items	N Sellers	% New Sellers
No SP	1,324,804	110,307	14.947
30% below Usual SP	540,988	48,265	16.063
20% below Usual SP	553,980	48,599	16.134
10% below Usual SP	553,842	48,295	16.024
Usual SP	555,735	48,729	15.906
10% above Usual SP	555,334	48,692	15.931
20% above Usual SP	587,740	48,877	16.042
30% above Usual SP	547,999	48,777	16.106

Table 2.5 displays the number of completed listings and sellers in each experimental condition. Figure B.3 in the appendix, which plots the assigned and realized probability that each seller is assigned to the "No Price Suggestion" group for each treatment saturation group, shows that the realized probability matches closely with the assigned probability.

For each listing, we observe its brand (if applicable), category, physical condition, listing price, shipping fee, the "usual" suggested price (i.e., the suggested price the seller would have seen in the absence of the experiment), and conditional on receiving a suggested price, the "treated" suggested price (i.e., the suggested price that the seller actually saw). We also observe usual and treated suggested price ranges, whether the item is sold, and if so, the sale date and selling price. Furthermore, we observe any adjustments to the listing price. For each platform user, we observe their search queries, the items they click on, and the corresponding timestamps. We use seller search history to assess potential interference bias in listing prices, and buyer click history to evaluate interference bias in selling outcomes (details in Section 2.6).

We check the randomization in the experiment by comparing the following seller attributes: the number of total past listings and months since the first listing. We estimate the following linear regression:

$$Y_i = \sum_{t \in T} \beta^t I_{i,t} + \alpha_{cat} + \epsilon_i, \tag{2.7}$$

where Y_i are the logged (plus one) seller's number of preexperiment listings and number of months since the seller's first listing. The variable α_{cat} is the fixed effect of the category of seller i's first listing in the experiment. We include the product category fixed effect because the likelihood that the seller is assigned to each experimental condition varies by the item's category (because of the random saturation design). Figure 2.1, which reports the estimated βs, shows no significant differences in seller attributes across experimental conditions.[25] The manipulation check is also satisfied (see Appendix B.4).

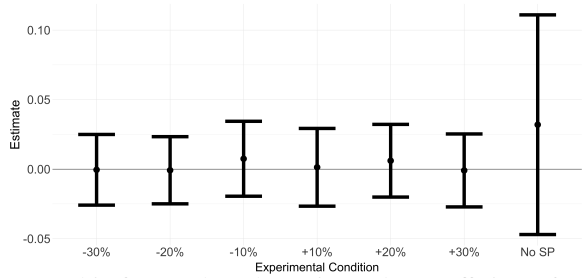
Table 2.6 displays the summary statistics of the listing prices, the "treated" suggested price, and the "usual" suggested price for all listings and the first listing created in a suggested-price-supported category during the experiment. The listing prices in our data range mainly from \$9 to \$90 and are typically higher than the usual suggested prices. The first listings in the experiment tend to have higher suggested and listing prices, suggesting that sellers lower their prices as they gain experience. To eliminate selection bias, we focus on sellers' first listings in the experiment for the remainder of the paper (unless noted otherwise).[26]

[25] In Figure B.16 in the appendix, we do not find significant differences in the likelihood of being a new seller across experimental conditions.

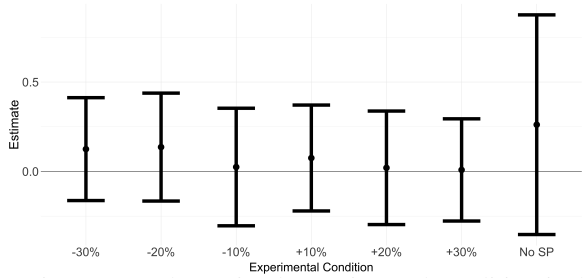
[26] Selection bias can arise because sellers self-select into creating additional listings based on their selling outcomes. For instance, a seller who quickly sells their first item might be more likely to list again and may price differently in subsequent listings. Conditioning on additional listings could therefore confound treatment effects with endogenous seller responses, leading to biased estimates.

Figure 2.1: Randomization Check

(a) Past Listings



(b) Months since First Listing



Notes. This figure plots the estimated β coefficients from Equation (2.7). The omitted experimental condition is the UsualSP condition. Standard errors are clustered at the item category level. The error bars represent 95% confidence intervals.

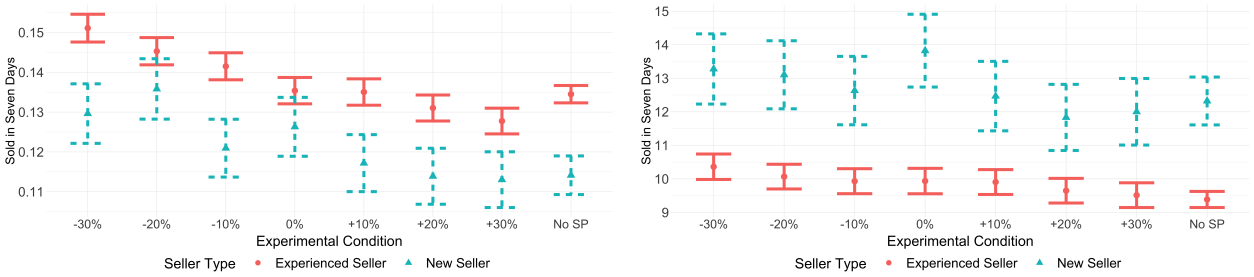
Table 2.6: Summary Statistics of Prices

	All Listings			
	10% Percentile	Median	Mean	90% Percentile
Listing Price	9	24	40.37	90
Treated Suggested Price	9	16	25.57	48
Usual Suggested Price	9	16	25.07	46
	First Listings			
	10% Percentile	Median	Mean	90% Percentile
Listing Price	12	37	71.27	180
Treated Suggested Price	10	20	39.59	89
Usual Suggested Price	10	20	38.68	86

Notes. The top panel reports summary statistics for all experimental listings, while the bottom panel focuses on sellers' first listing. The treated suggested price is the one shown to sellers in the experiment. The usual suggested price is what they would have seen outside the experiment.

Figure 2.2 reports selling outcomes for new and experienced sellers across experimental conditions. Overall, 13.5% of items are sold within seven days.[27] For items that are sold, the median time to sell is 9.7 days, with an average of 21.8 days. Although new sellers tend to list items with higher usual suggested prices, they are 1.7 percentage points less likely to sell their items within seven days compared to experienced sellers (Figure 2.2a). In addition, new sellers earn about $2 more in revenue per listing than experienced sellers, because they list more valuable items.

Figure 2.2: The Likelihood of Sale and Revenue in Seven Days by Seller Type



(a) Likelihood of Sale in Seven Days

(b) Seven-Day Revenue

Notes. This figure plots the sample mean probability that a seller's first experimental listing sells within seven days (top panel) and the mean revenue earned within the first seven days (bottom panel). The error bars represent 95% confidence intervals of the sample mean.

2.5 Experiment Results

2.5.1 Treatment Effects on Listing Price

We measure the treatment effects on listing price with the following linear regression:

$$\log(ListPrice_i) = \sum_{t \in T} \left(\beta^t I_{it} + \delta^t (I_{it} \times NewSeller_i)\right) + \gamma NewSeller_i + \alpha_{cat} + \alpha_{itemcond} + \epsilon_i, \quad (2.8)$$

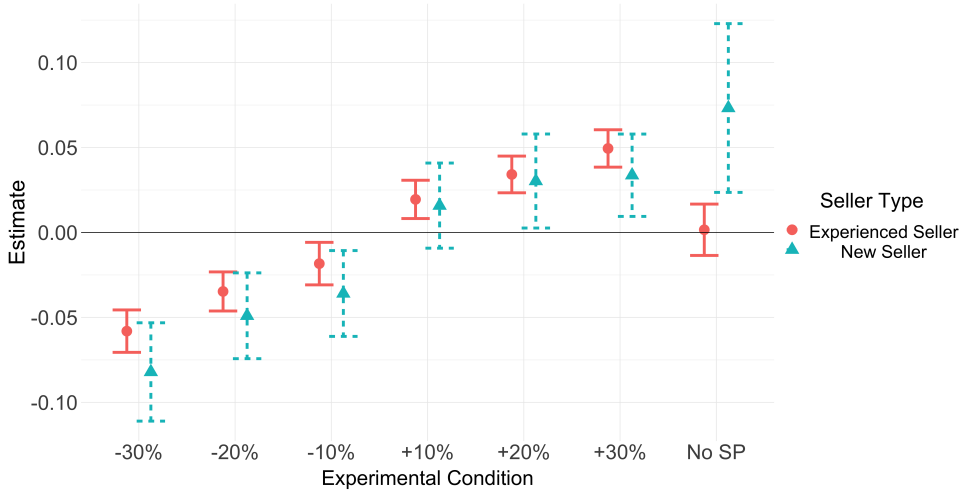
where i denotes a seller. Recall that we consider only the seller's first listing. The dependent variable is the seller's (logged) listing price. The variable I_{it} is an indicator for whether seller i is in experimental condition t. Since new and experienced sellers differ in pricing behavior, we estimate heterogeneous treatment effects by interacting I_{it} with an indicator for new sellers. The omitted experimental condition is the "Usual Suggested Price," so β^t and δ^t capture the effect of

[27] We choose a seven-day window because it aligns with the platform's revenue calculation period. Figure B.17 in the appendix plots the distribution of days until sold for all listings created during the experiment. Our results are robust to longer time windows, such as 14 and 28 days.

treatment t relative to this baseline. To improve statistical precision, we include fixed effects for the item category and physical condition. We cluster standard errors at the product category level.

Figure 2.3, which reports the treatment effects for new and experienced sellers, shows two main results. First, comparing NoSP condition with UsualSP condition, we find that the *absence* of the suggested price has a positive and significant effect on listing prices by new sellers, who price 7.3% ($6.5) higher, but has no effect on listing prices by experienced sellers. Second, listing prices move in the same direction as suggested prices. A 30% increase in suggested price increases the listing price by 3.4% and 5.0% for new and experienced sellers, respectively; a 30% decrease in suggested price decreases the listing price by 8.2% and 5.8% for new and experienced sellers, respectively.[28] Furthermore, relative to NoSP condition, sellers who receive a suggested price are 2–8 percentage points, or 7%–35%, more likely to choose a listing price within the suggested price range (Table B.6 in the appendix).[29] This finding suggests that suggested prices can anchor seller behavior, increasing the likelihood that sellers price their items within the recommended range.

Figure 2.3: Treatment Effects on Listing Price



Notes. This figure plots the estimated effects for new and experienced sellers using Equation (2.8). The dependent variable is the logged listing price. The error bars represent 95% confidence intervals. The estimates, in table format, are reported in Table B.7 Column 1 in the appendix.

[28]The difference in treatment effects is not statistically significant when comparing new sellers (i.e., sellers who have not created any listings before the experiment) to experienced sellers (i.e., sellers who have created at least one listing before the experiment). We revisit treatment effects by heterogeneity in experience in more depth in Section 2.5.1.1.

[29]As noted earlier, a seller's revenue also depends on the shipping fee, raising the potential concern that sellers might strategically adjust both listing prices and shipping fees. For instance, sellers with lower suggested prices might lower their listing price to be within the suggested range but compensate by increasing the shipping fee. We do not find evidence of this kind of behavior—there are no treatment effects on the shipping fee (see Figure B.18 in the appendix).

2.5.1.1 Heterogeneous Treatment Effects on Listing Price

Suggested prices present information that is likely to be more useful in situations characterized by greater uncertainty. At the seller level, we proxy uncertainty using seller experience: sellers with more past listings are likely to face less uncertainty about the "right" listing price. This is consistent with findings from Huang et al. (2022), who document similar patterns in the context of conventional retailers. At the item level, the newer the item, the lower the uncertainty around the "right" listing price. For example, it might be easier to choose a listing price for a new pair of Nike shoes than a slightly worn pair of Nike shoes. In this section, we quantify the heterogeneity of the treatment effect along these dimensions.

We convert the treatment condition—how much the treated suggested price differs from the usual suggested price—into a continuous variable Δ_i^s, which takes one of the following values: $\{-0.3, -0.2, -0.1, 0, +0.1, +0.2, +0.3\}$.[30] There is no analogous continuous variable for the NoSP condition, so we do not include that experimental condition in this analysis. For simplicity, we assume a linear effect of suggested price on listing price, consistent with the non-parametric treatment effects in Figure 2.3. We estimate the following linear regression:

$$Y_i = \gamma_1 \Delta_i^s + \gamma_2 (\Delta_i^s \times X_i) + \alpha_{cat} + \alpha_{itemcond} + \epsilon_i, \tag{2.9}$$

where X_i is one of the following: the logged number of past listings since account creation as a proxy for seller experience or an indicator for whether the item is in "New" condition. In all specifications where a variable is log-transformed, we add one before taking the logarithm to avoid the $\log(0)$ problem, except for price, which is never zero. All remaining variables remain the same as Equation (2.8).

Table **??** reports the estimates of Equation (2.9). These results are consistent with our conjecture. A 30% increase in suggested prices increases listing prices by 5.4% on average (Column 1), replicating the results in Figure 2.3. Importantly, suggested prices have a larger effect on listing prices for sellers who are less experienced (Column 2) and whose items are not in "New" condition (Column 3). We find a similar result when we include both variables in one regression (Column 4).[31]

We also consider other proxies for pricing uncertainty: the number of similar products, and whether the seller has sold an item in the same category prior to the experiment. A larger set of similar products may reduce uncertainty by offering more reference points for pricing. Additionally, sellers with prior sales experience in the same category are likely more knowledgeable about

[30] For example, if seller i is in the "−30% Usual Suggested Price" condition, $\Delta_i^s = -0.3$.

[31] Suggested prices have a larger effect on listing prices for sellers who create multiple listings within a short time frame (e.g., one hour or one day). These sellers also have lower listing prices relative to sellers who create a single listing. We thank an anonymous reviewer for suggesting this analysis.

demand and pricing dynamics, and therefore have less pricing uncertainty. We find that the impact of suggested prices on listing prices is indeed stronger for sellers with fewer similar products and without prior sales experience in the category, consistent with the notion that suggested prices are more influential when sellers face higher pricing uncertainty. We report the estimates in Table B.9 in the appendix.

2.5.1.2 Sellers' Pricing Strategy

In this section, we explore how sellers choose the listing price given the suggested price and its range. So far, we have shown that suggested prices are more influential when sellers are more uncertain, implying that suggested prices are informative. One mechanism through which this information is processed is by anchoring, where individuals tend to use recently encountered numbers as the input of their decision making (Tversky and Kahneman, 1974). In our setting, the auto-filled suggested price, the minimum, and the maximum of the suggested price range might serve as anchors.

First, sellers might anchor on the auto-filled price if they face high cognitive costs when choosing a listing price. However, like Foroughifar and Mehta (2023) who document low adoption of Airbnb's smart pricing algorithm, only 3.2% of sellers choose the auto-filled price. Second, we find that few (1.8%) anchor on the minimum suggested price.

The third possibility is that sellers pick the maximum of the suggested price range. We find that 13.2% sellers do so. Figure 2.4 illustrates the distribution of listing prices relative to the maximum of the treated suggested price range for each experimental condition. For NoSP condition, we use the maximum of the usual suggested price range they would have seen absent the experiment as the denominator. The figure shows that across all experimental conditions, except for NoSP condition, a notable percentage of sellers set their listing price to be the maximum of the suggested price range, regardless of the specific suggested price. This suggests that sellers tend to anchor on the maximum suggested price.

When are sellers most likely to anchor on the maximum suggested price? To explore this, we focus on sellers who are in UsualSP and NoSP conditions, where the only difference is whether the seller sees a suggested price.[32] Table 2.8 shows how the likelihood of choosing the maximum suggested price varies by seller experience and whether the item is in "New" condition. The coefficients on these interaction terms indicate that less experienced sellers and items that are not in "New" condition are more likely to use this pricing strategy. These results imply that sellers are more likely to anchor on the maximum suggested price when they are less certain about their item's value. Overall, these findings are consistent with the fact that anchors are more influential when the individual has less knowledge about the correct value (Smith et al., 2013; Smith and Windschitl,

[32]In contrast, other conditions show sellers a different suggested price than they would have seen absent the experiment. This confounds the effect of seeing a suggested price with the suggested price level.

Figure 2.4: Ratio of Listing Prices to Max Suggested Prices



Notes. This figure plots the distribution of the ratio of the listing price to the maximum of the seller's suggested price range for each seller's first item. For the NoSP condition, the maximum suggested price is the price they would have seen if they were in the UsualSP condition.

2015; Turner and Schley, 2016).

Sellers' pricing may also be affected by any value within the suggested price range, as 29.8% of sellers select a listing price in this range. Consistent with this, we still find a statistically significant positive treatment effect of the suggested price on the listing price when the listing price is not equal to the maximum suggested price (Table B.10 in the appendix). However, the treatment effect on this subset is 50% smaller in magnitude than on the entire sample, implying that a large portion of the effects of the suggested price are driven by anchoring on the maximum suggested price.

Further evidence that suggested prices act as anchors is that receiving a suggested price decreases the use of round listing prices by 6.6 percentage points, or 18.1%, on average (Table 2.9 Column 2). Without an anchor, sellers tend to default to round numbers, which are more cognitively accessible (Dehaene and Mehler, 1992) and may signal inexperience (Mason et al., 2013). Sellers with less experience are more likely to choose a round listing price, as shown by the negative effect of the number of past listings in Column (3). This tendency diminishes when they receive a suggested price, again suggesting that information is more influential for less experienced sellers.[33]

[33]Backus et al. (2019) find that on eBay, sellers use round listing prices to signal weaker bargaining power. While this may occur in our context, our findings suggest that some portion of sellers choose round numbers due to the lack

Table 2.8: Treatment Effects on Selecting the Maximum Suggested Price

	I(Picked Max SP)
Usual SP	0.1276
	(0.0055)
	[< 0.001]
log(Past Listings)	0.0013
	(0.0003)
	[< 0.001]
New Item) × Usual SP	-0.0275
	(0.0045)
	[< 0.001]
log(Past Listings)× Usual SP	-0.0029
	(0.0008)
	[< 0.001]
Category FE	Yes
Condition FE	Yes
Observations	159,033
R^2	0.0514

Notes. The dependent variable is an indicator for whether the item's listing price equals the maximum of the suggested price range. Independent variables are indicators for the experimental condition and their interactions with the number of the seller's previous listings and whether the item is in "New" condition. The sample of this analysis is restricted to sellers who are in UsualSP and NoSP experimental conditions. The baseline experimental condition is the NoSP. Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Finally, suggested prices make pricing more salient. New and experienced sellers are significantly less likely to subsequently adjust their listing prices if they do not receive a suggested price (Figure B.19 in the appendix).[34]

2.5.2 Treatment Effects on Selling Outcomes

We next examine whether suggested prices impact demand and seller revenue. We estimate Equation (2.8) with two dependent variables: an indicator for whether the item is sold within seven days, and the logged item-level revenue, which is the selling price of the item if sold within seven days and zero otherwise.[35] Recall that items can be sold at their listing price or through bargaining. We do not distinguish between these two purchase methods because we do not find any treatment

of information—if round prices were strategically chosen to convey information, it is unlikely that providing price information would affect the use of round numbers.

[34]We also find that sellers who receive higher suggested prices are more likely to adjust their listing prices. One possible explanation is that these sellers realize their listings are priced relatively high, which may reduce the likelihood of sale.

[35]We use a seven-day window to standardize the selling window. Our effects are robust to using alternative time windows, such as 14 days and 28 days. We report the treatment effects on selling outcomes for these time windows in Figure B.20 in the appendix.

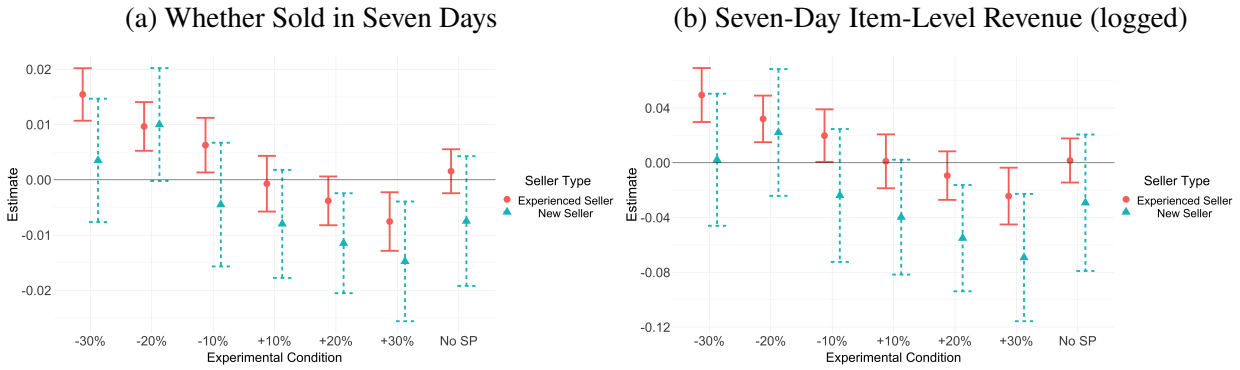
Table 2.9: Treatment Effects on Whether Listing Price Ends in 0

	List Price Ends in 0		
	(1)	(2)	(3)
ReceiveSP	-0.0602	-0.0660	-0.0882
	(0.0033)	(0.0031)	(0.0088)
	[< 0.001]	[< 0.001]	[< 0.001]
log(N Past Listings)			-0.0106
			(0.0017)
			[< 0.001]
ReceiveSP × log(N Past Listings)			0.0058
			(0.0017)
			[< 0.001]
Constant	0.4099		
	(0.0077)		
	[< 0.001]		
Category FE		Yes	Yes
Physical Condition FE		Yes	Yes
Observations	450,541	450,541	450,541
R^2	0.0029	0.0271	0.0281

Notes. Columns (1)–(2) report OLS estimates from $y_i = \beta ReceiveSP_i + \alpha_{cat} + \alpha_{itemcond} + \epsilon_i$, where the outcome is whether seller i's first experimental listing ends in zero. $ReceiveSP_i$ is an indicator for whether i is in an experimental condition that receives a suggested price. Column (3) adds an interaction with the i's number of preexperiment listings (log + 1). Standard errors in parentheses, clustered at the category level. P-values in square brackets.

effects on whether the item is sold via bargaining, conditional on being sold (see Figure B.21 in the appendix).

Figure 2.5: Treatment Effects on Selling Outcomes



Notes. This figure plots the treatment effect on selling outcomes for new and experienced sellers. All effects are relative to the UsualSP condition. Error bars represent 95% confidence intervals. We report the estimates in Table B.7 Columns (2) and (3) in the appendix.

Figure 2.5a shows that lower suggested prices increase the likelihood of sale for experienced sellers. For these sellers, a 30% reduction in the suggested price leads to a 1.6 percentage point (11.3%) increase in the likelihood of sale in seven days. For new sellers, we find that higher-than-usual suggested prices significantly decrease the likelihood of sale, but the effect is not statistically significant for lower suggested prices. The insignificance may be due to large confidence intervals, resulting from relatively fewer observations for new sellers. The linear trends in the treatment effects are similar in magnitude between new sellers and experienced sellers, as reported in Table B.11 in the appendix. Figure 2.5b displays the treatment effects on the item-level revenue.[36] Despite having lower listing prices, items that have lower suggested prices generate more revenue, resulting from the boost in the probability of the item being sold. For experienced sellers, a 30% reduction in the suggested price generates about 5% more revenue relative to UsualSP condition. For new sellers, the treatment effects of "+20%" and "+30%" conditions (relative to UsualSP condition) are statistically significant, indicating that higher suggested prices decrease revenue. In sum, the treatment effects on revenue show that lower suggested prices can increase both platform and seller revenue, which occurs because the higher likelihood of selling outweighs the lower selling price.

Given these treatment effects, we can compare platform revenues under decentralized pricing (i.e., no pricing guidance from the platform) and suggested pricing. We use the outcomes in the NoSP condition to inform the revenues under decentralized pricing. For the same set of products,

[36]Recall that we define revenue to be the final transaction price. However, the seller's realized revenue may be less, as this can include the shipping fee, which might be paid by the seller. The treatment effects are robust to excluding shipping fees borne by the seller (Figure B.22 in the appendix).

we estimate counterfactual listing prices and calculate average revenue per item between the NoSP and all other experimental conditions using an instrumental variable approach (details in Appendix B.5). This exercise requires some assumptions. First, the pricing schemes do not differentially affect attrition. Our experiment results support this assumption, as we find no differences in treatment effects on listing completion rates (Figure B.15 in the appendix). Second, suggested prices impact selling outcomes only through listing prices. Third, suggested prices monotonically impact listing prices. Given that we use an instrumental variable approach, we can only estimate how the pricing schemes impact compliers, those whose listing prices are changed by the suggested price.

We find that suggested prices help sellers generate more revenue (except in the highest price suggestion treatment conditions) relative to decentralized pricing. The average revenue per item for sellers is higher (by 0.5% to 11.7%) across all conditions. For new sellers, the average revenue per item is higher (by 0.1% to 5.1%) for all conditions except for the "+20%" and "+30%" conditions. We also compare average revenue per item under centralized pricing by assuming that all sellers adopt the suggested price and no change in seller attrition. Under this assumption, centralized pricing generates nearly twice the revenue of decentralized pricing.

2.5.3 Seller-Level Outcomes

Our analysis so far has focused only on the first listing. However, seller-level outcomes are also of interest because they can provide additional insights, such as whether suggested prices lead the seller (and the platform) to earn more revenue overall or list more items.

We estimate the treatment effects on multiple seller-level outcomes using the following regression:

$$Y_i = \sum_{t \in T} \alpha_t I_{i,t} + \alpha_{cat} + \epsilon_i, \tag{2.10}$$

where $I_{i,t}$ is an indicator for whether seller i is in experimental condition t. We report the effects relative to UsualSP condition. The dependent variable Y_i is one of several outcomes: the logged average listing price of i's items, the proportion of i's items that sell within seven days, the logged total revenue that i earns over all their items, or the number of listings that i creates. These outcomes include all items that seller i creates during the experiment. We include category fixed effects α_{cat} based on the category in which seller i has the most listings to improve the precision of our estimates.

Table 2.10 reports the estimates of Equation (2.10). The treatment effects on average listing price, the proportion of items that sell within seven days, and seven-day revenue (in Columns 1–3, respectively) are similar to the item-level effects. A 30% decrease in suggested prices reduces average listing prices by 5.9%, increases the likelihood of sale in seven days by 9.1% (1.1 percentage point), and increases the seller's total revenue by 7% relative to receiving the usual suggested price.

The last column displays the treatment effects on the (logged) number of listings sellers create. On one hand, lower suggested prices may increase revenue per item, which would encourage sellers to list more. On the other hand, lower suggested prices could reduce the number of listings by signaling that the items are less valuable. We do not find evidence supporting either of these effects. This null result may reflect the nature of the platform: sellers are often casual users looking to declutter and may be less strategic in deciding what to sell. Importantly, this demonstrates that suggested prices do not impact sellers' platform usage.[37]

Table 2.10: Seller-Level Treatment Effects

	Avg Listing Price (1)	Prop. Sold (2)	Total Revenue (3)	N Listings (4)
Condition = −30%	-0.0589	0.0108	0.0695	-0.0004
	(0.0056)	(0.0015)	(0.0139)	(0.0074)
	[< 0.001]	[< 0.001]	[< 0.001]	[0.9520]
Condition = −20%	-0.0355	0.0101	0.0367	-0.0034
	(0.0054)	(0.0014)	(0.0124)	(0.0077)
	[< 0.001]	[< 0.001]	[0.0033]	[0.6626]
Condition = −10%	-0.0236	0.0030	0.0077	0.0003
	(0.0048)	(0.0015)	(0.0145)	(0.0072)
	[< 0.001]	[0.0385]	[0.5956]	[0.9710]
Condition = +10%	0.0133	-0.0027	-0.0268	-0.0010
	(0.0051)	(0.0016)	(0.0158)	(0.0075)
	[0.0096]	[0.0802]	[0.0900]	[0.8953]
Condition = +20%	0.0268	-0.0062	-0.0368	0.0087
	(0.0048)	(0.0014)	(0.0134)	(0.0072)
	[< 0.001]	[< 0.001]	[0.0063]	[0.2270]
Condition = +30%	0.0432	-0.0088	-0.0748	-0.0016
	(0.0053)	(0.0016)	(0.0150)	(0.0058)
	[< 0.001]	[< 0.001]	[< 0.001]	[0.7841]
Condition = NoSP	0.0099	-4.46×10^{-5}	-0.0248	0.0030
	(0.0093)	(0.0014)	(0.0166)	(0.0112)
	[0.2871]	[0.9753]	[0.1343]	[0.7917]
DV Logged	Yes		Yes	Yes
Category FE	Yes	Yes	Yes	Yes
Observations	450,541	450,541	450,541	450,541
R^2	0.2775	0.0504	0.0552	0.1059

Notes. This table reports the OLS estimates of Equation (2.10). The baseline is UsualSP condition. The dependent variables are all derived from the listings created during the experiment. The dependent variables are $\log(y)$ in Columns (1) (the number of listings created during the experiment is at minimum one), and $\log(y+1)$ in Columns (3) and (4). Standard errors in parentheses, clustered at the seller level. P-values in square brackets.

[37]Treatment effects for new sellers, reported in Table B.12 in the appendix, are similar in magnitude and direction but are noisier due to fewer observations. One notable difference is that, like the item-level effects, not receiving a suggested price reduces listing prices by 3.5% for new sellers, relative to receiving the usual suggested price.

2.6 Testing for Interference

Random saturation design. We test for potential interference bias in treatment effects on the item's listing price and revenue using the random saturation design. First, we conduct the test proposed by Cai et al. (2022): in a linear regression framework $Y_i = \alpha_i + \gamma_i \rho_i + \epsilon_i$, where ρ_i is the proportion of treated units in i's neighborhood, if $\gamma = 0$, then SUTVA holds. That is, there is no interference if control units' outcomes do not vary based on the proportion of treated neighbors. Extending this to our setting, we estimate the following linear regression for sellers in our control group—UsualSP condition:

$$y_i = \alpha_{itemcond} + \beta PropNoSP_i + \gamma log(SP_i) + \epsilon_i, \tag{2.11}$$

where $PropNoSP_i$ is the proportion of sellers in NoSP condition, determined by seller i's treatment saturation group.[38] The dependent variables are the logged listing price and the logged seven-day revenue per item. We control for $log(SP_i)$, the logged (usual) suggested price for seller i's listing, to explain more variation in y_i, thereby reducing noise in our estimates. Recall that the usual suggested price is the suggested price that the seller would have seen outside of the experiment and so is unaffected by our treatment. We include item physical condition fixed effects but, unlike previous specifications, omit category fixed effects because the product's category is perfectly collinear with its treatment saturation group. An estimate of $\beta \neq 0$ would imply the violation of SUTVA.

We report both linear and nonlinear specifications of $PropNoSP_i$ in Table 2.11. In Columns (1) and (3), the baseline is $PropNoSP = 0.65$, so the coefficients are interpreted as the difference relative to $PropNoSP = 0.65$. All nonlinear effects on listing price (Column 1) are not statistically significantly different from zero, and the linear effect (Column 2) is small in magnitude, in addition to being statistically insignificant. The effects on revenue are also not statistically significant, except for the effect of $PropNoSP = 0.09$ (Column 3). Recall that we had expected that there is greater potential for interference bias as the proportion of NoSP sellers increases; these estimates do not show a clear or consistent pattern in how changes in $PropNoSP$ impact revenue. Therefore, based on the test proposed by Cai et al. (2022), we find no evidence of indirect effects on listing prices. Moreover, the evidence for indirect effects on revenue is inconclusive, as the effect for one group out of four is statistically different from the baseline.

A key limitation of this analysis is the inherent difficulty of demonstrating that an effect is truly zero, as the 95% confidence intervals are wide and thus may include potentially large effects (relative to our estimated treatment effects). Therefore, we add another layer of analysis by following the approach similar to Filmer et al. (2023), Burke et al. (2019), and Egger et al. (2022) to measure the

[38] $PropNoSP_i$ can directly translate to the proportion of treated sellers because the proportions of sellers allocated to the other treatment conditions are symmetric.

Table 2.11: Listing Prices by Treatment Saturation: Control Sellers

	log(ListingPrice)		log(Seven-Day Revenue)	
	(1)	(2)	(3)	(4)
PropNoSP = 0.02	-0.0110		0.0659	
	(0.0274)		(0.0402)	
	[0.6875]		[0.1019]	
PropNoSP = 0.09	-0.0236		-0.1078	
	(0.0306)		(0.0461)	
	[0.4404]		[0.0197]	
PropNoSP = 0.125	-0.0319		-0.0192	
	(0.0291)		(0.0384)	
	[0.2737]		[0.6176]	
PropNoSP = 0.3	-0.0493		0.0412	
	(0.0368)		(0.0430)	
	[0.1812]		[0.3383]	
PropNoSP		0.0044		0.0285
		(0.0460)		(0.0703)
		[0.9230]		[0.6854]
log(SP)	0.8731	0.8734	0.2965	0.2922
	(0.0114)	(0.0113)	(0.0321)	(0.0362)
	[< 0.001]	[< 0.001]	[< 0.001]	[< 0.001]
Condition FE	Yes	Yes	Yes	Yes
Observations	48,729	48,729	48,729	48,729
R^2	0.5172	0.5170	0.0352	0.0329

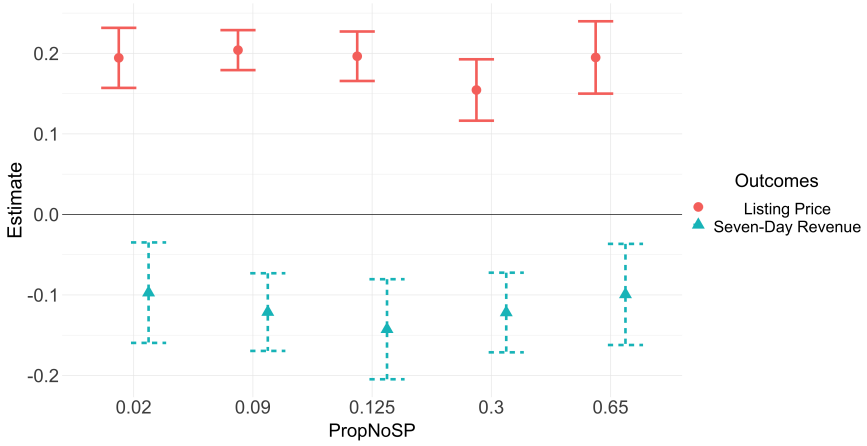
Notes. This table reports the OLS estimates of Equation (2.11). Columns 1 and 3 report PropNoSP as dummy variables, where the baseline is $PropNoSP = 0.65$. In Columns 2 and 4, PropNoSP is expressed linearly. The sample for these estimates includes only sellers in the Usual Suggested Price condition. Standard errors in parentheses, clustered at the category level. P-values in square brackets.

extent to which treatment proportions impact our treatment effects. We measure the heterogeneity of our treatment effects with respect to the proportion of NoSP sellers. This analysis differs from that of Cai et al. (2022), which measures indirect treatment effects using only the control group as it, by construction, should not experience the direct effect(s). In contrast, this additional test quantifies treatment effects in the presence of potential indirect effects, allowing us to assess the robustness of our conclusions to such interference. To this end, we estimate the following linear regression:

$$y_i = \alpha_{itemcond} + \sum_{j=\{0.02,0.09,..\}} \Big(\beta_j(\Delta_i^s \times 1\{PropNoSP_i = j\}) + \delta_j 1\{PropNoSP_i = j\}\Big) + \gamma \log(SP_i) + \epsilon_i. \quad (2.12)$$

As in Equation (2.9), Δ_i^s is the continuous deviation in suggested price, ranging from -0.3 to $+0.3$, for seller i. The coefficient β_j is the effect of the suggested price for the treatment saturation group with the proportion of NoSP sellers equal to j. Figure 2.6 plots the estimates of β_j for all treatment saturation groups. For both outcomes (logged listing price and revenue), we do not find evidence of interference, as the effect of the suggested price does not vary significantly across groups. We cannot rule out the possibility, however, that our design may be underpowered and hence unable to detect a bias. Nevertheless, our core results remain robust across various treatment proportions: lower suggested prices decrease listing prices and increase revenue. Accounting for potential interference by considering the full range of 95% confidence intervals across groups, the estimated effect of the suggested price on logged listing price ranges from 0.12 to 0.24, and on logged revenue from –0.20 to –0.035.

Figure 2.6: Treatment Effects of Suggested Price Levels by Saturation Group

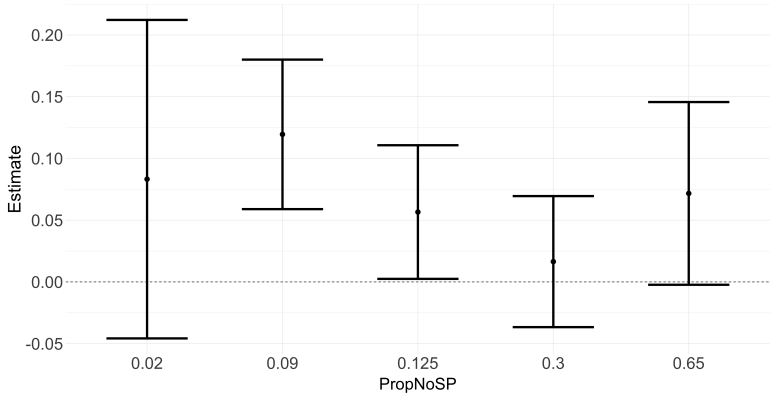


Notes. This figure presents the estimates of γ_j in Equation (2.12). The dependent variables are logged. The estimates are reported in Table B.13 in the appendix. Error bars represent 95% confidence intervals. The standard errors used to generate the confidence intervals are clustered by category.

We also measure the heterogeneity in the treatment effect of no suggested prices, relative to the

usual suggested prices, for new sellers. Recall that in Section 2.5.1.1 we find that not receiving a suggested price leads new sellers to set higher listing prices. We examine whether the significant effect observed remains robust to potential indirect effects. We estimate a linear regression similar to Equation (2.12), but instead of Δ_{it}^{s}, we include $NoSP_i$, an indicator for whether the seller is in the NoSP condition. Because we are interested in only the effect of no suggested price, we include only sellers in the NoSP and UsualSP conditions. Figure 2.7 plots the effects of no suggested price for each treatment saturation group. Although all coefficients are positive, only two are statistically significant at the 5% level. These results provide suggestive evidence that removing suggested prices increases listing prices among new sellers. However, the limited statistical power, due to fewer observations for new sellers relative to experienced sellers, prevents us from concluding that this effect holds consistently across all saturation levels.

Figure 2.7: Treatment Effect of No Suggested Price by Saturation Group: New Sellers



Notes. This figure presents the treatment effect of no suggested prices relative to the usual suggested prices. The dependent variable is the listing price (logged). Error bars represent 95% confidence intervals. The standard errors used to generate the confidence intervals are clustered by category. Table B.14 in the appendix reports the coefficient estimates.

Sellers who do not search. Interference in listing price treatment effects may arise if sellers observe other sellers' prices, primarily through search. To mitigate this concern, we restrict the analysis to sellers who did not view other listings within 24 hours before completing their own listing. One concern with restricting the sample is that it could introduce selection bias if treatment affects clicking behavior. However, we find no evidence that treatment influences whether sellers—new or experienced—click on other listings.[39] Moreover, clicking on other items before listing is uncommon: 91% of sellers did not click on items in the same category, and 75.8% did not click on any items at all in the 24 hours before listing. Treatment effects on listing price for this

[39]See Figure B.5 in the appendix. All coefficients are null except for a single significant estimate for experienced sellers under the 10% increase condition, which we attribute to multiple comparisons.

restricted sample are similar in magnitude to those in the full sample (Figure B.6 in the appendix) and are robust to longer pre-listing windows (48 and 72 hours).

Because sellers may observe prices through search (or browsing) even without clicking, we further examine search behavior. First, we do not find significant differences in sellers' tendency to search for products in the same category before listing, as shown in Figure B.7 in the appendix. Second, as a robustness check, we examine treatment effects for sellers who did not search within their listing category.[40] The treatment effects, reported in Figure B.8 in the appendix, are similar to the main results.

Buyers' consideration sets. Interference can occur when an item's price affects the outcomes of other items in different treatment groups (e.g., by affecting whether the item is sold). Such interference will bias the estimated treatment effects on revenue. To detect this, we test whether treatment effects vary with the treatment status of similar items (i.e., items in the same consideration set). We expect interference to be more likely when items are more substitutable.

We construct the consideration set for each item using buyers' clickstream data. We assume that items clicked within the same search session belong to the same consideration set (e.g., Ursu, 2018b). For each item, we compute the share of other items in its consideration sets that fall into each experimental condition.[41] We then average these shares across all consideration sets containing the item. In our sample, 46% of items co-occur with at least one other experimental item. Table B.3 in the appendix reports the average co-occurrence probabilities across experimental conditions.

To test whether treatment effects vary depending on the treatment conditions of other items in the focal item's consideration set, we estimate the following regression:

$$\log(Revenue_i + 1) = \sum_{t \in T} \beta_1^t I_{i,t} + \sum_{t \in T} \Big(\sum_{\tau \in T} \beta_2^{t,\tau} I_{i,t} \times Prop_i^\tau \Big) + \sum_{\tau \in T} \delta_\tau Prop_i^\tau + \alpha_{cat} + \alpha_{itemcond} + \epsilon_i. \tag{2.13}$$

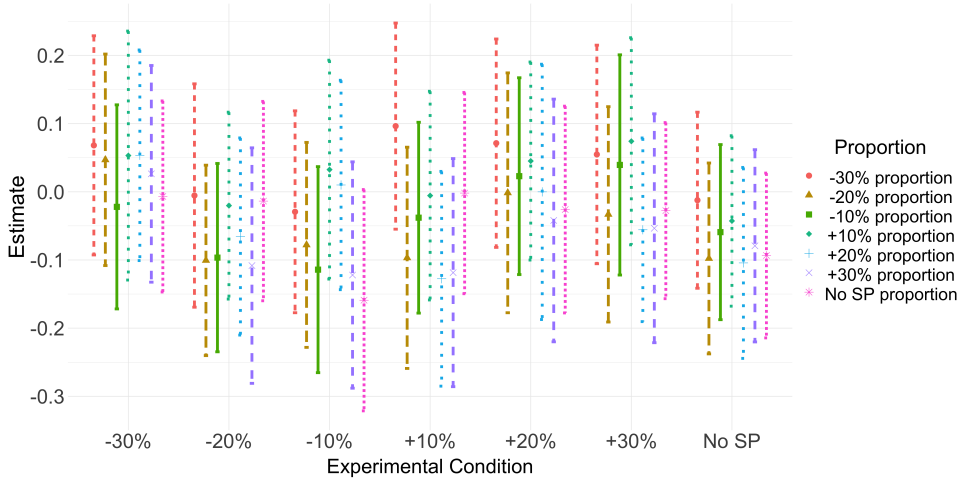
The term $Prop_i^\tau$ is the proportion of items in seller i's consideration set that belong to the experimental condition τ. The term T represents the set of all conditions. All other variables are defined as in Equation (2.8). The coefficients of interest, $\beta_2^{t,\tau}$, capture whether the treatment effect of t varies with exposure to items in condition τ; $\beta_2^{t,\tau} = 0$ implies no such variation. In order for the estimates of $\beta_2^{t,\tau}$ to be unbiased, item j's experimental condition must not impact the proportion of items in

[40] We define a seller as not searching within the same category if none of the first twelve items in the search results (the initial screen) belong to the same listing category. This excludes 13.23% of sellers.

[41] For example, suppose a user views item j alongside four others: one that is not in the experiment (e.g., the item was created before the experiment started), one in the NoSP condition, and two in the "+30% Suggested Price" condition. Then j's consideration set consists of 25% "Usual Suggested Price" items, 25% "No Suggested Price" items, and 50% "+30% Suggested Price" items, and 0% for the remaining conditions.

other experiment conditions in j's consideration set. For example, items in the "+30% Suggested Price" condition should have, on average, the same proportion of items from NoSP condition in their consideration set as those in UsualSP condition. Table B.3 in the appendix supports this assumption, showing similar co-occurrence proportions across conditions. Figure 2.8 reports the estimates of $\beta_2^{t,\tau}$. The estimates are not statistically different from zero, indicating little evidence of interference. We find similar results for whether the item is sold within seven days (Figure B.9 in the appendix).

Figure 2.8: Heterogeneity in Treatment Effects on Revenue by Other Items' Experimental Conditions



Notes. This figure plots the OLS estimates of $\beta_2^{t,\tau}$ of Equation (2.13). The dependent variable is seven-day revenue (logged). All effects are relative to UsualSP condition. Error bars represent 95% confidence intervals.

Treatment effects over time. Finally, we assess interference by examining whether treatment effects vary over time. Sellers listing early in the experiment are, *ceteris paribus*, less likely to be exposed to other treated sellers than those listing later. If interference exists, treatment effects should therefore change over the course of the experiment. We find no such evidence: treatment effects on listing prices and revenue in the first week are similar in magnitude and statistically indistinguishable from those in the last week (see Figures B.10, B.11, and B.12 in the appendix).

In sum, across all interference checks, our qualitative conclusions remain unchanged: lower suggested prices decrease listing prices and increase revenue. These results boost our confidence that interference bias in our setting is not substantial enough to affect our qualitative conclusions regarding the impact of suggested prices on listing prices.

2.7 Follow-Up Experiment: Getting Closer to the Full Equilibrium

Our results indicate that a lower suggested price—30% below the usual suggested price, in particular—increases both item-level and seller-level revenue, relative to the usual or higher suggested prices. However, these effects might disappear when the entire platform shifts to a lower suggested price, as such lower-priced items might not stand out in a buyer's consideration set if *all* items reduce their prices.

To investigate whether a lower suggested price continues to generate more revenue for the seller and the platform if the entire platform transitions to lower price suggestions, we ran a second field experiment. This experiment has three experimental conditions: (1) NoSP, (2) UsualSP, and (3) "−30% Suggested Price." We randomly assign 80% of eligible sellers to the "−30% Suggested Price" condition, 10% of sellers to UsualSP, and the remaining 10% of sellers to NoSP condition. The "−30%" treatment is applied to 80% of sellers, as opposed to 100% of sellers, to retain a control group to estimate treatment effects against.

This second experiment ran for 13 weeks, from December 8, 2023 to March 8, 2024. During the experiment, 834,571 sellers created a total of 12,779,673 listings. We use Equation (2.8) to estimate treatment effects on listing prices and selling outcomes. As before, we focus on the first listing that each seller creates during this follow-up experiment, but our findings are robust to including all listings.

Figures 2.9a, 2.9b and 2.9c report the effects on logged listing price, whether the item is sold within seven days, and the logged seven-day revenue, respectively. We find that a 30% decrease in the usual suggested price results in a 6.5% and 5.8% decrease in the listing price for experienced sellers and new sellers respectively. The difference between these two types of sellers is not significant. A 30% decrease in the usual suggested price increases the probability of the item selling within seven days by 1.1 percentage points (7.3%) for experienced sellers and 0.7 percentage points (5.3%) for new sellers, and increases revenue by 3.4% and 1.8% for experienced sellers and new sellers respectively.

Receiving no suggested price increases listing prices by 4.7% for new sellers. Consequently, new sellers earn less revenue (significant at the 10% level). Similar to the results from our main experiment, the absence of suggested prices does not influence experienced sellers.

In addition, I examined the heterogeneity in treatment effects in the second experiment by comparing outcomes for new and experienced sellers. I estimate the following regression:

$$Y_i = \alpha + \beta T_i + \lambda_{\text{cat}} + \lambda_{\text{cond}} + \epsilon_i \tag{2.14}$$

separately for new and experienced sellers, where I consider listing price (logged) and seven-day revenue (logged plus one) as the dependent variables Y_i. T_i denotes the treatment indicator, and λ_{cat}

Figure 2.9: Treatment Effects in the Second Experiment



Notes. This figure plots the treatment effect for new and experienced sellers in the second experiment. The dependent variables are the logged listing price, an indicator for whether the item sold within seven days, and the logged item-level seven-day revenue, respectively. The error bars represent 95% confidence intervals.

and λ_{cond} are category and item condition fixed effects, respectively.

As shown in Figure 2.10, a lower suggested price decreases listing prices and increases revenue for both new and experienced sellers. However, the mere presence of a suggested price affects listing prices and revenue only for experienced sellers; it does not have a statistically significant effect on revenue for new sellers. This lack of significance could due to limited sample size of new sellers.

To further unpack this heterogeneity, I divide the sample into equally sized groups based on the total number of past listings. The results, displayed in Figure 2.11, show that a lower suggested price consistently reduces listing prices and increases revenue across all experience levels, with the effects being larger for less experienced sellers. The presence of a suggested price, by contrast, has no statistically significant effect on revenue on average, with the exception of sellers in the middle range of experience.

Figure 2.10: Exp 2: Heterogeneous Treatment Effects by New and Experienced Sellers

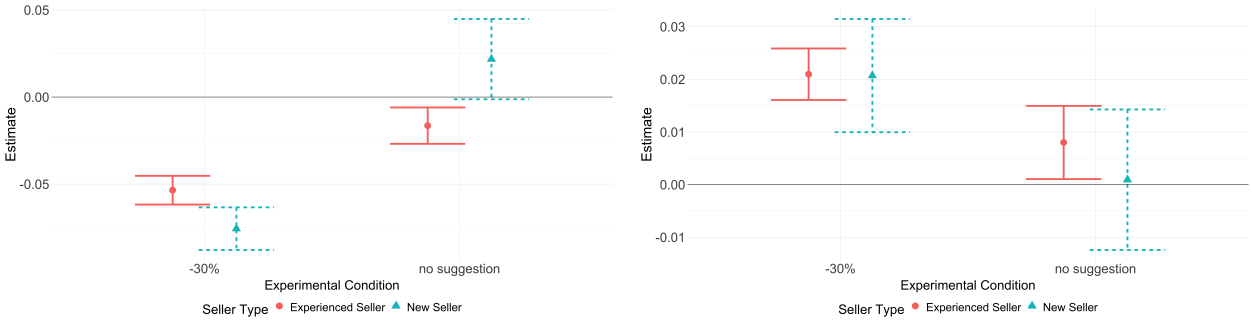
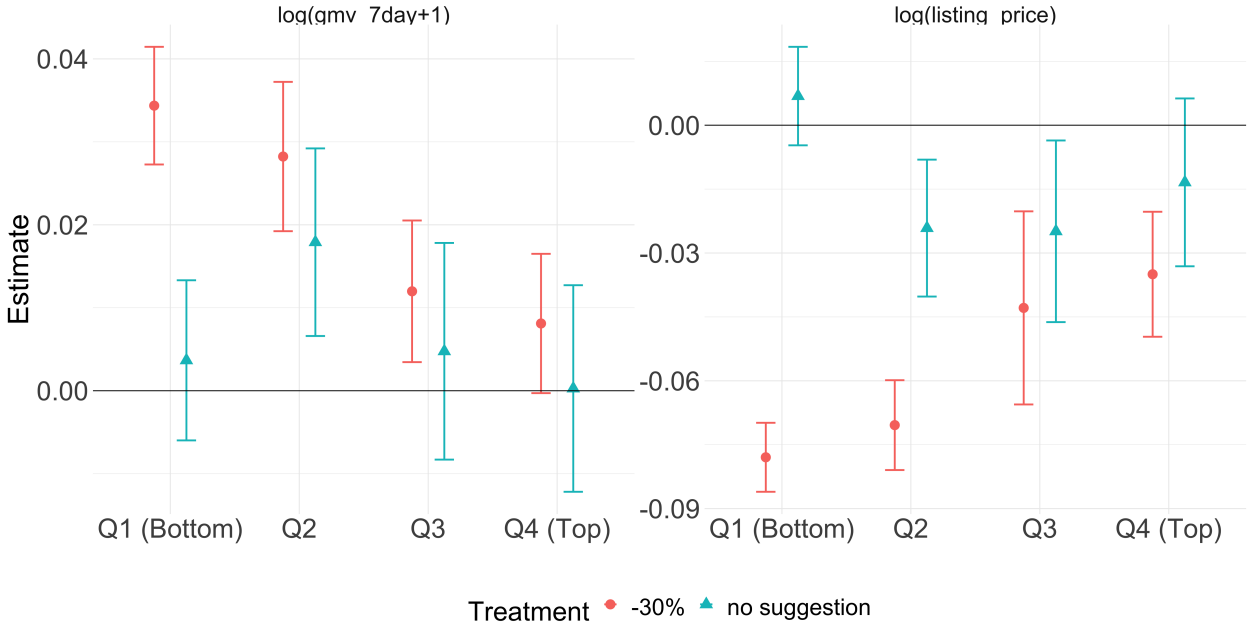


Figure 2.11: Exp 2: Heterogeneous Treatment Effects by Quartile



The treatment effects in the second experiment are similar in magnitude to those in the first experiment. Overall, the results from this experiment address the common concern that field experiments can only measure partial equilibrium effects. After this second experiment, Mercari made a platform-wide update to reduce algorithm-generated suggested prices by 30%.

2.8 Conclusion

Many platforms, including Mercari and eBay, provide price suggestions to sellers. This practice has become increasingly important, especially as P2P platforms typically lack direct control over pricing. Suboptimal pricing can negatively affect platform revenue, which relies on successful transactions. Despite the growing prevalence of suggested pricing, there is limited understanding of how it influences listing prices and its broader downstream effects.

By collaborating with Mercari to run two field experiments, we measure the extent to which platform-suggested pricing impacts sellers' pricing and subsequent selling outcomes. We find a statistically significant effect of suggested prices on sellers' listing prices. On average, a 30% increase in the suggested price leads to a 5% increase in the listing price. Suggested prices have a stronger influence on listing prices when sellers face greater uncertainty about how to price their items (e.g., less experienced sellers and items not in "New" physical condition). In the absence of guidance, inexperienced sellers systematically set listing prices that are too high. Suboptimal pricing by sellers has also been documented in other settings (e.g., Pan and Wang, 2021; Huang

et al., 2022; Huang, 2022). The fact that mispricing can be influenced by suggested prices implies that this mispricing is in part due to a lack of knowledge about demand, rather than characteristics inherent to inexperienced sellers, such as over-optimism or impatience. Suggested prices can help mitigate this mispricing among inexperienced sellers. Overall, our results suggest that suggested prices serve an informative role, functioning as anchors.

These results also imply that while suggested prices influence listing behavior, the risk of inducing collusion is lower than under centralized pricing policies. Specifically, the effect on listing prices is modest—a 30% increase in suggested prices results in only a 5% average increase in listing prices—making seller collusion less likely compared to more heavy-handed pricing strategies. Furthermore, in our setting, the risk of collusion is further mitigated by the design of Mercari's pricing algorithm, which is based only on historical prices within the platform and does not incorporate data from competitors.

Moreover, we find that the effects of suggested prices on listing prices are substantial enough to influence seller revenue. In our setting, lower suggested prices lead to higher revenue for both sellers and the platform, despite the reduction in listing prices. Combined with the finding that suggested prices do not affect the number of listings created, this suggests that platform-suggested pricing can serve as an effective lever for influencing prices, while avoiding the downsides associated with more heavy-handed approaches, such as centralized pricing (Gaineddenova, 2022; Filippas et al., 2023). When we scale the experiment to nearly the entire platform, the lower suggested price continues to yield the highest revenue, reinforcing the robustness of our findings and suggesting they are likely to hold in full equilibrium. As a result, the platform permanently adopted the revenue-maximizing suggested price identified in our experiment.

Our study has several limitations. First, we do not observe sellers' costs, which prevents us from measuring the impact of suggested prices on sellers' profits. Thus, we are unable to provide insights into the profit-maximizing pricing strategies that sellers might adopt. In the context of our study, we focus on seller revenue as a relevant metric, because it is reasonable to assume that Mercari sellers, many of whom are non-professionals selling used goods, face sunk costs. However, this assumption may not hold in other settings with professional sellers (e.g., Amazon). Relatedly, we do not observe sellers' objective functions. One might expect that sellers' pricing strategies, as well as the extent to which they are influenced by suggested prices, depend on their underlying objective functions. For example, more patient sellers may set higher prices, while impatient sellers may prefer to price below the suggested level to increase the likelihood of a quick sale. Second, we can only observe listing and pricing behaviors on our platform, not on competing platforms. Some sellers, especially experienced ones, cross-list on multiple platforms, and their multi-homing behaviors may influence their pricing and selling outcomes. Third, our study focuses exclusively on Mercari, a platform characterized by a high proportion of inexperienced sellers and

used items. As demonstrated in our model (Appendix B), the effect of the suggested price on listing prices depends on sellers' beliefs about the pricing algorithm's objective. The effect sizes that we measure are specific to sellers' beliefs about Mercari's pricing algorithm. Unfortunately, we do not observe sellers' beliefs. However, Mercari, like most platforms, does not reveal the details of the algorithm's objective. We expect that our findings are likely to generalize to other platforms where seller uncertainty is high, both about pricing and the pricing algorithm's objective. While our model helps address this limitation by providing predictions under several alternative beliefs, the extent to which these findings generalize across platforms remains an open question. Finally, the platform subsequently lowered its suggested prices by -30% as a result of our experiments. However, if sellers systematically respond to lower suggested prices by reducing their own prices, and the platform subsequently retrains its algorithm on these updated transaction data, the predicted prices may continue to decline. This dynamic feedback could, over time, generate downward pressure on overall market prices and potentially reduce both seller and platform revenues. The long-run equilibrium outcome would depend on how the platform retrains its model, how frequently updates occur, and how sellers adapt to perceived changes in pricing autonomy. While exploring these long-run feedback effects is beyond the scope of the current paper, we view it as an important direction for future research, particularly as many platforms increasingly rely on adaptive machine learning–based pricing algorithms. We hope that future research can address these limitations.

APPENDIX A

Appendix to Chapter 1

A.1 Randomization, Manipulation, and Interference Checks

A.1.1 Randomization Check

For the personalized discovery-stage experiment, I compare user activity in the 60 days prior to the experiment. I conduct a t-test to check the balance between the treatment and control conditions. As shown in Table A.1 Panel (A), I find no systematic differences between the two conditions in discovery-stage clicks, query-stage clicks, and total orders.

For the personalized query-stage experiment, I analyze user behavior over the 30 days preceding the experiment, including whether the user was active, the number of queries they submitted, and their total spending. As shown in Table A.1 Panel (B), there are no systematic differences between the treatment and control conditions.

Table A.1: Randomization Check

Panel (A): Personalized Discovery-Stage Experiment					
	Non-Personalized		Personalized		
	Mean	SE	Mean	SE	p-value
Discovery-Stage Clicks	2.107	0.008	2.107	0.008	0.986
Query-Stage Clicks	27.784	0.059	27.787	0.059	0.968
Total Orders	0.008	0	0.008	0	0.88
Panel (B): Personalized Query-Stage Experiment					
	Non-Personalized		Personalized		
	Mean	SE	Mean	SE	p-value
Whether Active	0.799	0.001	0.801	0.001	0.188
Queries	1.907	0.004	1.914	0.004	0.246
Spending	0.775	0.0037	0.773	0.004	0.65

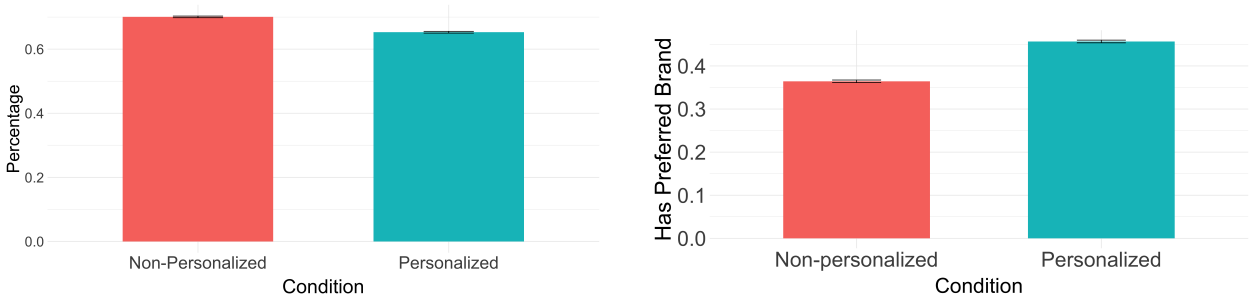
Notes. Panel (A) presents the summary statistics of user activities in the past 60 days before the personalized discovery-stage experiment. Panel (B) presents the summary statistics of user activities in the past 30 days before the personalized query-stage experiment. Both queries and spending use $\log(x + 1)$ transformation.

A.1.2 Manipulation Check

Recall that in the personalized discovery-stage experiment, customers assigned to the non-personalized condition are shown products selected primarily on popularity. As popularity is proxied by historical performance, these product discoveries should be disproportionately drawn from category–brand pairs with high past revenue. To assess this, I compute, among the top 12 recommended items (i.e., those visible on the initial screen), the share that belong to top-selling category–brand combinations.[1] Figure A.1a reports the share of items that fall within the top 10% of brands by revenue within each category. As expected, the non-personalized condition includes a higher proportion of high-revenue category–brand items than the personalized condition.

In the personalized query-stage experiment, I examine whether the ranking reflects customers' inferred preferences. Section 1.4.2 presents results for price as an illustrative case; here I focus on brand preferences inferred from users' past transactions. Figure A.1b reports the mean of an indicator equal to one if at least one of the top 12 query results (i.e., the initial screen) matches the user's inferred preferred brand. Customers in the personalized condition are more likely to be shown items from their preferred brands than customers in the non-personalized condition. This pattern is consistent with the ranking incorporating inferred brand tastes.

Figure A.1: Manipulation Checks



(a) Personalizing Discovery Stage

(b) Personalizing Query Stage

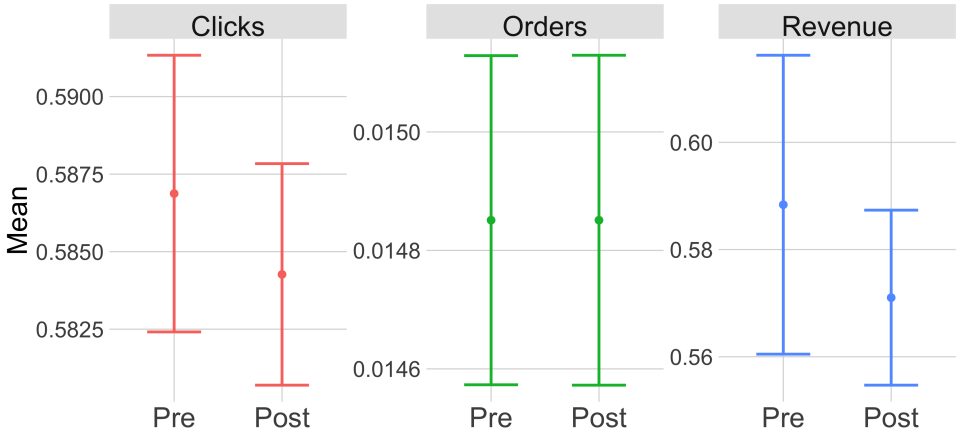
Notes. Panel (a) plots the average share of items among the top 12 discovery products that belong to the top 10% brands by revenue within a category. Panel (b) plots the average of an indicator equal to one if at least one of the top 12 query results matches the user's inferred preferred brand. The analysis uses the initial screen (first 12 items) of the first query session for each user during the experiment. Error bars represent 95% confidence intervals.

[1] Because the experiment was conducted over two years ago at the time of this study, not all discovery-stage recommendations are available due to data storage limitations. Discovery items were recorded only when users generated an interaction event (e.g., clicking a listing or switching tabs), which does not necessarily imply a click on a specific item. I therefore restrict this manipulation check to sessions in which discovery items were recorded. The main treatment-effect estimates using this restricted sample remain consistent with the full-sample results reported in Section 1.4.1.2.

A.1.3 Interference Discussion

Personalized Discovery-Stage Experiment SUTVA is violated (i.e., there is "interference") when the treatment assignment of one user influences the outcomes of other users in the experiment. To check potential interference, I focus on users in the control condition and compare their activity before and after the experiment. If there is no interference, control users should exhibit consistent patterns in clicks, orders, and revenue across both periods, because they do not experience any change. Specifically, I compare each control user's behavior in the discovery stage prior to the experiment with their first session following the experiment's launch. Figure A.2 shows no significant differences in these outcomes, thereby mitigating concerns about interference.

Figure A.2: Comparisons of Users in Control Condition Pre- and Post-Experiment

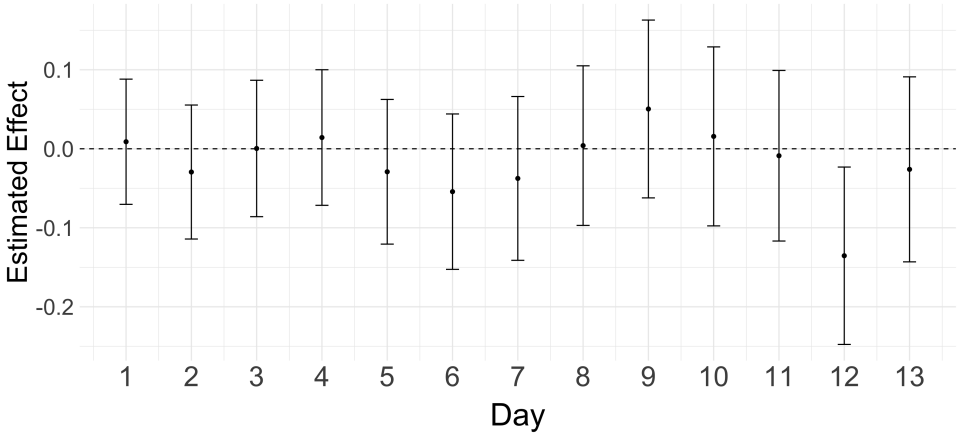


Notes. This figure compares discovery-stage outcomes for users in the control condition before versus after the experiment, including the number of clicks, the number of orders, and total revenue. The pre-period captures each user's last activity within the two weeks leading up to the experiment, and the post-period corresponds to their first discovery engagement afterward. The analysis includes only users who see product discoveries in both periods. The error bar represents the 95% confidence interval.

In addition, I examine the treatment effects of personalization over time by analyzing results separately by day. If interference were present, I would expect treatment effects to vary across days, as the likelihood of interference increases with more users being assigned to the experiment (e.g., Fong et al., 2026). However, as shown in Figure A.3, the treatment effects remain stable across days, further alleviating concerns about interference.

Personalized Query-Stage Experiment First, in this experiment, only a small share of users were assigned to each experimental condition: 10% to the treatment condition and 10% to the control condition. This low saturation rate reduces the likelihood of substantial interference bias

Figure A.3: Treatment Effects of Personalized Discovery Stage by Day



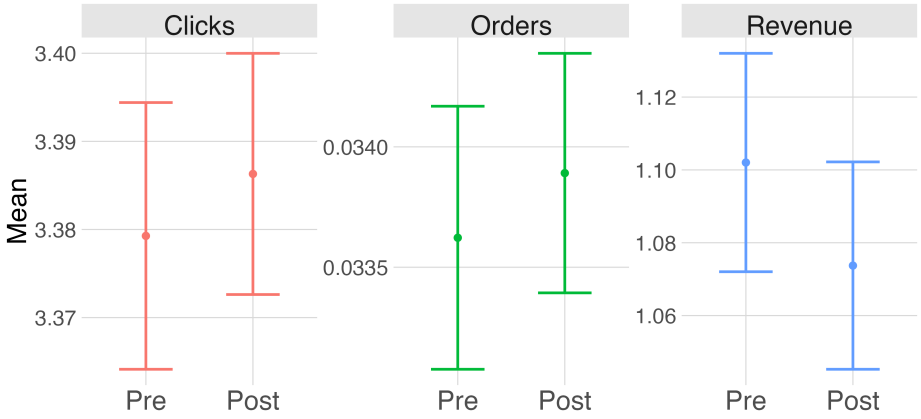
Notes. This figure plots the the estimation results of β^t in the following equation: $\log(Revenue + 1) = \beta_0 + \beta_1 D_t + \beta_2 Personalized_t + \sum_t \beta^t D_t \times Personalized_i + \epsilon_{it}$, where D_t indicates the day. The omitted day is the first day (Day = 0). Standard errors are in parentheses, clustered at the individual level. The error bar represents the 95% confidence interval.

(e.g., Fong, 2024).

Second, following the approach above, I test for interference by examining users who are consistently exposed to the status quo ranking algorithm both before and after the experiment. Specifically, I focus on the 70% of users assigned to the status quo ranking in both periods and compare their clicks, orders, and revenue before versus after the experiment. Figure A.4 shows that these outcomes remain stable across periods, providing evidence against interference.

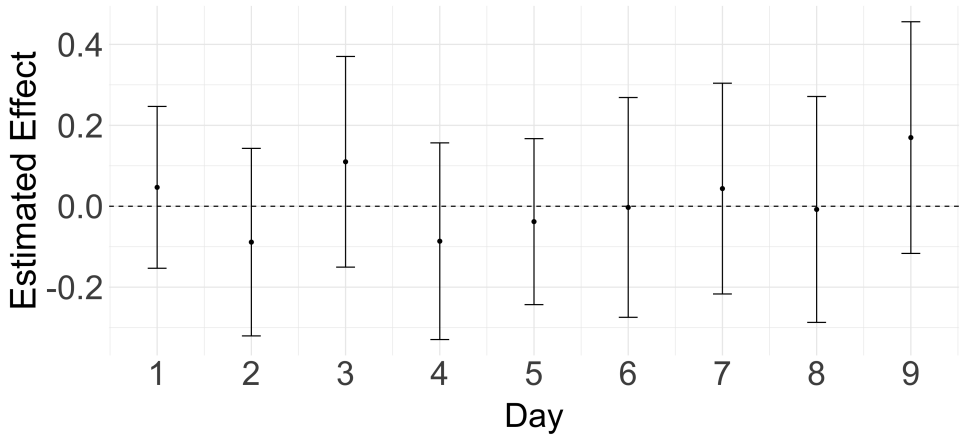
Third, similarly, I estimate the treatment effects of personalized query-stage ranking over days. Figure A.5 plots these estimates for each day during the experiment. None of the effects differ significantly from those on the first day, further minimizing concerns about interference.

Figure A.4: Comparisons of Users in Status Quo Condition Pre- and Post-Experiment



Notes. This figure presents the number of clicks, orders, and total revenue at the query stage for users in the status quo condition. The pre-period reflects each user's last activity within the 12 days preceding the experiment (matching the duration of the personalized query-stage experiment), and the post-period captures their first session. The analysis is restricted to users who conduct a search query in both periods. The error bar represents the 95% confidence interval.

Figure A.5: Treatment Effects of Personalized Query Stage by Day



Notes. This figure plots the the estimation results of β^t in the following equation: $\log(Revenue + 1) = \beta_0 + \beta_1 D_t + \beta_2 Personalized_t + \sum_t \beta^t D_t \times Personalized_i + \epsilon_{it}$, where D_t indicates the day. The omitted day is the first day (Day = 0). Standard errors are in parentheses, clustered at the individual level. The error bar represents the 95% confidence interval.

A.2 Additional Evidence and Discussion

A.2.1 Spillovers from Discovery Stage to Query Stage

I use the experimental condition in the personalized discovery-stage experiment as an instrumental variable to examine the relationship between the discovery and query stages. The experimental condition exogenously influences outcomes in the discovery stage and is uncorrelated with unobserved factors that affect outcomes in the query stage. Furthermore, it impacts the query stage only indirectly, through its effect on the discovery stage.

I estimate the following two-stage equations:

$$Y_i^D = \kappa_0 + \kappa_1 Personalized_i + \xi_i, \tag{A.1}$$

$$Y_i^Q = \varphi_0 + \varphi_1 \hat{Y}_i^D + \epsilon_i, \tag{A.2}$$

where Y_i^D and Y_i^Q denote customer i's outcomes in the discovery and query stages, respectively, and $\hat{Y}_i^D$ is the predicted value from Equation (A.1). I examine three dependent variables: the number of clicks, orders, and logged revenue. Estimation results are presented in Table A.2. I find that an increase in discovery-stage outcomes (clicks, orders, and revenue) leads to a decline in the corresponding outcomes in the query stage. This pattern suggests a negative spillover effect from the discovery stage to the query stage.

Table A.2: Relationship between Discovery Stage and Query Stage

	Query Clicks (1)	Query Orders (2)	Log(Query Rev) (3)
Constant	4.504***	0.0250***	0.0736***
	(0.0738)	(0.0024)	(0.0074)
Discovery Clicks	-3.120***		
	(0.1723)		
Discovery Orders		-0.7868***	
		(0.2444)	
Log(Discovery Rev)			-0.8287***
			(0.2387)
1-stage F-stats	841.6	43.9	38.3
Observations	3,050,146	3,050,146	3,050,146

Notes. This table shows the estimation results of Equations (A.1) and (A.2). Standard errors are in parentheses, clustered at the individual level. * $p < 0.1$, ** $p < 0.5$, *** $p < 0.01$.

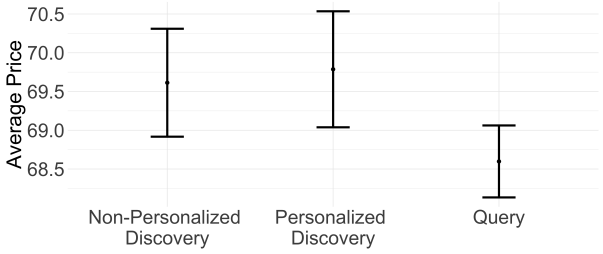
A.2.2 Discovery versus Query Stages

A.2.2.1 Price and Item Condition

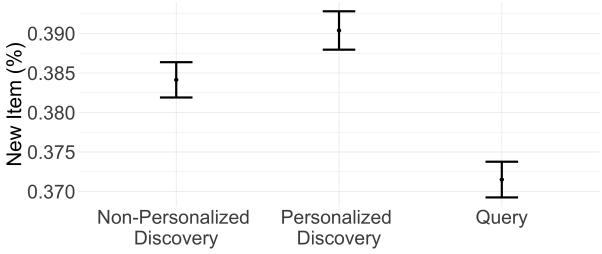
I present the average price and the proportion of new items at each stage. Figure A.6a shows that average prices are similar under personalized and non-personalized discovery stage, and are higher than the prices shown in the query stage. Figure A.6b shows that items shown in the personalized discovery stage are more likely to be new, suggesting that customers tend to prefer new products. In addition, the proportion of new items is higher in the discovery stage than in the query stage.

Figure A.6: Price and Item Condition Across Stages

(a) Average Price



(b) Proportion of New Items



A.2.2.2 Product Variety

I examine the variety of products presented to customers. I measure product variety using two standard metrics in the literature (e.g., Chen et al., 2024; Holtz et al., 2020): Shannon entropy and the Herfindahl-Hirschman Index (HHI).

Shannon entropy quantifies the dispersion of a distribution. For a discrete random variable Y with support $\{y_1, \ldots, y_n\}$ and associated probabilities $P(y_i)$, entropy is defined as:

$$H(Y) = -\sum_{i=1}^{n} P(y_i) \log_2 P(y_i), \tag{A.3}$$

where the logarithm is base 2. Entropy is maximized under a uniform distribution, which corresponds to maximal variety, and it decreases as the distribution becomes more concentrated.

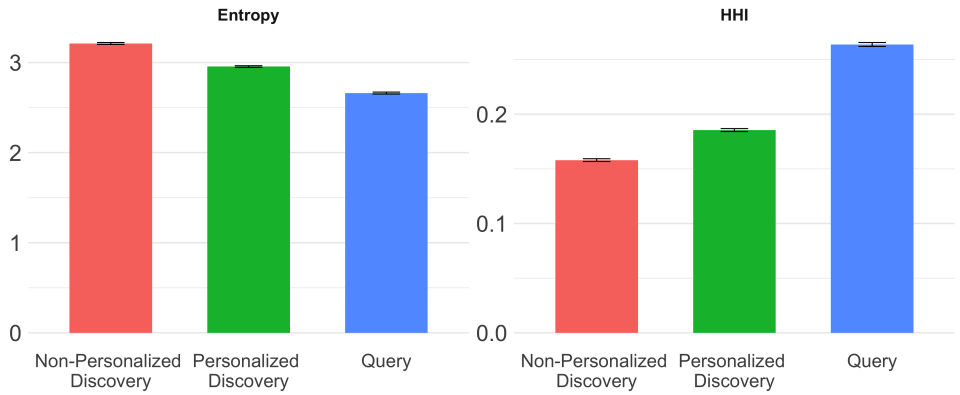
HHI captures concentration by summing the squared probabilities:

$$HHI(Y) = \sum_{i=1}^{n} (P(y_i))^2 . \tag{A.4}$$

Higher HHI values indicate greater dominance by a cluster and thus lower product variety.

Using Equations (A.3) and (A.4), I compute Shannon entropy and HHI at the session level, focusing on the top 12 items shown on the initial screen for both stages. I define each brand-physical condition combination as a cluster. Figure A.7 presents these two metrics for the personalized discovery stage, the non-personalized discovery stage, and the query stage. As shown, personalizing discovery stage reduces product variety, as indicated by lower entropy and higher HHI. Even with personalization, the discovery stage still exhibits greater variety than the query stage, suggesting that it exposes users to a broader range of products.

Figure A.7: Shannon Entropy and HHI



Notes. This figure presents the Shannon entropy and HHI of the top 12 products, using data from the personalized discovery-stage experiment. Higher entropy and lower HHI indicate greater variety.

A.2.3 Query Analysis

A.2.3.1 Inferring Demand States via Query Specificity

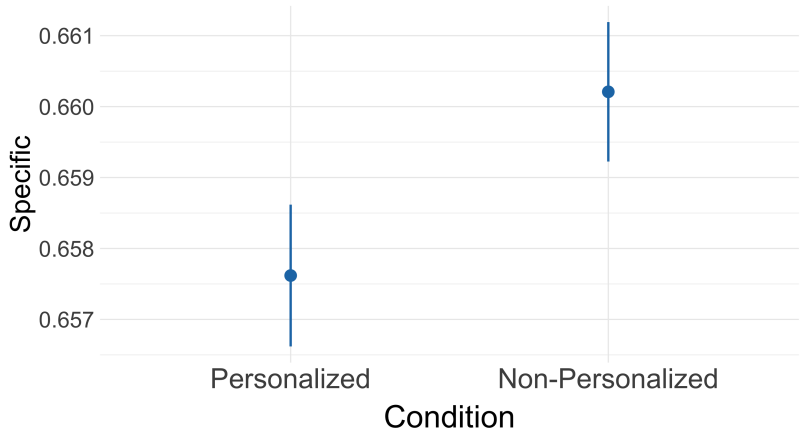
Yuan et al. (2025) identify two distinct customer states when visiting a platform: demand fulfillment and demand formation. Customers in the fulfillment state have well-defined needs and tend to compensate for less relevant discovery-stage recommendations by submitting more specific queries. In contrast, customers in the demand formation state rely more heavily on product discovery, and reduced relevance at the discovery stage leads them to submit more generic queries. These contrasting patterns offer a lens for inferring whether a customer is in a demand formation or fulfillment state. I adopt this framework to identify customer states.

I examine how personalization at the discovery stage influences the specificity of users' search queries. I follow the rule-based approach outlined in Yuan et al. (2025) to measure query specificity. Each query is decomposed into two types of components: generic words and decoration words.

Generic words refer to terms that describe a product's basic type or category (e.g., *jeans*, *headphones*), while decoration words include descriptive modifiers such as brand (*Lululemon*, *LEGO*), color (*yellow*, *red*), physical condition (*new*), material (*cotton*, *leather*), or size (*plus size*). I compile a comprehensive dictionary of decoration terms derived from the product attributes available on the platform. A query is classified as *generic* if it contains only generic words, and as *specific* if it includes at least one decoration word. For example, the query *"black UGG boots"* contains a generic term (*boots*) along with decoration words (*UGG* as the brand and *black* as the color), and is categorized as specific.

In Figure A.8, I present query specificity for users in the personalized and non-personalized conditions. The figure shows that users in the non-personalized condition submit more specific queries. This pattern suggests that many customers arrive with well-defined demand and compensate for less personalized discovery-stage recommendations by refining their query inputs.

Figure A.8: The Impact of Personalizing Discovery Stage on Search Query Specificity

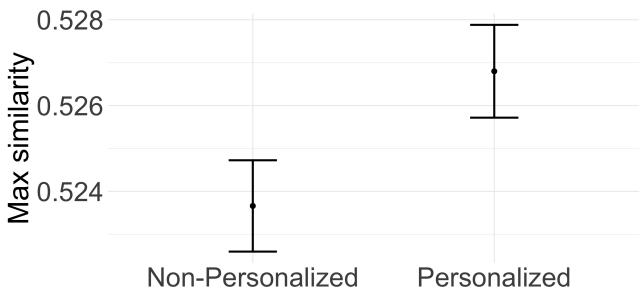


Notes. This figure presents the query specificity in the personalized and non-personalized conditions from the personalizing discovery-stage experiment. The error bar represents the 95% confidence interval.

A.2.3.2 Correlation between Discovery Products and Query

I examine how closely the products shown in the discovery stage align with the query a customer submits later in the same session. To quantify this alignment, I compute a semantic similarity score between each discovery-stage item's title and the session's query using Sentence-BERT embeddings and cosine similarity. Figure A.9 shows that the maximum similarity is higher under the personalized condition than the non-personalized condition. This pattern suggests that a broader product assortment does not increase customer interest or trigger spontaneous query initiation.

Figure A.9: Correlation between Discovery Products and Query by Condition



Notes. This figure presents the correlation between discovery-stage products and the subsequent query in the personalized and non-personalized conditions using data from the discovery-stage personalization experiment. The error bar represents the 95% confidence interval.

A.2.4 Stratification by Discovery-Stage Purchase

To examine whether and how users continue into the query stage differently depending on whether the discovery stage offers a good match, I stratify users by whether they purchase at the discovery stage and compare outcomes between the personalized and non-personalized conditions within each stratum. Table A.3 reports effects on query initiation and total revenue across both stages. Among users who purchase at the discovery stage, personalization reduces the likelihood of initiating a query, with no corresponding change in total revenue. Among users who do not purchase at the discovery stage, personalization also reduces query initiation and lowers total revenue. These results suggest that when discovery-stage personalization fails to provide a good match, customers are more likely to exit rather than continue the journey.

Table A.3: Personalizing Discovery Stage on Query and Total Revenue by Stratification

	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Purchase at Discovery Stage					
Whether Query	0.321	0.308	-4.189	-2.442	0.015
Total Revenue	43.578	43.809	0.526	0.375	0.708
No Purchase at Discovery Stage					
Whether Query	0.580	0.556	-4.130	42.046	< 0.001
Total Revenue	0.568	0.544	-4.134	2.370	0.018

Notes. This table reports the effects of personalizing the discovery stage on query initiation and total revenue, stratified by whether the customer makes a purchase at the discovery stage. I report the mean values of outcomes for the control condition and the treatment condition. The difference reflects the mean difference between the treatment and control conditions. The final two columns report t-statistics and p-values from two-tailed tests for differences in means.

A.3 Ranking Model at the Query Stage

In the re-ranking stage, I employ LambdaMART (Burges, 2010), a widely used pairwise learning-to-rank algorithm that combines the LambdaRank framework with Multiple Additive Regression Trees (MART). LambdaMART leverages gradient-boosted decision trees and uses a cost function derived from LambdaRank to optimize ranking performance. As one of the most effective learning-to-rank algorithms, LambdaMART has been widely adopted in both academic research and industry applications. In marketing literature, Yoganarasimhan (2020) finds that LambdaMART outperforms alternative ranking models in terms of improvements in Normalized Discounted Cumulative Gain (NDCG) metrics. In industry, LambdaMART powers Microsoft's Bing search engine.[2]

Training Data The training data include item-level features, contextual features, and user-level features (for the personalized model). Item-level features capture static attributes (e.g., category, brand) and recent interaction metrics over the past 30 days (e.g., cumulative likes). Contextual features capture market conditions at the time a query is submitted, such as real-time inventory for relevant products. User-level features include behavioral variables (e.g., item views in the past 30 days) and demographic information (e.g., inferred age). The training sample comprises all query-stage activity on the platform over a two-week period.

LambdaMART predicts item relevance via a ranking score. I construct this score from user interactions (e.g., purchases, likes), placing greater weight on higher-intent actions such as purchases. The weights are calibrated using historical user activity data.

Training Process The model is trained by tuning several hyperparameters to balance flexibility and overfitting. These include regularization terms (which control model complexity), the learning rate (which determines the step size during optimization), and the number of leaves per tree (which affects the model's ability to capture interaction effects). I also tune the maximum tree depth, the minimum number of observations per leaf, the number of estimators, the bagging fraction, and the feature-subsampling ratios. The model is trained to optimize NDCG over the top 100 positions. For evaluation, I split the data into 80% for training and 20% for testing.

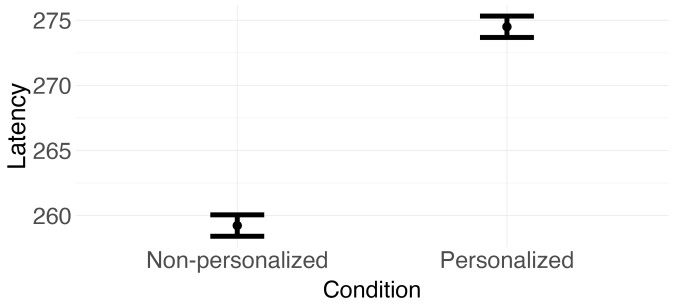
[2]Microsoft Research, "RankNet: A Ranking Retrospective," https://www.microsoft.com/en-us/research/blog/ranknet-a-ranking-retrospective/, accessed March 2026.

A.4 Latency

The Effect of Personalization on Latency As shown in Figure A.10, in the personalized query-stage experiment, average latency is higher in the personalized condition (274 milliseconds) compared to the non-personalized condition (259 milliseconds). Based on the discussion with the company, this increase is primarily driven by the additional time required to retrieve and compute user-specific features.[3]

Since the conclusion of this experiment, the company has significantly expanded its infrastructure investments to improve system performance. These latency results are thus specific to my experimental period. However, they also highlight another cost of personalization—the continual infrastructure investments required to maintain responsiveness at scale.

Figure A.10: The Impact of Personalized Query Stage on Latency



Notes. This figure compares latency under personalized and non-personalized conditions from the personalized query-stage ranking experiment. The error bar represents the 95% confidence interval.

The Effects of Latency on Customers From the customer's perspective, increased latency can lead to longer loading times, potentially degrading the overall shopping experience. To quantify the relationship between backend latency and observed loading time, I estimate the following regression:

$$Y_k = \iota_0 + \iota_1 latency_k + \epsilon_k, \tag{A.5}$$

where $latency_k$ denotes the backend latency (in milliseconds, log-transformed) for session k, and Y_k represents the outcome of interest. This analysis uses data from the personalized query-stage experiment. I first consider Y_k as the customer-observed loading time for session k. Since clickstream data is recorded at the second level, observed loading times are rounded to the nearest

[3]As the personalized discovery-stage experiment was conducted more than two years prior to my data access, backend latency logs from that period are no longer available.

full second.[4] 37.8% of sessions have an observed loading time of 0 seconds, and 55.8% register a loading time of 1 second.[5] The estimated coefficient on latency, $\hat{\iota_1} = 0.20$ (standard error = 0.05), indicates a statistically significant and positive relationship: longer backend latency is associated with longer observed loading times.

Next, I investigate the impact of latency on consumer engagement by estimating Equation (A.5) with three outcome variables: the number of clicks, the number of orders, and logged revenue per session. In Table A.4, a 1% increase in latency is associated with a 4.6% decrease in clicks, a 0.4% decrease in orders, and a 0.1% decrease in revenue. These findings demonstrate the economic significance of minimizing latency in personalization systems.

Table A.4: Effects of Search Query Latency

	Clicks	Orders	Log(Rev)
	(1)	(2)	(3)
Constant	4.101***	0.0703***	0.2149***
	(0.1420)	(0.0091)	(0.0267)
Log(Latency)	-0.0464*	-0.0038**	-0.0130***
	(0.0258)	(0.0016)	(0.0048)
Observations	101,419	101,419	101,419
R^2	0.00004	0.0001	0.0001

Notes. This table reports the estimated effects of backend search latency (log-transformed) on consumer engagement in the query stage. Standard errors are in parentheses, clustered at the individual level. * $p < 0.1$, ** $p < 0.5$, *** $p < 0.01$.

[4]The latency dataset is not directly linked to individual user sessions, so I match latency to customer sessions based on timestamp and exclude ambiguous cases (e.g., concurrent queries by different users at the same timestamp).

[5]Sessions with loading times above 3 seconds (fewer than 1%) are excluded as outliers.

A.5 Robustness Checks of Alternative Algorithm Designs

A.5.1 Alternative Discovery-Stage Model

I evaluate an alternative discovery-stage recommendation model because the "quality" of recommendations may influence consumer decisions. To do so, I draw on a separate field experiment run by the company that tested a more advanced personalized recommendation algorithm on the discovery page. Relative to the baseline model, this approach represents a substantial upgrade along several dimensions. It uses a transformer-based architecture that incorporates visual product representations (e.g., images) and a richer set of user engagement signals. It also introduces a forecasted-serendipity component that predicts future user actions and tailors recommendations with greater contextual relevance. Consistent with these design changes, the new model improves standard ranking metrics, including recall, precision, and NDCG.

The experiment evaluating this enhanced algorithm was conducted over a two-week period in Spring 2023. During the experiment, 50% of users were randomly assigned to the status quo personalized discovery-stage algorithm (the treatment condition in Section 1.4.1), while the remaining 50% of users received the more advanced algorithm at the discovery stage.

I report the experimental results in Table A.5. I find that customers exposed to a more advanced algorithm click on 45.5% more items, place 17.4% more orders, and generate 16.2% more revenue within the discovery stage compared to those in the status quo condition. However, they are 1.6% less likely to initiate a query, click on 2% fewer items, place 3.7% fewer orders, and generate 2.8% less revenue in the query stage. The total orders and revenue across two stages remain unchanged.

Table A.5: Treatment Effects of More Advanced Personalized Discovery Stage

Variable	Personalized	More Personalized	Change (%)	t-statistic	p-value
Discovery Clicks	0.216	0.247	14.245	19.799	< 0.001
Discovery Orders	0.005	0.006	20.132	9.904	< 0.001
Discovery Revenue	0.230	0.284	23.614	5.858	< 0.001
I(Initiate a Query)	0.600	0.592	-1.322	-12.595	< 0.001
Query Clicks	3.427	3.366	-1.796	-8.096	< 0.001
Query Orders	0.038	0.037	-3.823	-5.156	< 0.001
Query Revenue	1.691	1.651	-2.323	-1.721	0.085
Total Clicks	3.643	3.612	-0.846	-3.986	< 0.001
Total Orders	0.043	0.043	-1.056	-1.502	0.133
Total Revenue	1.921	1.936	0.783	0.607	0.544

Notes. This table presents the treatment effects of a more advanced personalization algorithm at the discovery stage. I report the mean values of outcomes under both the control condition (baseline personalization) and the treatment condition (personalization using a more advanced algorithm). The reported difference reflects the mean outcome in the treatment condition minus that in the control condition. The final two columns display the corresponding t-statistics and p-values from two-tailed tests for differences in means.

A.5.2 Alternative Ranking Model

The effectiveness of personalized ranking also depends in part on the "quality" of the underlying ranking algorithm. For instance, if the baseline non-personalized algorithm is already highly optimized, the marginal benefit of personalization may be limited. In contrast, greater gains may arise when the baseline algorithm has more room for improvement.

I use the status quo ranking model as the base algorithm and develop a personalized counterpart that additionally incorporates user-specific features. Recall that in the personalized search experiment, among the remaining 80% of users not included in the experimental conditions, 70% were randomly assigned to a non-personalized status quo ranking model and 10% were assigned to a personalized version of the same model that integrates user-specific features.

Table A.6 reports the treatment effects of personalization under the status quo ranking algorithm. Personalization leads to statistically significant improvements across all query-stage outcomes: clicks increase by 1.7%, orders by 6.9%, and revenue by 7.3%. These results convey two main takeaways. First, the choice of ranking algorithm itself influences user engagement and platform revenue. For example, the LambdaMART and status quo models produce different revenue outcomes. Second, holding the algorithm constant, incorporating user-level data can enhance overall revenue.

Table A.6: The Effects of Personalized Query Ranking (Alternative Model)

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Search Clicks	2.464	2.506	1.670	3.209	0.001
Search Orders	0.025	0.027	6.912	3.799	< 0.001
Search Revenue	0.866	0.929	7.334	2.220	0.026
Rec Clicks	0.821	0.824	0.392	0.565	0.572
Rec Orders	0.002	0.002	-2.873	-0.447	0.655
Rec Revenue	0.045	0.050	11.195	0.775	0.438
Total Clicks	3.285	3.329	1.350	3.182	0.001
Total Orders	0.027	0.029	6.324	3.601	< 0.001
Total Revenue	0.911	0.980	7.526	2.337	0.019

Notes. This table presents the treatment effects of personalized query rankings based on the status quo algorithm. I report the mean values of outcomes for the control condition (non-personalized status quo model) and the treatment condition (personalized status quo model). The difference reflects the mean difference between the treatment and control conditions. The final two columns report t-statistics and p-values from two-tailed tests for differences in means.

A.6 Belief Estimation

A.6.1 Conducting a Query

Beliefs about Query Stage. To improve the precision of the estimated empirical distribution, I group similar queries using natural language processing techniques. Specifically, I first embed every query string into a TF-IDF vector space over unigrams and bigrams. I then run k-means clustering on those embeddings. Next, I choose the number of clusters by maximizing the average silhouette score under a cosine-distance metric. The silhouette score measures how similar an object is to its own cluster compared to other clusters. For example, "PS5" and "PlayStation 5" belong to the same query group. These procedures ultimately give me a total of 39 query groups.

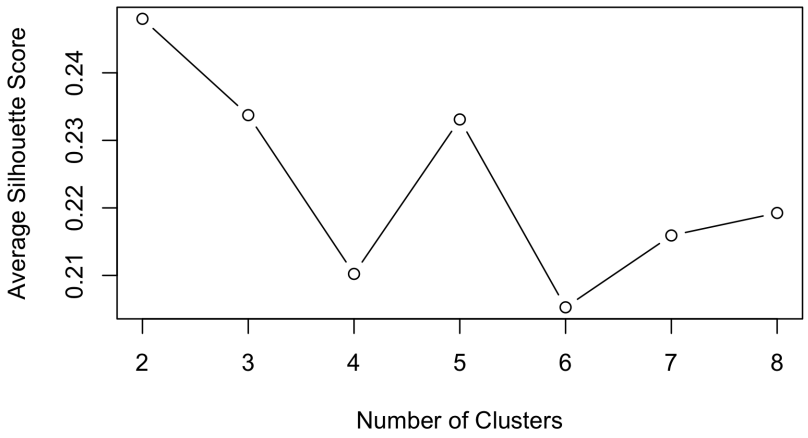
I then empirically estimate the distribution of product attributes for each query group across sessions. For customers in the non-personalized condition, expectations are formed based on the empirical distribution of products observed across all non-personalized users. For those in the personalized condition, I segment them based on their historical activities, including preferences (e.g., price level, brand), activity levels (e.g., item clicks in the past 30 days), purchase histories, and inferred demographic age. Customers' expectations are assumed to follow the empirical distribution of products seen by customers within their respective segments. To group customers by segments, I again apply k-means clustering to group individuals in the personalized condition based on observed user attributes and use the silhouette scores to assess the quality of the clustering. I iterate the number of clusters from 2 to 8 and find that the silhouette score is maximized when the segment is 2 (Figure A.11). Based on the inferred segments, customers in one group are generally more active, with greater past engagement (e.g., item clicks, queries, and purchases), while those in the other group are less active. For example, the median number of purchased products since account creation is 7 for the less active segment and 49 for the more active segment. I would expect the optimal number of clusters to be larger when considering all customers on the platform.

Finally, to reduce computational burden, I follow Koulayev (2014) and Ghose et al. (2019) and use a bootstrap sampling procedure. Specifically, in each draw I sample 48 products from the empirical distribution, conditional on the query and the personalization condition (and, in the personalized condition, conditional on the segment as well).[6]

Beliefs based on Discovery Stage Recall that when the discovery-stage products do not match the customer's intent, the customer evaluates whether to query based on the expected payoff from the query-stage environment. When the discovery-stage products match the customer's intent, I treat the discovery stage as an additional signal about the latent payoff u_i^q. I define intent-match if at

[6]The top 48 products account for 74% of clicks and 80% of purchases.

Figure A.11: Silhouette Score of Clusters



least three products in the discovery stage match the query submitted by the customer.[7] When intent matches, customers may expect the query stage to display products similar to those shown in the discovery stage. To capture this, I construct an empirical distribution of discovery-stage products that approximates what a customer could plausibly encounter at the query stage. Specifically, I pool products shown in the discovery stage across sessions and stratify them by position. Within each position, I further restrict attention to products with similar attributes, yielding a position-specific empirical distribution of comparable discovery-stage items. I then apply a bootstrap procedure analogous to the one described above: in each draw, I construct a synthetic discovery by independently sampling one product for each position from the corresponding position-specific empirical distribution.

A.6.2 Inspecting a Product

I estimate the following equation to examine the relationship between a product's hidden attributes and its visible attributes:

$$Z_j = \phi_0 + \phi_1 \log(price_j) + category_j + brand_j + \eta_j, \tag{A.6}$$

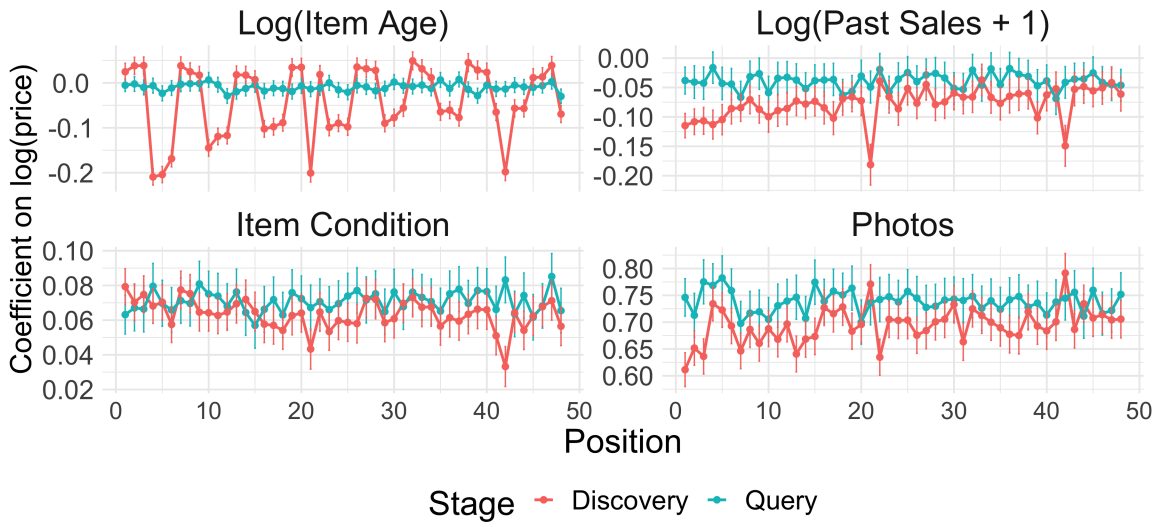
where Z_j are the hidden attributes, including the number of photos, item physical condition (measured on a scale from 1 to 5, where 1 indicates poor condition and 5 indicates new), item age and its seller's past sales. The visible attributes include price, category, and brand. The error term

[7] I also try definitions of intent-match based on one or two products, and the results are similar.

η_j is assumed to have a mean of zero.

Figure A.12 illustrates the estimation results by plotting the estimated (logged) price coefficients. The results suggest that higher-priced items tend to include more photos, be in better physical condition, and be listed by less experienced sellers (i.e., those with fewer past sales). In addition, the coefficients exhibit variation across both stages and positions.

Figure A.12: Beliefs about Hidden Attributes



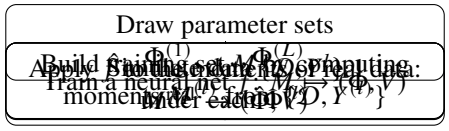
Notes. This figure displays the estimated price coefficients from Equation (A.6) for the discovery stage and query stage respectively. Error bars represent 95% confidence intervals.

A.7 Search Model Estimation

A.7.1 Model Estimation using Neural Networks

I estimate the structural search model using the NNE approach proposed by Wei and Jiang (2025), which leverages a neural network to learn a mapping from datasets to the structural parameters. Figure A.13 illustrates the steps to train a neural network to recover the parameter vector.

Figure A.13: Neural Network Training Illustration



The process can be described as follows.

1. **Draw Parameter Sets:** Generate L independent draws of model parameters $\Phi^{(1)}, \ldots, \Phi^{(L)}$ uniformly from a predefined parameter space. In my application, I set L = 20,000.

2. **Simulate Data:** For each draw $\Phi^{(l)}$, simulate consumer behavior using the structural search model to generate the data $\{\mathcal{D}, Y^{(l)}\}$. The dataset $\mathcal{D}$ includes item attributes (X, Z) and cost variables (Pos, $Personalized$). The simulated outcomes $Y^{(l)}$ include inspection, query submission, and purchase.

3. **Compute Moments:** From each simulated dataset $\{\mathcal{D}, Y^{(l)}\}$, compute a vector of data moments $M^{(l)}$ that summarize the key data patterns. I construct the following data moments at the option-level (including items and query), session-level, and individual-level: means and variances of $Y^{(l)}$ and cross-covariances between $\mathcal{D}$ and $Y^{(l)}$. As suggested by Wei and Jiang (2025), NNE is robust to redundant moments, because the neural net can learn from the training set which moments are informative about the parameters.

4. **Train a Neural Network:** Collect all moment-parameter pairs $(M^{(l)}, \Phi^{(l)})_{l=1}^{L}$ into a training set, where M is an input to the neural net and Φ is the output. The neural network is trained to learn the inverse mapping from empirical data moments M to model parameters Φ and covariance matrix V. Following Wei and Jiang (2025), I use a cross-entropy loss to teach the network to map from M back to (Φ, V). The neural network architecture is defined using a sequence of layers. The input layer normalizes the input data using symmetric rescaling. This is followed by a fully connected layer with 64 hidden nodes and a ReLU activation function. The output layer is another fully connected layer, with the number of outputs matching the dimensionality of the output data. The initial learning rate is set to 0.01 and updated using

a piecewise schedule, decreasing every 1,000 iterations. The network is trained using the Adam optimizer over a maximum of 500 epochs with a mini-batch size of 500 samples. I use a 90/10 split for training and testing, and model validation is conducted every 500 iterations. As demonstrated by Wei and Jiang (2025), the accuracy of NNE is not sensitive to the specific architecture or hyperparameter choices of the neural network. I experiment with different hyperparameter settings and find very consistent predictive results.

5. **Apply to Real Data:** Compute the moments M^{real} on actual data $\{\mathcal{D}, Y^{\text{real}}\}$. Feed M^{real} into the trained network to obtain both the point estimate $\hat{\Phi}$ and the statistical accuracy for the point estimate $\hat{V}$.

A.7.2 Monte Carlo Simulation

I generate a dataset consisting of 3,000 customers and each customer has three sessions. In each session, customers follow a two-stage sequential search process: first a discovery stage, then a query stage. Each stage displays 24 products. Customers submit one of the two queries available. Customers are randomly assigned, with equal probability, to either a personalized query result or a non-personalized query result and are categorized into one of two segments (Segment 1 or Segment 2). For simplicity, product attributes and rankings are randomly generated.

Each product has three visible attributes: price, category and brand, and four hidden attributes: number of photos, product age, item condition, and sellers' past sales. Logged prices are drawn from a normal distribution: $\log(\text{Price}) \sim \mathcal{N}(3.5, 1)$. Category is a dummy variable indicating whether the item belongs to the same category as the customer's purchase intent. The probability that an item is in the relevant category is 0.3 in the discovery stage and 0.9 in the query stage. There are five brands, each with the following probabilities of being chosen in the discovery stage: 0.05, 0.15, 0.15, 0.20, and 0.25. In the query stage, the corresponding probabilities are 0.10, 0.02, 0.10, 0.05, and 0.08. The remaining probability mass in each stage corresponds to unbranded items. The number of photos is drawn from a discrete distribution over $\{1, 2, 3, 4, 5\}$ with probabilities $P = (0.10, 0.05, 0.25, 0.40, 0.30)$, respectively. Item condition is also sampled from a discrete distribution over $\{1, 2, 3, 4, 5\}$. Logged product age follows $\mathcal{N}^+(4, 1)$ and sellers past sales (logged) follow $\mathcal{N}^+(0, 1)$; both are normal distributions truncated below at zero.

Consumer utility depends on both visible and hidden product attributes. I follow the procedures described in Section 1.5.2.2 to estimate customers' beliefs about product utility conditional on visible attributes and beliefs about the utility of conducting a query. Customers also face two types of costs discussed in Section 1.5.2.3: (1) an inspection cost that varies with a product's position in the list and (2) a query cost that depends on the personalization condition.

Customers begin their decision process in the discovery stage, choosing whether to inspect an

item, conduct a search query, or exit (i.e., select the outside option). If they conduct a query, they exit the discovery stage and proceed to the query stage, where they view a new set of 24 products related to the query they submit and decide which to inspect. Ultimately, customers choose to purchase the most preferred product among all inspected products across both stages or opt for the outside option.

Table A.7 reports the results of the estimation. Column (1) lists the true parameter values, and Column (2) presents the corresponding estimates from the model. Overall, the estimated parameters align closely with the true values.

A.7.3 Model Fit

To evaluate model fit, I compare predicted and observed customer outcomes using parameter-free moments: the average number of inspections per session, the percentage of sessions with a search query, and the percentage of sessions with a purchase. Table A.8 shows that the model captures these moments well.

Table A.7: Monte Carlo Simulation Results

Variables	(1) True	(2) Estimated
Preferences		
Mean Preferences		
Constant	-3.0	-3.098 (0.066)
Price	-0.4	-0.443 (0.033)
Same Category	0.8	0.828 (0.020)
Brand 1	0.4	0.412 (0.038)
Brand 2	0.4	0.448 (0.048)
Brand 3	0.5	0.552 (0.057)
Brand 4	0.3	0.349 (0.089)
Brand 5	0.1	0.141 (0.036)
N Photos	0.3	0.292 (0.090)
Item Age	-0.2	-0.238 (0.011)
Physical Condition	0.3	0.293 (0.019)
Seller Past Sales	0.3	0.286 (0.010)
Inspection Base (Discovery)	-3.0	-3.110 (0.055)
Position	1.0	0.964 (0.064)
Inspection Base (Query)	-4.0	-3.841 (0.256)
Query Base	-1.5	-1.487 (0.027)
Personalization	0.8	0.879 (0.126)
Preferences Heterogeneity (SD)		
Price	0.2	0.200 (0.058)
N Photos	0.1	0.094 (0.062)
Item Age	0.1	0.100 (0.057)
Physical Condition	0.1	0.094 (0.058)
Seller Past Sales	0.1	0.114 (0.065)
Costs		
Inspection Base (Discovery)	-3.0	-3.260 (0.281)
Position	1.0	0.964 (0.137)
Inspection Base (Query)	-4.0	-3.941 (0.256)
Query Base	-1.5	-1.587 (0.244)
Personalization	0.8	0.829 (0.019)

Notes. The estimation is based on 3,000 customers, each with 3 sessions (9,000 sessions in total). Standard errors are in parentheses.

Table A.8: Model Fit

Moments	Predicted	Observed
Session level		
Inspections	1.614	1.672
Query Percentage	0.847	0.851
Orders	0.031	0.031
Individual level		
Inspections	4.192	4.326
Orders	0.067	0.070

A.8 Utility-Based Ranking Models

At each stage, the platform ranks candidate items for customer i using a stage-specific match index,

$$v_{ij}^{g} \;=\; \omega^{g} u_{ij} \;+\; (1-\omega^{g})\bar{u}_{j}, \qquad g \in \{D, Q\}, \tag{A.7}$$

where $g = D$ denotes the discovery-stage product list and $g = Q$ denotes the query-stage product list. Individual-level utility is $u_{ij} = X_j'\alpha_i + Z_j'\beta_i$, and $\bar{u}_j = X_j'\bar{\alpha} + Z_j'\bar{\beta}$ is the utility implied by mean preference coefficients across customers. Equivalently, $\bar{u}_j$ can be interpreted as a baseline score that captures demand on average and thus proxies for popularity. The parameter $\omega^g \in [0, 1]$ governs the degree of personalization at stage g: larger values place more weight on individual tastes, while smaller values place more weight on average tastes. In particular, $\omega^g = 1$ corresponds to a fully personalized ranking, $\omega^g = 0$ to a non-personalized ranking, and intermediate values to partial personalization. In the simulations, I implement non-personalized, partially personalized, and fully personalized variants by setting $\omega^g \in \{0, 0.5, 1\}$ at the relevant stage.

At the discovery stage, the non-personalized condition selects products from the cohort of items shown to customers who arrive around the same time, which approximates the market's "popular" products at the moment of arrival. In contrast, the personalized condition selects products from a candidate set constructed using customers with similar historical activity. In both conditions, the products are selected and ranked according to Equation (A.7). At the query stage, the platform first retrieves up to 100 products relevant to the user's query. These retrieved products are then re-ranked according to Equation (A.7).

A.9 Additional Tables and Figures

Table A.9: User-Level Treatment Effects of Personalization at Discovery Stage

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Discovery Clicks	0.421	0.490	16.545	52.351	< 0.001
Discovery Orders	0.011	0.012	6.044	7.914	< 0.001
Discovery Revenue	0.537	0.565	5.044	3.758	< 0.001
Query Clicks	3.100	2.956	-4.632	-27.764	< 0.001
Query Orders	0.0224	0.0216	-3.665	-6.674	< 0.001
Query Revenue	0.789	0.750	-5.024	-5.315	< 0.001
Total Clicks	3.517	3.443	-2.115	-13.802	< 0.001
Total Orders	0.033	0.033	-0.322	-0.687	0.492
Total Revenue	1.306	1.294	-0.962	-1.159	0.246

Notes. The table shows the treatment effects of personalized discovery stage on outcomes at the user level. I take the mean outcomes across sessions in the experiment for each user. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

Table A.10: User-Level Treatment Effects of Personalization at Query Stage

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Discovery Clicks	0.791	0.793	0.255	0.320	0.749
Discovery Orders	0.002	0.002	0.963	0.163	0.871
Discovery Revenue	0.042	0.042	-0.582	-0.058	0.954
Query Clicks	2.242	2.303	2.708	5.132	< 0.001
Query Orders	0.023	0.025	9.222	4.775	< 0.001
Query Revenue	0.812	0.873	7.582	2.264	0.024
Total Clicks	3.032	3.094	2.048	4.595	< 0.001
Total Orders	0.025	0.027	8.680	4.674	< 0.001
Total Revenue	0.854	0.915	7.148	2.212	0.027

Notes. The table shows the treatment effects of personalized query stage on outcomes at the user level. I take the mean outcomes across sessions in the experiment for each user. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

Table A.11: Treatment Effects of Personalized Query Stage on Discovery-Stage Outcomes and Total Outcomes

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Discovery Clicks	0.846	0.844	-0.300	-0.241	0.81
Discovery Orders	0.002	0.002	-1.271	-0.140	0.888
Discovery Revenue	0.045	0.045	0.968	0.060	0.952
Total Clicks	3.086	3.175	2.874	5.173	< 0.001
Total Orders	0.027	0.030	9.549	4.144	< 0.001
Total Revenue	0.883	0.982	11.231	3.098	0.002

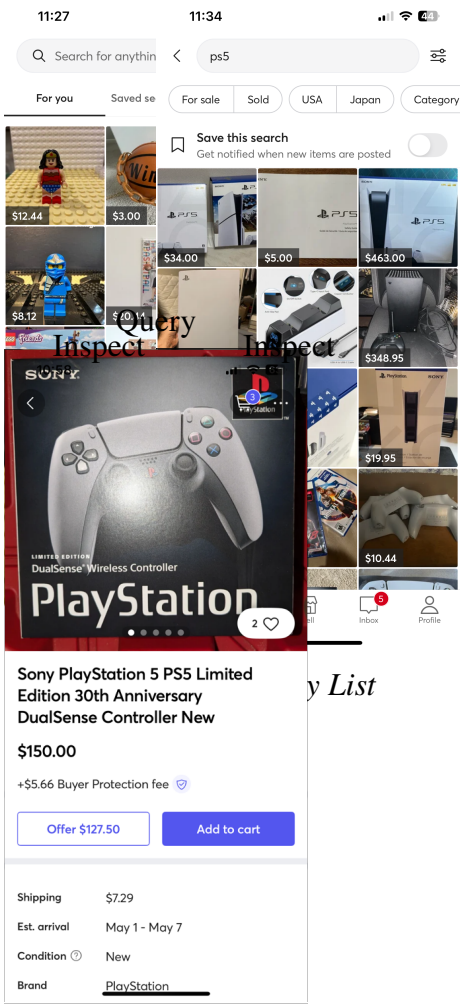
Notes. This table shows the treatment effects of the personalized query stage. I report the mean of outcomes in both personalized and non-personalized conditions. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

Table A.12: Effects of Refinement on Customer Outcomes

	Orders	
	(1)	(2)
Constant	0.0459***	0.0477***
	(0.0008)	(0.0009)
Sorted	0.0443***	
	(0.0071)	
Personalized	0.0031***	0.0029**
	(0.0011)	(0.0012)
Sorted × Personalized	0.0028	
	(0.0093)	
Filtered		-0.0024
		(0.0024)
Filtered × Personalized		0.0018
		(0.0033)
Observations	187,818	187,818
R^2	0.0011	0.00005

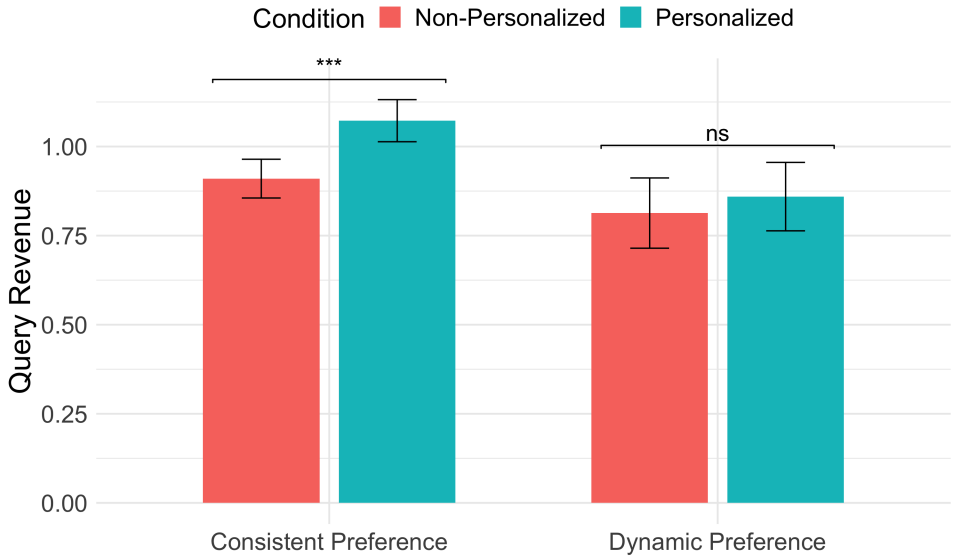
Notes. This table shows the effects of applying sorting and filters on the customer's purchase decisions. Standard errors are in parentheses, clustered at the individual level. * $p < 0.1$, ** $p < 0.5$, *** $p < 0.01$.

Figure A.14: An Illustration of Platform Layout



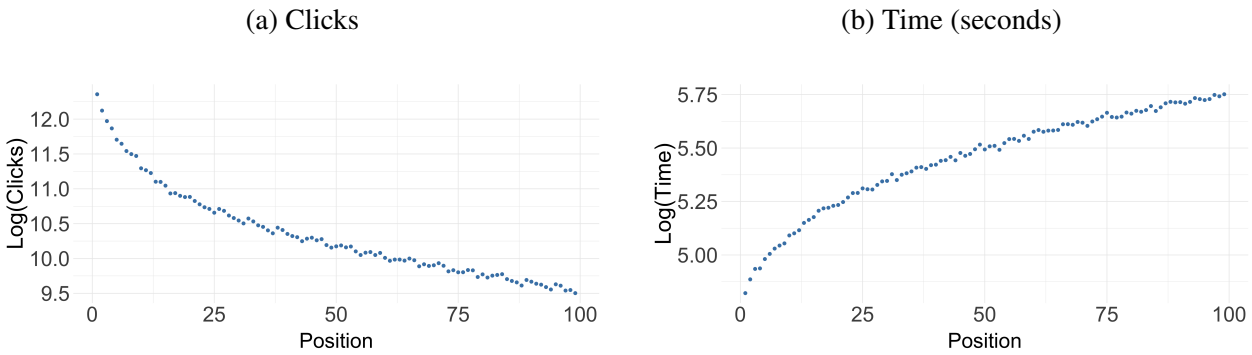
Notes. This figure shows an example of product display on Mercari in the discovery and query stages. In both lists, customers see each product's image and price. They can inspect a product by clicking on it, which takes them to the product page.

Figure A.15: Heterogeneous Effects of Personalizing Query Stage



Notes. This figure plots the average revenue at the query stage for customers in the personalized and non-personalized conditions, separately for those with consistent and dynamic preferences. The sample is restricted to customers who clicked more than one item in the 30 days prior to the personalized query stage experiment. Error bars represent 95% confidence intervals.

Figure A.16: Clicks and Time by Item Position



Notes. Panel (a) presents the logged number of clicks by query-result position; panel (b) presents the logged average time (in seconds) from the start of the session to the click.

APPENDIX B

Appendix to Chapter 2

B.1 Recreating the Suggested Price Algorithm

We lack complete information about the algorithm and training data to generate the suggested price. Thus, we can replicate it based on only observed item attributes available to us (as researchers). We predict the suggested price using the item's category, brand, physical condition, and listing day. To assess how well these item attributes can account for the suggested price, we partition our data into training (80% of the data) and validation subsets (20% of the data). We display the replication results in Table B.1. We report the R^2 from the OLS regression, Root Mean Squared Error (RMSE), and the matching rate which is calculated by comparing the predicted suggested price by our OLS model with the suggested price generated by the firm.

Table B.1: Suggested Price Reproduction

Input	R^2	RMSE	Matching Rate
category	0.29	26.94	0.35
brand	0.29	28.85	0.40
category + condition	0.29	26.78	0.36
brand + condition	0.30	28.69	0.40
category_condition	0.32	26.29	0.36
category + condition + brand	0.46	25.38	0.40
category_condition_brand	0.50	23.70	0.48
category_condition_brand + day	0.50	23.70	0.48

Notes. Matching rate is defined as the proportion of observations in the validation set where the predicted suggested price equals the true suggested price.

B.2 A Model of Seller Responses to Suggested Prices

In this section, we present a simple theoretical model to illustrate why suggested prices may impact listing prices. A key feature of the model is that it allows for variation in sellers' beliefs about the objective of the suggested pricing algorithm. We show that the effect of suggested prices on listing behavior depends on these beliefs. We consider three possible seller beliefs about the suggested price: (a) it reflects the output of an algorithm that minimizes prediction error between true selling prices and estimated prices based on historical sales data of similar items; (b) it is set to maximize the platform's revenue; and (c) it is set to maximize the platform's revenue when the platform prioritizes quick sales (i.e., an impatient platform). We focus on these three beliefs because belief (a) reflects the true objective of the algorithm, belief (b) is a plausible alternative that sellers might reasonably hold, and belief (c) has been explicitly hypothesized by sellers on online forums such as Reddit.[1]

We assume that the seller has one unit to sell. His profit π given price p is

$$\pi(p) = e^{-rt} \cdot \big((1-\alpha)p - c\big), \tag{B.1}$$

where r is the discount rate, t is the time to sell, c is the marginal cost, and α is the platform's commission rate. For simplicity, we ignore α in the seller's profit maximization, as it is fixed. Let the time to sell depend on p, such that $t = \beta * p$, where β is the mapping from price to the time to sale. We assume that $\beta > 0$, implying that the higher the price, the longer it takes to sell.

The seller's profit-maximizing price is then

$$\frac{\partial \pi}{\partial p} = e^{-r\beta p} \cdot (1 - (\beta \cdot r \cdot (p - c))) \tag{B.2}$$

$$p^* = c + \frac{1}{\beta r}. \tag{B.3}$$

The intuition is that (1) the lower the discount rate r (i.e., the more patient the seller is), the higher the price (holding all else constant), and (2) the longer it takes to sell (i.e., higher β), the lower the price (holding all else constant).

We assume that the seller does not know β with certainty, but holds a belief $\tilde{\beta}$. Therefore, the price that the seller chooses is

$$p = c + \frac{1}{\tilde{\beta} r}. \tag{B.4}$$

So far, we have not specified how suggested prices impact p. In what follows, we formalize this

[1] An example of such a Reddit post: https://www.reddit.com/r/Mercari/comments/rhyjw5/where_dose_mercari_get_their_suggested_pricing/.

relationship under each belief about the platform's objective.

B.2.1 Beliefs about Platform's Objective

Provide Accurate Information about Time to Sale Recall that Mercari provides sellers with a suggested price and a range. The suggested price is the value such that the time it takes for the product to sell is "just right." Prices greater than the maximum suggested price and lower than the minimum suggested price lead to "slow" and "fast" sales, respectively. Therefore, one can think of the suggested price as information about how long it takes for this item to sell in a reasonable amount of time. The following describes sellers' beliefs about how the suggested price is generated.

Let β^f be the platform's estimate of β. Given the platform's information about the time to sell, the platform's suggested price (when maximizing the seller's profit) is

$$p^s = c + \frac{1}{\beta^f r}. \tag{B.5}$$

Note that the seller assumes that the platform's suggested prices are determined based on the seller's marginal cost and discount rate. This implies that the seller believes that the platform knows his marginal cost and discount rate, which is a strong assumption. The assumption is more likely to hold when the seller has priced many items on Mercari, as the platform then has more data on the seller.

From Equation (B.5), the seller can then back out β^f, such that $\beta^f = \frac{1}{(p^s - c)r}$. This inference implicitly assumes that the seller believes the platform's objective is to maximize the average seller's profit. The seller then updates his beliefs $\tilde{\beta}$ in the following manner:

$$\tilde{\beta} = (1 - \lambda)\beta^0 + \lambda\beta^f, \tag{B.6}$$

where β^0 is seller's own estimate of β. We choose this functional form for simplicity. Alternatively, one could model belief updating using a Bayesian approach. The parameter λ denotes how much weight the seller puts on the platform's signal. λ can vary based on several factors, such as the seller's experience (e.g., more experienced sellers put more weight on his own information), and how accurate the seller believes the platform's estimate of his discount rate and costs to be. Then, the seller's listing price is

$$p = c + \frac{1}{\left((1 - \lambda)\beta^0 + \frac{\lambda}{(p^s - c)r)}\right)r}. \tag{B.7}$$

Increase Platform Revenue We assume that sellers believe the platform suggests the price that maximizes platform revenue. We abstract away from the possibility that the platform adjusts suggested prices strategically based on how sellers respond to them. For example, suppose a seller

tends to list items at a 20% markup over the suggested price. The platform could anticipate this behavior and intentionally set a lower suggested price, so that the seller's final listing price aligns with the true revenue-maximizing price. We assume that the seller does *not* consider this type of adjustment in their beliefs. In other words, the seller believes that the platform's suggested price is the true revenue-maximizing price.

The platform's revenue per unit is

$$\pi(p) = e^{-r^f t} \cdot (\alpha \cdot p), \tag{B.8}$$

where r^f is the platform's discount factor, t is the time to sale (t is very large if the item is never sold), and α is the platform's commission rate. We again specify $t = \beta \cdot p$. The platform's revenue-maximizing price, which is also the suggested price, is

$$\frac{\partial \pi}{\partial p} = \alpha e^{-r^f \beta p}(1 - r^f \beta p), \tag{B.9}$$

$$p^s = \frac{1}{\beta r^f}. \tag{B.10}$$

The more impatient the platform is (higher r^f) and the longer it takes to sell (higher β), the lower the price.

Given this belief about the objective function and the observed p^s, the seller infers that $\beta^f = \frac{1}{p^s r^f}$. He updates $\tilde{\beta}$ accordingly and sets the following listing price:

$$\tilde{\beta} = (1 - \lambda)\beta^0 + \lambda \frac{1}{p^s r^f}, \tag{B.11}$$

$$p = c + \frac{1}{\left((1 - \lambda)\beta^0 + \lambda \frac{1}{p^s r^f}\right) r}. \tag{B.12}$$

Prioritize Quick Sales The seller might believe that the platform is impatient (i.e., has a high discount factor). This belief is consistent with anecdotal evidence from online forums, where sellers often describe suggested prices as "too low."[2] Under this belief, the seller assumes the platform's objective follows the same structure as Equation (B.10), but with a higher discount rate r^f, which suggests that the platform is more impatient.

[2]Example: https://www.reddit.com/r/Mercari/comments/kdeqtt/how_do_we_all_feel_about_the_suggested_price_and/

B.2.2 Seller Competition

Platform suggested prices can impact seller pricing by influencing their beliefs about competitors' prices. To formalize this mechanism, we consider an extension with a focal seller i selling good j competing against a single rival k selling an identical product. Both sellers set prices simultaneously, so that when seller i chooses his price, he does not know seller k's price. Seller i interprets the platform-suggested price as a noisy but informative signal of the competitor's intended price, with the logic being that he assumes that k will generally follow the suggested price. For simplicity, we assume that i does not take into account that k may adopt the same strategy as i.

Let i believe that k's selling price is

$$p_k = p^s + \epsilon. \tag{B.13}$$

We assume that ϵ is mean zero such that $E[p_k] = p^s$. To account for seller competition, we modify seller i's profit function to the following:

$$\pi(p_i, p_k) = \big((1-\alpha)\cdot p_i - c\big)\cdot D_i(p_i, p_k), \tag{B.14}$$

where $D_i(p_i, p_k)$ is a linear demand function:

$$D(p_i, p_k) = \delta - \gamma p_i + \psi p_k + \phi(p_k)\cdot(p_k - p_i). \tag{B.15}$$

The function $\phi(p_k)$ captures how undercutting a competitor's price can affect demand. If $\phi(p_k) > 0$, the greater the increase in demand from undercutting seller k's price. The extent of the marginal effect on demand can depend on the other seller's price. A 10% lower price can boost demand more for higher priced goods relative to lower priced goods. In a digital marketplace setting, the marginal benefit of undercutting may come from the ranking algorithm, such as if lower priced products are promoted more or have greater visibility. Note that, for simplicity, demand is specified as a linear function, implying that multiple units can be sold. Our results can also generalize to cases when there is a single unit.

Given the demand and profit functions, the price that maximizes seller i's profit is

$$p_i = \frac{\delta + \psi p_k + \phi(p_k)p_k}{2\left(\gamma + \phi(p_k)\right)} + \frac{c}{2(1-\alpha)}. \tag{B.16}$$

Because seller i does not know p_k, he sets his price based on $E[p_k]$, which we assume equals the

suggested price p^s. Therefore, the above equation can be written as

$$p_i = \frac{\delta + \psi p^s + \phi(p^s)p^s}{2\left(\gamma + \phi(p^s)\right)} + \frac{c}{2(1-\alpha)}. \tag{B.17}$$

B.2.3 How Do Sellers Respond to Changes in Suggested Prices?

We describe how listing prices respond to suggested prices under different beliefs that sellers might hold.

Provide accurate information about time to sale. If the seller believes that the pricing algorithm's objective is to provide accurate information to sellers about β, then following from Equation (B.7), the change in the listing price given a change in the suggested price is the following:

$$\frac{\partial p}{\partial p^s} = \frac{\lambda}{\left(\lambda\beta^0 r \cdot (c - p^s) + \beta^0 r(p^s - c) + \lambda\right)^2} \tag{B.18}$$

$$= \frac{\lambda}{\left(\beta^0 \cdot r \cdot (1-\lambda) \cdot (p^s - c) + \lambda\right)^2} \; > \; 0. \tag{B.19}$$

Increase platform revenue. Under the belief that the algorithm's objective is to maximize platform revenue, then following from Equation (B.12), the change in p with respect to p^s is

$$\frac{\partial p}{\partial p^s} = \frac{\lambda}{\left(\beta^0 \cdot p^s \cdot r^f \cdot (1-\lambda) + \lambda\right)\right)^2} \; > \; 0. \tag{B.20}$$

Prioritize quick sales. The change in p with respect to p^s, under the belief that the platform prioritizes quick sales, is the same as in Equation (B.20), except that the seller believes r^f is high. When the seller believes that the more the platform discounts future revenue (i.e., r^f is high), the less suggested prices impact listing prices, holding all else constant.

When does a seller respond more to a suggested price if they believe the platform is providing accurate information, compared to if they believe the platform is maximizing its own revenue? Assuming that λ, β^0 are constant under both beliefs about the algorithm's objectives, Equation (B.19) is greater than Equation (B.20) when

$$r^f p^s > r(p^s - c). \tag{B.21}$$

This inequality holds under the following conditions:[3]

[3]Because "the platform prioritizes quick sales" is a special case of the broader belief that the platform seeks to maximize revenue, the same implications hold.

(1) If $c > 0$ and $r^f \geq r$. That is, the seller has a positive marginal cost and believes the platform is less patient (i.e., has a higher discount rate) than he is.

(2) If $c = 0$ and $r^f > r$. That is, the seller has a zero marginal cost and believes that the platform is less patient than he is.

(3) If $r^f < r$ and $p^s < \frac{rc}{r-r^f}$. That is, the seller believes the platform is more patient than he is. Furthermore, $p^s < \frac{rc}{r-r^f}$ is more likely to hold when the seller's marginal cost is high. Intuitively, under the belief that the platform provides accurate information, a high suggested price implies that the seller perceives a high marginal cost.

If $c = 0$ and $r^f = r$, then $\frac{\partial p}{\partial p^s}$ is the same under both objectives. This is because, when marginal costs are zero, the seller and the platform share the same objective: to maximize revenue. These conditions also imply that when the platform is less patient than the seller (i.e., $r^f > r$), the likelihood that Equation (B.21) holds *decreases* as the seller's marginal cost approaches zero. Intuitively, as marginal costs decline, the seller's and the platform's objectives become more aligned, since the platform's revenue becomes a constant fraction of the seller's revenue.

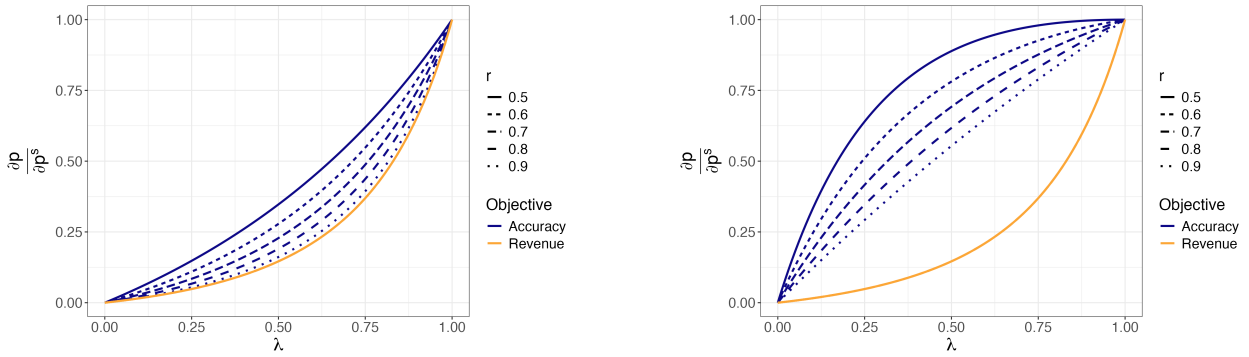
We believe that, in our setting, condition (3), where the seller believes that the platform is more patient than he is, is unlikely to hold. As stated in the main paper, Mercari relies on transaction volume to sustain its operations, whereas most sellers on the platform are not professional retailers and do not depend on sales as their primary source of income. Therefore, they can afford to wait longer, making it reasonable to assume that sellers perceive themselves as more patient than the platform.

We illustrate the effect of the suggested price on the listing price using the simulations shown in Figure B.1. We focus on cases where $r^f \geq r$ (i.e., the platform is at least as patient as the seller) for the reasons described above. The figure illustrates that responsiveness to suggested prices is higher when the seller believes the platform provides accurate information, thereby supporting condition (1). In addition, comparing the two panels, we observe that as marginal costs decrease, the effects under both belief systems become more similar, especially when the seller is equally as patient as the platform, thus illustrating condition (2).

Incorporating seller competition. We next consider how suggested prices influence listing prices when there is seller competition and the focal seller believes that other sellers choose the suggested price. Equation (B.17) shows the seller's profit-maximizing listing price. Taking the derivative with respect to p^s gives us

$$\frac{\partial p_i}{\partial p^s} = \frac{(\gamma + \phi(p^s)) \cdot (\psi + \phi'(p^s)p^s + \phi(p^s)) - \phi'(p^s) \cdot (\delta + \psi p^s + \phi(p^s)p^s)}{2(\gamma + \phi(p^s))^2}. \tag{B.22}$$

Figure B.1: Model Simulations: Beliefs about the Platform's Objective



(a) Low Marginal Cost ($c = 0.1$)

(b) High Marginal Cost ($c = 1$)

Notes: This figure plots the derivative of the listing price with respect to the suggested price under the beliefs about objectives that the platform provides accurate information about the time to sell and the platform maximizes its own revenue (Equations (B.18) and (B.20), respectively). In both panels, $\beta^0 = 2$, $p^s = 1.5$, $r^f = 0.9$.

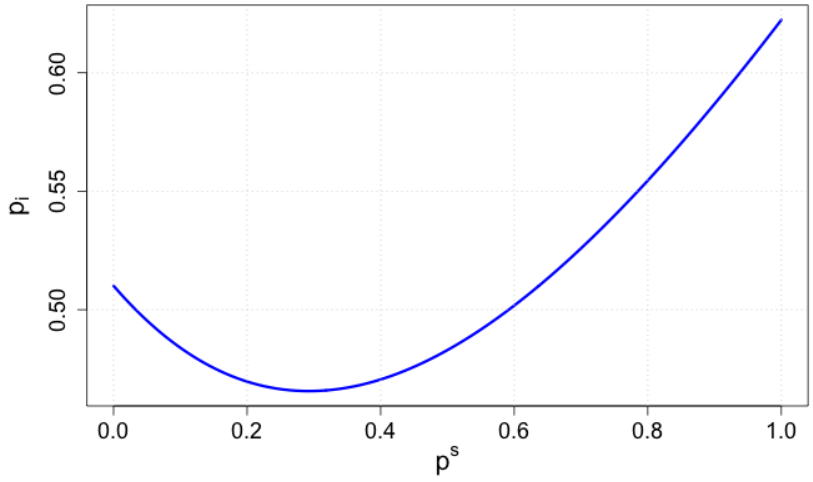
The above equation shows that when $\phi'(p^s)$ is large enough, the derivative of p_i with respect to p^s can be negative, implying that higher suggested prices decrease seller i's price. Figure B.2 presents a numerical example where p_i can decrease with p^s.

A large $\phi'(p^s)$ means that the benefit from undercutting increases with the suggested price. In digital marketplaces like Mercari where products can be ordered by price, this marginal benefit can arise if improved rankings and visibility with lower prices, and if improved rankings are especially beneficial for products with higher suggested prices. Intuitively, when sellers undercut by greater amounts, they get less profit per sale but also receive a larger boost in demand due to greater visibility and a lower price. When the returns to visibility rise quickly with the suggested price (as captured by a steep $\phi'(p^s)$), the demand gain from undercutting can outweigh the lost profit.

Overall, the model (without seller competition) demonstrates that regardless of the seller's belief about the algorithm's objective, sellers who place less weight on their private information tend to increase their listing prices in response to an increase in suggested prices. The extent to which listing prices respond to suggested prices depends on the seller's beliefs about the pricing algorithm's objective. In addition, once the model accounts for seller competition, there are areas in the parameter space that lead to decreases in listing prices with increasing suggested prices.[4]

[4]Although the model can rationalize this behavior, we find no empirical evidence that listing prices decrease with the suggested price.

Figure B.2: Listing Price as a Function of Suggested Prices under Seller Competition



Notes. The figure plots p_i given a suggested price p^s as specified in Equation (B.17), where $\phi(p^s) = e^{2p^s}$, $\gamma = 0.1$, $\psi = 0.1$, $\delta = 1$, $c = 0.1$, and $\alpha = 0.1$.

B.3 Data Construction

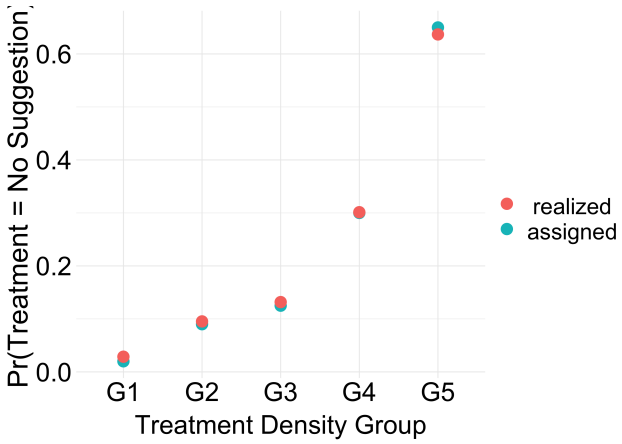
Our main dataset includes all items created by sellers in the experiment during the experimental period. We exclude items with a listing price exceeding $500 (these account for 0.8% of the total listings). Furthermore, we remove 148 items from NoSP condition that received a suggested price as these items were incorrectly assigned to their treatment conditions. Our results remain robust to including these items. Additionally, we drop items for which we do not observe the usual suggested price (i.e., the suggested price that the seller would have seen without the experiment), which constitute 0.2% items. Moreover, there are 3,930 (0.86%) items whose treatment saturation level is missing, and we drop these observations.

Following the data cleaning rules outlined above, our main sample contains 5,220,422 items, with 450,541 of them being the first items created by sellers in our experiment.

B.4 Manipulation Checks

We check whether the treatment saturation groups produced variation in the proportion of users in the "No Suggestion" group. Figure B.3 plots the percentage of sellers in the "No Suggestion" group for each treatment saturation group. We can observe that the realized treatment saturation matches closely with our predetermined treatment saturation.

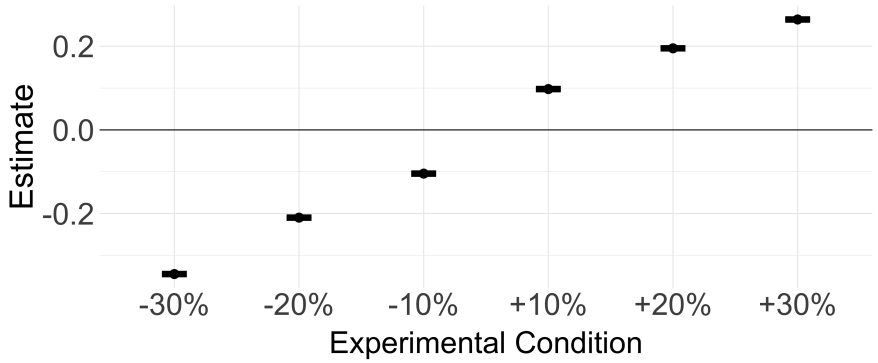
Figure B.3: Assigned and Realized Treatment Saturation



Notes. This figure plots the assigned and realized probability of a seller being randomized into the "No Suggestion" group across treatment saturation groups.

In addition, as shown in Figure B.4, in each experimental condition, the treated suggested price (i.e., the suggested price that a seller sees) is the desired ratio to the usual suggested price.

Figure B.4: Realized Deviation in Suggested Price across Experimental Conditions



B.5 Pricing Schemes Comparison

In this section, we detail how we compute the average revenue per item under decentralized pricing, platform-suggested prices, and centralized pricing. Using the measured treatment effects, we do a back-of-the-envelope calculation to compare the platform's revenue under each pricing regime. This involves estimating the causal impact of listing prices on whether the item sells, which can be done by using the experimental conditions as instruments for the listing price.[5] The first- and second-stage equations are reported below, respectively.

$$log(LP_{ij}) = \sum_{t \in T} \alpha_t I_{i,t} + \tau_{cat} + \tau_{cond} + \zeta_j \tag{B.23}$$

$$Sold_{ij} = \beta \widehat{log(LP_{ij})} + \tau_{cat} + \tau_{cond} + \epsilon_j, \tag{B.24}$$

where $I_{i,t}$ is the indicator for whether seller i—who creates the listing for item j—is in experimental condition t, τ_{cat} is the category of item j, τ_{cond} is the physical condition of item j, LP_{ij} is the listing price of item j, and $Sold_{ij}$ is an indicator for whether the item j is sold in seven days. Using the estimates of β, we can predict the revenue for an item j by $\widehat{Revenue}_{ij} = \widehat{Sold}_{ij} \times LP_{ij}$.

The NoSP condition allows us to calculate the average revenue per item under a decentralized pricing scheme, as sellers in this condition do not receive any price guidance. We calculate the revenue per item of the first items created during the experiment by sellers in this condition.

The other conditions allow us to calculate the average revenue per item under the suggested and centralized pricing schemes. The revenue per item under the suggested pricing scheme is calculated by using the observed listing prices. The revenue per item under the centralized scheme is calculated by assuming that the item's listing price is the treated suggested price. We emphasize that we report the average revenue per item, which implies that our calculations do not account for the possible lower seller retention under centralized pricing schemes, which has been documented by Filippas et al. (2023). Therefore, our estimates for centralized pricing should be interpreted as upper bounds.

Table B.2 Panel (A) shows that having suggested prices generates a higher average revenue per item than not having suggested prices (decentralized pricing) for all levels of suggested prices in our experiment. For new sellers, gains in revenue from suggested prices over decentralized prices are realized only at suggested prices that are below +10% of the usual suggested prices (Panel (B)). We also find that centralized pricing, regardless of the price, leads to the highest average revenue. In fact, new sellers would experience a higher percentage increase in revenue under the centralized pricing scheme (assuming no change in seller usage of the platform). Recall that listing prices are

[5]The experimental condition is randomly assigned. We assume that the experimental condition influences whether the item is sold only through its listing price. The F-statistic of the first stage is 84.1.

typically higher than the suggested price, especially for new sellers, and suggested price adoption rates are low (i.e., only 2% sellers choose the listing price to be the auto-filled suggested price). As a result, under a centralized scheme, the listing price would be reduced, which in turn increases the likelihood of sale and, consequently, revenues.

As we mentioned earlier, our centralized pricing revenue estimates are upper bounds because they do not account for reductions in seller retention. How much of a drop in seller retention does there need to be for suggested pricing to outperform centralized pricing? We can obtain this value by comparing the average revenue per item under suggested pricing to centralized pricing. Centralized pricing leads to between 56% to 58% greater revenue than suggested pricing in all suggested price levels. This means that in order for suggested pricing to outperform centralized pricing in terms of total revenue, centralized pricing must cause 56–58% of items to not be listed. Note that for these calculations, we assume that there is no differential attrition by types of items, such as by their expected revenue or price. To provide a sense of magnitude, Filippas et al. (2023) document that switching to centralized pricing decreased retention by 31%, reduced the availability of their vehicles by 19%, and canceled 20% more transactions.

These simulations demonstrate that (1) suggested pricing increases revenue over decentralized pricing, and (2) centralized pricing can yield the highest revenue for both the seller and the platform. The advantage of suggested over centralized pricing, however, lies in the fact that suggested prices do not negatively affect seller retention. This implies that suggested pricing can be an effective tool for influencing seller pricing.

Table B.2: Predicted Revenue under Different Pricing Systems

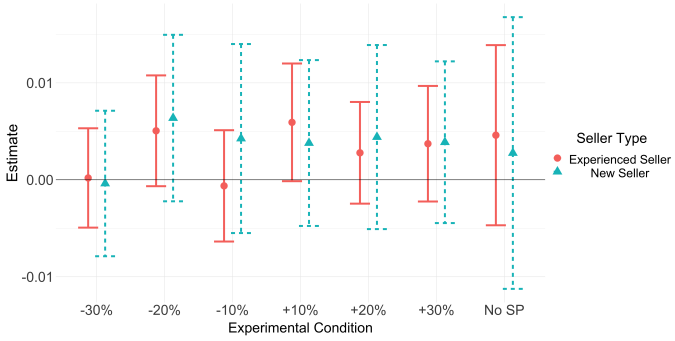
Panel (A). All Sellers		
Treatment Condition	Suggested Pricing	Centralized Pricing
Usual Suggested Price	10.346	18.152
−30% Suggested Price	11.019	18.862
−20% Suggested Price	10.663	18.094
−10% Suggested Price	10.468	17.979
+10% Suggested Price	10.193	17.472
+20% Suggested Price	9.918	16.868
+30% Suggested Price	9.9572	17.157
Treatment Condition	Decentralized Pricing	
No Suggested Price	9.868	
Panel (B). New Sellers		
Treatment Condition	Suggested Pricing	Centralized Pricing
Usual Suggested Price	11.576	24.388
−30% Suggested Price	12.007	25.361
−20% Suggested Price	11.749	24.626
−10% Suggested Price	11.590	24.378
+10% Suggested Price	11.435	22.982
+20% Suggested Price	11.162	22.815
+30% Suggested Price	11.269	23.686
Treatment Condition	Decentralized Pricing	
No Suggested Price	11.419	

B.6 Testing for Interference

This section provides additional results that support the "Testing for Interference" section in the main paper.

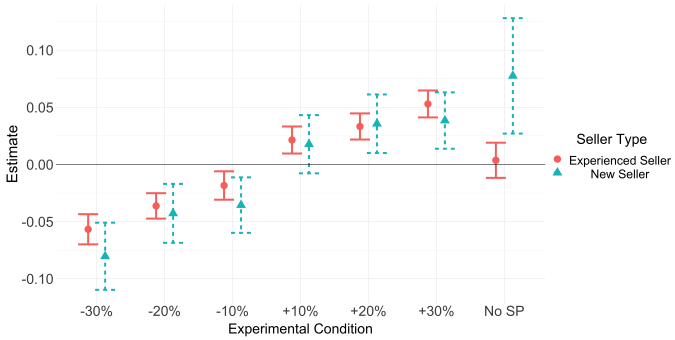
B.6.1 Seller Search Behaviors

Figure B.5: Treatment Effect on Whether Seller Clicked on Any Other Items



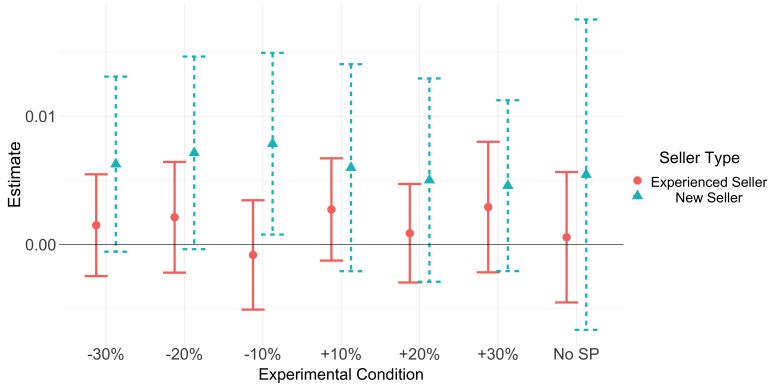
Notes. The dependent variable is an indicator for whether the seller clicked on any listings in the 24 hours before creating their first listing in the experiment. The baseline is the UsualSP condition. Error bars represent 95% confidence intervals.

Figure B.6: Treatment Effects on Listing Price: No Items in the Same Category Clicked Before Listing Creation



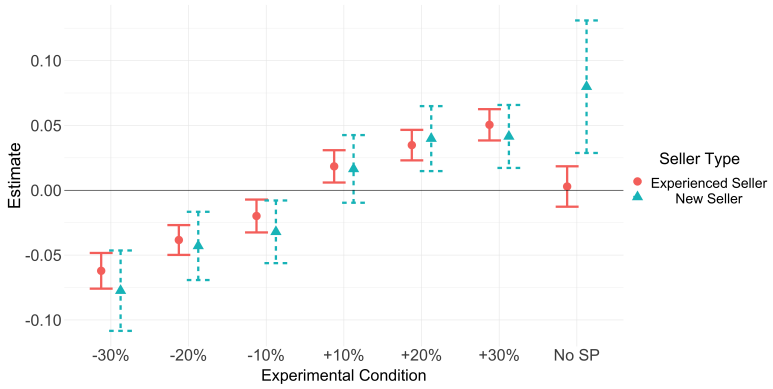
Notes. The dependent variable is the logged listing price. The baseline is UsualSP condition. Error bars represent 95% confidence intervals. These effects are estimated using a subsample of sellers who, within the 24 hours before creating their first listing in the experiment, did not click on any listings within the same category as their first listing.

Figure B.7: Treatment Effects on Search the Same Category before Listing



Notes. This figure presents the treatment effects on seller searching behaviors of the same category 24 hours before the listing creation. We define a seller as searching within the same category if any of the first 12 items in the search results belong to the same category as the item listed by them. Error bars represent 95% confidence intervals.

Figure B.8: Treatment Effects on Listing Price on Sellers Who Do Not Search the Same Category



Notes. This figure presents the treatment effects on the logged listing price, focusing on sellers who do not search for items in the same category as their own listing within 24 hours before the listing creation. We define a seller as searching within the same category if any of the first 12 items in the search results belong to the same category as the item listed by them. Error bars represent 95% confidence intervals.

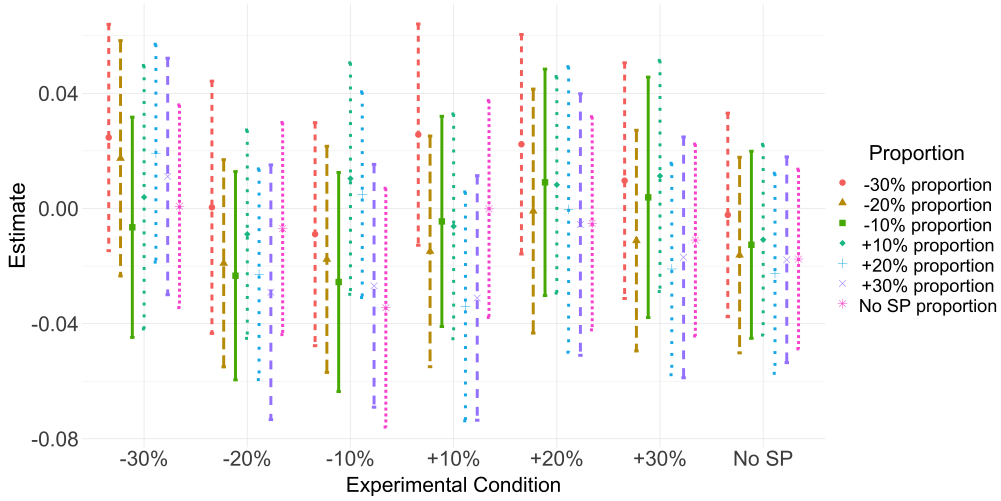
B.6.2 Buyers' Consideration Sets

Table B.3: Experimental Condition Co-Occurrence Matrix

Experimental Condition	−30%	−20%	−10%	0%	+10%	+20%	+30%	No SP
−30%	0.1105	0.1094	0.1092	0.1093	0.1099	0.1073	0.1071	0.2373
−20%	0.1110	0.1102	0.1094	0.1082	0.1111	0.1066	0.1091	0.2345
−10%	0.1101	0.1082	0.1082	0.1095	0.1100	0.1073	0.1081	0.2386
0%	0.1107	0.1088	0.1094	0.1094	0.1114	0.1056	0.1083	0.2363
+10%	0.1087	0.1096	0.1088	0.1093	0.1092	0.1090	0.1096	0.2358
+20%	0.1101	0.1095	0.1086	0.1092	0.1087	0.1083	0.1081	0.2376
+30%	0.1102	0.1091	0.1093	0.1090	0.1101	0.1085	0.1072	0.2366
No SP	0.1077	0.1078	0.1068	0.1071	0.1080	0.1066	0.1062	0.2500

Notes. Rows are the item's experimental condition, and columns are the other items' experimental condition. Element $\{i, j\}$ is the average probability that an item in condition i co-occurs with at least one item in condition j.

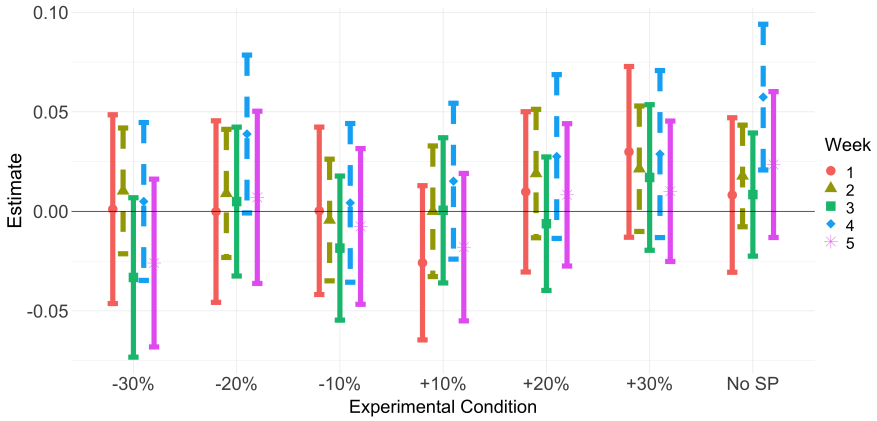
Figure B.9: Heterogeneity in the Treatment Effect on Whether Sold in Seven Days by Other Items' Experimental Conditions



Notes. This figure reports the OLS estimates of $\beta_2^{t,\tau}$ of Equation (2.13). The dependent variable is whether the item is sold in seven days. All effects are relative to UsualSP condition. Error bars represent 95% confidence intervals.

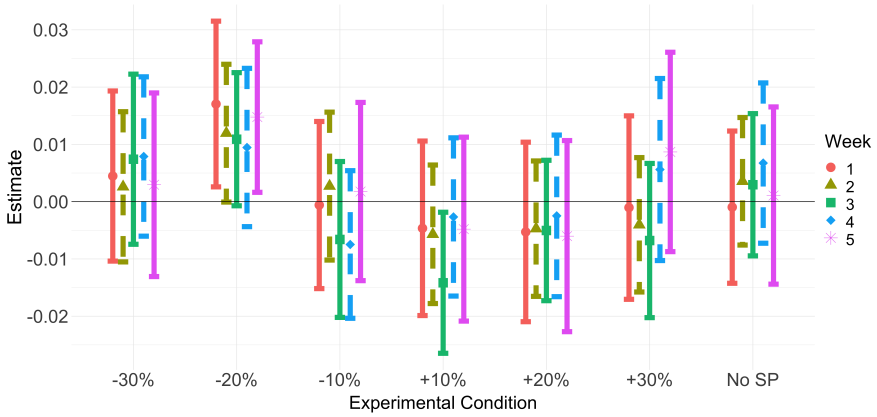
B.6.3 Treatment Effects by Weeks

Figure B.10: Treatment Effect on Listing Price by Week



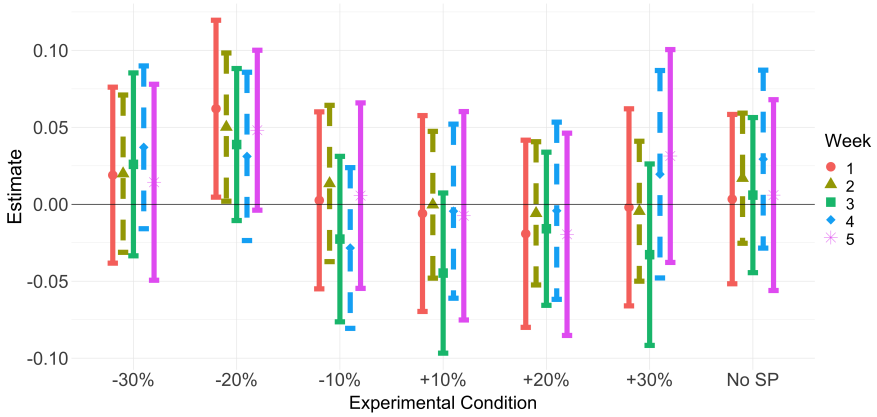
Notes. The dependent variable is the listing price (logged). The baseline is UsualSP condition. Error bars represent 95% confidence intervals.

Figure B.11: Treatment Effect on Whether Sold in Seven Days by Week



Notes. The dependent variable is whether sold in seven days. The baseline is UsualSP condition. Error bars represent 95% confidence intervals.

Figure B.12: Treatment Effect on Seven-Day Revenue by Week



Notes. The dependent variable is the logged seven-day revenue per item. The baseline is UsualSP condition. Error bars represent 95% confidence intervals.

B.7 Additional Figures and Tables

Table B.4: Correlation between Price Suggestion and Likelihood of Sale in 14 and 28 Days

	I(Sold in 14 Days) (1)	I(Sold in 28 Days) (2)
I(Has Price Suggestion)	0.0165	0.0215
	(0.0015)	(0.0016)
	[< 0.001]	[< 0.001]
Item Category × Physical Condition FE	Yes	Yes
Observations	1,407,287	1,407,287
R^2	0.1689	0.1774

Notes. This table reports the OLS estimates of Equation (2.1). The dependent variables are indicators of whether the item is sold within 14 and 28 days. Standard errors in parentheses, clustered at the seller level. P-values in square brackets.

Table B.5: Distribution of Suggested Prices for Listings in Top Categories

p10	p25	p50	p75	p90	N Items
9	12	16	24	37	52,965
11	15	23	35	60	51,051
11	14	19	29	48	37,728
28	40	70	121	172	36,039
12	16	20	34	55	35,156
10	11	14	19	30	34,986
8	11	19	33	53	32,756
9	9	11	14	19	30,562
8	9	10	13	19	28,489
10	12	15	21	31	21,263

Notes. This table reports the 10th, 25th, 50th, 75th, and 90th percentiles of suggested prices for the top 10 categories with the largest number of items in the preexperiment sample. The last column indicates the number of items in each category. The table shows that, for instance, in the most popular category, 50% of suggested prices fall between $12 and $24, with a median of $20. In the experiment, prices would lie within a range of $14 to $26 for this median price point. Similarly, in the second most popular category, 50% of suggested prices range from $15 to $35, with a median of $23. In the experiment, prices for the median listing would fall between $16 and $30.

Table B.6: Treatment Effects on Whether Listing Price is Within Suggested Price Range

	Within Suggested Price Range						
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
−30%SP	0.0369						
	(0.0030)						
	[< 0.001]						
−20%SP		0.0382					
		(0.0030)					
		[< 0.001]					
−10%SP			0.0245				
			(0.0029)				
			[< 0.001]				
UsualSP				0.0759			
				(0.0034)			
				[< 0.001]			
+10%SP					0.0252		
					(0.0035)		
					[< 0.001]		
+20%SP						0.0307	
						(0.0030)	
						[< 0.001]	
+30%SP							0.0230
							(0.0029)
							[< 0.001]
Category FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Physical Condition FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Mean (Adopted Range)	0.14	0.14	0.19	0.22	0.33	0.30	0.32
Observations	158,572	158,906	158,602	159,084	158,999	159,184	159,036
R^2	0.2604	0.2595	0.2555	0.2614	0.2635	0.2636	0.2699

Notes. This table reports OLS estimates of this equation: $y_i = \beta * I_{it} + \alpha_{cat} + \alpha_{itemcond} + \epsilon_i$. The dependent variable is an indicator of whether seller i's listing price falls within the suggested price range. The independent variable I_{it} is an indicator for whether i is in the experimental condition t. The omitted condition is the NoSP condition. We estimate this equation for each of the conditions relative to the NoSP condition. For example, Column 1 includes observations from "−30% Suggested Price" and NoSP conditions. We observe the usual suggested price and its range for NoSP condition, so that we can observe whether the listing price falls within the usual suggested price range. Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Table B.7: Treatment Effects on Listing Price and Selling Outcomes

	log(Listing Price) (1)	I(Sold in Seven Days) (2)	log(Seven-Day Revenue) (3)
Experienced Seller	-0.2348	0.0123	0.0149
	(0.0252)	(0.0061)	(0.0253)
	[< 0.001]	[0.0438]	[0.5559]
Experienced Seller × −30%	-0.0580	0.0155	0.0495
	(0.0064)	(0.0024)	(0.0101)
	[< 0.001]	[< 0.001]	[< 0.001]
Experienced Seller × −20%	-0.0347	0.0097	0.0321
	(0.0059)	(0.0022)	(0.0087)
	[< 0.001]	[< 0.001]	[< 0.001]
Experienced Seller × −10%	-0.0183	0.0063	0.0199
	(0.0064)	(0.0025)	(0.0098)
	[0.0043]	[0.0119]	[0.0424]
Experienced Seller × +10%	0.0195	-0.0007	0.0011
	(0.0057)	(0.0026)	(0.0100)
	[< 0.001]	[0.7877]	[0.9124]
Experienced Seller × +20%	0.0342	-0.0038	-0.0093
	(0.0055)	(0.0023)	(0.0090)
	[< 0.001]	[0.0985]	[0.3014]
Experienced Seller × +30%	0.0495	-0.0075	-0.0243
	(0.0056)	(0.0027)	(0.0106)
	[< 0.001]	[0.0055]	[0.0219]
Experienced Seller × No SP	0.0016	0.0016	0.0017
	(0.0077)	(0.0020)	(0.0082)
	[0.8354]	[0.4237]	[0.8358]
New Seller × −30%	-0.0821	0.0035	0.0022
	(0.0148)	(0.0057)	(0.0246)
	[< 0.001]	[0.5395]	[0.9287]
New Seller × −20%	-0.0490	0.0100	0.0222
	(0.0129)	(0.0052)	(0.0236)
	[0.0001]	[0.0550]	[0.3467]
New Seller × −10%	-0.0359	-0.0045	-0.0239
	(0.0129)	(0.0057)	(0.0248)
	[0.0054]	[0.4305]	[0.3357]
New Seller × +10%	0.0158	-0.0080	-0.0397
	(0.0128)	(0.0050)	(0.0214)
	[0.2172]	[0.1099]	[0.0633]
New Seller × +20%	0.0303	-0.0115	-0.0550
	(0.0141)	(0.0046)	(0.0198)
	[0.0317]	[0.0124]	[0.0054]
New Seller × +30%	0.0337	-0.0147	-0.0692
	(0.0124)	(0.0055)	(0.0237)
	[0.0065]	[0.0075]	[0.0035]
New Seller × No SP	0.0732	-0.0075	-0.0292
	(0.0253)	(0.0060)	(0.0254)
	[0.0038]	[0.2117]	[0.2499]
Category FE	Yes	Yes	Yes
Physical Condition FE	Yes	Yes	Yes
Observations	450,541	450,541	450,541
R^2	0.2910	0.0389	0.0445

Notes. This table reports the OLS estimates of Equation (2.8). The dependent variables are the item's listing price (logged), whether sold in seven days, and seven-day revenue (logged after adding one). Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Table B.8: Heterogeneous Treatment Effects on Listing Price

	log(Listing Price)			
	(1)	(2)	(3)	(4)
Δ^s	0.1817	0.1971	0.2053	0.2177
	(0.0100)	(0.0120)	(0.0152)	(0.0164)
	[< 0.001]	[< 0.001]	[< 0.001]	[< 0.001]
New Item		0.1400		0.1531
		(0.0287)		(0.0269)
		[< 0.001]		[< 0.001]
$\Delta^s \times$ New Item		-0.0467		-0.0457
		(0.0169)		(0.0166)
		[0.0058]		[0.0060]
log(N Past Listings)			-0.0523	-0.0537
			(0.0038)	(0.0038)
			[< 0.001]	[< 0.001]
$\Delta^s \times$ log(N Past Listings)			-0.0063	-0.0056
			(0.0030)	(0.0030)
			[0.0360]	[0.0610]
Category FE	Yes	Yes	Yes	Yes
Observations	340,234	340,234	340,234	340,234
R^2	0.2843	0.2880	0.2976	0.3020

Notes. This table reports the OLS estimates of Equation (2.9). To estimate the main effect of "New Condition", we do not include the item's physical condition fixed effect in the equation. The dependent variable is the item's listing price (logged). Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Table B.9: Heterogeneous Effects on Listing Price: Other Measures of Pricing Uncertainty

	log(Listing Price)	
	(1)	(2)
Δ^s	0.2170	0.1875
	(0.0123)	(0.0094)
	[< 0.001]	[< 0.001]
log(N Similar Products)	0.0341	
	(0.0062)	
	[< 0.001]	
$\Delta^s \times$ log(N Similar Products)	-0.0143	
	(0.0041)	
	[< 0.001]	
SoldBefore		-0.0491
		(0.0209)
		[0.0188]
$\Delta^s \times$ SoldBefore		-0.0737
		(0.0355)
		[0.0375]
Category FE	Yes	Yes
Physical Condition FE	Yes	Yes
Observations	340,234	340,234
R^2	0.2946	0.2921

Notes. This table reports the OLS estimates of Equation (2.9). The dependent variables are both the item's listing price (logged). Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Table B.10: Treatment Effects on Listing Price When Listing Price is Not Equal to Max Suggested Price

	log(Listing Price)	
	(1)	(2)
Δ^s	0.1815	0.0803
	(0.0100)	(0.0107)
	[< 0.001]	[< 0.001]
Category FE	Yes	Yes
Physical Condition FE	Yes	Yes
Observations	340,234	295,432
R^2	0.2919	0.2749

Notes. This table reports the OLS estimates of Equation (2.8) for the entire sample (Column 1) and only for items for which the listing price is not equal to the maximum treated suggested price (Column 2). The omitted experimental condition is UsualSP. Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Table B.11: Linear Treatment Effects on Selling Outcomes

	I(Sold in 7 Days)	log(Seven-Day Revenue)
	(1)	(2)
$\Delta^s \times$ Experienced Seller	-0.0368	-0.1157
	(0.0038)	(0.0149)
	[< 0.001]	[< 0.001]
$\Delta^s \times$ New Seller	-0.0363	-0.1373
	(0.0081)	(0.0360)
	[< 0.001]	[< 0.001]
Category FE	Yes	Yes
Physical Condition FE	Yes	Yes
Observations	340,234	340,234
R^2	0.0391	0.0448

Notes. This table reports the OLS estimates of the linear treatment effect $\Delta^s = \{-0.3, -0.2, .., 0.3\}$ on whether the listing sold within seven days and its revenue within seven days. We omit observations in NoSP condition because there is no linear analog of the treatment effect. Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Table B.12: Seller-Level Treatment Effects: New Sellers

	Avg Listing Price (1)	Prop. Sold (2)	Total Revenue (3)	N Listings (4)
−30%	-0.0858	0.0075	0.0542	0.0115
	(0.0126)	(0.0041)	(0.0318)	(0.0150)
	[< 0.001]	[0.0696]	[0.0897]	[0.4441]
−20%	-0.0539	0.0103	0.0350	0.0232
	(0.0121)	(0.0038)	(0.0346)	(0.0177)
	[< 0.001]	[0.0066]	[0.3124]	[0.1905]
−10%	-0.0369	-0.0021	-0.0007	0.0192
	(0.0135)	(0.0038)	(0.0340)	(0.0156)
	[0.0062]	[0.5812]	[0.9836]	[0.2195]
+10%	0.0031	-0.0028	-0.0163	0.0111
	(0.0122)	(0.0036)	(0.0301)	(0.0154)
	[0.7984]	[0.4367]	[0.5891]	[0.4703]
+20%	0.0231	-0.0107	-0.0319	0.0372
	(0.0136)	(0.0031)	(0.0302)	(0.0163)
	[0.0897]	[< 0.001]	[0.2917]	[0.0217]
+30%	0.0340	-0.0122	-0.0736	0.0282
	(0.0126)	(0.0035)	(0.0300)	(0.0165)
	[0.0068]	[< 0.001]	[0.0145]	[0.0873]
NoSP	0.0355	-0.0030	-0.0254	0.0195
	(0.0140)	(0.0029)	(0.0336)	(0.0252)
	[0.0113]	[0.2997]	[0.4492]	[0.4395]
DV Logged	Yes		Yes	Yes
Category FE	Yes	Yes	Yes	Yes
Observations	70,089	70,089	70,089	70,089
R^2	0.3161	0.0824	0.0656	0.1191

Notes. This table reports the OLS estimates of Equation (2.10) for new sellers only. The baseline is UsualSP condition. Standard errors in parentheses, clustered at the seller level. P-values in square brackets.

Table B.13: Treatment Effect of Suggested Price Level by Saturation Group

	log(Listing Price) (1)	log(Seven-Day Revenue) (2)
$\Delta^s \times$ PropNoSP = 0.02	0.1944	-0.0972
	(0.0190)	(0.0318)
	[< 0.001]	[0.0023]
$\Delta^s \times$ PropNoSP = 0.09	0.2040	-0.1213
	(0.0127)	(0.0246)
	[< 0.001]	[< 0.001]
$\Delta^s \times$ PropNoSP = 0.125	0.1964	-0.1426
	(0.0157)	(0.0317)
	[< 0.001]	[< 0.001]
$\Delta^s \times$ PropNoSP = 0.3	0.1545	-0.1218
	(0.0194)	(0.0252)
	[< 0.001]	[< 0.001]
$\Delta^s \times$ PropNoSP = 0.65	0.1949	-0.0994
	(0.0229)	(0.0320)
	[< 0.001]	[0.0019]
PropNoSP = 0.02	-0.0323	0.0735
	(0.0234)	(0.0325)
	[0.1679]	[0.0237]
PropNoSP = 0.09	-0.0338	-0.0733
	(0.0284)	(0.0351)
	[0.2350]	[0.0369]
PropNoSP = 0.125	-0.0421	0.0224
	(0.0276)	(0.0283)
	[0.1263]	[0.4265]
PropNoSP = 0.3	-0.0631	0.0728
	(0.0375)	(0.0353)
	[0.0925]	[0.0387]
log(SP)	0.8815	0.2990
	(0.0114)	(0.0287)
	[< 0.001]	[< 0.001]
Physical Condition FE	Yes	Yes
Observations	340,234	340,234
R^2	0.5169	0.0354

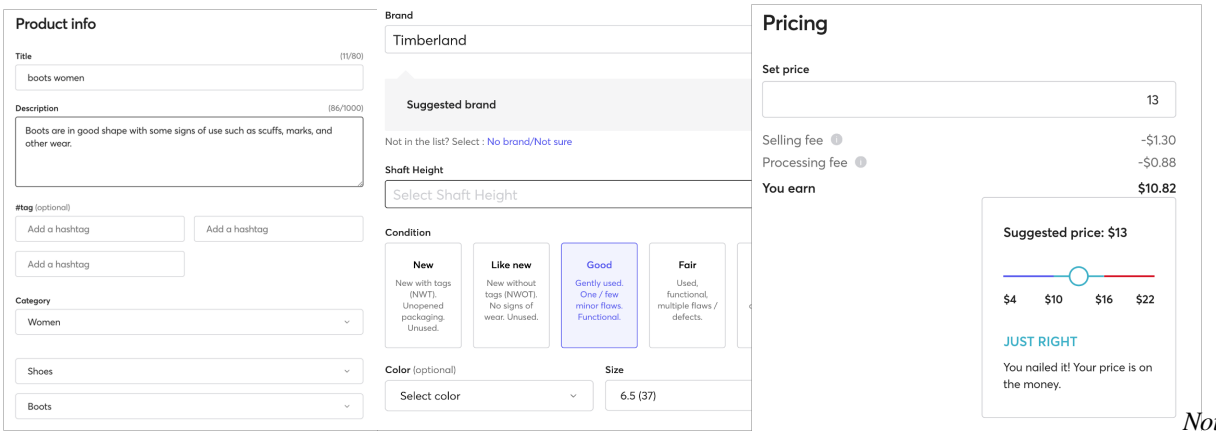
Notes. OLS estimates of Equation (2.12). The variable Δ^s denotes the continuous treatment effect, varying from -0.3 to $+0.3$. Standard errors are clustered by category. Unlike previous specifications, we omit category fixed effects because the product's category is perfectly collinear with its treatment saturation group. Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Table B.14: Treatment Effect of No Suggested Price by Treatment Saturation Group: New Sellers

	log(Listing Price) (1)
PropNoSP = 0.09	0.0187
	(0.0559)
	[0.7364]
PropNoSP = 0.125	-0.0071
	(0.0528)
	[0.8929]
PropNoSP = 0.3	-0.0373
	(0.0692)
	[0.5907]
PropNoSP = 0.65	0.0697
	(0.0786)
	[0.3757]
NoSP × PropNoSP = 0.02	0.0832
	(0.0658)
	[0.2053]
NoSP × PropNoSP = 0.09	0.1195
	(0.0309)
	[< 0.001]
NoSP × PropNoSP = 0.125	0.0566
	(0.0276)
	[0.0410]
NoSP × PropNoSP = 0.3	0.0164
	(0.0271)
	[0.5449]
NoSP × PropNoSP = 0.65	0.0717
	(0.0377)
	[0.0573]
log(SP)	0.6473
	(0.0250)
	[< 0.001]
Physical Condition FE	Yes
Observations	24,239
R^2	0.3663

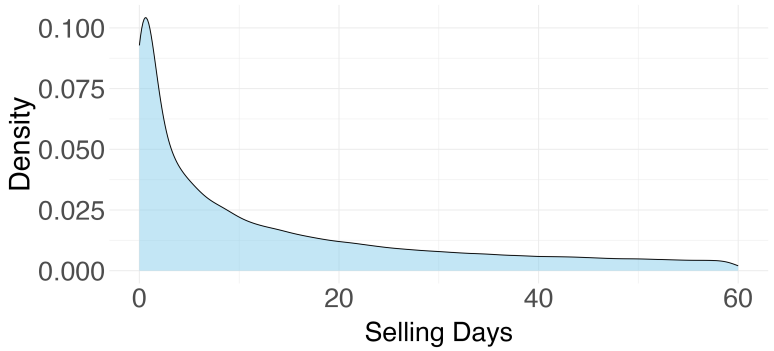
Notes. We omit category fixed effects because the product's category is perfectly collinear with its treatment saturation group. Standard errors in parentheses, clustered at the category level. P-values in square brackets.

Figure B.13: Example Listing Creation Page



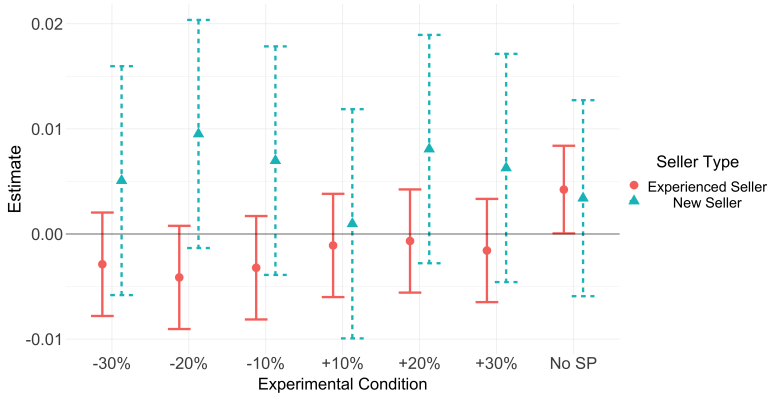
Notes: This figure displays the process for creating a listing. The seller first enters the product information, and then advances to the "Pricing" page.

Figure B.14: Distribution of Selling Time (Preexperiment Data)



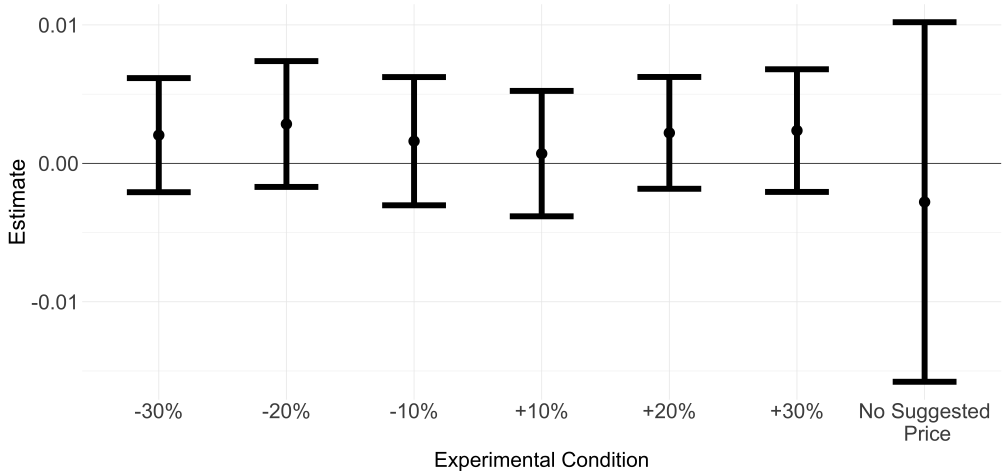
Notes. This figure plots the distribution of selling days of listings created between March 1, 2023 and March 14, 2023. We focus on products that are sold within 60 days of creation, which is the 80th percentile in the selling days.

Figure B.15: Treatment Effects on Whether Listing Creation Is Completed



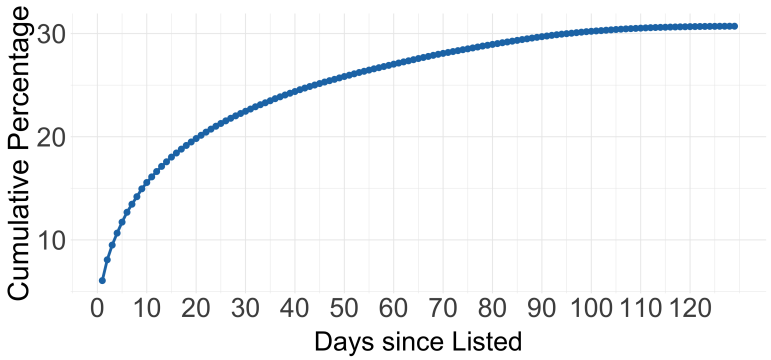
Notes. The dependent variable is an indicator for whether the seller completes their first listing. Error bars represent 95% confidence intervals.

Figure B.16: Randomization Check of Whether New Seller



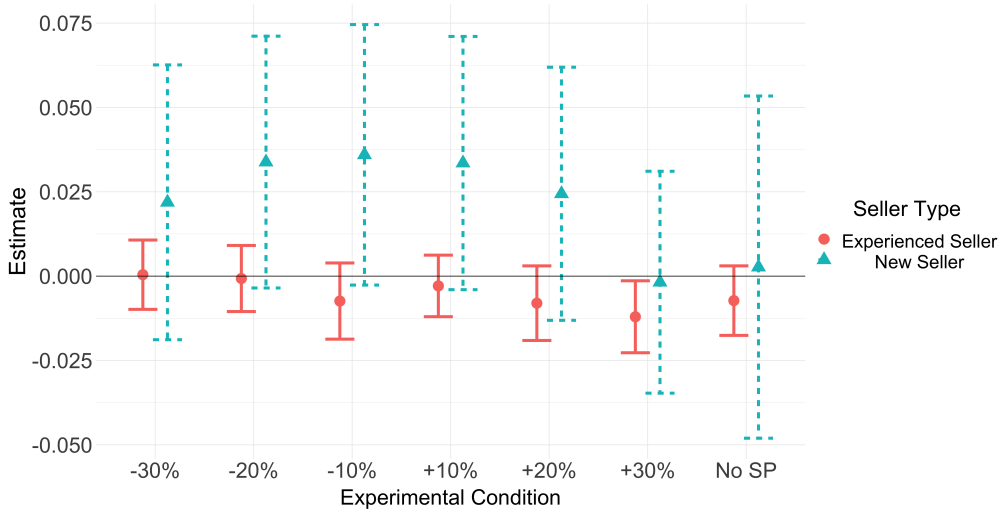
Notes. The dependent variable is a dummy variable indicating whether the seller is a new seller. Standard errors are clustered at the item category level. The error bars represent 95% confidence intervals.

Figure B.17: Relationship between Time since Listed and Fraction of Listings Sold



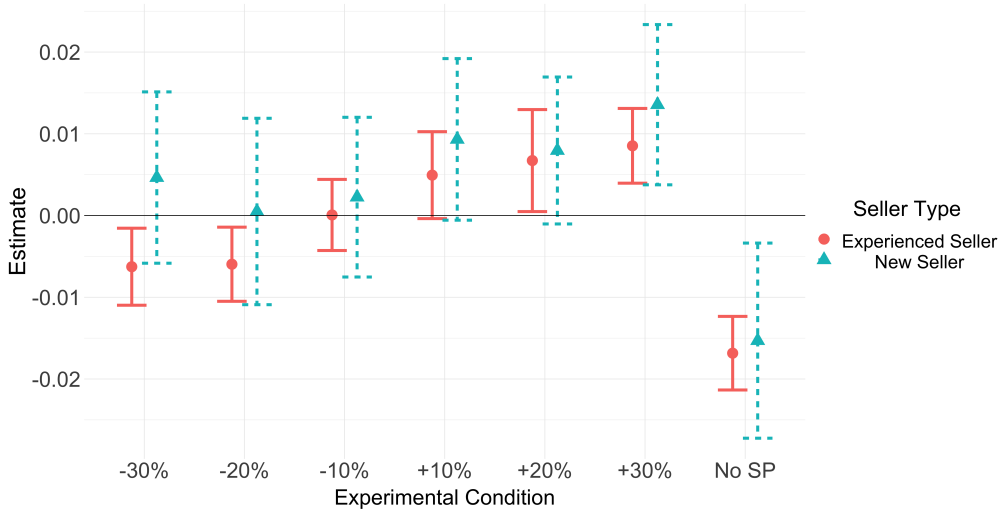
Notes. This figure displays the relationship between time since listing creation and the fraction of listings that are sold among all listings in the experiment. The x-axis is the days since listing creation and the y-axis is the percentage of items sold.

Figure B.18: Treatment Effects on Shipping Fees



Notes. This figure reports the treatment effects on shipping fees. Error bars represent 95% confidence intervals.

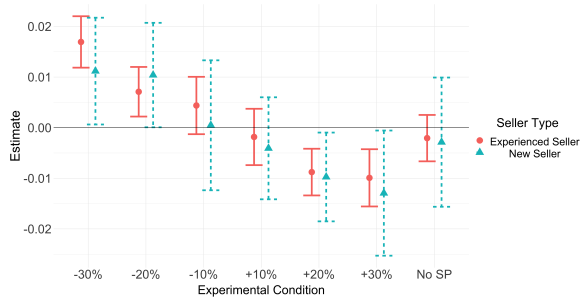
Figure B.19: Treatment Effects on Listing Price Adjustments



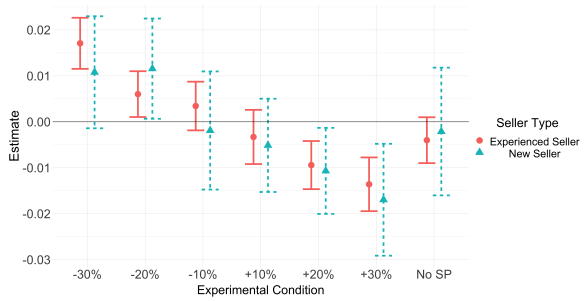
Notes. This figure reports the treatment effects on whether the seller edits their listing price in the seven days after listing creation. Error bars represent 95% confidence intervals.

Figure B.20: Treatment Effects on Item-Level Likelihood of Sale and Revenue in 14 Days and 28 Days

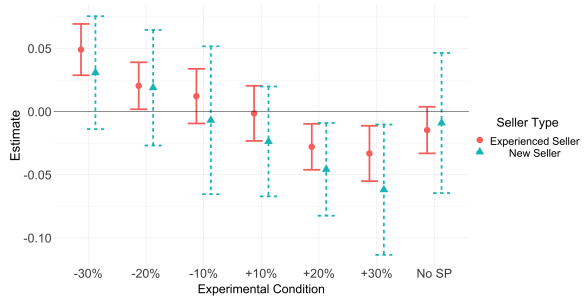
(a) 14-Day Likelihood of Sale



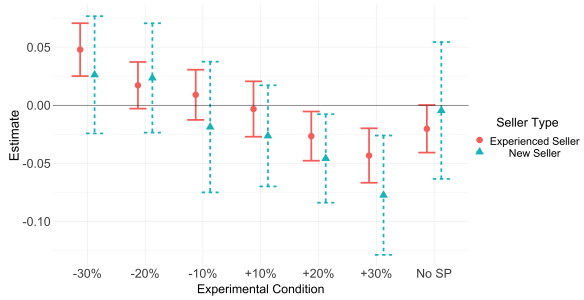
(b) 28-Day Likelihood of Sale



(c) 14-Day Revenue

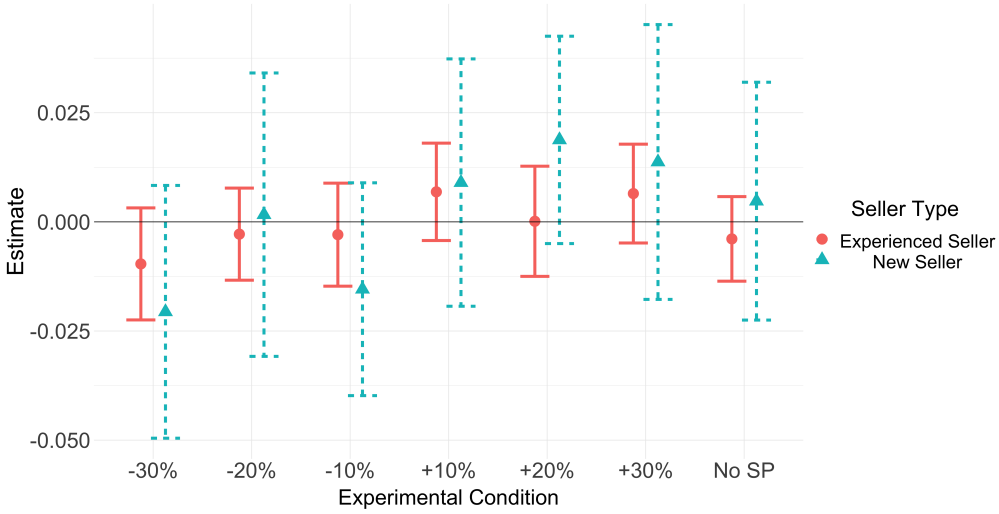


(d) 28-Day Revenue



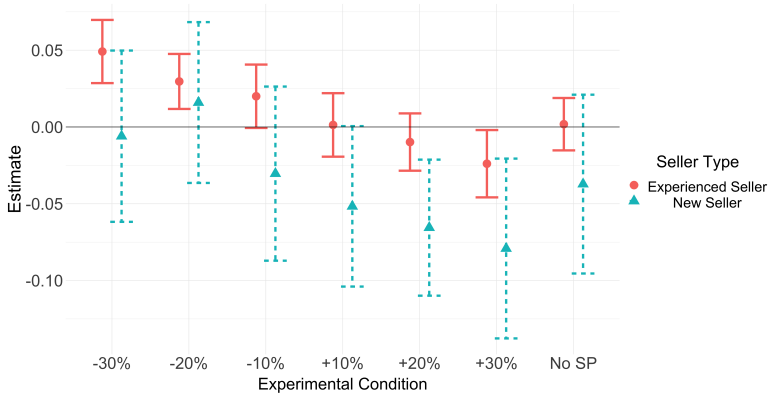
Notes. This figure reports the treatment effects on 14-day and 28-day selling outcomes. Error bars represent 95% confidence intervals.

Figure B.21: Treatment Effects on Whether the Item is Sold through Bargaining



Notes. This figure reports the treatment effects on whether the item is sold through bargaining, conditional on being sold within seven days since listing creation. Error bars represent 95% confidence intervals.

Figure B.22: Treatment Effects on Revenue (Deducting the Seller-Paid Shipping Fees)



Notes. This figure presents the treatment effects on revenue after deducting the shipping fees (if paid by the seller). Error bars represent 95% confidence intervals.

BIBLIOGRAPHY

Abraham, Mark and Edelman, David C (2024). "Personalization Done Right The five dimensions to consider and how AI can help". *Harvard Business Review*, 103 (11-12), 104–115.

Aparicio, Diego and Simester, Duncan (2022). "Price frictions and the success of new products". *Marketing Science*, 41 (6), 1057–1073.

Backus, Matthew, Blake, Thomas, and Tadelis, Steven (2019). "On the empirical content of cheap-talk signaling: An application to bargaining". *Journal of Political Economy*, 127 (4), 1599–1628.

Baird, Sarah, Bohren, J Aislinn, McIntosh, Craig, and Özler, Berk (2018). "Optimal design of experiments in the presence of interference". *Review of Economics and Statistics*, 100 (5), 844–860.

Banerjee, Abhijit, Chattopadhyay, Raghabendra, Duflo, Esther, Keniston, Daniel, and Singh, Nina (2021). "Improving police performance in Rajasthan, India: Experimental evidence on incentives, managerial autonomy, and training". *American Economic Journal: Economic Policy*, 13 (1), 36–66.

Blake, Thomas and Coey, Dominic (2014). "Why marketplace experimentation is harder than it seems: The role of test-control interference". In "Proceedings of the Fifteenth ACM conference on Economics and Computation", 567–582.

Bronnenberg, Bart J, Kim, Jun B, and Mela, Carl F (2016). "Zooming in on choice: How do consumers search for cameras online?" *Marketing Science*, 35 (5), 693–712.

Burges, Christopher JC (2010). "From ranknet to lambdarank to lambdamart: An overview". *Learning*, 11 (23-581), 81.

Burke, Marshall, Bergquist, Lauren Falcao, and Miguel, Edward (2019). "Sell low and buy high: arbitrage and local price effects in Kenyan markets". *The Quarterly Journal of Economics*, 134 (2), 785–842.

Cai, Chencheng, Pouget-Abadie, Jean, and Airoldi, Edoardo M (2022). "Optimizing randomized and deterministic saturation designs under interference". *arXiv preprint arXiv:2203.09682*.

Chen, Guangying, Chan, Tat, Zhang, Dennis, Liu, Senmao, and Wu, Yuxiang (2024). "The effects of diversity in algorithmic recommendations on digital content consumption: A field experiment". *Available at SSRN 4365121*.

Chen, Yuxin and Yao, Song (2017). "Sequential search with refinement: Model and application with click-stream data". *Management Science*, 63 (12), 4345–4365.

Chen, Zirou, Shi, Mengze, and Zhong, Zemin (Zachary) (2025). "Predictive Accuracy, Consumer Search, and Personalized Recommendation". *Available at SSRN 4298841*.

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

Compiani, Giovanni, Lewis, Gregory, Peng, Sida, and Wang, Peichun (2024). "Online search and optimal product rankings: An empirical framework". *Marketing Science*, 43 (3), 615–636.

Dang, Chu (Ivy), Ursu, Raluca, and Chintagunta, Pradeep K. (2025). "Going Back to Move Forward? How Search Revisits on a Website We Built, and in Field Data, Inform Us about Search Outcomes". *Available at SSRN 3626451*.

De los Santos, Babur and Koulayev, Sergei (2017). "Optimizing click-through in online rankings with endogenous search refinement". *Marketing Science*, 36 (4), 542–564.

Dehaene, Stanislas and Mehler, Jacques (1992). "Cross-linguistic regularities in the frequency of number words". *Cognition*, 43 (1), 1–29.

Dietvorst, Berkeley J, Simmons, Joseph P, and Massey, Cade (2015). "Algorithm aversion: people erroneously avoid algorithms after seeing them err." *Journal of Experimental Psychology: General*, 144 (1), 114.

Donnelly, Robert, Kanodia, Ayush, and Morozov, Ilya (2024). "Welfare effects of personalized rankings". *Marketing Science*, 43 (1), 92–113.

Egger, Dennis, Haushofer, Johannes, Miguel, Edward, Niehaus, Paul, and Walker, Michael (2022). "General equilibrium effects of cash transfers: experimental evidence from Kenya". *Econometrica*, 90 (6), 2603–2643.

Fang, Xing, Kim, SunAh, and Chintagunta, Pradeep K (2026). "Too many or too few? Information cues in recommender systems and consequences for search and purchase behavior". *Journal of Marketing*, 90 (1), 9–28.

Farronato, Chiara, Fradkin, Andrey, Larsen, Bradley, and Brynjolfsson, Erik (2020). "Consumer protection in an online world: An analysis of occupational licensing". Technical report, National Bureau of Economic Research.

Filippas, Apostolos, Jagabathula, Srikanth, and Sundararajan, Arun (2023). "The limits of centralized pricing in online marketplaces and the value of user control". *Management Science*.

Filmer, Deon, Friedman, Jed, Kandpal, Eeshani, and Onishi, Junko (2023). "Cash transfers, food prices, and nutrition impacts on ineligible children". *The Review of Economics and Statistics*, 105 (2), 327–343.

Fong, Jessica (2024). "Effects of market size and competition in two-sided markets: Evidence from online dating". *Marketing Science*, 43 (5), 971–985.

Fong, Jessica, Manchanda, Puneet, and Song, Yu (2026). "How Effective is Suggested Pricing?: Experimental Evidence from an E-Commerce Platform". *Available at SSRN 4993457*.

Fong, Jessica, Natan, Olivia, and Pantle, Ranmit (2024). "Consumer Inferences from Product Rankings: The Role of Beliefs in Search Behavior". *Available at SSRN*.

Fong, Nathan M (2017). "How targeting affects customer search: A field experiment". *Management Science*, 63 (7), 2353–2364.

Foroughifar, Mohsen and Mehta, Nitin (2023). "The challenges of deploying an algorithmic pricing tool: Evidence from Airbnb". *University of Toronto Working Paper*.

Fradkin, Andrey, Bhole, Monica, and Horton, John (2023). "Competition avoidance vs herding in job search: evidence from large-scale field experiments on an online job board".

Gaineddenova, Renata (2022). "Pricing and efficiency in a decentralized ride-hailing platform". *Working Paper*.

Gandhi, Amit and Houde, Jean-François (2019). "Measuring substitution patterns in differentiated-products industries". *NBER Working paper*, (w26375).

Ghose, Anindya, Ipeirotis, Panagiotis G, and Li, Beibei (2012). "Designing ranking systems for hotels on travel search engines by mining user-generated and crowdsourced content". *Marketing Science*, 31 (3), 493–520.

Ghose, Anindya, Ipeirotis, Panagiotis G, and Li, Beibei (2019). "Modeling consumer footprints on search engines: An interplay with social media". *Management Science*, 65 (3), 1363–1385.

Giné, Xavier and Mansuri, Ghazala (2018). "Together we will: experimental evidence on female voting behavior in Pakistan". *American Economic Journal: Applied Economics*, 10 (1), 207–235.

Goić, Marcel, Jerath, Kinshuk, and Kalyanam, Kirthi (2022). "The roles of multiple channels in predicting website visits and purchases: Engagers versus closers". *International Journal of Research in Marketing*, 39 (3), 656–677.

Goli, Ali, Lambrecht, Anja, and Yoganarasimhan, Hema (2023). "A bias correction approach for interference in ranking experiments". *Marketing Science*.

Greminger, Rafael P (2022). "Optimal search and discovery". *Management Science*, 68 (5), 3904–3924.

Gu, Chris and Wang, Yike (2022). "Consumer online search with partially revealed information". *Management Science*, 68 (6), 4215–4235.

Gu, Wei, Luo, Jing, Yu, Xiaoru, Zhang, Wenqing, and Li, Baixun (2023). "Dynamic decisions between sellers and consumers in online second-hand trading platforms: Evidence from C2C transactions". *Transportation Research Part E: Logistics and Transportation Review*, 177, 103257.

Hodgson, Charles and Lewis, Gregory (2025). "You can lead a horse to water: Spatial learning and path dependence in consumer search". *Econometrica*, 93 (4), 1299–1332.

Holtz, David and Aral, Sinan (2020). "Limiting bias from test-control interference in online marketplace experiments". *arXiv preprint arXiv:2004.12162*.

Holtz, David, Carterette, Ben, Chandar, Praveen, Nazari, Zahra, Cramer, Henriette, and Aral, Sinan (2020). "The engagement-diversity connection: Evidence from a field experiment on spotify". In "Proceedings of the 21st ACM Conference on Economics and Computation", 75–76.

Holtz, David, Lobel, Felipe, Lobel, Ruben, Liskovich, Inessa, and Aral, Sinan (2024). "Reducing interference bias in online marketplace experiments using cluster randomization: Evidence from a pricing meta-experiment on Airbnb". *Management Science*.

Honka, Elisabeth, Hortaçsu, Ali, and Vitorino, Maria Ana (2017). "Advertising, consumer awareness, and choice: Evidence from the US banking industry". *The RAND Journal of Economics*, 48 (3), 611–646.

Honka, Elisabeth, Hortaçsu, Ali, and Wildenbeest, Matthijs (2019). "Empirical search and consideration sets". In "Handbook of the Economics of Marketing", volume 1, 193–257. Elsevier.

Honka, Elisabeth, Seiler, Stephan, and Ursu, Raluca (2024). "Consumer search: What can we learn from pre-purchase data?" *Journal of Retailing*, 100 (1), 114–129.

Hosanagar, Kartik, Fleder, Daniel, Lee, Dokyun, and Buja, Andreas (2014). "Will the global village fracture into tribes? Recommender systems and their effects on consumer fragmentation". *Management Science*, 60 (4), 805–823.

Huang, Yufeng (2022). "Pricing frictions and platform remedies: the case of Airbnb". *Available at SSRN 3767103*.

Huang, Yufeng, Ellickson, Paul B, and Lovett, Mitchell J (2022). "Learning to set prices". *Journal of Marketing Research*, 59 (2), 411–434.

Hudgens, Michael G and Halloran, M Elizabeth (2008). "Toward causal inference with interference". *Journal of the american statistical association*, 103 (482), 832–842.

Hui, Xiang, Saeedi, Maryam, Spagnolo, Giancarlo, and Tadelis, Steven (2023). "Raising the bar: Certification thresholds and market outcomes". *American Economic Journal: Microeconomics*, 15 (2), 599–626.

Imbens, Guido W and Rubin, Donald B (2015). *Causal Inference in Statistics, Social, and Biomedical Sciences*. Cambridge University Press.

Jameei Osgouei, Ata, Ching, Andrew T, Ratchford, Brian T, and Shahrokhi Tehrani, Shervin (2026). "Estimating Position and Social Influence Effects in Online Search". *Marketing Science*, 45 (1), 203–223.

Jiang, Zhenling, Chan, Tat, Che, Hai, and Wang, Youwei (2021). "Consumer search and purchase: An empirical investigation of retargeting based on consumer online behaviors". *Marketing Science*, 40 (2), 219–240.

Johari, Ramesh, Li, Hannah, Liskovich, Inessa, and Weintraub, Gabriel Y (2022). "Experimental design in two-sided platforms: An analysis of bias". *Management Science*, 68 (10), 7069–7089.

Jussupow, Ekaterina, Benbasat, Izak, and Heinzl, Armin (2020). "Why are we averse towards algorithms? A comprehensive literature review on algorithm aversion".

Kaye, Aaron P (2024). "The Personalization Paradox: Welfare Effects of Personalized Recommendations in Two-Sided Digital Markets". *Working Paper*.

Kim, Jun B, Albuquerque, Paulo, and Bronnenberg, Bart J (2010). "Online demand under limited consumer search". *Marketing science*, 29 (6), 1001–1023.

Korganbekova, Malika and Zuber, Cole (2023). "Balancing user privacy and personalization". *Working Paper*.

Koulayev, Sergei (2014). "Search for differentiated products: identification and estimation". *The RAND Journal of Economics*, 45 (3), 553–575.

Leung, Michael P (2020). "Treatment and spillover effects under network interference". *Review of Economics and Statistics*, 102 (2), 368–380.

Li, Hannah, Zhao, Geng, Johari, Ramesh, and Weintraub, Gabriel Y (2022a). "Interference, bias, and variance in two-sided marketplace experimentation: Guidance for platforms". In "Proceedings of the ACM Web Conference 2022", 182–192.

Li, Xitong, Grahl, Jörn, and Hinz, Oliver (2022b). "How do recommender systems lead to consumer purchases? A causal mediation analysis of a field experiment". *Information Systems Research*, 33 (2), 620–637.

Liu, Lan and Hudgens, Michael G (2014). "Large sample randomization inference of causal effects in the presence of interference". *Journal of the american statistical association*, 109 (505), 288–301.

Lu, Zipei and Kannan, PK (2026). "AI for Customer Journeys: A Transformer Approach". *Journal of Marketing Research*, 63 (1), 1–26.

MacKenzie, Ian, Meyer, Chris, and Noble, Steve (2013). "How retailers can keep up with consumers". *McKinsey & Company*, 18 (1), 1–10.

Mason, Malia F, Lee, Alice J, Wiley, Elizabeth A, and Ames, Daniel R (2013). "Precise offers are potent anchors: Conciliatory counteroffers and attributions of knowledge in negotiations". *Journal of Experimental Social Psychology*, 49 (4), 759–763.

Misra, Kanishka, Schwartz, Eric M, and Abernethy, Jacob (2019). "Dynamic online pricing with incomplete information using multiarmed bandit experiments". *Marketing Science*, 38 (2), 226–252.

Nakamura, Shotaro, Hasanain, Syed Ali, and Tariq, Adeel (2024). "Spillovers and Adjustments under Information and Search Frictions in a Developing Online Marketplace". *Available at SSRN 4971204*.

Offer-Westort, Molly and Dimmery, Drew (2021). "Experimentation for homogenous policy change". *arXiv preprint arXiv:2101.12318*.

Ostuni, Vito, Kofler, Christoph, Nilange, Manjesh, Lamkhede, Sudarshan, and Zylberglejd, Dan (2023). "Search personalization at netflix". In "Companion Proceedings of the ACM Web Conference 2023", 756–758.

Padilla, Nicolas, Ascarza, Eva, and Netzer, Oded (2024). "The customer journey as a source of information". *Quantitative Marketing and Economics*, 1–40.

Pan, Qi and Wang, Wen (2021). "Costly price adjustment and automated pricing: The case of Airbnb". *Available at SSRN 4077985*.

Pariser, Eli (2011). *The filter bubble: How the new personalized web is changing what we read and how we think*. Penguin.

Rayner, Keith (1998). "Eye movements in reading and information processing: 20 years of research." *Psychological bulletin*, 124 (3), 372.

Sinclair, Betsy, McConnell, Margaret, and Green, Donald P (2012). "Detecting spillover effects: Design and analysis of multilevel experiments". *American Journal of Political Science*, 56 (4), 1055–1069.

Smith, Adam N, Seiler, Stephan, and Aggarwal, Ishant (2023). "Optimal price targeting". *Marketing Science*, 42 (3), 476–499.

Smith, Andrew R and Windschitl, Paul D (2015). "Resisting anchoring effects: The roles of metric and mapping knowledge". *Memory & cognition*, 43, 1071–1084.

Smith, Andrew R, Windschitl, Paul D, and Bruchmann, Kathryn (2013). "Knowledge matters: Anchoring effects are moderated by knowledge level". *European Journal of Social Psychology*, 43 (1), 97–108.

Song, Michelle (2021). "How Do Personalized Recommendations Affect Consumer Exploration: A Field Experiment". *Working paper*.

Tchetgen, Eric J Tchetgen and VanderWeele, Tyler J (2012). "On causal inference in the presence of interference". *Statistical methods in medical research*, 21 (1), 55–75.

Turner, Brandon M and Schley, Dan R (2016). "The anchor integration model: A descriptive model of anchoring effects". *Cognitive Psychology*, 90, 1–47.

Tversky, Amos and Kahneman, Daniel (1974). "Judgment under Uncertainty: Heuristics and Biases: Biases in judgments reveal some heuristics of thinking under uncertainty." *Science*, 185 (4157), 1124–1131.

Ursu, Raluca, Seiler, Stephan, and Honka, Elisabeth (2024). "The sequential search model: A framework for empirical research". *Quantitative Marketing and Economics*, 1–49.

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

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

Wan, Xiang, Kumar, Anuj, and Li, Xitong (2024). "How do product recommendations help consumers search? Evidence from a field experiment". *Management Science*, 70 (9), 5776–5794.

Wang, Xuewen and Liu, Jia (2026). "Modeling Consumer Search with Refinement under Non-Compensatory Preference Rules". *Working paper*.

Wei, Yanhao and Jiang, Zhenling (2025). "Estimating parameters of structural models using neural networks". *Marketing Science*, 44 (1), 102–128.

Wei, Zaiyan and Lin, Mingfeng (2017). "Market mechanisms in online peer-to-peer lending". *Management Science*, 63 (12), 4236–4257.

Weitzman, Martin L. (1979). "Optimal Search for the Best Alternative". *Econometrica*, 47 (3), 641–654.

Williams, Kevin R (2022). "The welfare effects of dynamic pricing: Evidence from airline markets". *Econometrica*, 90 (2), 831–858.

Wu, Ruiqi, Huang, Yufeng, and Li, Nan (2023). "Platform Information Design and Competitive Price Targeting". *Available at SSRN 4619584*.

Yavorsky, Dan, Honka, Elisabeth, and Chen, Keith (2021). "Consumer search in the US auto industry: The role of dealership visits". *Quantitative Marketing and Economics*, 19, 1–52.

Yoganarasimhan, Hema (2020). "Search personalization using machine learning". *Management Science*, 66 (3), 1045–1070.

Yoon, Song-Oh and Simonson, Itamar (2008). "Choice set configuration as a determinant of preference attribution and strength". *Journal of Consumer Research*, 35 (2), 324–336.

Yuan, Zhe, Chen, AJ Yuan, Wang, Yitong, and Sun, Tianshu (2025). "How recommendation affects customer search: A field experiment". *Information Systems Research*, 36 (1), 84–106.

Zhan, Ruohan, Han, Shichao, Hu, Yuchen, and Jiang, Zhenling (2024). "Estimating Treatment Effects under Recommender Interference: A Structured Neural Networks Approach". *arXiv preprint arXiv:2406.14380*.

Zhang, Luna, Ursu, Raluca, Honka, Elisabeth, and Yao, Oliver (2023). "Product discovery and consumer search routes: Evidence from a mobile app". *Available at SSRN*, 4444774.