Consumer Inferences from Product Rankings: The Role of Beliefs in Search Behavior*

Jessica Fong[1], Olivia R. Natan[2], and Ranmit Pantle[3]

[1]University of Michigan

[2]University of California, Berkeley

[3]Northwestern University

July 21, 2025

Abstract

In online markets, consumers tend to search and purchase prominently positioned products. We develop an experimental paradigm to distinguish between two mechanisms driving this behavior: position-specific search costs and beliefs about expected returns to search. Using incentivized experiments, we find that both mechanisms exist, and short-term randomization of rankings alone does not separate the two mechanisms. Failing to account for beliefs leads to biased estimates of search costs and incorrect consumer welfare predictions under alternative recommendation systems, such as platform self-preferencing. We discuss solutions for estimating unbiased search costs in real-world search settings.

JEL: D83, C93, L81, D84

*We are grateful for comments from Anocha Aribarg, Chu Ivy Dang, Andrey Fradkin, Rafael Greminger, Sam Hirshman, Aaron Kaye, Bowen Luo, Brett Gordon, Przemek Jeziorski, Ilya Morozov, Sarah Moshary, Yeşim Orhun, Gautam Rao, Heiner Schumacher, Katja Seim, Raluca Ursu, Caio Waisman, Anna Tuchman, and audiences and seminar participants at Dartmouth Tuck, the University of Michigan Dearborn, CU Boulder, Wharton, UCLA, FTC, the CODE 2024 conference, the CESifo Economics of Digitization workshop, the Marketing Science 2024 conference, the UTD Bass FORMS 2025 conference, the Paris Digital Economics 2025 conference, and IIOC 2025. Contact information: Jessica Fong (jyfong@umich.edu), Olivia R. Natan (olivia.natan@berkeley.edu), Ranmit Pantle (ranmit.pantle@kellogg.northwestern.edu).

1 Introduction

Due to the vast array of products available online, recommendation systems are increasingly influential in digital markets, shaping both consumer choice and competition among firms (Armstrong et al., 2009). By determining which products appear and in what order, recommendation systems can help consumers discover products more easily but can also be used to manipulate consumers' choices in ways that could harm them. For example, consumers are more likely to search and purchase products that appear towards the top of the search results (De los Santos and Koulayev, 2017; Ursu, 2018; Compiani et al., 2023). Platforms may exploit this by placing more profitable products in these positions (Lam, 2023; Peitz, 2023). The potential for consumer harm due to platform manipulation raises important questions about how recommendation systems steer consumer behavior.

To evaluate how recommendation systems impact consumer behavior and welfare, we need to understand *why* consumers are more likely to search recommended products. Does searching recommended products take less effort, or do consumers believe recommended products are "better?" Prior literature has often attributed position effects to position-specific search costs, which represent the effort to search down the page.[1] At the same time, consumers might believe that more prominently ranked products are higher surplus because platforms typically prioritize better products in their rankings (e.g., Kaye, 2024).

In this paper, we show that (1) both mechanisms can exist and (2) failing to account for consumer beliefs about their returns to search for recommended products leads to biased estimates of search costs, which in turn leads to incorrect behavioral and welfare predictions under alternative recommendation systems. Accurately predicting welfare under alternative recommendation systems (e.g., ranking algorithms) requires that we understand the source of position effects. If position effects are cost-driven, prominently ranking lower-quality products reduces consumer surplus by increasing search costs or reducing chosen product quality. If position effects are belief-driven, consumers may better direct their search by learning to avoid positions with lower expected returns.

Position-specific costs and beliefs about search payoffs are difficult to separately iden-

[1] Exceptions include Nocke and Rey (2023) and Athey and Ellison (2011) which attribute position effects to beliefs; position correlates with the product's utility that is observed prior to search.

tify with observational data, as they both shift search behavior in the same way, even when product rankings are randomized. Because typical field experiments do not disclose changes to the ranking algorithm, consumers may still hold beliefs formed before the experiment—that the top-ranked products have higher expected value.[2] As a result, any position effects exhibited during the experiment may still be driven by beliefs in addition to search costs, despite randomization. To overcome this challenge, we design an incentive-compatible experiment that allows us to measure search costs by shutting off the beliefs mechanism. In the experiment, participants search for a product with two vertical attributes—one visible and one observed only after costly search—in a setting similar to that of Weitzman (1979). Participants perform this search task repeatedly so we can measure learning about the ranking algorithms.

Product display order and whether participants are informed about it vary by experimental condition. In the Random-Informed condition, products are randomly ordered and participants are told this, allowing identification of position-specific search costs, as no position-specific beliefs should exist. In the Random condition, products are also randomly ordered, but participants are not informed, so both search costs and prior beliefs may influence behavior. In the Strong condition, products are generally sorted by the hidden attribute, favoring high-payoff items more prominently, and participants are again uninformed. Here, position effects may stem from search costs, prior beliefs, and learned beliefs. By estimating search costs from Random-Informed, we can isolate belief-driven effects in Random and Strong. This experiment shows that both position-specific costs and beliefs exist, and beliefs can play a significant role in driving position effects.

To better understand how beliefs can confound search costs even when rankings are randomized, we conduct another study that replicates the design of typical search ranking experiments. We change the ranking algorithm mid-experiment without informing participants. In one condition, the ranking algorithm changes from *Strong* to *Random*. This parallels typical experiments where the ordering changes to *Random* without the consumers' knowledge. In such experiments, search costs are estimated from the *Random* observations, but consumers may still hold position-specific beliefs learned from the *Strong* algorithm.

[2]For the remainder of this paper, "beliefs" refers to beliefs about the expected payoff from search.

In another condition, the algorithm switches from *Strong* to one where the two top-ranked products have the lowest value of the hidden attribute. This arrangement, which we refer to as the *Pref* algorithm, reflects the concern that ranking algorithms might prioritize products that are better for the platform or firm but worse for the consumer, i.e., self-preferencing.

We find that the larger position effects observed under the *Strong* algorithm (compared to Random-Informed) persist after the algorithm switches from *Strong* to *Random*. This persistence suggests that their beliefs formed during the *Strong* algorithm periods carry over to the *Random* algorithm periods. Consequently, the conventional approach of randomizing without disclosure can misattribute position effects generated by beliefs to costs.

Separating cost-driven and belief-driven position effects is critical for evaluating how product position impacts consumer surplus. To generate welfare predictions, we estimate a sequential search model (Weitzman, 1979) that allows for position-specific costs and Bayesian learning about position-specific payoffs. We demonstrate how failing to account for the evolution of beliefs leads to wrong welfare predictions. When the ranking algorithm shifts from *Strong* to *Pref*, counterfactual search behavior estimated from a model where search costs are estimated from short-term experiments with incorrect consumer beliefs predicts a 3.3% loss in consumer surplus, while search costs estimated from a model that accounts for learning predicts a 0.84% loss—a fourfold difference.

To separate beliefs and search costs in practice, experiments should be run long enough for consumers to fully update their beliefs to the new algorithm, and only the search behavior from after beliefs have fully updated should be used to estimate search costs. However, this could require the experiment to run for a long time, which might not be feasible. An alternative solution is to assume, in model estimation, that consumers hold correct position-specific beliefs *prior to* the experiment (e.g., Kaye, 2024). However, we find that if consumers learn slowly, as in our setting, assuming correct beliefs can lead to incorrect welfare predictions in the short run and even to the opposite direction of search costs. Therefore, assuming correct pre-experiment beliefs is best suited to settings where consumers are likely to have them, such as in settings where consumers are experienced with the search environment and the product market. Another solution is to incorporate surveyed beliefs in model estimation.

In addition to improving methods for evaluating recommendation algorithms, our paper

highlights the importance of accounting for beliefs more broadly when experimentally estimating model primitives. We demonstrate this issue for estimating search costs, but it can arise in other settings, such as when measuring advertising effectiveness. In short-term field experiments, consumer responses may reflect preexisting beliefs about the brand or product. For example, if they view an ad as part of a temporary promotion, their behavior might not predict responses to a permanent increase in advertising. A similar issue can occur with pricing: If a discount is perceived as temporary, the estimated price elasticity might not reflect behavior under a new pricing equilibrium. The solutions we propose—informing consumers of policy changes, eliciting pre-experimental beliefs (Aridor et al., 2024), or assuming correct beliefs for experienced consumers—can apply to these settings as well.

Relevant Literature and Contribution This paper contributes to four related areas of literature. First, our paper relates to existing work on consumer search (De los Santos et al., 2012; Moraga-González et al., 2023; Dinerstein et al., 2018). We focus on position effects in product search and advertising, which are well documented across many domains: users click more often on options near the top of the list in a paid keyword search (e.g., Jeziorski and Segal (2015), Ghose and Yang (2009), Agarwal et al. (2011), Jeziorski and Moorthy (2018), and Yao and Mela (2011)) as well as in product search (e.g., Ghose et al. (2014), De los Santos and Koulayev (2017), Chen and Yao (2017), Ursu (2018), Compiani et al. (2023), and Donnelly et al. (2024)). In fact, these effects are so commonplace that work in these areas accounts for position effects wherever possible (Ursu et al., 2025), sometimes even assuming the existence of the effect ex ante (Choi and Mela, 2019; Lam, 2023; Derakhshan et al., 2022). Our work builds on these findings by taking a closer look at whether these position effects result from equilibrium inferences about firm behavior by consumers or whether they reflect fundamentals of information processing.

Second, we relate to the empirical literature on consumer search behavior with learning (Rothschild, 1978; Koulayev, 2013; De Los Santos et al., 2017). Much of this literature focuses on learning about prices (e.g., Wu et al., 2022, Jindal and Aribarg, 2021), match values (Ursu et al., 2020), and the distribution of product quality (Hu et al., 2019). Other work considers how consumers extrapolate beliefs from previously searched options to unsearched

options that have similar attributes (Hodgson and Lewis, 2023) or across attributes within products (Gardete and Hunter, 2024). We build on this literature by incorporating learning about the structure of the search environment: the informativeness of product rankings.

Third, this paper relates to a growing body of work in economics and quantitative marketing that studies consumer search in controlled laboratory settings. Papers in this area investigate a wide variety of search behavior, including the role of revisiting products during a search (Dang et al., 2022), social influence and product arrangement (Jameei Osgouei et al., 2023), how advertising impacts the search funnel (Morozov and Tuchman, 2024), and heuristic approximations of optimal information acquisition (Gabaix et al., 2006). These works also study undirected search behaviors, including the role of contextual information (Karle et al., 2025) and learning about the distribution of product values (Casner, 2021). We contribute to this literature by creating an experiment design that can distinguish between position-specific beliefs and position-specific costs.

Finally, we relate to work incorporating beliefs into structural models (see DellaVigna (2018) for a review).[3] These papers generally consider new markets or first-time consumers, scenarios in which agent behavior can be observed "from the beginning." Doraszelski et al. (2018) and Huang et al. (2022) study firm pricing following deregulation in the UK energy and Washington State liquor markets, respectively. Firm behavior eventually converges to an equilibrium, although the speed of adjustment can be heterogeneous across firms (e.g., Goldfarb and Xiao, 2011). For consumers, models of learning about brand value (see Ching et al. (2013) for a summary) show the importance of dynamic beliefs in explaining several features of consumer behavior, including inertia, and imply incomplete information long after consumers enter a market. In education markets, households might hold inaccurate beliefs about school quality, which leads to too little search (Agte et al., 2024). Together, these studies suggest that assuming correct expectations might not hold in settings where a stable equilibrium has yet to be reached. We add to these works by studying the confounding role of beliefs when estimating structural primitives using short-term experiments.

The paper proceeds as follows. In Section 2, we describe a motivating model of directed search and highlight the difficulty of separating position-specific search costs and position-

[3]The standard approach to beliefs is to impose rational expectations about market conditions (e.g., Aguirregabiria, 2021, Bresnahan and Reiss, 1991, Aguirregabiria and Mira, 2007, among many others).

specific beliefs. In Section 3, we describe the design and results of Study 1. In Section 4, we describe the design and results of Study 2, outline the structural model, and describe our estimation results. We discuss counterfactuals in Section 5. Section 6 concludes.

2 Identifying Beliefs in Consumer Search

To illustrate how to separately identify beliefs about recommendation quality from search costs, we first describe the consumer search model. We consider a directed sequential search framework like that of Weitzman (1979) and Kim et al. (2010). Consumers face a finite, ordered set of product alternatives $j \in 1, \dots J$, such as one page of search results on a retail website.[4] The utility that a consumer receives from choosing product j is

$$u_j = a_j + b_j, \tag{1}$$

where a_j and b_j are vertical attributes which are valued equally by all consumers.[5] We assume the distributions of a_j and b_j are independent, and consumers are aware of this. The attribute a_j is costlessly observed prior to search (e.g., a_j is visible without clicking on the product's page). The attribute b_j is observed if and only if the consumer clicks on the product, incurring a cost c_j.[6] The distribution of b_j is known to consumers: $b_j \sim N(\mu_j, \sigma_b^2)$.

The consumer's reservation utility is the value at which the consumer is indifferent between the expected benefit of searching another product and choosing among the already-searched products. Formally, these reservation utilities, z_j, are the solutions to

$$c_j = \mathcal{B}(z_j) = \int_{z_j}^{\infty} (u_j - z_j) f(u_j) du_j. \tag{2}$$

[4]We do not incorporate consumers' pagination decisions for simplicity, which may be interpreted as consumers exclusively studying the first page of results. See Greminger (2022) for an approach to handling scrolling and pagination in search. Our results on non-identification of costs and beliefs also apply to the decision to expand the awareness set.

[5]We abstract from heterogeneity in valuation and horizontal differentiation to parallel our experimental paradigm. We discuss how to extend our model and experimental results to horizontal preferences over differentiated products in Appendix A.1.

[6]Consistent with the literature, we define a product as having been "searched" if the consumer clicks on it to view the hidden attribute (b_j). Products which are not searched are referred to as "unsearched products." In this section, we set aside heterogeneity in search costs across consumers, which we will directly account for in the empirical model.

Because the post-search utility term b_j has a normal distribution, we can use the solution derived in Kim et al. (2010). The reservation utility is

$$z_j = a_j + \mu_j + \zeta(\frac{c_j}{\sigma_b})\sigma_b, \tag{3}$$

where ζ is a scalar function.[7]

Under these assumptions, optimal search and choice behavior follows three rules:

1. *Selection rule:* Products are searched in decreasing order of reservation utilities.

2. *Stopping rule:* Consumers stop searching when the highest utility found so far is greater than the maximum reservation utility of all unsearched products.

3. *Choice rule:* After ending their search, consumers choose the option with the maximum utility among the searched set.

Thus far, we have only reiterated the basic model outlined in Weitzman (1979) and specified using a normally distributed error term (Kim et al., 2010) but have not yet discussed how position effects arise. We now describe the mechanisms that lead consumers to search products toward the top of the page. For the rest of this paper, we refer to "more prominently ranked" or "top-ranked" products as those positioned toward the top of the page.

One simple explanation for position effects is that top-ranked products could have higher a_j values. However, this does not explain why consumers prefer to search top-ranked products, as position effects exist even after controlling for a_j. Similar to previous studies (e.g., Ursu, 2018, Lam, 2023, Ghose et al., 2014), we later show that position effects exist even when a_j is controlled for. Therefore, we focus on two alternative mechanisms.

First, the search costs c_j, which are model primitives, can increase with rank $r(j)$. Higher search costs for less prominently ranked products might reflect the difficulty of navigating through ordered products (e.g., click costs), or they might reflect the higher attention costs to search among products lower down the list.

[7] In particular, "the function $\zeta(x)$ solves the following implicit equation $x = (1 - \Phi(\zeta))(\lambda(\zeta) - \zeta)$ where Φ is the cumulative standard normal distribution and λ is the standard normal hazard rate, $\phi(\zeta)/(1 - \Phi(\zeta))$, in which ϕ is the standard normal probability distribution function" (Kim et al., 2010, p. 1011).

Second, consumers might expect top-ranked products to yield greater post-search utility b_j. This belief can arise due to the design of many e-commerce platforms, where products are sorted by popularity or other hidden attributes. For example, Kaye (2024) documents that online travel agency Expedia's default algorithm places hotels with better hidden attributes (location desirability) toward the top of the page. As a result, consumers' beliefs about the returns to searching top-ranked hotels might be higher. This mechanism implies that consumers believe that μ_j decreases with $r(j)$.

A third potential mechanism is that the variance of the hidden attribute, σ_b, varies by position. This implies that consumers know that products in top-ranked positions have greater variance in the hidden attribute and therefore can yield greater returns to search. Ursu (2018) does not find empirical support for this mechanism. We exclude this mechanism from our experiment and later show that it does not impact our main conclusions.[8]

In summary, rank enters the reservation utilities in two places: in costs ($c_{r(j)}$) and in beliefs about the expected value of the hidden attribute b_j ($\mu_{r(j)}$). Taking these mechanisms into account, the reservation utility for product j is then

$$z_j = a_j + \mu_{r(j)} + \zeta(\frac{c_{r(j)}}{\sigma_b})\sigma_b. \tag{4}$$

Below, we show conditions under which position-specific beliefs can and cannot be separately identified from position-specific search costs. To demonstrate non-identification, we show that position-specific costs and position-specific beliefs can yield identical reservation utilities. This is sufficient to show non-identification in our setting because under the Weitzman sequential search model, search order and stopping behavior are fully characterized by reservation utilities and searched payoffs. Choices among searched options identify preferences for product attributes (Ursu et al., 2025), but we consider a setting with exclusively vertical differentiation and assume that utility is fully observed after search, so choice is deterministic after search. As a result, the choice rule does not provide additional identification.

Our first cases, which enable identification, require *assuming the structure of consumer*

[8]We consider robustness to different beliefs about the variance of the hidden attribute in our estimates in Section 4.3.

information in full. The assumptions in Cases 1a and 1b are often applied in practice in observational data.

Case 1a: No Correlation between Position and Beliefs Suppose that the consumer believes that rankings do not contain information, such that μ_j is independent of $r(j)$. Consider two alternatives j and k that are ex-ante identical ($\mu_j = \mu_k = \mu$,$a_j = a_k$, $\sigma_{b,r(j)} = \sigma_{b,r(k)} = \sigma_b$) except for position ($r(j) < r(k)$). Because a_j and a_k are observed and do not depend on any utility parameters to be estimated, any difference in search behavior between these items must be driven by position-specific search costs. The probability that the consumer searches j before k is

$$\begin{aligned} Pr(\text{search } j \text{ before } k) &= Pr(z_j > z_k) && (5)\\ &= Pr(a_j + \mu_j + \zeta(\frac{c_{r(j)}}{\sigma_b})\sigma_b > a_k + \mu_k + \zeta(\frac{c_{r(k)}}{\sigma_b})\sigma_b) && (6)\\ &= Pr(\zeta(\frac{c_{r(j)}}{\sigma_b}) > \zeta(\frac{c_{r(k)}}{\sigma_b})). && (7) \end{aligned}$$

Because $\zeta(\cdot)$ is monotonic, this implies that product j is searched first here if and only if $c_j < c_k$. Variation in the order of first product searched alone identifies cost differences across products. If consumers search products higher on the page first, under these assumptions, the model only produces this pattern if more prominent positions have lower search costs.

Case 1b: Correct Expectations about Position-Specific Payoffs Suppose that the consumer has correct beliefs about the ranking, such that their expectation for rank $r(j)$ is equal to $\mu_{r(j)}$. Consider a pair of alternatives j and k where $a_j + \mu_{r(j)} = a_k + \mu_{r(k)}$. Then, a consumer searches j before k with probability

$$\begin{aligned} Pr(\text{search } j \text{ before } k) &= Pr(z_j > z_k) && (8)\\ &= Pr(a_j + \mu_{r(j)} + \zeta(\frac{c_{r(j)}}{\sigma_b})\sigma_b > a_k + \mu_{r(k)} + \zeta(\frac{c_{r(k)}}{\sigma_b})\sigma_b) && (9)\\ &= Pr(\zeta(\frac{c_{r(j)}}{\sigma_b}) > \zeta(\frac{c_{r(k)}}{\sigma_b})). && (10) \end{aligned}$$

Because $\zeta(\cdot)$ is monotonic, this again implies that product j is searched first here if and only if $c_j < c_k$. Variation in the order of first product searched alone identifies cost differences across products for pairs of products where $a_j + \mu_{r(j)} = a_k + \mu_{r(k)}$. If consumers search products higher on the page first, under these assumptions, the model only produces this pattern if more prominent positions have lower search costs. This case implies that if the researcher *correctly* assumes that consumers have correct beliefs, the correct search costs can be identified.

Case 1c: Observed Expectations about Position-Specific Payoffs Search costs can also be identified if the researcher knows consumers' position-specific beliefs. For example, these beliefs can be obtained through survey methods. This case is identical to Case 1b, where instead of correct expectations, the expectations at each rank are stated beliefs $\hat{\mu}_{r(j)}$, which may not equal the true value $\mu_{r(j)}$.

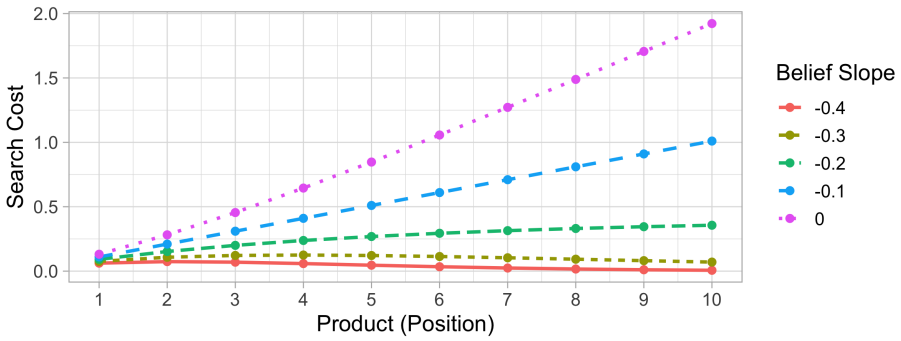
Case 2: Unobserved Expectations about Position-Specific Payoffs Position-specific beliefs and costs are unidentified outside of the cases above, where the researcher observes or assumes consumers' beliefs. To illustrate, we simulate ten products in an ordered list and show that various combinations of beliefs and costs yield the same reservation utilities, and thus, search order and continuation/stopping. We draw attributes a_j and b_j from independent normal distributions, and the products are randomly ordered. We fix position-specific search costs to be $c_j = c_0 + \beta^c \cdot r(j)$. We also generate position-specific beliefs, with consumers believing that top-ranked products have higher b_j: $\mu_{r(j)} = 3 + \beta^b \cdot r(j)$.[9] Note that these consumers do not have correct beliefs, similar to a field experimental setting where the products are randomly ordered but beliefs have not yet adjusted. These beliefs and search costs determine reservation utilities for each product j, as in Equation 4.

To show non-identification, we then solve for the costs c_j given these reservation utilities, holding the slope of position-specific beliefs (β^b) fixed at different values. Figure 1 shows that various combinations of costs and belief slopes can generate the same reservation utilities. Steeper position-specific beliefs (more negative β^cs) require less steep position-specific costs to explain reservation utilities (and identical implied search behavior). If

[9]Specifically, $a_j \sim N(5,1)$ and $b_j \sim N(3,1)$. In addition, $c_0 = 0.01$, $\beta^c = 0.1$, and $\beta^b = -0.1$

the researcher's assumptions about consumers' beliefs differ from their true beliefs, the estimated search costs are biased. In this example, the belief slope that generates the data is -0.1. If the researcher assumes consumers to have correct beliefs about the environment ($\beta^c = 0$) (i.e., that consumers realize that the products are randomly ordered), they would overestimate the slope of position-specific search costs. If consumers' true beliefs differ from the researcher's assumptions, a sequential search model produces biased search costs because many combinations of costs and belief slopes can generate the same search behavior.

Figure 1: Non-identification of Position-Specific Beliefs and Search Costs



Notes: This figure plots the search costs (c_j) for each product position for a given value of the belief slope β^b that generate the same reservation utilities, holding the set of products fixed. The true values of β^c and β^b that generate the reservation utilities are $\beta^c = 0.1$ and $\beta^b = -0.1$. The blue dashed line represents the values originally used to generate the data.

Although the two mechanisms can produce the same search behavior, they yield different predictions of behavior under counterfactual ranking algorithms. If position effects are due to position-specific costs, the search cost for a product in rank r is fixed regardless of the ranking algorithm because search costs are considered to be policy invariant. To illustrate, suppose that we observe position effects when products are sorted from best to worst. If we attribute position effects solely to search costs, a reversal in ranking order to worst-to-best would still lead consumers to search higher-ranked (but worse) products first. Conversely, if consumers learn from experience and we attribute position effects to beliefs, we would expect consumers to eventually learn to search the bottom-ranked products first. These two mechanisms have different implications for consumer welfare. If position effects are primarily cost-driven, rankings that prominently feature lower-quality products will

substantially decrease welfare, as consumers either incur higher total search costs or end up with worse products. If position effects are due to beliefs, and beliefs adjust, consumers will learn to avoid positions with lower-quality products. This implies that consumers are more susceptible to platform steering if position effects are primarily driven by search costs. Understanding to what extent each mechanism influences position effects is important both for welfare evaluation and for managerial purposes, where accurate prediction is the goal.

Demonstrating how beliefs can confound search cost estimates is especially challenging in field settings because it requires beliefs to be either observed or imposed by the econometrician.[10] Therefore, we develop a series of controlled experiments that enable us to separately measure the roles of beliefs and costs in consumers' responses to search rankings, allowing us to quantify each mechanism's contribution to position effects. We also document the speed with which consumers form beliefs about hidden attributes, both when the ranking algorithm is constant and when it changes. We use the data from the experiments to estimate a structural model of search behavior, simulate several alternative ranking algorithms, and evaluate their impact on consumer welfare. We can compare welfare effects when the roles of costs and beliefs are accurately attributed versus when position effects are mistakenly solely attributed to costs.

Although our controlled experiments avoid the challenges of field implementation, they differ from real-world platforms along a number of dimensions. We make several restrictions which simplify our design and estimation. First, we assume uncorrelated pre- and post-search attributes, which eliminates the need to model how the pre-search attribute is informative of the post-search attribute. However, this simplification deviates from realistic attribute correlations (e.g., prices and reviews are correlated). Extending to the case where a_j and b_j are correlated is straightforward. In short, a setting with correlated attributes is equivalent to a setting with uncorrelated attributes but with a lower variance in the hidden attribute. We also test the importance of this restriction in a supplemental experiment, which is described in more depth in Appendix A.1, along with other robustness checks. Second, the ranking algorithm sorts only by hidden attributes, which allows for cleaner

[10] Beliefs can be observed via incentive-compatible surveys of real consumers on the platform or imposed by disclosing the algorithm. While these are not impossible, platforms often do not have the incentive to collaborate to run such interventions, as they reveal to consumers that they are in an experiment.

inference about belief formation. Third, we abstract from horizontal preferences by using vertical payoffs, which simplifies modeling and does not affect the generalizability of our main conclusions. Incorporating horizontal differentiation would require estimating heterogeneous preferences over attributes, add noise to our estimates, and increase the sample size needed to precisely measure effects.

3 Study 1: Separating Costs and Beliefs

The objectives of this study are to empirically demonstrate that position-specific search costs and beliefs both produce position effects in product search behavior and to show that our experimental design allows us to separately identify each mechanism. We design a simple search environment that abstracts from unobserved preference heterogeneity and experimentally removes the role of beliefs about rankings in one condition.[11]

3.1 Design

Participants must select one product from a vertical list of ten products. Each product j has two vertical attributes, "bonus A" (a_j), which is visible before search (i.e., before clicking on the product), and "bonus B" (b_j), which is observed only after search. These bonuses are independent, with $a_j \sim TN(20, \sigma = 4)$ and $b_j \sim TN(40, \sigma = 12)$, which are truncated normal distributions bounded below by zero.[12] Independence of bonuses facilitates identifying search costs by position and simplifies the participants' optimal search problem.

Participants are informed about the mean, standard deviation, and 95% confidence interval of b_j and that bonus A and bonus B are drawn independently.[13] For each product searched, they incur a search cost of one point, and they earn $a + b$ points from the product

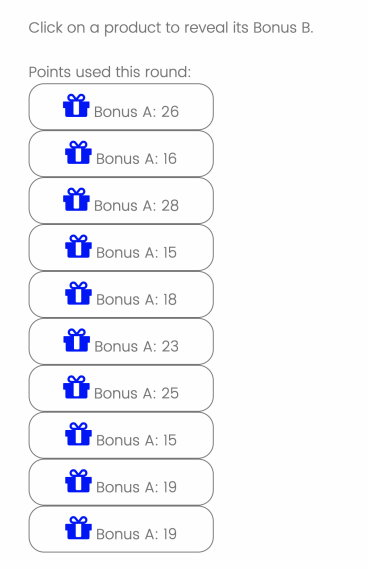
[11] Study 1 is pre-registered. The pre-registration is available at https://aspredicted.org/SWN_MK9.

[12] We use a truncated normal to avoid negative bonus Bs, slightly deviating from the stylized model's assumption of normality. Empirically, this distribution is almost equivalent to the normal distribution, as 0.034% of draws from the normal distribution are less than zero. Given the extremely low probability of truncation, we do not believe using the truncated normal implies meaningful differences from the normal.

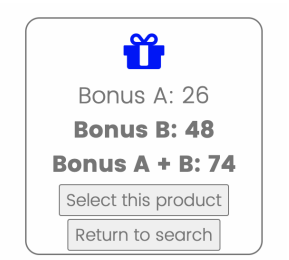
[13] We use the following language in the instructions to participants: "Bonus B has an average value of 40 points, and a standard deviation of 12. Put otherwise, 95% of products have their Bonus B between 16 and 64," and "The value of Bonus A is not informative of the value of Bonus B. In other words, Bonus A and B are not correlated. Products with high Bonus A can have low Bonus B, and vice versa." Full instructions are included in Appendix H.1.

they choose minus their search costs. Revisits are costless. There is no outside option, so participants must search at least one product. Figure 2 shows the task as displayed to the participants. All products are visible on a standard-size laptop or desktop screen. In this setting, position-specific search costs may represent effort costs, such as clicking on a product lower on the page. They may also be attributed to the general tendency to read top down, with position-specific search costs capturing attention-related cognitive effort.

Figure 2: Illustration of Search Task



(a) Search task



(b) Bonus B reveal

Notes: Panel (a) shows an example of the list of products that a participant sees. Panel (b) shows an example of what appears once the participant clicks on a product.

Participants complete the incentivized search task ten times.[14] The bonuses are redrawn for each task independently. After completing all tasks, participants receive a bonus payment of the total points earned on all tasks. For every 400 points, they receive $1 of bonus pay.[15] Participants also may earn an additional 40 bonus points ($0.10) for accurately answering a question at the end of the search tasks that elicits their beliefs about bonus B

[14]Before these ten tasks, participants begin with a practice round, followed by three comprehension checks. The first two test understanding of the bonus calculation; participants who fail both twice are disqualified. The third checks understanding that bonus A and B are uncorrelated.

[15]These values calibrate bonus payments to be between $1.50 and $2.50. These payments are in addition to their base pay.

by position.[16] We also collect participants' self-reported age, gender, and online shopping frequency. We expect that online shopping frequency may be relevant as frequent shoppers may have greater exposure to ordered product lists and have stronger position-specific prior beliefs.

We recruited US adult participants using desktop or laptop devices through Prolific.com. Participants were randomized into one of three conditions, which differ only in their ranking algorithm and its disclosure: Strong, Random, and Random-Informed.[17] In the Strong condition, bonus B is strongly correlated with the product's rank, such that products with higher bonus Bs are more prominently ranked, and participants are not informed of the correlation between bonus B and rank.[18] In the Random condition, products are randomly ordered, and participants are not informed of the ranking algorithm. In the Random-Informed condition, products are randomly ordered, and participants are informed of this.[19]

Why include these conditions? The Random-Informed condition is designed to isolate the rank-specific search costs. Participants in this condition are informed that products are randomly ordered, so any tendency to search more prominently positioned products is driven by search costs rather than by beliefs. Therefore, the Random-Informed condition allows us to estimate position-specific search costs. Any residual position effects (after accounting for rank-specific search costs measured from Random-Informed) in the Random and Strong conditions can be attributed to beliefs.

We include the Random condition to enable comparison to Random-Informed, which allows us to detect whether participants enter the experiment with position-specific prior beliefs.[20] This is important to understand in our setting because it provides empirical support for how we model learning, which we describe in Section 4.3. Additionally, no

[16] We ask participants about two products that have the same bonus A but in positions 1 and 3 in the rankings. We elicit their belief about the probability that the product in position 1 has a higher bonus B than the product in position 3. The correct answer depends on the participant's ranking algorithm.

[17] We report randomization and manipulation checks in Appendix G.3.1.

[18] To generate these imperfectly informative or "noisy" rankings, we draw an additional, independent noise term $\sim N(0, 5)$ and add it to product bonus B values, then sort in descending order of the perturbed value.

[19] This study was preregistered and run with a coding error in a fourth condition, where the ranking algorithm is supposed to switch from Strong to Random after five tasks. We dropped this condition from our analysis in Study 1 and re-ran this condition in Study 2.

[20] A reason to expect participants might have position-specific prior beliefs in our study is that if they frequently shop online, where products are typically ordered by their attributes, that behavior might carry over to other settings.

differences in search behavior between Random and Random-Informed would indicate no effect of algorithm disclosure per se. This allows us to ensure that any additional position effects observed in Strong, relative to those observed in Random-Informed, are due to the ordering of products and not due to participants not knowing the algorithm.

3.2 Experimental Results

After excluding participants who repeatedly failed comprehension checks and those who completed the survey too quickly, our final dataset contains 961 participants.[21] Across all search tasks where participants searched at least two products, participants chose the product with the largest bonus in 93% of tasks. This supports our assumption that participants choose the product with the highest payoff.

Table 1 reports the summary statistics for each condition. In all conditions, the median number of products searched (clicked on) is 1, while the mean ranges from 2.1 to 2.2 products.[22] Participants search the most in the Random condition and the least in the Strong condition, but these differences are not statistically significant. However, the total bonus earned significantly varies across conditions. Participants in Strong earn significantly more than participants in the other conditions.

We draw three conclusions from Figure 3, which plots the probability that participants search each rank across conditions, grouped by the first task, tasks 2–5, and tasks 6–10. First, the presence of position effects in Random-Informed, which is mostly clearly seen in panels 2 and 3, indicates that position-specific search costs exist. Recall that in Random-Informed, we inform participants that the products are randomly ordered, essentially "shutting off" beliefs. Second, panels 2 and 3 show that participants in Strong are more likely to search the first product than those in Random-Informed. This difference in search propensity provides evidence for position-specific beliefs about the returns to search. Note that these position effects are not driven by differences in pre-search payoffs across rank because bonus A is independent of rank. The lack of differences in position effects across conditions

[21] As noted in our pre-registration, we omit participants who completed the survey in less than the first percentile, which is 3.3 minutes.

[22] The search intensity in our setting is similar to that of other experimental search papers. For example, Morozov and Tuchman (2024) find that the average participant searches 1.9 products.

Table 1: Summary Statistics Across Conditions (Study 1)

	Condition Mean (SD)		
	Random	Random-Informed	Strong
No. searches	2.22 (2.04)	2.14 (1.77)	2.11 (1.82)
Bonus earned	66.41 (11.67)	67.18 (10.94)	68.42 (11.4)
Click max(a) first	0.41 (0.49)	0.44 (0.5)	0.41 (0.49)
Time spent per task	13.32 (16.71)	13.4 (17.99)	13.88 (19.43)
No. tasks	3,300	3,170	3,140
No. participants	330	317	314

Notes: Table reports mean and standard deviation by condition for each variable. No. searches is the number of products searched in a task. Bonus earned is the bonus of selected product minus the nominal search costs per task. Click max(a) first is a dummy variable which is equal to one if the participant clicked on the product with the highest bonus A value first in a task. Time spent per task is the seconds spent on a task from page load until final click. These statistics are summarized across participants and tasks within a condition.

in the first search task, as shown in the first panel of Figure 3, suggests that participants in the Strong condition are learning through experience that the first product offers the greatest returns to search. Third, we find no evidence of position-specific prior beliefs, as there are no differences in position effects between the Random and Random-Informed conditions.[23]

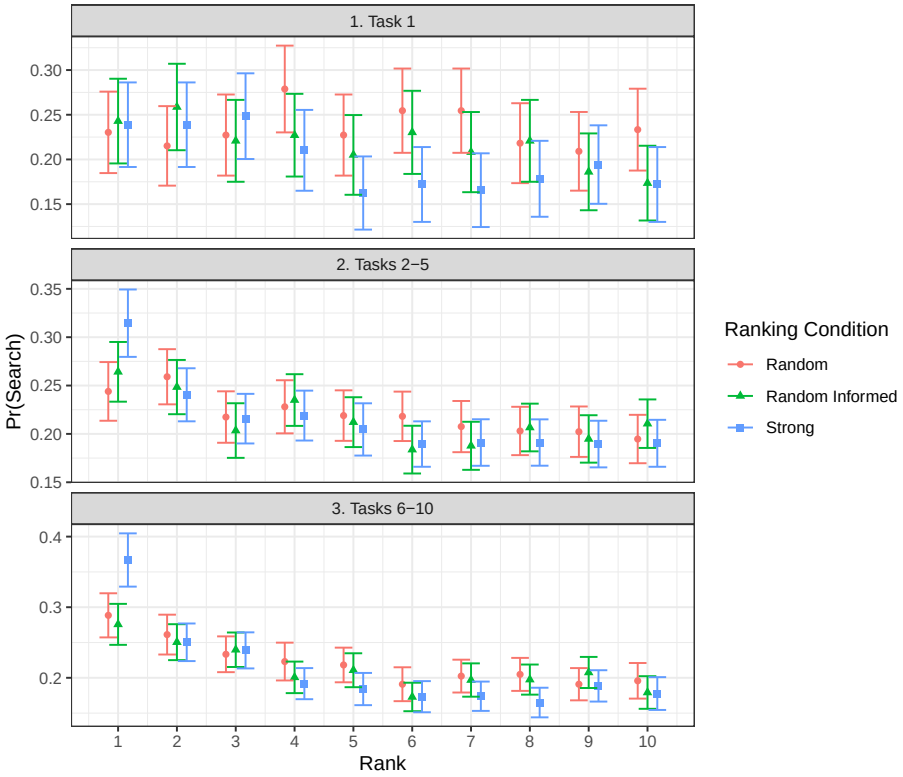
Consistent with evidence from search rates, participants in all conditions forgo higher certain payoffs (bonus As) to search top-ranked products (Figure 4).[24] Additionally, participants in all conditions display signs of inattention: they forgo bonus A points when searching any product, as shown in Figure 4, forgoing over three points of certain payoff on average. We account for this inattention in our search model in Section 4.3.

We find evidence of learning. We observe that participants in the Strong condition become better at finding products with higher bonus Bs. If the participant learns which positions yield higher bonus Bs, they should become more likely to search those positions

[23]We also do not find heterogeneity in position effects by online shopping frequency, which suggests that frequent online shoppers do not have different position-specific prior beliefs. We present this evidence in Table G.3.

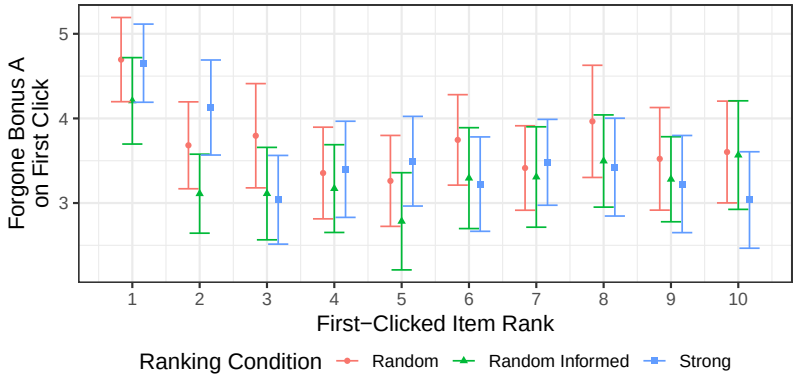
[24]Recall that the bonus As are independent of position, and the participant's total bonus for the search task depends on bonus A + bonus B. Position does not contain information about bonus B in Random-Informed, so, according to the model in Weitzman (1979), participants should search products with higher bonus As first to maximize their certain payoff, if there are no position-specific costs.

Figure 3: Search Probability by Rank and Condition (Study 1)



Notes: Figure shows the probability that each item is searched by rank and condition across participants and task groups. 95% confidence intervals are based on standard errors clustered at the participant level.

Figure 4: Forgone Bonus A by Rank of First-Clicked Item (Study 1)



Notes: This figure plots the average forgone bonus A of the first-clicked item by its rank. The forgone bonus A is defined as the maximum bonus A of all products in the task minus the bonus A of the first searched product. Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level.

Table 2: Learning to Find Higher Bonus Bs (Study 1)

Dependent Variable:	b_B of First Clicked
Model:	(1)
Variables	
Search Task	0.0146
	(0.0710)
Search Task × Condition = Random	0.1143
	(0.0989)
Search Task × Condition = Strong	0.2824***
	(0.1052)
Fixed-effects	
Participant (961)	Yes
Fit statistics	
Observations	9,610
R^2	0.13796
Within R^2	0.00226

Clustered (Participant) standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Notes: This table reports the OLS estimates of Equation 11. The dependent variable is the bonus B of the product that the participant searches first in each task. Thus, each observation is at the participant–task level. The baseline condition is Random-Informed.

first. To test this, we estimate the following linear regression:

$$b_{it} = \alpha_i + \beta_1 SearchTask_{it} + \beta_2 (SearchTask_{it} \times Condition_i) + \epsilon_{it}. \tag{11}$$

The dependent variable b_{it} is the bonus B of the first searched product in task t. The variable $SearchTask_{it}$ denotes the task number; $Condition_i$ is participant i's experimental condition; α_i are participant fixed effects.[25] Table 2 reports the OLS estimates of Equation 11. The coefficients of the main effect of $Task$ and its interaction with $Condition = Random$ are not statistically significant. These results suggest that, as expected, participants in Random-Informed and Random do not learn to find better bonus Bs with experience. However, the statistically significant interaction with Strong indicates that participants in the Strong condition do learn to identify better bonus Bs through experience. We expect that

[25]Note that this specification does not include the main effect of $Condition_i$ because it is collinear with participant fixed effects.

our results on learning potentially serve as an upper bound versus non-lab settings, where post-search utility requires further evaluation or even consumption experience. In field settings, attributes can be more difficult for consumers to evaluate and are rarely presented as univariate numerical payoffs. We discuss these differences in more detail in Appendix A.1.

Given that we do not find evidence of position-specific prior beliefs, the differences between Strong and Random-Informed conditions reveal the role of beliefs in driving position effects. To quantify this, we compare the difference in the probability of searching the first versus tenth item across these conditions to answer: "How much of this difference in search rates across position can be attributed to costs?" The difference in search rates in the final task in Random-Informed is 50% as large in Strong, so we attribute half of the position effects to costs and half to beliefs.[26]

In summary, the results from Study 1 highlight three main results. First, participants' search propensities vary significantly with rank. This is consistent with prior work, which suggests that our experimental setting can replicate features of field search settings. Second, these position effects are a combination of both costs and beliefs, and beliefs are a major component of position effects in the Strong condition. Third, participants adjust their search behavior over time in a manner consistent with learning about the underlying algorithm from repeated search and updating their beliefs to be closer to the ranking algorithm.

4 Study 2: Consumer Responses to Algorithm Changes

The results from Study 1 imply that position effects are not static, unchangeable costs—rather, they arise in part from consumers' beliefs, which may evolve in equilibrium. In Study 2, we seek to understand the implications of these equilibrium effects on search behavior when there are changes in the ranking algorithm. We also aim to demonstrate that failing to account for consumer beliefs biases search cost estimates, leading to inaccurate predictions of consumer welfare under alternative ranking policies. We focus on changes in the ranking algorithm for several reasons. First, analyzing practices like steering (Lee and Musolff, 2023; Lam, 2023), where certain products are placed at the top of rankings, often

[26] We compare only the final task because this provides the Strong participants more opportunities to learn by experience.

requires considering counterfactual rankings. If consumers can learn the ranking algorithm, they are less likely to be misled or manipulated into specific search and demand patterns. The faster consumers learn the new algorithm, the less effectively the platform can steer behavior. Conversely, the greater the influence of search costs on position effects, the more significant the impact of platform steering on consumers. Careful measurement of both belief evolution and rank-specific costs is critical for evaluating behavior and welfare under alternative platform designs.

Second, this study directly compares our approach with typical paradigms for estimating position-specific search costs. As previously mentioned, short-term field experiments that randomize rankings without disclosure can lead to incorrect estimates of the search costs if beliefs are unaccounted for. We include a condition that mimics this paradigm, Strong-to-Random, that allows us to compare how counterfactual ranking algorithms impact search cost estimates and consumer surplus when position effects are correctly attributed to both beliefs and search costs relative to when they are misattributed to search costs only.

4.1 Design

This study's design closely resembles that of Study 1, with a few changes. Participants complete fifteen tasks, rather than ten, which provides more opportunities to learn the algorithm. We also change the number of products per search task from ten to twenty. We do this because in Study 1, 10% of participants searched all ten products in at least one task. For these tasks, we can only estimate the upper bound of search costs. With twenty products, fewer participants search all items, which improves estimation precision.[27] In this study, participants might be required to scroll to see all products, depending on their screen size. In line with the Weitzman model, we assume that participants view the bonus A of all products. Greminger (2022) considers an alternative search model that accounts for scrolling costs. We abstract away from this extension but discuss its implications in Appendix A.1.

[27] There are a few smaller changes. Rather than compensating participants for all tasks, we inform participants that their bonuses are determined based on performance in a randomly selected eight of the fifteen tasks. We also re-calibrate the point conversion to 500 points equals \$1. We include participant instructions in Appendix H.2.

Table 3: Study 2 Design

Condition	Ranking Algorithm		
	Tasks 1–5	Tasks 6–10	Tasks 11–15
Random-Informed	*Random*	*Random*	*Random*
Strong	*Strong*	*Strong*	*Strong*
Strong-to-Random5	*Strong*	*Random*	*Random*
Strong-to-Random10	*Strong*	*Strong*	*Random*
Strong-to-Pref5	*Strong*	*Pref*	*Pref*
Strong-to-Pref10	*Strong*	*Strong*	*Pref*

Notes: Ranking algorithms are defined as follows. *Random* sorts products randomly. *Strong* is defined as in Study 1, where products are sorted noisily in descending bonus B. *Pref* first sorts products as in Strong, but then puts the two lowest bonus B items in the first two positions (so that the third position has the highest expected value).

Participants are randomized into one of six conditions, summarized in Table 3. Two of the conditions are Random-Informed and Strong; these are identical to those in Study 1, and the ranking algorithms remain constant across search tasks. The remaining conditions include an undisclosed change in the ranking algorithm after a set number of search tasks in the *Strong* algorithm. In Strong-to-Random5, the ranking algorithm is the same as the Strong condition for the first five tasks and switches to *Random* for the remaining ten tasks. Strong-to-Random10 switches from the *Strong* algorithm to *Random* after the first ten tasks. These two conditions mimic the typical experimental paradigm used to measure search costs, in which the algorithm switches from the platform's default to *Random* without disclosure to consumers. The remaining two conditions are Strong-to-Pref5 and Strong-to-Pref10, which represent a switch to platform steering. In these conditions, the first five and ten tasks, respectively, use the *Strong* algorithm and thereafter switch to the *Pref* algorithm, which is the following: we first sort the products using the *Strong* algorithm, and then move the two products with the lowest values of bonus B to the first and second positions. Recall that bonus A is independent of rank and bonus B, so the products sorted by the *Strong* and *Pref* algorithms are indistinguishable to participants before search. The average bonus B of the product in position 1 is 60, 40, and 20 under the *Strong*, *Random*, and *Pref* algorithms, respectively. To mimic most digital platforms' practice, we do not inform participants of the change in the ranking algorithm.

4.2 Experimental Results

We collected data from 3,569 participants on Prolific. After removing participants who did not pass the comprehension checks or completed the survey too quickly (less than the first percentile, which is 4.2 minutes), our remaining sample includes 3,533 participants.[28]

We report summary statistics in Table 4. Compared with Study 1, participants search less (1.92 products in Random-Informed in Study 2 compared to 2.14 in Random-Informed in Study 1). However, the mean bonus earned across the studies in Random-Informed is similar (67.18 and 67.16 in Studies 1 and 2, respectively).

Table 4: Summary Statistics Across Conditions (Study 2)

	Condition Mean (SD)					
	RI	Str	Str to P (10)	Str to P (5)	Str to R (10)	Str to R (5)
No. searches	1.92 (1.6)	1.78 (1.44)	1.86 (1.5)	1.97 (1.58)	1.92 (1.64)	1.99 (1.61)
Bonus earned	67.16 (11.29)	69.97 (11.63)	68.95 (12.11)	68.98 (11.93)	68.66 (11.69)	68.1 (11.38)
Click max(a) first	0.31 (0.46)	0.29 (0.45)	0.31 (0.46)	0.32 (0.47)	0.29 (0.46)	0.3 (0.46)
Time spent per task	13.41 (15.38)	12.02 (14.32)	12.92 (17.93)	13.57 (19.81)	12.51 (14.81)	13.53 (27.55)
No. tasks	8,595	9,195	8,835	8,910	8,070	9,390
No. participants	573	613	589	594	538	626

Notes: "Str" stands for Strong, "P" for Pref, "R" for Random, and "RI" for Random-Informed. Table reports mean and standard deviation by condition for each variable. No. searches is the number of products searched in a task. Bonus earned is the bonus of selected product minus the nominal search costs per task. Click max(a) first is a dummy variable which is equal to one if the participant searched the product with the highest ex-ante bonus value first on a task. Time spent per task is the seconds spent on a task from page load until final click. These statistics are summarized across participants and tasks within a condition.

Search behavior in Study 2 is similar to that in Study 1, so we briefly summarize it. Figure 5 reports the probability of searching each position for the first five products.[29] Because the ranking algorithm changes after five or ten search tasks in some conditions, we report results by each "task group" separately. Participants are more likely to search top-ranked products in Random-Informed, demonstrating the presence of position-specific search costs. In the Strong condition, position effects are more pronounced, especially in tasks 6–15, where participants have had more opportunity to learn the ranking algorithm. In fact, all other conditions have steeper position effects, averaged over all tasks.

[28]This study, including this data cleaning process, is pre-registered on AsPredicted.org (#159069). The pre-registration is available at https://aspredicted.org/3RP_ZVB.

[29]We report the search probability for all ranks in Figure G.3 in the online appendix. We also report the regression version in Table G.1 in the online appendix.

Figure 5 also shows that undisclosed randomization (in Strong-to-Random) leads to incorrect beliefs in the short term. In tasks 10–15 (Panel 3), participants in the Strong-to-Random conditions are more likely to search the first product relative to those in Random-Informed, despite the products being sorted randomly in both conditions. This suggests that beliefs learned under the *Strong* algorithm are carried over after the algorithm switch.

In this study, we directly elicit prior beliefs about expected returns to search. Participants report no significant differences in beliefs about rank-specific payoffs across conditions prior to the search tasks. None of the conditions differs from Random-Informed in the fraction of participants reporting that their beliefs about bonus B are identical across positions. We use these results to motivate the calibration of prior beliefs in Section 4.3.

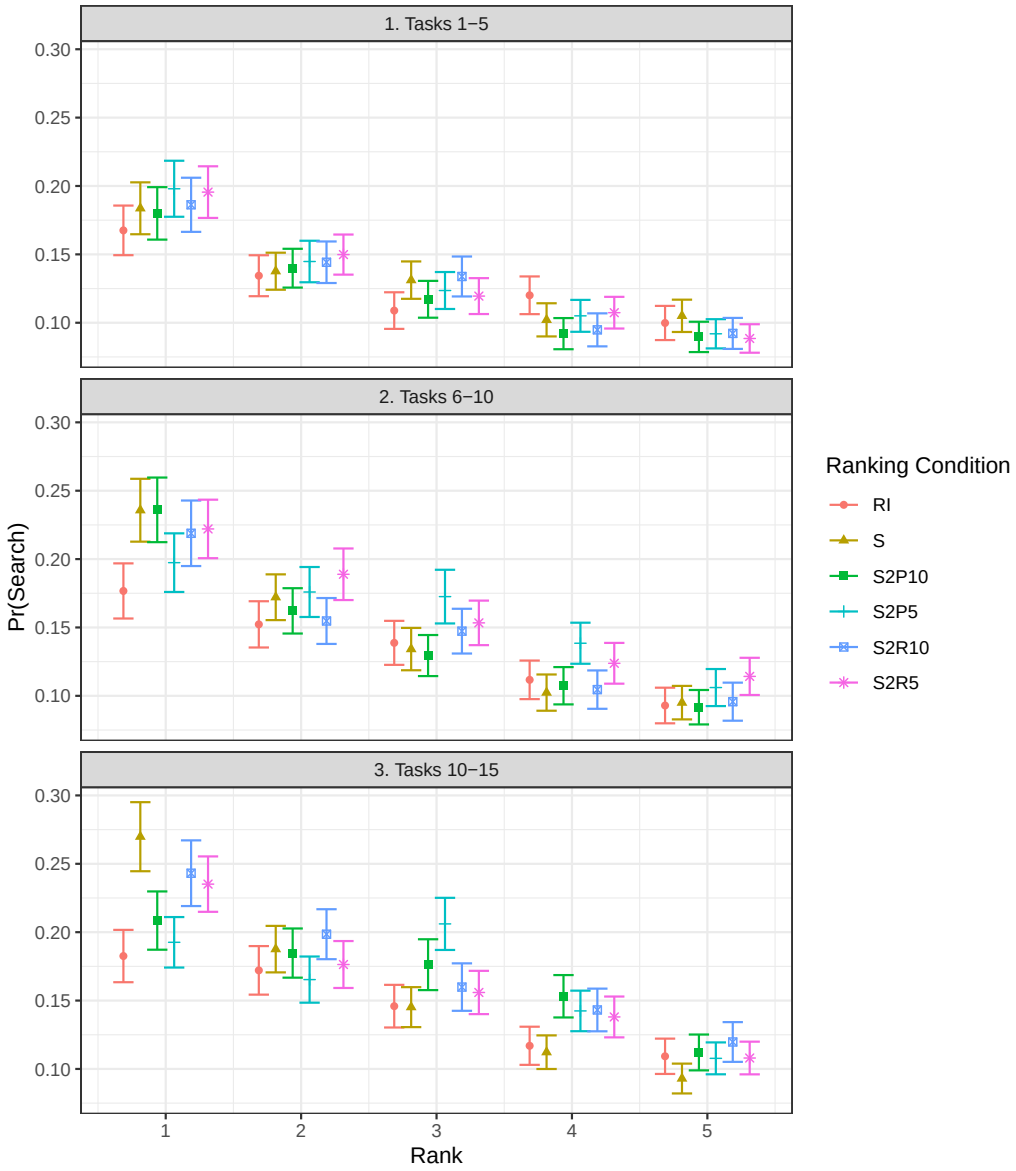
4.2.1 Evidence of Learning

We document that individuals learn the relationship between the product's position and its bonus B. These patterns inform how we model learning in Section 4.3. Position effects are more pronounced in Strong than in Random-Informed in later tasks but are similar in the early tasks (1–5), as shown in Figure 5. Participants appear to learn that products with higher bonus Bs are positioned toward the top in Strong. Figure 6, which plots the first searched bonus B by condition and search task, replicates our results from Study 1 that participants become better at directing their search with more experience. The bonus B of the first-searched product per task does not increase with experience in Random-Informed, but it does in Strong. More efficient search behavior translates into higher earnings, as shown in column 1 of of Table 5. Participants in the Strong condition earn 4.4 more points in the last task compared to the first. This increase in earnings is primarily due to increases in the bonus B of the selected product, as shown by the similar coefficient magnitudes in column 2.[30]

Participants also learn from algorithm changes. The bottom panel of Figure 5 shows that participants in Strong-to-Pref5 are significantly more likely to search the third-ranked

[30]Participants in Random-Informed earn 2.8 more points in the last task compared to the first. Improvement in earnings in Random-Informed could be due to either better stopping or continuation decisions. It is *not* due to selecting better alternatives because bonus Bs are randomly ordered. The bonus B of the first-searched product per task does not increase with experience in Random-Informed (Figure 6).

Figure 5: Search Probability by Rank by Condition (Study 2)



Notes: Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level. "RI" refers to Random-Informed, "S" refers to Strong, "S2P10" and "S2P5" refer to Strong-to-Pref10 and Strong-to-Pref5, respectively, and "S2R10" and "S2R5" refer to Strong-to-Random10 and Strong-to-Random5, respectively. Graph is censored to show only the first five products.

product relative those in Strong. This suggests that these participants correctly learned that in tasks 10–15, the third-ranked product yields the highest expected bonus B. We also find that participants in Strong-to-Pref5 are significantly more likely to search the third product first relative to Random-Informed and Strong-to-Random5 (Table G.2).

Table 5: Change in Earned Bonuses Across Tasks (Study 2)

Dependent Variables:	Earnings	Selected Bonus B
Model:	(1)	(2)
Variables		
Search Task	0.1448***	0.1218***
	(0.0271)	(0.0259)
Search Task × Condition = Strong	0.0890**	0.1121***
	(0.0378)	(0.0371)
Search Task × Condition = Strong2Pref10	0.1848***	0.2190***
	(0.0546)	(0.0528)
Search Task × Condition = Strong2Pref5	0.3101**	0.3570***
	(0.1382)	(0.1352)
Search Task × Condition = Strong2R10	0.1637***	0.1581***
	(0.0561)	(0.0542)
Search Task × Condition = Strong2R5	0.0523	0.0562
	(0.1374)	(0.1358)
Fixed-effects		
Condition (6)	Yes	Yes
Fit statistics		
Observations	35,160	35,160
R^2	0.01285	0.01573
Within R^2	0.00497	0.00497

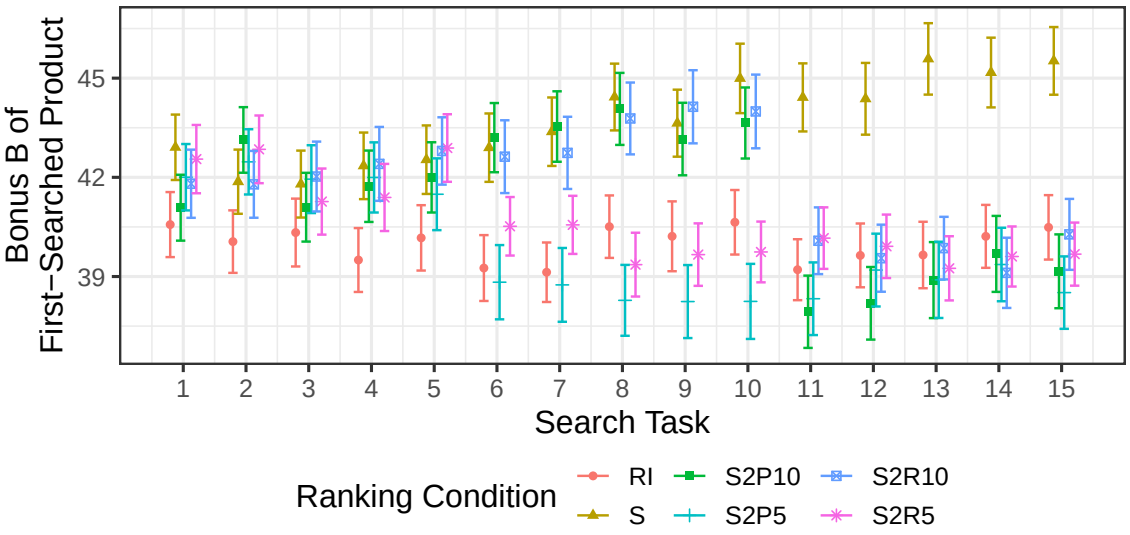
Clustered (Participant) standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Notes. This table reports the OLS estimates of $Earnings_{it} = \alpha_i + \alpha_{cond} + \beta_1 SearchTask_{it} + \beta_2(SearchTask_{it} \times Condition_i) + \epsilon_{it}$, where the earnings for a task is the bonus A + bonus B of the selected product minus the number of products searched. Variable α_i is the individual fixed effect, α_{cond} is the treatment condition fixed effect, and $SearchTask_{it}$ is the task number, which ranges from 1 to 20. We estimate this only for observations in conditions where the algorithm does not change or before the algorithm changes in conditions where it is supposed to. For example, the algorithm switches from *Strong* to *Random* after ten tasks in the condition Strong-to-Random10; we estimate this regression using only tasks 1–10 from participants in the Strong-to-Random10 condition.

4.3 Model and Estimation

We adapt the stylized sequential search model described above in Section 2 to our setting to include consumer learning and consumer heterogeneity. We estimate this model using a

Figure 6: Bonus B of First Search by Task and Condition (Study 2)



Notes: Error bars represent 95% confidence intervals which are based on standard errors clustered at the participant level. "RI" refers to Random-Informed, "S" refers to Strong, "S2P10" and "S2P5" refer to Strong-to-Pref10 and Strong-to-Pref5, respectively, and "S2R10" and "S2R5" refer to Strong-to-Random10 and Strong-to-Random5, respectively.

Bayesian estimator similar to that of Morozov (2023).[31]

Model of Learning We incorporate learning about position-specific payoffs into the model by assuming that participant i updates beliefs about the expected value of $b_{jt}|r_{tj}$ in a Bayesian fashion. Before engaging in the first task, i holds a prior belief about the expected bonus B for each product j, which varies with the product's rank. With a slight abuse of notation, we denote $b_{itj}|r_{itj}$ with b_{itr}. We assume that the prior is drawn from a normal distribution with mean b^0_{ir} and variance σ^2_0. Note that σ^2_0 is neither product- nor task-specific and is assumed to be homogeneous across participants. We also assume that $b^0_{ir} = 40$ because (1) participants were informed that the mean of bonus B (across ranks) is 40, and (2) this is the modal self-reported prior belief in all conditions.

Once i searches product r in task $t = 1$, the participant observes b_{itr} and then updates

[31]This estimator is more numerically stable and can better accommodate heterogeneity compared to frequentist methods (Morozov, 2023).

the expected value of bonus B in rank r to

$$\frac{\sigma_j^2 b_{ir}^0 + \sigma_0^2 b_{itr}}{\sigma_0^2 + \sigma_j^2}, \tag{12}$$

where $\sigma_j^2 = 144$ is the variance of bonus B. We assume σ_j^2 is constant across ranks. This assumption holds in the Random-Informed condition, where participants are explicitly told this. However, it does not empirically hold in the other conditions. For example, in the Strong-to-Pref5 condition, the variance of bonus B in rank 1 is higher than that of rank 20 because the bonus B of rank 1 changes from the highest to the lowest after five search tasks. However, because participants are not informed of this change, we assume that σ_j^2 is constant across ranks to simplify estimation. This assumption does not significantly impact our estimates.[32] Equation 12 also assumes that learning does not spill over to nearby positions (e.g., observing a high value of bonus B in position 2 does not inform beliefs about position 1). This assumption is supported by empirical evidence. Participants who see a high draw of b_r are more likely to search position r in the next task, but not positions $r+1$ or $r-1$. We report these results in Appendix D.

Given Equation 12, we can express participant i's expected value of bonus B for a product in rank r at task t as

$$\frac{1}{\sigma_0^{-2} + n_{itr}\sigma_j^{-2}} \left(\frac{b_{ij}^0}{\sigma_0^2} + \frac{n_{itr}\bar{b}_{itr}}{\sigma_j^2} \right), \tag{13}$$

where n_{itr} is the number of times that i has sampled the product in rank r between tasks 1 to $t-1$, and $\bar{b}_{itr}$ is the mean of the sampled bonus Bs in rank r in tasks 1 to $t-1$. We can further simplify Equation 13 to be expressed as a weight on the prior and the weight on the samples:

$$E[b_{itr}] = b_{ir}^0 \frac{m_{itr}}{1 + m_{itr}} + \bar{b}_{itr} \frac{1}{1 + m_{itr}}, \tag{14}$$

where $m_{itr} \equiv \frac{\sigma_j^2}{n_{itr}\sigma_0^2}$.

This specification of beliefs implies that we attribute differences in learning rates to differences in the variance of prior beliefs. In other words, if participants behave as if they

[32] We re-estimate the model with alternative assumptions on σ_j and find qualitatively similar estimates.

completely update ($E[b_{itr}] = \bar{b}_{itr}$), then our model assumes that σ_0^2 is very large. Furthermore, because we assume that σ_0^2 is homogeneous, differences in beliefs across participants arise only due to different sampling of bonus Bs ($\bar{b}_{itr}$).

Estimation We parameterize participant i's reservation utility of product j in task t (described in Equation 4) as

$$z_{itj} = a_{itj} + E[b_{itr}] + \delta_{itj}, \tag{15}$$

where a_{itj} is the pre-search, product-specific component of utility, and $E[b_{itj}|r_{itj}]$ is the expected bonus B of product j in rank r. Following Morozov (2023), instead of directly estimating search costs, we estimate the *search propensity*, denoted by δ_{itj}, which simplifies estimation. Search propensities capture rank-specific differences in the likelihood the consumer searches j due to search costs.[33] As mentioned in Section 2, we do not use the choice rule for identification or estimation.

Participant i's propensity to search product j in task t is specified as

$$\delta_{itj} = \beta_i^0 + \beta_i^1 \mathbb{1}\{r_{itj} = 1\} + \beta_i^r \Big(\mathbb{1}\{r_{itj} \neq 1\} \cdot (r_{itj} - 2) \Big) + \epsilon_{itj}. \tag{16}$$

The term ϵ_{itj} represents idiosyncratic differences in search propensities that are uncorrelated with position and cannot be rationalized by bonus A, such as differences in attention costs (e.g., seeing the third product first). Realizations of ϵ can explain why participants in the Random-Informed condition search products that have a lower bonus A, even in positions that are more costly to search on average (Figure G.4 in the online appendix). Assuming that ϵ_{itj} is drawn from a normal distribution with mean zero and variance σ_ϵ^2, we augment the data by drawing δ_{itj} directly from $\delta_{itj} \sim N(\beta_i^0 + \beta_i^1 \mathbb{1}\{r_{itj} = 1\} + \beta_i^r (\mathbb{1}\{r_{itj} \neq 1\} \cdot (r_{itj} - 2)), \sigma_\epsilon^2)$.

[33] We assume that participants are myopic across tasks—that is, they search in each task to maximize payoffs for that particular task and do not actively search more in earlier tasks to learn about position-specific distributions. Figure G.2 in the online appendix shows that the number of products searched does not vary significantly by search task under fixed ranking algorithms, which supports this assumption. Note that like Weitzman (1979), our model ignores product revisits (i.e., consumers viewing a product multiple times in the same search task). Only a small fraction (2.95%) of search tasks in Study 2 involved revisits prior to choice.

We refer to the intercept, β_i^0, as the baseline search propensity. This term represents i's propensity to search the first-ranked product, which also includes i's search costs regardless of the product's rank, such as the one point search cost imposed by the study and the opportunity cost of time. Holding all else constant, a larger β_i^0 increases the total number of searches the participant performs. Note, we do not identify this term for users who never search more than one item per task, because we force search of at least one item.

The terms β_i^1 and β_i^r capture the search-cost-driven effects of *rank* on search propensity. We allow for flexibility in the functional form by separately estimating a coefficient (β_i^1) for the first-ranked product. Position-varying search costs after the first position are captured by β_i^r. A more negative value of β_i^r, or a more positive value of β_i^1, implies that i is more likely to search top-ranked products.

Returning to the full reservation utility, we assume that participants in the Random-Informed condition do not update their beliefs about bonus B because the algorithm is disclosed. Thus, the reservation utility is

$$z_{itj} = a_{itj} + (b_{ij}^0 \frac{m_{itj}}{1+m_{itj}} + \bar{b}_{itj} \frac{1}{1+m_{itj}}) \cdot \mathbb{1}\{Cond_i \neq RI\}$$
$$+ 40 \cdot \mathbb{1}\{Cond_i = RI\} + \overbrace{\beta_i^0 + \beta_i^1 \mathbb{1}\{r_{itj}=1\} + \beta_i^r \Big(\mathbb{1}\{r_{itj} \neq 1\} \cdot (r_{itj}-2)\Big) + \epsilon_{itj}}^{\delta_{itj}}. \quad (17)$$

We further decompose individual coefficients to allow for observed and unobservable heterogeneity. Because each participant completes only fifteen tasks, estimating a separate β for each participant would yield noisy estimates. We therefore assume that $\beta_i := [\beta_i^0, \beta_i^1, \beta_i^r]$ is drawn from a shared distribution using a hierarchical model and depends on participant characteristics X_i, as follows:

$$\beta_i \sim N(\Delta X_i, V_\beta). \quad (18)$$

The matrix X_i includes (a) an intercept term, (b) an indicator for whether participant i passed all comprehension checks on the first attempt ($Comp_i$), and (c) an indicator for whether i's self-reported online shopping frequency is greater than once a week ($ShopFreq_i$). We allow β_i to vary by $Comp_i$ and $ShopFreq_i$ because participants that passed the com-

prehension checks on the first attempt exhibit stronger position effects and search more products than those who do not, and participants who shop online more frequently may have position-specific prior beliefs from their extensive experience with online product rankings.[34] The matrix Δ contains the coefficients that shift the mean value of each β element by demographics, and V_β is the covariance matrix of β_i across individuals, which reflects the correlations in unobserved heterogeneity.

We estimate this model using a Gibbs sampler that incorporates a Metropolis-Hastings step. Algorithm 1 provides an overview of our estimator. We are able to estimate search propensities using a Bayesian regression because the parameters β_i^0, β_i^1, and β_i^r enter the search propensity linearly (Equation 17). However, σ_0^2 does not enter the reservation value linearly. Rather, our specification results in a posterior density for the parameters that we can no longer directly sample from. Therefore, our estimation requires the Metropolis-Hastings step.[35] We delegate further estimation details to Appendix B. We show that we can correctly recover model parameters using simulated data in Appendix B.2.

Algorithm 1 Gibbs Sampler Overview

1: **for** $m = 1, \ldots, M$ **do**
2: Update reservation utilities $z^{(m)}|\beta^{(m-1)}, \sigma_\epsilon^{(m-1)}, \sigma_0^{(m-1)}$
3: Update search propensity coefficients $\beta^{(m)}|z^{(m)}, \sigma_\epsilon^{(m-1)}, \sigma_0^{(m-1)}$
4: Update search propensity coefficient shifters $\Delta^{(m)}, V_\beta|z^{(m)}, \beta^{(m-1)}, \sigma_0^{(m-1)}$
5: Update error variance $\sigma_\epsilon^{(m)}|z^{(m)}, \beta^{(m-1)}, \sigma_0^{(m-1)}$
6: Update prior uncertainty $\sigma_0^{(m)}|z^{(m-1)}, \beta^{(m-1)}$
7: **end for**

4.3.1 Estimation Results

The results of this estimation are presented in Table 6. The predicted search behavior using the model's estimates fit the data well. We report this in Appendix E.

The estimated prior variance is 11, which reflects the initial uncertainty participants have about position effects (recall that the prior mean is $b = 40$ for all ranks). Consumers exhibit

[34] We report the reduced form effects on search by these attributes in Appendix Section G.2.

[35] Specifically, we use a random walk proposal distribution: At the kth iteration, we draw ${\sigma_0^2}^{prop} = TN({\sigma_0^2}^{(k-1)}, h0)$. We then accept the proposal (${\sigma_0^2}^{(k)} = {\sigma_0^2}^{prop}$) with a probability proportional to the ratio of the posterior probabilities of the proposed and current values of σ_0^2. Otherwise, we reject the proposal (${\sigma_0^2}^{(k)} = {\sigma_0^2}^{(k-1)}$).

position effects in Random-Informed, which we attribute to search costs: $\Delta_0^r = -0.27$ and $\Delta_0^1 = 3.2$. This confirms that search costs increase with position. Search costs also vary by participants' attributes—those who passed the comprehension checks on the first attempt search more ($\Delta_{comp}^0 = 9.4$), and participants who frequently shop online search less ($\Delta_{shop}^0 = -1.5$). In order to compute search costs, we invert the augmented draws of δ: $c_{itj} = \sigma_b \zeta^{-1}(\frac{\delta_{itj}}{\sigma_b})$. These costs combine the effort of processing information and the physical effort of clicking and reading. The implied average search cost is \$0.05 for the first item and \$0.07 for the twentieth item.[36] Note that the dollar amounts are low, but the median time spent per task is nine seconds.

5 Impacts of Ranking Algorithms on Consumer Surplus

In this section, we use the model estimates from Study 2 to show that failing to account for beliefs leads to incorrect predictions about search behavior and biases welfare evaluation, even when rankings are randomized. We also propose solutions to estimate search costs when it is not feasible to implement a Random-Informed experimental condition.

First, we compare behavior and welfare predictions with and without accounting for consumer beliefs. Because of growing concerns around the potential harms of platform steering, we focus on welfare implications of a scenario where the platform switches from a *Strong* ranking algorithm to the *Pref* algorithm. Recall that the *Pref* algorithm places the two items with the lowest bonus Bs in the top two positions on the page, and the remaining products are arranged in noisily decreasing order of the hidden attribute.

We generate welfare predictions that account for consumer beliefs by separately estimating the learning speed (using the estimator described above) and an auxiliary search model without beliefs and without learning on the Random-Informed data in Study 2, in which we attribute all position effects to search costs.[37] This provides us with the correct

[36] We conduct a robustness check for the specification of the search propensity, where instead of a position 1-specific term, $\delta_{itj} = \beta_i^0 + \beta_i^r \log(r_{itj}) + \epsilon_{itj}$. The model estimates, reported in Table A.1 in the web appendix, are similar. In particular, the credible interval for β^r does not include zero. The model's prediction of search behavior is robust to alternative assumptions about how participants understand σ_b: namely, that the variance is lower when products are sorted. However, these estimates produce a smaller learning parameter σ_0^2, which implies faster learning.

[37] We use only tasks 6 and 7 to ensure there is an apples-to-apples comparison with the data used to estimate the auxiliary model without beliefs or algorithm disclosure, which we describe next.

Table 6: Estimation Results

Variable	Coefficient	Mean (SD)
Learning Parameter		
Prior Variance	σ_0^2	11 (0.47)
Rank Coefficient β^r		
Intercept	Δ_0^r	-0.27 (0.019)***
Comp FE	Δ_{comp}^r	0.025 (0.021)
ShopFreq FE	Δ_{shop}^r	-0.0098 (0.02)
Pos 1 Coefficient β^1		
Intercept	Δ_0^1	3.2 (0.37)***
Comp FE	Δ_{comp}^1	0.64 (0.39)
ShopFreq FE	Δ_{shop}^1	0.31 (0.37)
Baseline Search Propensity β^0		
Intercept	Δ_0^0	-28 (0.63)***
Comp FE	Δ_{comp}^0	9.4 (0.71)***
ShopFreq FE	Δ_{shop}^0	-1.5 (0.65)**
Heterogeneity		
Rank Coefficient	$\sqrt{V}_{\beta^r}$	0.51 (0.0092)
Pos1 Coefficient	$\sqrt{V}_{\beta^1}$	8.9 (3.4)
Mean Search Propensity	$\sqrt{V}_{\beta^0}$	18 (9.1)
Reservation Utility	σ_ϵ	16 (2.1)

Notes: We report whether the 90% (*), 95%(**), and 99% (***) credible intervals exclude zero for the search propensities and their shifters. Posterior means and standard deviations are based on the thinned chain which drops the first 1,000 draws of the chain and keeps every tenth draw thereafter.

position-specific search costs. We then use these estimates to simulate search behavior for consumers over 35 tasks. In the first five tasks, the products are ordered by the *Strong* algorithm, and by the *Pref* algorithm in the remaining tasks. We allow for Bayesian updating of position-specific beliefs about bonus B, and we use the variance of the prior beliefs as reported in Table B.5. Simulating search behavior provides us with each consumer's surplus per task, which is bonus A + B of the chosen product minus the total search cost.

We compare the average surplus per task to that generated by a model that does not account for consumer beliefs, estimated using data where the algorithm was not disclosed. The goal of this auxiliary model is to mimic inference drawn from typical empirical settings that randomize product ordering without disclosure, such as the commonly used Expedia field experiment data (e.g., the data used in Ursu (2018), Compiani et al. (2023), and others) and on other platforms (Derakhshan et al., 2022). In such settings, the data from the random rankings are used to estimate search costs. To parallel this, we estimate the search model without position-specific beliefs using data generated from the Strong-to-Random5 condition immediately following the switch to (undisclosed) *Random* rankings—tasks 6 and 7.[38] We assume that all position effects in the *Random* portion of the Strong-to-Random5 condition are due to search costs and simulate behavior and surplus.[39] We summarize the data sources of each estimate in Table 7, and describe the method for generating these surplus predictions in detail in Appendix F.1

Table 7: Overview of Counterfactual Models

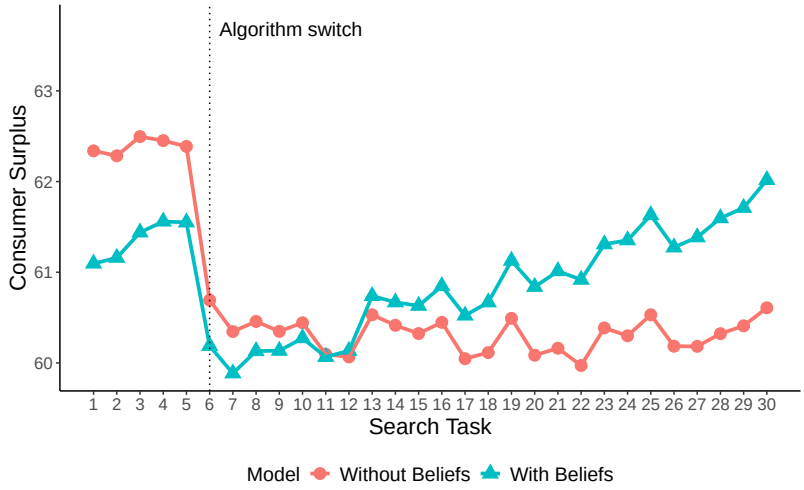
Counterfactual	**Parameter(s)**	**Data source**	**Beliefs model**
With beliefs	σ_0	Full study 2	Bayesian updating (§4.2)
	β, V_β, σ_ϵ	RI tasks 6 and 7	$E[b\|r] = 40\ \forall r$
Without beliefs	β, V_β, σ_ϵ	S2R5 tasks 6 and 7	$E[b\|r] = 40\ \forall r$

Notes. This table shows which data and which models are used in each counterfactual. "Data source" denotes the data that we use to estimate those parameters. "Beliefs model" denotes the assumptions about beliefs used when estimating those parameters. In the counterfactual without beliefs, there is no σ_0 because the model does not allow for learning.

[38] We choose Strong-to-Random5 instead of Strong-to-Random10 to demonstrate that consumers form rank-specific beliefs even after a short exposure of just five search tasks. We also estimate the same model using Strong-to-Random10, which produces stronger position effects due to beliefs, and thus, a larger error in estimated search costs and a larger misprediction of welfare.

[39] We report results for both auxiliary models in Table F.1 in the online appendix.

Figure 7: Consumer Surplus With and Without Accounting for Beliefs



Notes: Figure plots the simulated average consumer surplus per consumer for each search task. We calculate the average surplus (bonus chosen minus incurred search costs) per consumer per task, given a draw of consumer parameters. We repeat this 200 times. The ranking algorithm switches from *Strong* to *Pref* after task 5. "Without Beliefs" refers to the model estimated on the Strong-to-Random5 data after the switch to *Random* rankings. "With Beliefs" refers to the model estimated using the learning parameter in the main model and search costs recovered from Random-Informed.

Figure 7, which reports the average surplus per consumer for each task, demonstrates that not accounting for beliefs overpredicts consumer surplus under the *Strong* algorithm. This occurs because under the *Strong* algorithm, consumers benefit from stronger position effects because top-ranked positions have higher hidden payoffs. The model without beliefs generates stronger position effects due to its (incorrect) steeper search costs with respect to position. Meanwhile, the model with beliefs has less steep (but correct) search costs. The difference in search costs between products 1 and 20, estimated from the model without beliefs, is 1.8 times as large as that estimated from the model that accounts for beliefs. The higher predicted surplus without beliefs is in part due to slow belief updating. In Appendix F.4, we show that after beliefs fully adjust, the model with beliefs predicts a higher consumer surplus than the one without.

Figure 7 also shows that after the algorithm switches to *Pref*, the model without beliefs underpredicts surplus. That is, failing to account for consumer learning predicts greater consumer harm from this particular form of self-preferencing. Intuitively, the underprediction of surplus occurs because in the model without beliefs, (1) the slope of position-specific

search costs is overestimated, so that consumers are predicted to exhibit stronger position effects, and (2) they do not learn about the hidden attribute over time. Therefore, in the model with beliefs, consumer surplus increases over time due to learning, leading the consumer to pick a product with a higher bonus while also reducing search costs.[40] Our model suggests that when accounting for beliefs, changing the algorithm from *Strong* to *Pref* reduces consumer surplus by 0.84% (95% CI: 0.2% − 1.48%) when comparing the average surplus per task for tasks 1–5 to 6–30.[41] However, the model without beliefs suggests that switching to the *Pref* algorithm reduces consumer surplus by 3.3% (95% CI: 2.52% − 4.17%), overstating the loss in surplus by four times relative to the model with beliefs. The model without beliefs underpredicts the level of consumer surplus by 2.3% at task 30.[42] In equilibrium (i.e., after beliefs fully adjust), under both counterfactuals, consumers are worse off when platforms self-preference. The model without beliefs also overstates the harm from the *Pref* algorithm, predicting that the *Pref* algorithm reduces consumer surplus by 3.2%, while the model with beliefs predicts a reduction of 1.6% (a difference of 2x).

Discussion The counterfactuals show that not accounting for beliefs can lead to incorrect search and surplus predictions. How can this issue be addressed in the field? One solution is to run the experiment for a long enough time for beliefs to adjust. One method to test whether beliefs have fully adjusted is to measure position effects over time and quantify search costs only after these effects have stabilized at the consumer level. This requires a panel data set in which each consumer can be observed over time. However, this solution may be impractical. If the algorithm is subtle or differences in hidden attributes are difficult for consumers to discern, it might take an impractical amount of time for consumers to fully update. For example, consumers might learn more quickly if the hidden attribute is price, but less quickly if the hidden attribute is more difficult to evaluate, such as the quality of experience goods. Another challenge with this approach is that consumers learn at different rates. Those with high search costs take longer to learn the algorithm because

[40] This can be seen in Figure F.1 in the online appendix, which plots the average of the selected product's bonus and total incurred search costs over search tasks.

[41] Note that this is an underestimate of the decrease in surplus because consumers have not completely learned the *Strong* algorithm by task 5, as evidenced by the increasing surplus per task for these tasks under the model with beliefs.

[42] This value is obtained by comparing the surplus at task 30 between the two models.

they conduct fewer searches. In settings where consumers have heterogeneous or high search costs, running the experiment long enough for beliefs to converge might not be feasible.

Another potential solution is to assume complete information about rankings (i.e., correct beliefs) in combination with a randomized experiment. That is, one can estimate the search model using data from an experiment in which listings are randomly ordered and consumers are not informed, but which also assumes that consumers know the correct rank-specific returns to search (i.e., the average hidden attribute value at each position) under the ranking algorithm before the switch (Kaye, 2024). We test whether this is a viable solution using our data from Study 2. We estimate the model using data from the first two tasks after a switch from *Strong* to *Random* and assume that consumers have correct beliefs (as per the *Strong* algorithm) about the bonus B for a product in rank r. We compare the surplus predictions using these estimates to predictions from model estimates where search costs are separately measured (Study 2 estimates).

When we impose correct beliefs under the "old" firm policy, estimated search costs are higher for top-positioned products. This occurs because we assume full knowledge of the algorithm, but as shown in Study 2, learning is slow. Position effects are therefore not as stark as they would be under full knowledge of the *Strong* algorithm (i.e., participants should be clicking on the top-ranked products much more). As a result, the model rationalizes this with search costs that decrease with position. When we simulate search under the *Pref* algorithm, these estimates predict that consumers are *better off* (+2.23% versus −0.84% under our model's estimates). This effect is driven by lower incurred search costs (−10.33% versus +4.25% under our model's estimates). The unexpected direction of position effects from search costs may be specific to our setting, as the accuracy of this assumption depends on how quickly participants learn. In settings where preferred products are less prominently ranked, ignoring beliefs would bias search costs downward. Faster learning or a higher share of participants with accurate pre-randomization beliefs improves the validity of the correct-belief assumption in estimating search costs.[43] In field settings, this approach works best when hidden attributes are easy to learn or consumers are highly experienced. Alternatively,

[43]Consistent with this, we repeat the exercise using tasks 11 and 12 from Strong-to-Random10, where participants had more experience under the *Strong* algorithm. When participants empirically have "more correct" beliefs, we find that the bias of β^r is indeed smaller.

firms can elicit beliefs directly via surveys and incorporate them into models.

Although our counterfactual focuses on how not accounting for beliefs impacts consumer surplus, this misestimation can also have important implications for firms. For example, if a platform overestimates consumers' position-specific search costs, it may conclude that consumers are less responsive to variation in product quality or price. Consequently, the platform might set higher prices or adopt less competitive ranking strategies under the assumption that consumers will not search further. This could result in lower customer engagement and reduced sales. Therefore, obtaining correct search costs are important not just for evaluating consumer welfare but also for the firm's revenue.

6 Conclusion

In this paper, we show that beliefs play an important role in customer responses to recommendation systems, such as product rankings. We provide a framework for separating belief-driven and cost-driven position effects. Our experimental findings demonstrate that consumer beliefs contribute significantly to position effects. If beliefs are overlooked, this can lead to incorrect search cost estimates and welfare implications of alternative recommendation systems. Although the magnitude of this decomposition between beliefs and costs is specific to our experimental design and setting, we show that consumers indeed learn about the ranking algorithm, and this learning leads to biased search cost estimates.

The bias in search cost estimates exists even when the listing order is randomized. Our findings highlight the shortcomings of the common approach of extrapolating results from short-term field experiments without accounting for consumer beliefs. We demonstrate that short-term experiments could mis-estimate consumer surplus and demand under different ranking policies if researchers fail to consider the gradual adaptation of consumer beliefs.

An important implication of our findings in the product positioning context is that search costs are typically overestimated because platforms typically order from best to worst. Higher search costs can lead to the conclusion that platforms are able to steer consumers' choices to a large extent simply through the product ranking algorithm. This implies that policies such as self-preferencing (or any policy that more prominently displays

products that are worse for consumers) must greatly reduce consumer welfare: Consumers will not adjust their search in response to learning about the algorithm, and they incur (incorrectly) high search costs for conducting the same search and purchase.

We propose practical solutions to disentangle beliefs from search costs in both experimental and field settings. While we focus on product rankings, similar issues arise when short-term experiments distort consumer beliefs in estimating other model primitives—such as advertising effects (Goli et al., 2025) or price elasticity (Anderson and Simester, 2004). In pricing, for example, short-term experiments often overstate price elasticities.

Another implication for evaluating recommendation systems is that belief adjustment is often slow. Imposing correct pre-experiment beliefs overlooks the fact that consumers learn gradually—and some, due to infrequent participation or high search costs, might never fully adjust. One solution is to elicit beliefs during data collection: direct measurement offers a short-term, data-driven way to isolate policy-invariant primitives like search costs (Manski, 2004). In this paper, we instead model belief evolution directly, capturing slow adjustment and making it possible to simulate long-run counterfactuals.

We emphasize that accounting for beliefs is especially important when the platform's ranking algorithm is not transparent to consumers (e.g., Amazon's "Featured" algorithm or Expedia's "Recommended" algorithm). However, there are search environments in which position might be less informative regarding the product's hidden attribute, such as when consumers are informed about the algorithm. Examples include price-ascending algorithms for undifferentiated products or algorithms that sort based on recency. In such cases, accounting for beliefs could be less important.

References

Agarwal, Ashish, Kartik Hosanagar, and Michael D Smith, "Location, location, location: An analysis of profitability of position in online advertising markets," *Journal of Marketing Research*, 2011, *48* (6), 1057–1073.

Agte, Patrick, Claudia Allende, Adam Kapor, Christopher Neilson, and Fernando Ochoa, "Search and Biased Beliefs in Education Markets," 2024. Working Paper.

Aguirregabiria, Victor, "Identification of firms' beliefs in structural models of market competition," *Canadian Journal of Economics/Revue canadienne d'économique*, 2021, *54* (1), 5–33.

_ **and Pedro Mira**, "Sequential estimation of dynamic discrete games," *Econometrica*, 2007, *75* (1), 1–53.

Anderson, Eric T and Duncan I Simester, "Long-run effects of promotion depth on new versus established customers: three field studies," *Marketing Science*, 2004, *23* (1), 4–20.

Aridor, Guy, Duarte Goncalves, Daniel Kluver, Ruoyan Kong, and Joseph Konstan, "The Informational Role of Online Recommendations: Evidence from a Field Experiment," 2024.

Armstrong, Mark, John Vickers, and Jidong Zhou, "Prominence and consumer search," *The RAND Journal of Economics*, 2009, *40* (2), 209–233.

Athey, Susan and Glenn Ellison, "Position auctions with consumer search," *The Quarterly Journal of Economics*, 2011, *126* (3), 1213–1270.

Bresnahan, Timothy F and Peter C Reiss, "Entry and competition in concentrated markets," *Journal of Political Economy*, 1991, *99* (5), 977–1009.

Casner, Ben, "Learning while shopping: an experimental investigation into the effect of learning on consumer search," *Experimental Economics*, 2021, *24* (1), 238–273.

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

Ching, Andrew T, Tülin Erdem, and Michael P Keane, "Learning models: An assessment of progress, challenges, and new developments," *Marketing Science*, 2013, *32* (6), 913–938.

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

Compiani, Giovanni, Gregory Lewis, Sida Peng, and Peichun Wang, "Online Search and Optimal Product Rankings: An Empirical Framework," *Marketing Science*, 2023.

Dang, Chu Ivy, Raluca Ursu, and Pradeep K Chintagunta, "Repeated Product Searches and Choice Elimination: Evidence from a Lab Study," 2022. Working Paper.

De los Santos, Babur, Ali Hortaçsu, and Matthijs R Wildenbeest, "Testing models of consumer search using data on web browsing and purchasing behavior," *American economic review*, 2012, *102* (6), 2955–2980.

De Los Santos, Babur, Ali Hortaçsu, and Matthijs R Wildenbeest, "Search with learning for differentiated products: Evidence from e-commerce," *Journal of Business & Economic Statistics*, 2017, *35* (4), 626–641.

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

DellaVigna, Stefano, "Structural behavioral economics," in "Handbook of Behavioral Economics: Applications and Foundations 1," Vol. 1, Elsevier, 2018, pp. 613–723.

Derakhshan, Mahsa, Negin Golrezaei, Vahideh Manshadi, and Vahab Mirrokni, "Product ranking on online platforms," *Management Science*, 2022, *68* (6), 4024–4041.

Dinerstein, Michael, Liran Einav, Jonathan Levin, and Neel Sundaresan, "Consumer price search and platform design in internet commerce," *American Economic Review*, 2018, *108* (7), 1820–1859.

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

Doraszelski, Ulrich, Gregory Lewis, and Ariel Pakes, "Just starting out: Learning and equilibrium in a new market," *American Economic Review*, 2018, *108* (3), 565–615.

Gabaix, Xavier, David Laibson, Guillermo Moloche, and Stephen Weinberg, "Costly information acquisition: Experimental analysis of a boundedly rational model," *American Economic Review*, 2006, *96* (4), 1043–1068.

Gardete, Pedro M and Megan Hunter, "Multiattribute search: Empirical evidence and information design," *Marketing Science*, 2024.

Ghose, Anindya and Sha Yang, "An empirical analysis of search engine advertising: Sponsored search in electronic markets," *Management Science*, 2009, *55* (10), 1605–1622.

_ , Panagiotis G Ipeirotis, and Beibei Li, "Examining the impact of ranking on consumer behavior and search engine revenue," *Management Science*, 2014, *60* (7), 1632–1654.

Goldfarb, Avi and Mo Xiao, "Who thinks about the competition? Managerial ability and strategic entry in US local telephone markets," *American Economic Review*, 2011, *101* (7), 3130–3161.

Goli, Ali, David H Reiley, and Hongkai Zhang, "Personalizing ad load to optimize subscription and ad revenues: Product strategies constructed from experiments on pandora," *Marketing Science*, 2025, *44* (2), 327–352.

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

Hodgson, Charles and Gregory Lewis, "You can lead a horse to water: Spatial learning and path dependence in consumer search," Technical Report, National Bureau of Economic Research 2023.

Hu, Mantian, Chu Dang, and Pradeep K Chintagunta, "Search and learning at a daily deals website," *Marketing Science*, 2019, *38* (4), 609–642.

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

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

_ **and Sridhar Moorthy**, "Advertiser prominence effects in search advertising," *Management Science*, 2018, *64* (3), 1365–1383.

Jindal, Pranav and Anocha Aribarg, "The importance of price beliefs in consumer search," *Journal of Marketing Research*, 2021, *58* (2), 321–342.

Karle, Heiko, Florian Kerzenmacher, Heiner Schumacher, and Frank Verboven, "Search Costs and Context Effects," *American Economic Journal: Microeconomics*, 2025, *17* (2), 127–161.

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

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

Koulayev, Sergei, "Search with dirichlet priors: estimation and implications for consumer demand," *Journal of Business & Economic Statistics*, 2013, *31* (2), 226–239.

Lam, H Tai, "Platform search design and market power," 2023. Working Paper.

Lee, Kwok Hao and Leon Musolff, "Entry Into Two-Sided Markets Shaped By Platform-Guided Search," 2023. Working Paper.

Manski, Charles F, "Measuring expectations," *Econometrica*, 2004, *72* (5), 1329–1376.

Moraga-González, José Luis, Zsolt Sándor, and Matthijs R Wildenbeest, "Consumer search and prices in the automobile market," *The Review of Economic Studies*, 2023, *90* (3), 1394–1440.

Morozov, Ilya, "Measuring benefits from new products in markets with information frictions," *Management Science*, 2023.

_ **and Anna Tuchman**, "Where does advertising content lead you? We created a bookstore to find out," *Marketing Science*, 2024.

_ **, Stephan Seiler, Xiaojing Dong, and Liwen Hou**, "Estimation of preference heterogeneity in markets with costly search," *Marketing Science*, 2021, *40* (5), 871–899.

Nocke, Volker and Patrick Rey, "Consumer search, steering and choice overload," *Journal of Political Economy*, 2023.

Osgouei, Ata Jameei, Andrew T Ching, Brian T Ratchford, and Shervin Shahrokhi Tehrani, "Estimating Position and Social Influence Effects in Online Search: An Empirical Generalized Weitzman Model," *Available at SSRN 4545610*, 2023.

Peitz, Martin, "How to apply the self-preferencing prohibition in the DMA," *Journal of European Competition Law & Practice*, 2023, *14* (5), 310–315.

Rothschild, Michael, "Searching for the lowest price when the distribution of prices is unknown," in "Uncertainty in Economics," Elsevier, 1978, pp. 425–454.

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

_ **, Qingliang Wang, and Pradeep K Chintagunta**, "Search duration," *Marketing Science*, 2020, *39* (5), 849–871.

Ursu, Raluca, Stephan Seiler, and Elisabeth Honka, "The sequential search model: A framework for empirical research," *Quantitative Marketing and Economics*, 2025, *23* (1), 165–213.

Weitzman, Martin, "Optimal search for the best alternative," *Econometrica*, 1979, *78* (8).

Wu, Xiaosong, Matthew S Lewis, and Frank A Wolak, "Search with learning in the retail gasoline market," *The RAND Journal of Economics*, 2022.

Yao, Song and Carl F Mela, "A dynamic model of sponsored search advertising," *Marketing Science*, 2011, *30* (3), 447–468.

Online Appendix

A Robustness Checks

A.1 Comparisons to Real-World Search Settings

Our experimental design decisions were made to simplify the experiment and model estimation. However, the simplification creates some divergence from real-world product search settings in several key ways. Below, we outline these differences and discuss their implications.

Correlated attributes. First, we impose that a_j and b_j are uncorrelated. As previously mentioned, this choice simplifies identification and estimation but deviates from real-world settings where pre- and post-search attributes are often correlated. It is straightforward to extend to a case where a_j and b_j are correlated. a_j and b_j having a correlation coefficient of ρ is equivalent to a search context with transformed variables $\tilde{a}_j$ and $\tilde{b}_j$, where $\tilde{a}_j = a_j + E[b_j|a_j]$ and $\tilde{b}_j = b_j - E[b_j|a_j]$. By definition, $\tilde{a}_j$ and $\tilde{b}_j$ are uncorrelated. The variance of $\tilde{b}_j$ is ${\sigma_b}^2(1-\rho^2)$ which is less than ${\sigma_b}^2$, since correlation reduces the uncertainty resolved by search. In Appendix C.1, we present a supplemental experiment with an alternative data generating process that mimics real-world correlations. We find that participants engage in similar search behaviors, in terms of position effects and search intensity, as in the case with uncorrelated attributes.

Ranking algorithm. Second, our Strong algorithm only sorts based on b_j and not a_j. Since a_j is observed by participants, we do not believe that learning about whether (and to what extent) the ranking algorithm prioritizes a_j is of material consequence. On the other hand, sorting only by b_j allows for cleaner identification – any incremental position effects in Strong are *only* due to beliefs.

Horizontal preferences. Third, by using bonuses instead of product attributes, participants in our setting have vertical preferences over product attributes. This simplifies model estimation because we do not need to estimate participants' preferences. However, in real search settings, consumers may have horizontal preferences. This may produce several differences from our paradigm. It makes identification more difficult for the econometrician, since search costs and preference heterogeneity may be confounded without the panel style data we collect (Morozov et al., 2021). Also, horizontal preferences may also yield differentiated products, which are more costly to evaluate. As a result, consumers may learn more slowly because the signals about the ranking algorithm are more obscured. Additionally, with horizontal preferences, the self-preferencing algorithm in Study 2 and our counterfactuals may not harm all consumers as much as if preferences were purely vertical. Some consumers may prefer the products that are placed more prominently (i.e., those with lower bonus Bs) under a self-preferencing algorithm. While this might impact the magnitude of our consumer welfare estimates in the counterfactuals, the presence of horizontal preferences does not impact the main conclusion that consumers form beliefs about the correlation between prominence and product value and account for these beliefs in search. In terms of the

model, if consumers' preferences are recovered, we can just as easily separate search costs and beliefs by using individual-specific $a_{ij} = \beta_i^a a_j$ and $b_{ij} = \beta_i^b b_j$ in the model.[44]

Explicit marginal search costs. Fourth, we impose an explicit marginal search cost: each product costs one point to search. This differs from real search settings where consumers do not pay monetary costs to search each product. We impose the marginal search cost because participants in an online experiment setting may face different outside options than those who are conducting search in field settings. A source of this difference is that study participants are paid for their time, while consumers in the field are not. The differences in outside options impact their marginal search costs, as these search costs are relative to the normalized outside option (but should not create differences in position-specific search costs). As a robustness check, we implemented a small pilot study that is similar to Study 2 but with zero explicit marginal search costs (i.e., products are free to search). We find that this study generates search behavior that is less similar to field settings and to optimal search. Participants in Random-Informed are less likely to search the product with the highest bonus A first (which is the optimal search strategy). Additionally, participants search more products relative to field settings, and are more likely to search all products relative to Study 2.[45] In summary, having an explicit marginal search cost in the online experiment setting is important because it generates search behavior that more closely mirrors real-world patterns and predictions from Weitzman search model. We describe the details of this pilot study with zero marginal search in Appendix C.2.

Greater returns to search. Fifth, in both studies, the fifth and ninety-fifth percentiles of bonus B are 20 and 60, which equates to a difference of approximately \$0.10 USD. A concern is that the incentives to search are not very high, leading to behavior that is not consistent with the Weitzman search model. In other words, the monetary incentive may be too small to motivate participants to search in a way that maximizes their payoffs. Also, in real search settings, price dispersion is likely much greater than in our study. To address this, we first note that each search task takes each participant on average 13 seconds to complete. This translates into an hourly rate of \$27.69. So, while the incentives are low, the tasks are also conducted quickly. Additionally, the estimated search costs are in line with these incentives. In Study 1, the estimated search cost for a product in the first position is \$0.05. This translates to an hourly cost of \$13.80, which is a reasonable rate, especially relative to our study's average base payment of \$9.30/hour. Furthermore, to ensure that our conclusions are not driven by low study incentives, we ran a follow-up study that "raises the stakes." This study is similar to Study 2, with a few exceptions: (1) we increased the payoffs from search by using a conversion rate of 200 points per dollar (instead of 500), (2) participants conduct ten search tasks (instead of fifteen) and only include the Strong and Random-Informed conditions (for cost considerations), (3) decreased the point value of the marginal search cost to 0.5 (instead of 1), and (4) payoffs were summed over all tasks (instead of eight out of fifteen). These changes translate into an increase in expected

[44] The variables β_i^a and β_i^b represent individual-specific preferences towards bonuses A and B.

[45] Participants search on average between 4.5 to 7.4 products across condition, which is more than the 1.78-1.99 products searched in Study 2 and in other field settings (e.g. Morozov, 2023; Ursu, 2018). Participants searched all products in 12% of search tasks, as opposed to 0.1% in Study 2.

returns to search by ∼4x because a point is worth more than twice as much as before, and we compensate the participant for more tasks. We find that while raising the stakes increases search intensity (as expected), it does not impact position effects relative to those in Study 2. Therefore, we conclude that our findings from our main studies are not impacted by lower incentives to search. We describe the "higher stakes" study in detail in Appendix C.3.

Ease of Attribute Evaluation. In our experiment, the total product payoff is presented for easy processing by participants, unlike many product market settings where quality must be assessed through further attention or through experience. We expect if attributes are more difficult to evaluate, the learning speed we observe is *faster* than in many real search settings.

Ability to Observe Pre-Search Attributes for All Products. Another difference between our setting, that of Study 1 in particular, and the real world setting is that in our setting, participants can observe the pre-search attribute, Bonus A, for most, if not all, products. In Study 1, participants with a standard screen size can view all ten products without scrolling. However, in the real world, the platform may have many products such that the assumption that the consumer observes the pre-search attribute for all products on the page does not hold. As a result, consumers may form position-specific beliefs about pre-search attributes, in addition to hidden attributes, which in turn can determine their search behavior. We abstract away from this possibility by assuming that participants form reservation utilities over all products. Although this may be a reasonable assumption in Study 1 because all products are visible, this assumption may be violated in Study 2, where some products are visible only after scrolling. Therefore, one can interpret Study 2 as more similar to real world search in this regard. Nevertheless, the Weitzman model can be extended to allow for costs associated with scrolling (as in Greminger (2022)) or turning to the next page.

A.2 Model Robustness Checks

Table A.1: Search Propensity Specification Robustness Check: Log(rank)

Variable	Coefficient	Mean (SD)
Learning Parameter		
	σ_0^2	9 (0.46)
Rank Coefficient β^r		
Intercept	Δ_0^r	-2.7 (0.14)***
Comp FE	Δ_{comp}^r	-0.043 (0.16)
ShopFreq FE	Δ_{shop}^r	-0.2 (0.15)
Baseline Search Propensity β^0		
Intercept	Δ_0^0	-25 (0.72)***
Comp FE	Δ_{comp}^0	9.8 (0.8)***
ShopFreq FE	Δ_{shop}^0	-1 (0.75)
Heterogeneity		
Rank Coefficient	$\sqrt{V}_{\beta^r}$	4.1 (0.54)
Mean Search Propensity	$\sqrt{V}_{\beta^0}$	21 (12)
Reservation Utility	σ_ϵ	16 (2)

Notes: This table reports the estimates of an alternative specification of the search propensity described in Equation 16. In this robustness check, the search propensity $\delta_{itj} = \beta_i^0 + \beta_i^r \log(r_{itj}) + \epsilon_{itj}$. We report whether the 90% (*), 95%(**), and 99% (***) credible intervals exclude zero for the search propensities and their shifters. Posterior means and standard deviations are based on the thinned chain which drops the first 1,000 draws of the chain and keeps every tenth draw thereafter.

B Bayesian Estimation Details

B.1 Estimator Details

We outline the implementation of our estimator for Random-Informed and subsequently for all other conditions. In all cases, the following inequalities must hold for reservation utilities:

1. Search order: $z_{i1} > z_{i2}, > ... > z_{iJ}$ (where $j = 1$ is the first searched, and J is the last searched)
2. Continuation: $max(a_{ik} + b_{ik}) < z_{ij}$ for all $k < j$, all $j \in \{2, ..J\}$ (max bonus of searched is less than the reservation value of next searched)
3. Stopping: $max_{k \in \{1,..J-1\}}(a_{ik} + b_{ik}) > z_{iJ'}$ (max bonus of searched is greater than the reservation values of all unsearched)

B.1.1 Random-Informed

To sample from the distribution of β and σ_ϵ, we augment the data by drawing the search propensities δ's, which allows us to have values for z which satisfy the above conditions. We do not impose any sign restrictions on β, so that it is possible that the position effects go in either direction. Let w_{itj} denote the vector of attributes that shift search propensities for product j, such that

$$\delta_{itj} = \beta_i^0 + \beta_i^1 \cdot \mathbb{1}\{r_{itj} = 1\} + \beta_i^r \cdot \Big(\mathbb{1}\{r_{itj} \neq 1\} \cdot (r_{itj} - 2)\Big) + \epsilon_{itj} \tag{B.1}$$

$$= \beta_i' w_{itj} + \epsilon_{itj}. \tag{B.2}$$

In this specification, w_{itj} is a vector of 1 , $\mathbb{1}\{r_{itj} = 1\}$, and $(\mathbb{1}\{r_{itj} \neq 1\} \cdot (r_{itj} - 2))$, where r_{itj} is the rank of product j for user i for task t. In alternative specifications, w includes 1 and $ln(r_{itj})$. Also, we denote pre-search component of the reservation utility by $\theta_j = a_j + E[b_j|r_j]$. Recall that in Random-Informed, $E[b_j] = 40$ for all j.

Since we have imposed a normal distribution on this error term, we use conjugate priors where possible (Normal for the mean parameters, Inverse Gamma for the variance). We estimate this model using a hybrid Gibbs sampler, with each step outlined below. Since Random-Informed participants have static expectations, we do not need to estimate a prior variance to match their learning. We start by explaining the steps for estimating Random-Informed parameters.

1. **Augment δ and update z.** Given β and σ_ϵ, the distribution of δ is fully defined. However, we have to impose censoring on the draws such that the search sequences are optimal at that set of draws. For ease of exposition, we drop subscripts i and t, as this holds for all tasks and all users. We also omit the superscript m for objects outside the step of the Gibbs sampler (e.g., when drawing δ, parameters β and σ_ϵ are already at their "most recent" value).
   (a) **Case: Search 1 product** For the searched product 1, we require that $z_1 > max(z_k|k > 1) \iff \delta_1^m > max(\theta_k + \delta_k^{m-1}|k > 1) - \theta_1$. Thus, draw δ_1^m from a

Truncated Normal $TN(\beta' w_1, \sigma_\epsilon^2 | lb = max(z_k^{m-1} | k > 1) - \theta_1, ub = \infty)$.
For unsearched products 2 through 10, the stopping rule provides the upper bound on the reservation utilities: $z_j < a_1 + b_1 \iff \delta_j < a_1 + b_1 - a_j - 40$, thus $\delta_j^m \sim TN(\beta' w_j, \sigma_\epsilon^2 | lb = -\infty, ub = a_1 + b_1 - a_j - 40)$.
Before proceeding, set $z_j^m = \theta_j + \delta_j^m$.

(b) **Case: Searches 2 through 9 products** Let K be the number of searches. For the first searched product 1, we require that $z_1 > max(z_k | k > 1) \iff \delta_1^m > max(\theta_k + \delta_k^{m-1} | k > 1) - \theta_1$. Thus, draw from a Truncated Normal $\delta_1^m \sim TN(\beta' W_1, \sigma_\epsilon^2 | lb = max(z_k^{m-1} | k > 1) - \theta_1, ub = \infty)$. Set $z_1^m = \theta_1 + \delta_1^m$

Next, for searched products j in 2 through $K-1$, $z_j^m < z_{j-1}^{m-1} \iff \delta_j^m < z_{j-1}^m - \theta_j$. In addition, there is a lower bound condition to satisfy: that the subsequent product searched has a lower z. This is $\delta_j^m > z_{j+1}^{m-1} - \theta_j$, so we have the following $\delta_j^m \sim TN(\beta' w_j, \sigma_\epsilon^2 | lb = z_{j+1}^{m-1} - \theta_j, ub = z_{j-1}^{m-1} - \theta_j)$. Set $z_j^m = \theta_j + \delta_j^m$

For the last-searched product K we have the same style upper bound but a different lower bound (continuation restriction + search order restriction). $z_K^m < z_{K-1}^{m-1} \iff \delta_K^m < z_{K-1}^{m-1} - \theta_K$, $z_K > max(z_k | k > K) \iff \delta_K^m > max(\theta_k + \delta_k^{m-1} | k > K) - \theta_K$ and $z_K^m > max_{j<K}(a_j + b_j) \iff \delta_K^m > max_{j<K}(a_j + b_j) - \theta_K$. This yields $\delta_K^m \sim TN(\beta' w_K, \sigma_\epsilon^2 | lb = max(max_{j<K}(BA_j + BB_j) - \theta_K, max(\theta_k + \delta_k^{m-1} | k > K) - \theta_K), ub = z_{K-1}^{m-1} - \theta_K)$. Set $z_K^m = \theta_K + \delta_K^m$.

Finally, for unsearched products l, the stopping rule and search order rule provide the upper bound $z_l^m < max_{j \le K}(a_j + b_j) \iff \delta_l^m < max_{j \le K}(a_j + b_j) - \theta_l$ and $z_l^m < z_K^{m-1} \iff \delta_l^m < z_K^{m-1} - \theta_l$. Draw $\delta_l^m \sim TN(\beta' w_l, \sigma_\epsilon^2 | lb = -\infty, ub = max(max_{j \le K}(a_j + b_j) - \theta_l), z_K^{m-1} - \theta_l)$ and update $z_l^m = \theta_l + \delta_l^m$

(c) **Case: Searches 10 products** This is the same as above case, but just uses the search orders for products 1 through 9. For the final searched product, the upper bound is based on product 9. The lower bound is based on only the continuation condition. After each product, update z with the new δ.

2. **Draw** β. Given the search propensities δ^m and the prior draws of σ_ϵ^{m-1},Δ^{m-1} and V_β^{m-1}, we update the regression coefficients β_i's using a Bayes regression step. The posterior distribution is

$$\beta_i^m | \sigma_\epsilon^2, w_i, V_\beta, \Delta \sim N((w_i' w_i / \sigma_\epsilon^2 + V_\beta^{-1})^{-1} (w_i' \delta_i / \sigma_\epsilon^2 + V_\beta^{-1} \Delta' X_i), (w_i' w_i / \sigma_\epsilon^2 + V_\beta^{-1})^{-1}) \tag{B.3}$$

where w_i is a matrix of values w_{itj} with rows for each j, t corresponding to user i.

3. **Draw** σ_ϵ.

$$\sigma_\epsilon^2 \sim IG(a + \frac{N}{2}, (\frac{1}{b} + \sum_{i,j,t} (\frac{(\delta_{itj} - w_{itj} \beta_i)^2}{2}))^{-1}) \tag{B.4}$$

Here N is the total number of observations, and prior value of $\sigma_\epsilon^2 \sim IG(a, b)$

4. **Draw Δ, V_β.**

$$vec(\Delta) \sim N(vec(\tilde{\Delta}), V_\beta \otimes (X'X + A)^{-1}) \tag{B.5}$$

$$V_\beta \sim IW(\nu_0 + Nusers, V_0 + (\beta - X\tilde{\Delta})'(\beta - X\tilde{\Delta}) + (\tilde{\Delta} - \bar{\Delta})'A(\tilde{\Delta} - \bar{\Delta})) \tag{B.6}$$

Here, β is a matrix that stacks β_i's across users, and $\tilde{\Delta} = (X'X + A)^{-1}(X'\beta + A\bar{\Delta})$. Prior value of $V_\beta \sim IW(\nu_0, V_0)$ and prior value of $vec(\Delta) \sim N(vec(\bar{\Delta}), V_\beta \otimes A^{-1})$.

B.1.2 Other Conditions

The previous subsection described estimation for Random-Informed only. In practice, we jointly estimate the model across all conditions by allowing participants in other conditions to learn about position specific means as detailed in Section 4.3. Recall that $z_{itj} = \theta_{itj} + \delta_{itj}$. We now let $\theta_{itj} = a_{itj} + E[b_{itj}|r_{itj}]$ for all conditions other than Random-Informed. This produces full reservation utilities:

$$\begin{aligned} z_{itj} = a_{itj} + (b^0_{ij}\frac{m_{itj}}{1+m_{itj}} + \bar{b}_{itj}\frac{1}{1+m_{itj}}) \cdot \mathbb{1}\{Cond_i \neq RI\} \\ + 40 \cdot \mathbb{1}\{Cond_i = RI\} + \overbrace{\beta^0_i + \beta^1_i \mathbb{1}\{r_{itj} = 1\} + \beta^r_i \Big(\mathbb{1}\{r_{itj} \neq 1\} \cdot (r_{itj} - 2)\Big) + \epsilon_{itj}}^{\delta_{itj}} \end{aligned} \tag{17}$$

where $m_{itj} \equiv \frac{\sigma^2_j}{n_{itj}\sigma^2_0}$, n_{itj} is the number of times that i has sampled the product in rank r_{itj} between tasks 1 to $t-1$, and $\bar{b}_{itj}$ is the mean of the sampled bonus Bs in rank r_{itj} in tasks 1 to $t-1$.

To estimate the prior variance parameter, σ^2_0, we need to make a non-conjugate Gibbs step, since σ^2_0 does not enter the reservation value linearly in Equation 17. At this step of the hybrid Gibbs sampler, we take a Metropolis-Hastings step. To implement this step, we use a random walk proposal distribution: at the mth iteration, we draw ${\sigma^2_0}^{prop} = TN({\sigma^2_0}^{(m-1)}, h0)$. We then accept the proposal (${\sigma^2_0}^{(m)} = {\sigma^2_0}^{prop}$) with a probability proportional to the ratio of the posterior probabilities of the proposed and current values of σ^2_0. Otherwise, we reject the proposal (${\sigma^2_0}^{(m)} = {\sigma^2_0}^{(m-1)}$).

B.1.3 Priors

Let p denote the length of vector β_i. Thus $p = 2$ in the linear specification and the log specification, and $p = 3$ in the specification with a separate effect for the first position. The values of the prior parameters we used for estimation are given in Table B.1. We base these priors on a now-omitted set of estimates on Study 1 data.

Table B.1: Priors

Parameter	Prior Variable	Value
Search Propensity(Δ)	$\bar{\Delta}$	$\begin{bmatrix} -22 & -0.36 & -0.23 \\ 18 & 2.7 & -0.36 \\ -2 & 1 & 0 \end{bmatrix}$
	A	I_5
Heterogeneity(V_β)	ν_0	5
	V_0	$\begin{bmatrix} 22^2 & 6.8^2 & 1.1^2 \end{bmatrix} \times I_p$
Search Propensity Shocks(σ_ϵ^2)	a	5
	b	1
Prior Uncertainty (σ_0^2)	Δ^0	17
	$\Delta_{\sigma_0^2}$	100

Notes: I_n denotes the identity matrix of dimension $n \times n$. p refers to the length of vector β_i. Prior versions of the paper included a structural model to estimate propensities and heterogeneity on study 1 data - these results were used to set prior means.

B.2 Simulations

B.2.1 Simulation 1: Random-Informed Only

To demonstrate that the proposed estimator can recover the correct parameters, we simulate search behavior based on the stimuli used in Random-Informed in Study 1. We take the actual data used ($N = 317$ participants, $T = 10$ tasks of $J = 10$ products each), remove outcomes, and simulate outcomes under a set of known parameters. We choose these parameters to approximately match the distribution of the number of searched products we observe in the data. We focus on the Random-Informed condition to show that the model works well to recover both baseline costs across positions and to recover the noise in behavior σ_ϵ.

We conduct three simulations. First, we simulate data and recover estimates under a DGP with exclusively unobserved heterogeneity (Simulation 1a). Second, we add observed shifters of mean search propensity and rank effect which match our empirical application: dummy variables for $Comp_i$ and $ShopFreq_i$ (Simulation 1b). Third, we allow for a distinct position effect in position 1, as opposed to exclusively a linear effect of rank (Simulation 1c).

In the first simulation, $z_{itj} = a_{itj} + E[b] + \beta_i^0 + \beta_i^r r_{itj} + \eta_{itj}$, where $\eta_{itj} \sim N(0, 15^2)$, $\beta_i^0 \sim N(-11, 20^2)$, and $\beta_i^1 \sim TN(-0.5, 1.5, \leq 0)$. All draws are i.i.d. across i, j, t or across i, depending on the unit over variation. This yields 55% of observations with only one search (versus 54% in our data).

In the second simulation, we allow for observable differences in search propensities (e.g., as in Equations 24 and 25 above). We simulate with parameters reported in Table B.3. Baseline propensity coefficients (equivalent to position 1) are drawn from a normal distribution, rank coefficients are drawn from a truncated normal distribution (with no values

greater than zero). In the third simulation, we keep the existing parameters from the second simulation, and we draw the position 1 fixed effects as normal draws with parameters reported in Table B.4.

We estimate the model using 5,000 draws from the Gibbs sampler. We apply diffuse priors, and start at random draws for each value. In practice, the starting draws do not impact the chain. We discard the first 2,000 draws, and we thin the subsequent chain by using every fifth draw. Both simulations have large degrees of heterogeneity across consumers.

We report the plotted posterior distributions in Figures B.1, B.2 and B.3. We also conduct inference on the thinned chain using every 10th draw. We are able to recover all parameters, even with large degrees of across-person heterogeneity.

Table B.2: Simulation 1a Parameter Estimates—Unobservable Heterogeneity

Parameter	Coef	Truth	Posterior Mean	Posterior SD
Baseline search propensity	Δ^0	-9.235	-9.502	1.332
Rank Coefficient	Δ^r	-1.374	-1.341	0.077
Unobs heterogeneity in (β^0)	λ_{Δ^0}	21.541	21.91	1
Unobs heterogeneity in (β^r)	λ_{Δ^r}	1.083	1.073	0.068
Standard deviation of ϵ	σ_ϵ	15	14.893	0.288

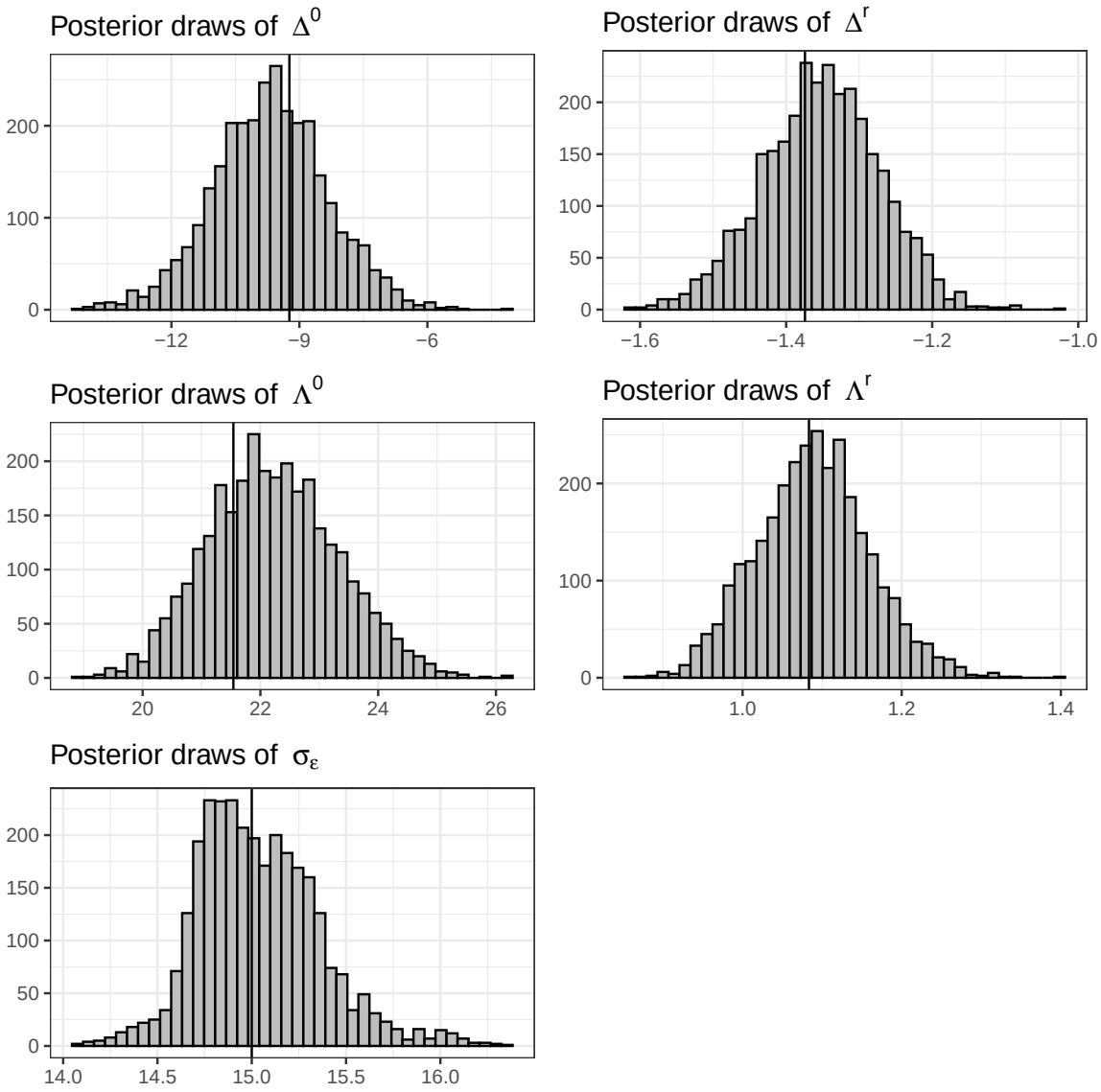
Table B.3: Simulation 1b Parameter Estimates—Observable and Unobservable Heterogeneity

Parameter	Coef	Truth	Posterior Mean	Posterior SD
Baseline search propensity	Δ^0	-14.575	-14.634	1.948
Baseline search propensity Comp FE	Δ^0_{comp}	10.993	10.865	2.287
Baseline search propensity Shop FE	Δ^0_{shop}	0.276	0.689	2.36
Rank Coefficient	Δ^r	-1.183	-1.022	0.114
Rank Coefficient Comp FE	Δ^r_{comp}	-0.282	-0.429	0.143
Rank Coefficient Shop FE	Δ^r_{shop}	-0.044	-0.1	0.119
Unobs heterogeneity in (β^0)	λ_{Δ^0}	20.522	20.216	0.963
Unobs heterogeneity in (β^r)	λ_{Δ^r}	1.032	0.947	0.063
Standard deviation of ϵ	σ_ϵ	15	14.784	0.292

B.2.2 Simulation 2: Learning

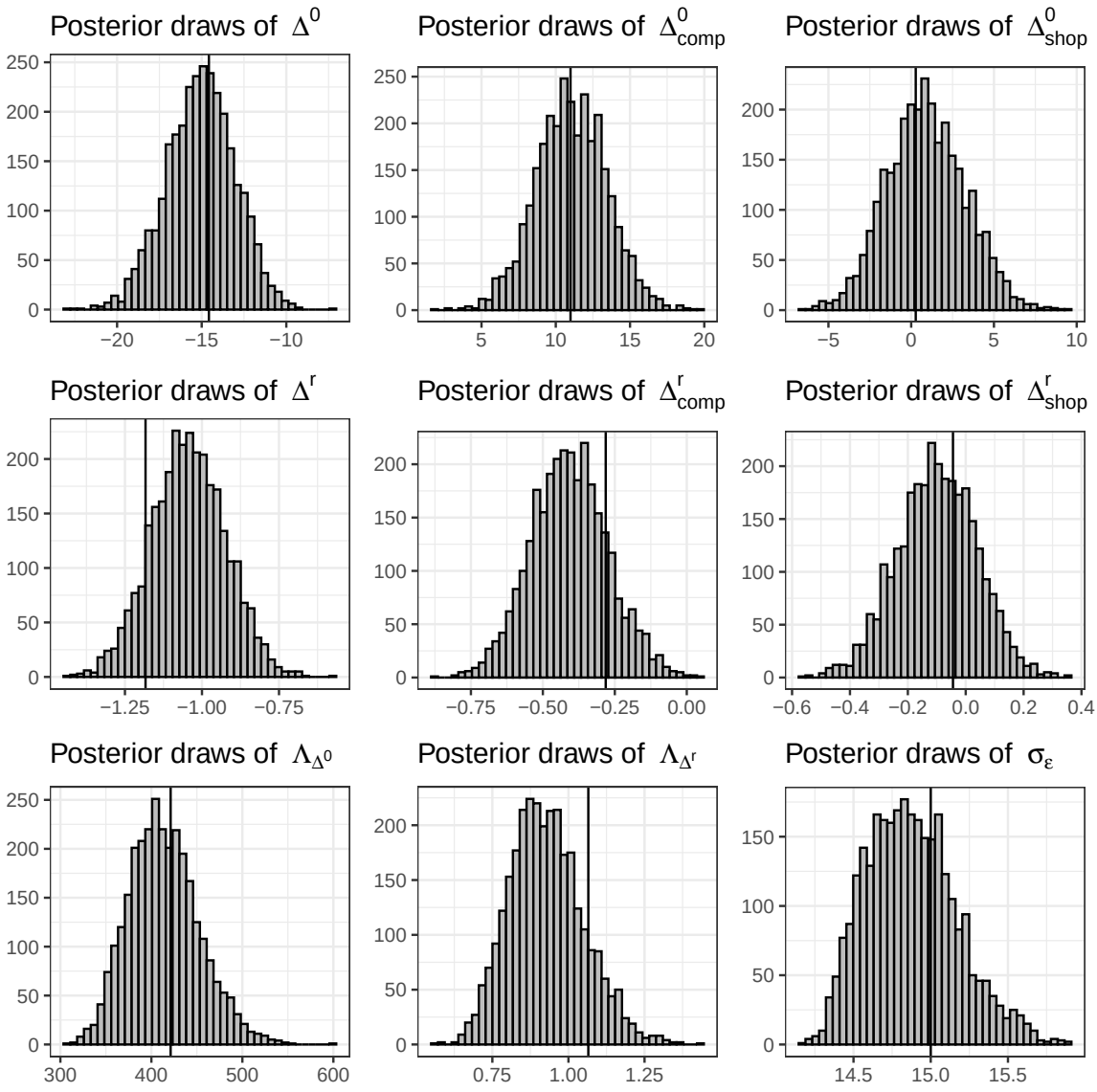
We simulate search behavior as per the learning model described in Study 2. We take the actual data used in the Strong condition ($N = 613$ participants, $T = 15$ tasks of $J = 20$ products each), remove outcomes, and simulate search under a set of known parameters. We choose the search propensity parameters to approximately match the estimates from Simulation 1 and select the prior variance parameter to reasonably replicate the rate of learning we observe in the data. We focus on the Strong condition to show that the model

Figure B.1: Simulation 1a Posterior Draws—Unobservable Heterogeneity Only



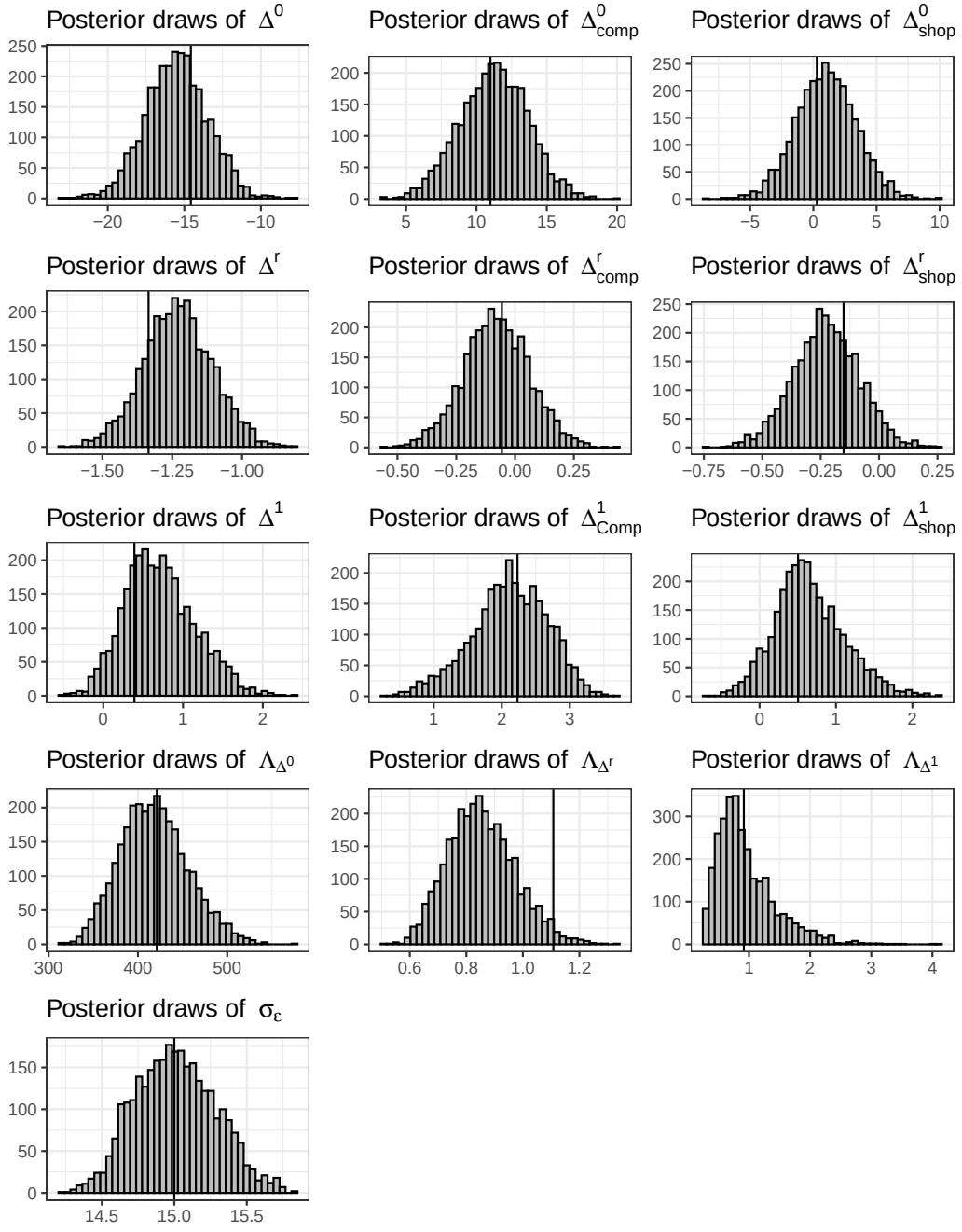
Notes: Each plot provides a histogram of posterior draws for mean search propensity (Δ^0), mean rank effect (Δ^1), the variance in search propensities (Λ^0), the variance in position effects (Λ^1), and the noise term on search decisions (σ^ϵ). The vertical black line corresponds to true value of the parameter.

Figure B.2: Simulation 1b Posterior Draws—Observable and Unobservable Heterogeneity



Notes: Each plot provides a histogram of posterior draws for mean search propensity (Δ^0), mean search propensity FEs (Δ^0_{comp}, Δ^0_{shop}), mean rank effect (Δ^r), mean rank effect fixed slope differences (Δ^r_{comp}, Δ^r_{shop}), the variance in search propensities (Λ_{Δ^0}), the variance in position effects (Λ_{Δ^r}), and the noise term on search decisions (σ^ϵ). The vertical black line corresponds to true value of the parameter.

Figure B.3: Simulation 1c Posterior Draws—Separate Postion 1 Effect



Notes: Each plot provides a histogram of posterior draws for mean search propensity (Δ^0), mean search propensity FEs (Δ^0_{comp}, Δ^0_{shop}), mean rank effect (Δ^r), mean rank effect fixed slope differences (Δ^r_{comp}, Δ^r_{shop}), position 1 rank effects (Δ^1, Δ^1_{comp}, Δ^1_{shop}), the variance in search propensities (Λ_{Δ^0}), the variance in position effects (Λ_{Δ^r}), and the noise term on search decisions (σ^ϵ). The vertical black line corresponds to true value of the parameter.

Table B.4: Simulation 1c Parameter Estimates—Separate Position 1 Effect

Parameter	Coef	Truth	Posterior Mean	Posterior SD
Baseline search propensity	Δ^0	-14.575	-15.759	2.253
Baseline search propensity Comp FE	Δ^0_{comp}	10.993	11.569	2.47
Baseline search propensity Shop FE	Δ^0_{shop}	0.276	0.961	2.375
Rank Coefficient	Δ^r	-1.335	-1.24	0.125
Rank Coefficient Comp FE	Δ^r_{comp}	-0.056	-0.084	0.141
Rank Coefficient Shop FE	Δ^r_{shop}	-0.152	-0.216	0.139
Position 1 Effect	Δ^1	0.39	0.692	0.377
Position 1 Comp FE	Δ^1_{comp}	2.231	2.047	0.55
Position 1 Shop FE	Δ^1_{shop}	0.504	0.788	0.487
Unobs heterogeneity in (β^0)	λ_{Δ^0}	20.522	20.543	0.963
Unobs heterogeneity in (β^r)	λ_{Δ^r}	1.053	0.937	0.066
Unobs heterogeneity in (β^1)	λ_{Δ^1}	0.957	0.889	0.239
Standard deviation of ϵ	σ_ϵ	15	15.075	0.357

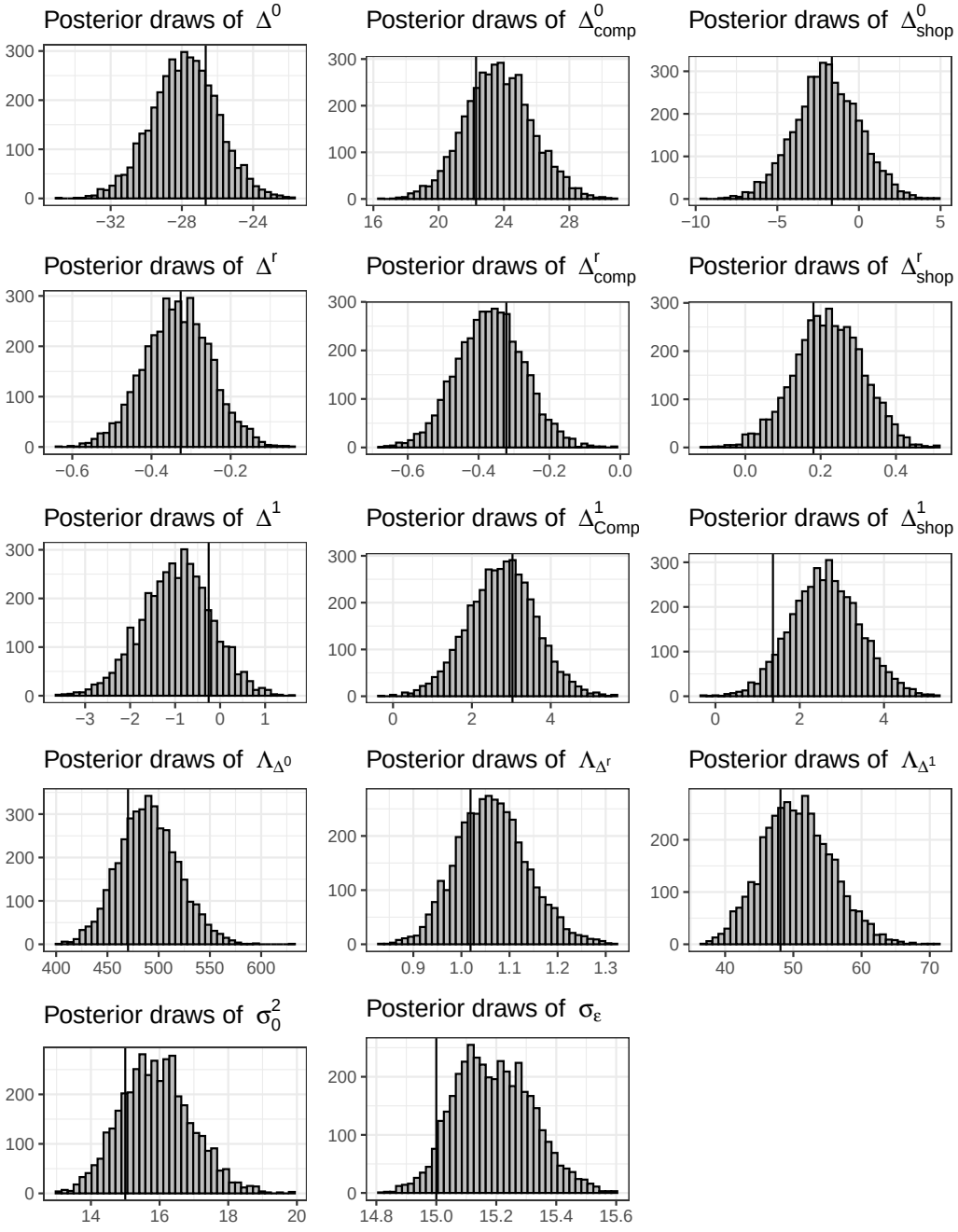
can recover the prior variance parameter (which governs the rate of learning), which is the additional parameter to be estimated in this model.

We estimate the model using 5,000 draws from the hybrid Gibbs sampler. We discard the first 1,000 draws and also conduct inference on the thinned chain using every 10th draw. We are able to recover all parameters, even with large degrees of across-person heterogeneity. The posterior distribution is plotted in Figure B.4 and the estimated mean (SD) of the parameters are reported in Table B.5.

Table B.5: Simulation 2 – Learning Model: Parameter Estimates

Parameter	Coef	Truth	Posterior Mean	Posterior SD
Baseline search propensity	Δ^0	-26.662	-27.383	1.857
Baseline search propensity Comp FE	Δ^0_{comp}	22.288	23.199	2.023
Baseline search propensity Shop FE	Δ^0_{shop}	-1.658	-1.959	1.859
Rank Coefficient	Δ^r	-0.326	-0.331	0.081
Rank Coefficient Comp FE	Δ^r_{comp}	-0.322	-0.37	0.092
Rank Coefficient Shop FE	Δ^r_{shop}	0.181	0.219	0.082
Position 1 Effect	Δ^1	-0.255	-0.879	0.776
Position 1 Comp FE	Δ^1_{comp}	3.03	2.716	0.842
Position 1 Shop FE	Δ^1_{shop}	1.365	2.591	0.774
Unobs heterogeneity in (β^0)	λ_{Δ^0}	21.683	21.987	0.802
Unobs heterogeneity in (β^r)	λ_{Δ^r}	1.009	1.029	0.035
Unobs heterogeneity in (β^1)	λ_{Δ^1}	6.936	7.097	0.341
Prior Variance	σ_0^2	15	15.963	1.062
Standard deviation of ϵ	σ_ϵ	15	15.181	0.149

Figure B.4: Simulation 2 Posterior Draws—Learning Model



Notes: Each plot provides a histogram of posterior draws for mean search propensity (Δ^0), mean search propensity FEs (Δ^0_{comp}, Δ^0_{shop}), mean rank effect (Δ^r), mean rank effect FEs (Δ^r_{comp}, Δ^r_{shop}), position 1 rank effects (Δ^1, Δ^1_{comp}, Δ^1_{shop}), the variance in search propensities (Λ_{Δ^0}), the variance in position effects (Λ_{Δ^r}), the prior variance (σ_0^2) and the noise term on search decisions (σ^ϵ). The vertical black line corresponds to true value of the parameter.

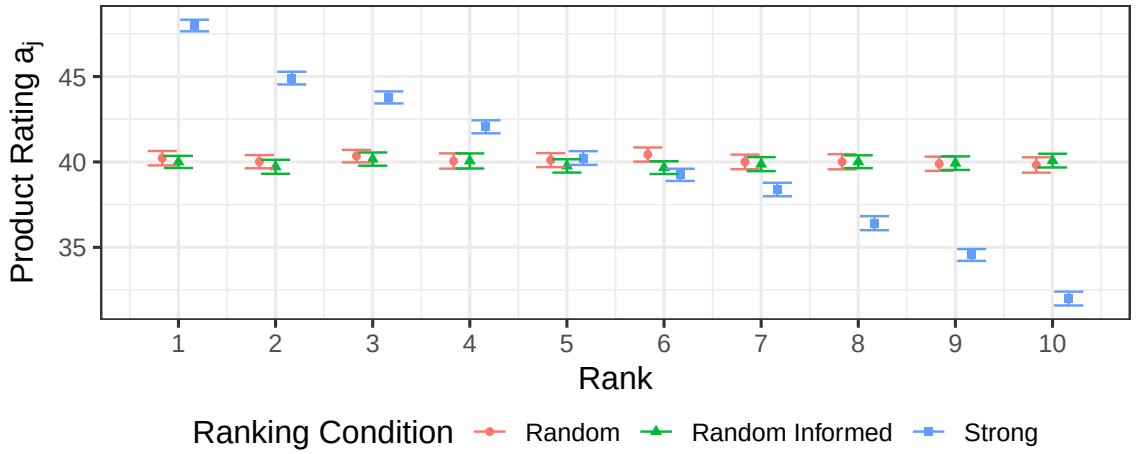
C Additional Studies

C.1 Study 1B: Correlated Attributes

C.1.1 Design

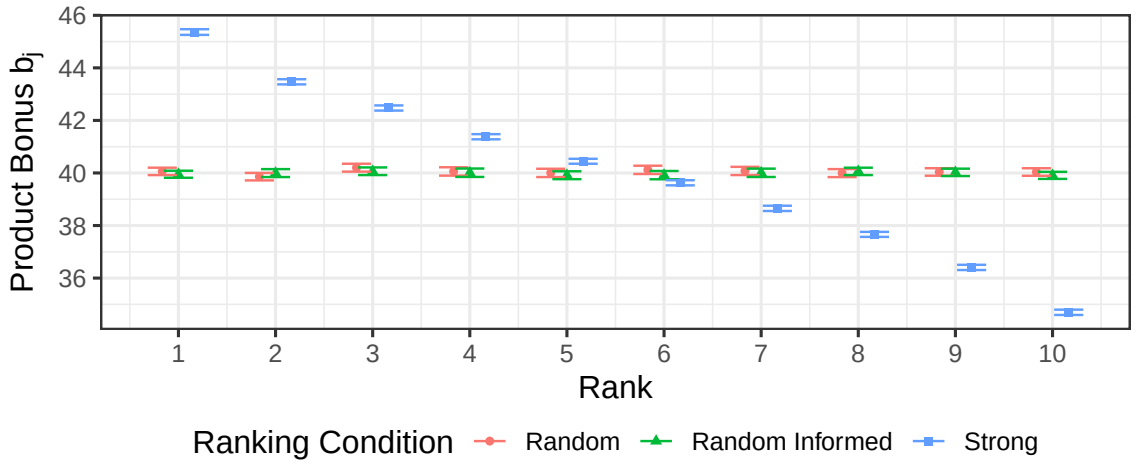
We constructed an alternative version of our first search task paradigm—Study 1B—which incorporates an additional feature of many search markets: informative pre-search signals. We redesign the attribute space to achieve this. We draw bonus b from a Normal distribution with mean 40 and standard deviation of 4. We eliminate bonus a as a component of the incentives and replace it with an informative rating (also called a here). The rating is a noisy but unbiased prediction of bonus b, which we create by adding $N(0, 10)$ draws. We show the average values by rank and condition of the rating a_j and the bonus payoff b_j in Figures C.1 and C.2.

Figure C.1: Rating by Rank by Condition (Study 1B)



Notes: Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level.

Figure C.2: Bonus by Rank by Condition (Study 1B)



Notes: Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level.

C.1.2 Results

Participants in Study 1B are more likely to start searching at the highest rating (max a) option first. Because options with high ratings (a_j) also have high payoffs (b_j), this difference is also reflected in shorter search sequences relative to Study 1.

Table C.1: Summary Statistics across Conditions (Study 1B)

	Condition Mean (SD)		
	Random	Random Informed	Strong
Num Searches	1.32 (0.71)	1.4 (0.93)	1.38 (0.87)
Bonus Earned	40.77 (3.75)	40.62 (3.8)	40.92 (3.98)
Click Max(a) First	0.69 (0.46)	0.71 (0.46)	0.69 (0.46)
N(Tasks)	2690	2860	2810
N(Participants)	269	286	281

Notes: Table reports mean and standard deviation by condition for each variable. Num Searches is the number of products searched in a task. Bonus Earned is the net bonus earned in a task (bonus of selected product net of nominal search costs). Click Max(a) First is a dummy variable which is equal to one if the participant clicked on the product with the highest rating first in a task. The prior statistics summarize across participants and tasks within a condition.

Despite searching less overall, participants in Study 1B display similar position effects to participants in Study 1 across conditions (Table C.2). We conclude from these results that the primitive rank-specific search costs are likely robust to alternative information structures.

Table C.2: Study 1B: Effect of Rank on Search by Condition

Dependent Variable:	1(Product Searched)					
Model:	(1)	(2)	(3)	(4)	(5)	(6)
Variables						
$r \times$ Condition = Random	-0.0037***	-0.0032***	-0.0037***	-0.0032***	-0.0037***	-0.0032***
	(0.0008)	(0.0007)	(0.0008)	(0.0007)	(0.0008)	(0.0007)
$r \times$ Condition = Random Informed	-0.0030***	-0.0030***	-0.0030***	-0.0030***	-0.0030***	-0.0030***
	(0.0009)	(0.0008)	(0.0009)	(0.0008)	(0.0009)	(0.0008)
$r \times$ Condition = Strong	-0.0282***	-0.0075***	-0.0282***	-0.0075***	-0.0282***	-0.0074***
	(0.0009)	(0.0009)	(0.0009)	(0.0009)	(0.0009)	(0.0009)
a_j (demeaned)		0.0129***		0.0129***		0.0130***
		(0.0002)		(0.0002)		(0.0002)
Fixed-effects						
Condition (3)	Yes	Yes	Yes	Yes		
Search Task (10)			Yes	Yes	Yes	Yes
Participant (836)					Yes	Yes
Fit statistics						
Observations	83,600	83,600	83,600	83,600	83,600	83,600
R^2	0.01933	0.18157	0.01939	0.18165	0.05091	0.21431
Within R^2	0.01925	0.18150	0.01925	0.18153	0.01988	0.18862

Clustered (Participant) standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

C.2 Study 2B: Zero Marginal Cost

In our studies in the main paper, we impose an explicit marginal search cost of 1 point. We impose this cost because participants who are paid to take a survey might weigh their outside option differently from those conducting a real search. However, a concern with imposing this marginal cost is that it creates another difference between our experimental setting and the field: in field settings, consumers do not pay an explicit search cost to click on products, but rather make the trade-off between searching again and choosing either the outside option or the best-searched alternative. In this section, we present results from a pilot study where we implemented a version of Study 2 that has zero explicit marginal search cost (i.e., it does not cost any points to search a product).

This pilot study includes 89 participants, randomized across 4 experimental conditions (Random-Informed, Strong, Strong-to-Random5, and Strong-to-Pref5). Like in Study 2, each participant performs each search task fifteen times, with twenty products in each search task. Table C.3 reports the summary statistics across conditions. The first row shows that participants in this pilot search more than those in the main Study 2, where participants search between 1.78–1.99 products on average. (Table 4, row 1). As a result, they spend longer per search task. In addition, pilot participants are less likely to search the product with the highest bonus A first; in Random-Informed, where this is the optimal strategy, participants search the product with the highest bonus A 20% of the time (row 3), as compared to 31% of the time in Study 2 (Table 4, row 3). This suggests that without the explicit marginal search cost, participants deviate more from optimal sequential search. Furthermore, in the pilot, participants searched all twenty products in 12% of the search tasks, compared to only 0.1% of tasks in the main Study 2. A high proportion of tasks where all products are searched presents a challenge for estimating search costs, as these cases only allow us to determine the upper bound of the search costs.

Table C.3: Summary Statistics across Conditions (Study 2, Zero Marginal Cost)

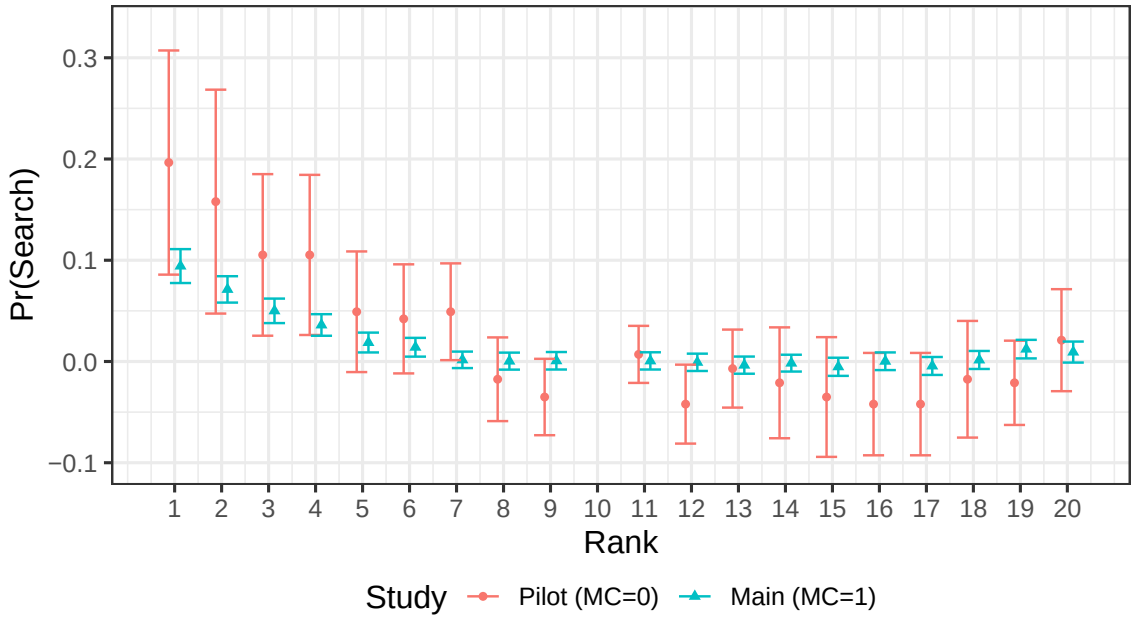
	Condition Mean (SD)			
	RI	Str	Str to P (5)	Str to R (5)
Num Searches	7.4 (7.92)	4.55 (5.15)	6.34 (6.84)	5.07 (5.58)
Bonus Earned	67.2 (10.52)	73.15 (10.43)	70.47 (12.03)	69.8 (10.43)
Click Max(a) First	0.2 (0.4)	0.23 (0.42)	0.18 (0.39)	0.22 (0.41)
Time Spent per Task	24.25 (24.57)	15.92 (16.8)	21.79 (21.6)	19.06 (18.3)
N(Tasks)	285	315	405	330
N(Participants)	19	21	27	22

Notes: "Str" stands for Strong, "P" for Pref, "R" for Random, and "RI" for Random-Informed. Table reports mean and standard deviation by condition for each variable. Num Searches is the number of products searched in a task. Bonus Earned is the total bonus of the selected product in a task. Click Max(a) First is a dummy variable which is equal to one if the participant searched the product with the highest ex-ante bonus value first in a task. Time Spent per Task is the seconds spent on a task from page load until final click. The prior statistics summarize across participants and tasks within a condition.

Figure C.3 plots the average probability of search by rank for the pilot and for main study 2. For ease of comparison, we include only the Random-Informed condition. Because

the average probability of search for each item is higher in the pilot, we plot the probability of search relative to the tenth-ranked product. The pilot's sample size is relatively small, so we do not have the statistical power to detect differences. However, it appears that position effects are stronger in magnitude when the explicit marginal cost is zero. That is, participants are more likely to search the first product relative to the tenth in the pilot than in the main study. A possible explanation is that in the pilot study, participants rely less on Bonus A to guide their search (consistent with the summary statistics above) because they do not feel constrained by how many products to search.

Figure C.3: Search Probability by Rank for Random-Informed



Notes: This figure plots the average probability of searching each product by its rank, relative to the tenth product, for participants in the Random-Informed condition. Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level.

Our results suggest that having zero explicit marginal search cost leads to search behavior that is *less* consistent with the search patterns observed in real-world contexts. In particular, it leads to greater search intensity compared to that reported in prior research (e.g., Morozov, 2023; Ursu, 2018), and it leads to search behavior that is less consistent with optimal behavior predicted by the Weitzman sequential search model.

Notably, the introduction of even a minimal marginal search cost—such as one point or 0.2 cents—has a surprisingly large impact on search intensity. We believe this effect is unique to the transition from zero to one point. Introducing a marginal cost of one point makes the trade-off between stopping and continuing search more salient to participants, thereby resulting in search behavior more consistent with the Weitzman model. We anticipate smaller differences in search behavior when increasing the marginal search cost from one to two points.

C.3 Study 2C: Higher Stakes

We implement a robustness check where we raised the incentives in the search task. This study is similar to Study 2, with a few exceptions: (1) we increased the payoffs from search by using a conversion rate of 200 points per dollar (instead of 500), (2) participants conduct ten search tasks (instead of fifteen) and only include the Strong and Random-Informed conditions (for cost considerations), (3) decreased the point value of the marginal search cost to 0.5 (instead of 1), and (4) payoffs were summed over all tasks (instead of eight out of fifteen). Otherwise, everything else, such as the distributions of the bonus points, are the same as in Study 2. This study includes 1,328 participants.

We compare the search behaviors with that of Study 2 by combining the data from the "higher stakes" study with the data from each participants' first ten tasks in Study 2. We then estimate the following regression:

$$y_{it} = \alpha_t + \beta_1 \mathbb{1}\{HigherStakes_i\} + \beta_2(\mathbb{1}\{HigherStakes_i\} \times Condition_i) + \epsilon_{it}. \quad \text{(C.1)}$$

We consider two dependent variables for participant i in task t: the number of products searched, and an indicator for whether i searches the product with the highest bonus A first. Recall that searching the product with the highest bonus A first is the optimal strategy in Random-Informed. Note that we do not include participant fixed effects in this specification because they are multicollinear with the $\mathbb{1}\{HigherStakes_{it}\}$. We report the estimates in Table C.4. Here, we find that participants search 0.27 more products per search task in the higher stakes study (column 1). This is expected because the returns to search are greater. However, we do not see differences in treatment effects on search depth, as indicated by the null interaction effect between condition and $\mathbb{1}\{HigherStakes_{it}\}$. This shows that having greater incentives does not have a differential effect on our treatment effects, indicating that our treatment effects should generalize to settings with higher incentives.[46]

The second column shows the effect on whether the participant searches the product with the highest bonus A first, as an indicator of optimal search (for Random Informed). Again, we do not find differences between the higher stakes study and Study 2, which suggests raising stakes does not alter the degree of optimality in search versus our main studies. That is, raising the incentives for attention does not seem to change the degree to which participants pay attention to bonus A, only their willingness to search further (since it translates to higher payoffs in dollar terms). However, we cannot rule out that even higher stakes would not generate behavior that more closely aligns with the Weitzman search model.

We also measure whether higher stakes differentially impacts position effects by estimating the following:

$$\mathbb{1}\{Searched_{itj}\} = \alpha_t + \beta(r_{itj} \times Condition_i \times \mathbb{1}\{HigherStakes_i\}) + \epsilon_{itj}. \quad \text{(C.2)}$$

This regression has a three-way interaction between the product j's rank r_{itj}, the partici-

[46] Note that this is not explicitly an experimental manipulation, since whether an observation was generated with higher stakes is not randomized within a single study. Instead, we combine non-overlapping samples from the same participant pool and compare the same experimental manipulation (RI vs Strong) across samples.

Table C.4: Differences in Search Behavior Between Study 2 and Study 2C (Higher Stakes)

Dependent Variables:	Search Depth	Searched Highest b_A First
Model:	(1)	(2)
Variables		
HigherStakes	0.2660***	-0.0187
	(0.0897)	(0.0180)
Condition = Strong	-0.1157**	-0.0228
	(0.0572)	(0.0171)
Condition = Strong × HigherStakes	-0.0312	-0.0141
	(0.1234)	(0.0246)
Fixed-effects		
Search Task (10)	Yes	Yes
Fit statistics		
Observations	21,910	21,910
R^2	0.00596	0.00243
Within R^2	0.00567	0.00190

Clustered (Participant) standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Notes: This table presents the OLS estimates of Equation C.1.

pant's condition, and whether the participant is in the higher stakes study. We report the estimates in Table C.5. We find that the higher stakes study does not show significantly different position effects than Study 2; the coefficients for the interaction between whether the participant is in the higher stakes study and rank, and with "Condition = Strong" are not statistically significant and close to zero. This demonstrates that our conclusions from our main study are robust to higher stakes.

Table C.5: Differences in Search Behavior Between Study 2 and Study 2C (Higher Stakes)

Dependent Variable:	1(Product Searched)
Model:	(1)
Variables	
r	-0.0029***
	(0.0003)
r × Condition = Strong	-0.0013***
	(0.0004)
r × HigherStakes	-0.0004
	(0.0004)
r × Condition = Strong × HigherStakes	-0.0002
	(0.0006)
Condition = Strong	0.0083
	(0.0058)
HigherStakes	0.0174**
	(0.0073)
Condition = Strong × HigherStakes	5.5×10^{-5}
	(0.0101)
Fixed-effects	
Search Task	Yes
Fit statistics	
Observations	438,200
R^2	0.00621
Within R^2	0.00618

Clustered (Participant) standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Notes: This table presents the OLS estimates of Equation C.2.

D Learning Across Positions

How do participants learn to find better bonus Bs? Do participants learn that certain ranks yield higher bonus Bs, and does this learning spill over (Hodgson and Lewis, 2023) to adjacent positions? For example, if a participant clicks on the second-ranked product and uncovers a large bonus B, are they more likely to search the second-ranked product in the next search task? If learning spills over to nearby ranks, we might expect the participant to also be more likely to search the products in ranks one or three in the next task. Understanding whether spillovers exist allows us to distinguish learning about specific actions (searching specific ranks) from learning about the environment (searching products in adjacent ranks with predicted good payoffs) to model learning. To test for rank-specific learning, we estimate the following linear regression:

$$\begin{aligned}\mathbb{1}\{\text{Searched Product } r\}_{i,t} = \alpha_t + \alpha_r + \beta_1 \mathbb{1}\{\text{Searched Product } r\}_{i,t-1} + \\ \beta_2 \mathbb{1}\{\text{Searched Product } r\}_{i,t-1} \times b_{i,t-1,r} + \beta_3 b_{i,t-1,r} + \\ \beta_4 a_{i,t,r} + \beta_5 NSearched_{i,t} + \epsilon_{i,t} \quad \text{(D.1)}\end{aligned}$$

The dependent variable $\mathbb{1}\{\text{Searched Product } r\}_{i,t}$ is an indicator for whether participant i searches the product in rank r in the task, t; $\mathbb{1}\{\text{Searched Product } r\}_{i,t-1}$ is an indicator for whether i searches the product in rank r in the previous task, $t-1$. The term $b_{i,t-1,r}$ is the bonus B of the product in rank r in task $t-1$. We control for $a_{i,t,r}$, the bonus A of the r-ranked product in task t. This accounts for the possibility that individual i clicks on r in the current task simply because it happens to have a high bonus A. We also control for the number of products searched in t ($NSearched_{i,t}$). This accounts for the possibility that i searches r simply because this individual searches many products. However, one may be concerned that $NSearched_{i,t}$ is influenced by actions at $t-1$. Therefore, we estimate another specification of the regression that includes only the first search in t, which allows us to drop $NSearched_{i,t}$. To measure whether learning spills over to nearby ranks, we also include alternative dependent variables: whether the individual searches the product $r+1$ and $r-1$.

We estimate this regression for participants in Study 2 in the Strong condition and for tasks where the individual searches only one product at task $t-1$.[47] We impose the former restriction for two reasons. First, we do not expect to find position-specific learning in Random-Informed because participants are informed that all products are ordered randomly. Second, other conditions (e.g., Strong-to-Random5) have varying number of tasks with the strong algorithm, and the degree of learning can vary across tasks; focusing on the Strong condition allows us to keep the number of tasks constant. We impose the latter restriction of single-search tasks because it allows us to attribute the effect to the bonus B of product r, as opposed to the bonus Bs of other positions the participant searched.

Table D.1 displays the OLS estimates of Equation D.1. Column 1 reports the estimates for whether the individual searches the identically positioned product in $t+1$ as in t. The coefficients of $\mathbb{1}\{\text{Searched Product } r\}_{i,t}$ and its interaction with b_{itr} imply that conditional on searching a product, the individual is more likely to search the product in the same rank

[47] Of all tasks in the Strong condition, 62% involve only one search.

Table D.1: Learning Across Positions (Study 2)

Dependent Variables:	Search r	Search $r+1$	Search $r-1$	Search First r	Search First $r+1$	Search First $r-1$
Model:	(1)	(2)	(3)	(4)	(5)	(6)
Variables						
1(r Searched in $t-1$)	-0.2105***	-0.0041	0.0083	-0.2027***	0.0218	0.0167
	(0.0260)	(0.0182)	(0.0171)	(0.0263)	(0.0170)	(0.0155)
1(r Searched in $t-1$) × Bonus B	0.0059***	0.0006	5.3×10^{-5}	0.0057***	-2.39×10^{-6}	-0.0003
	(0.0007)	(0.0004)	(0.0004)	(0.0007)	(0.0004)	(0.0004)
Bonus B	-0.0003**	9.89×10^{-5}	-8.01×10^{-5}	-0.0002*	0.0001	-4.82×10^{-5}
	(0.0001)	(0.0001)	(0.0001)	(0.0001)	(0.0001)	(0.0001)
Next Bonus A	0.0099***	0.0098***	0.0100***	0.0074***	0.0072***	0.0075***
	(0.0005)	(0.0005)	(0.0005)	(0.0004)	(0.0004)	(0.0004)
Num Searched t	0.0495***	0.0496***	0.0490***			
	(0.0001)	(0.0004)	(0.0003)			
Fixed-effects						
Search Task (14)	Yes	Yes	Yes	Yes	Yes	Yes
r	Yes	Yes	Yes	Yes	Yes	Yes
Fit statistics						
# r	20	19	19	20	19	19
Observations	106,800	101,460	101,460	106,800	101,460	101,460
R^2	0.08758	0.07539	0.07983	0.03872	0.02299	0.03129
Within R^2	0.07135	0.06913	0.06277	0.02691	0.02024	0.01896

Clustered (Participant) standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

in the subsequent task if the searched product has a higher bonus B in the current task. Columns 2 and 3 report whether participants who discover that product r has a higher bonus B are more likely to search product $r+1$ or $r-1$ in the next task, respectively. The coefficients of the interactions in columns 2 and 3 are closer to zero and not statistically significant. Columns 4–6 repeat these regressions with an indicator for searching that item first as the dependent variable and drops the independent variable $NumSearchedt + 1$. Similarly, we find that the learning about one position does not spill over onto the adjacent products across tasks (columns 5 and 6, very small and n.s. coefficient on the interaction). Therefore, we find evidence of position-specific learning but no spillovers across positions. These results are consistent with individuals forming beliefs about the bonus B of product r based on past realizations of bonus B for product r, and such realizations do not impact beliefs about nearby products. These results motivate how we model learning.

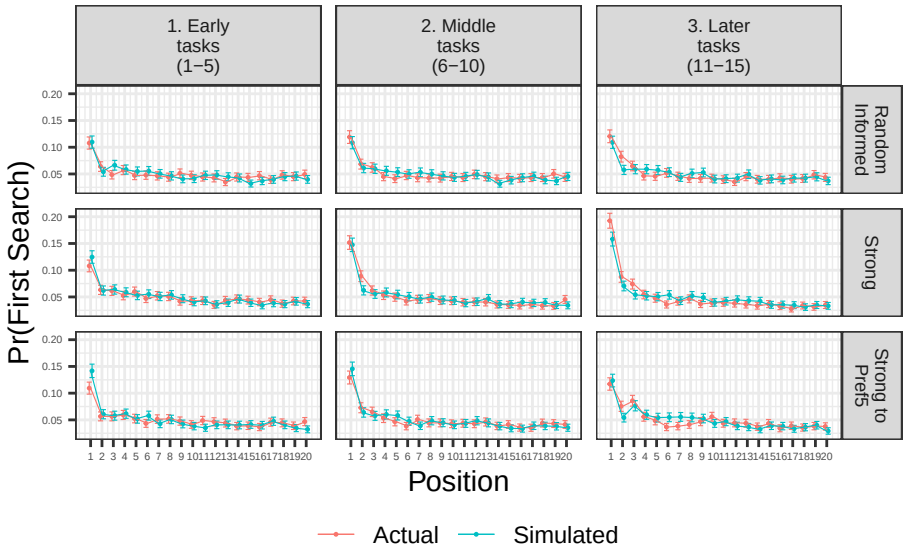
The lack of learning spillovers to other positions may not generalize to field settings. For example, on Amazon, one may infer that if the top products have higher prices, the bottom ones have lower prices. This means that consumers form beliefs about the correlation between hidden attributes and position, rather than form position-specific beliefs. This can easily be accommodated in the model, where the term $E[b_{itj}|r_{itj}]$ in Equation 15 can be parameterized as $\gamma \cdot r_{itj}$, and consumers update beliefs about γ. Our general conclusion—that beliefs can confound the identification of search costs—are not specific to settings without learning spillovers across positions. However, we expect that in learning with spillovers, consumers may learn faster, and therefore update their beliefs to the new equilibrium more quickly.

E Model Fit

How well do the model predictions fit the observed search patterns? We simulate search behavior based on the model described in Section 4.3 using the estimated parameters and then compare the first search probability by rank with the corresponding probabilities observed in the actual data. Figure E.1 plots this comparison, for conditions Random-Informed, Strong and Strong-to-Pref5. The figure shows that the model recovers underlying parameters reasonably well. The simulated and actual search patterns are very similar for Random-Informed. In Strong, the simulated search predicts an increase in probability of first searching the top-ranked product—matching what we see in the actual data.[48] Finally, in Strong-to-Pref5, the model is also able to predict the 'peak' beginning to form at rank 3 in tasks $11-15$, consistent with the observed search patterns and the underlying algorithm. The plots for the other conditions are in Figure E.2.

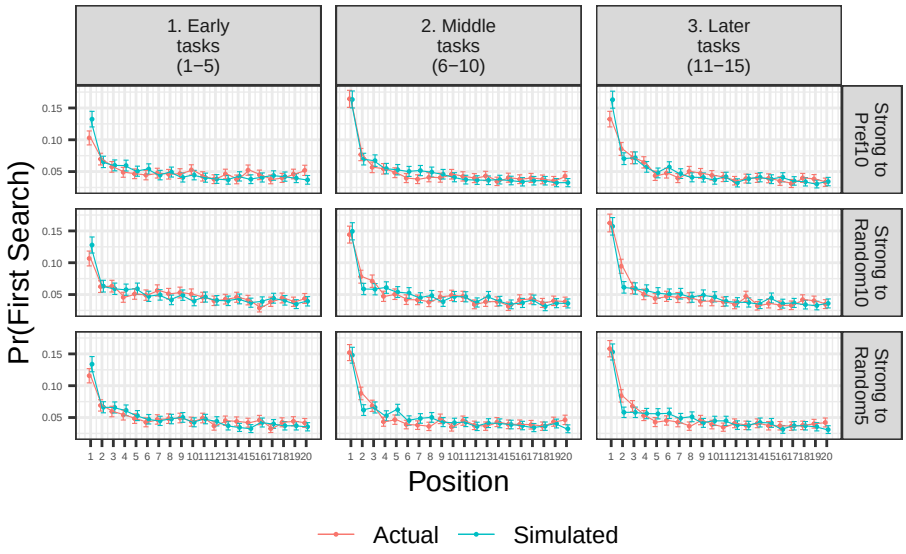
[48]The probability of searching rank 1 in tasks 11–15 in *Strong* is slightly understated by the model (middle right panel in Figure E.1). This suggests that there may be some form of learning that occurs which is not captured by our model. However, given the trade-off between accuracy and model tractability, we feel that our model reasonably captures the essential elements of the consumer search process.

Figure E.1: Actual versus Model-Predicted Search Patterns (Study 2)



Notes: The above figures plot the actual observed first click probability by rank versus the same probability as predicted by the model described in the preceding section, based on estimated parameters. The left, middle and right panels show the search probabilities for tasks 1–5, 6–10 and 11–15 respectively, to illustrate the evolution in search probabilities as participants learn about the underlying algorithm and update their beliefs.

Figure E.2: Actual versus Model-Predicted Search Patterns (Study 2)



Notes: The above figures plot the actual observed first click probability by rank vs the same probability as predicted by the model described in the preceding section, based on estimated parameters. The left, middle and right panels show the search probabilities for tasks $1-5$, $6-10$ and $11-15$ respectively, to illustrate the evolution in search probabilities as participants learn about the underlying algorithm and update their beliefs.

F Counterfactuals

In this section, we describe in detail how we obtain our surplus predictions.

F.1 Comparing Surplus with and without Belief Adjustments

We simulate behavior for a simulated set of 594 consumers with the same attributes (e.g., comprehension check accuracy and shopping frequency) as those in our Study 2 sample. We then predict consumer surplus for thirty search tasks for each consumer using the estimates from models that account or do not account for position-specific beliefs. In the first five of the thirty tasks, the ranking algorithm is *Strong*, and in the remainder of the tasks, the algorithm is *Pref*. We generate the bonuses for each of these tasks by sampling from the pool of tasks under the same algorithm that generated the bonuses in the experiment. We first describe how we generate the surplus predictions with beliefs (and thus, learning) and without beliefs.

1. Estimate the model described in Section 4.3 using the data generated from tasks 6 and 7 in the Strong-to-Random5 condition. We assume away the possibility that beliefs have adjusted over the course of the first five tasks under *Strong* rankings: the expected belief about bonus B is 40 for all ranks. We refer to these estimates as the "incorrect" estimates (i.e., conflating beliefs with costs).

2. Estimate the model described in Section 5 using the data generated from tasks 6 and 7 in the Random-Informed condition. In other words, this is the sequential search model in which all position effects are attributed to search costs, and the expected belief about bonus B is 40 for all ranks. We refer to these estimates as the "correct" estimates." We re-estimate this model using the same set of tasks, as opposed to using the estimates from Random-Informed in Section 4.3, to provide a fair comparison to the model in Step 1.

3. For each product in each task, draw ϵ_{itj} using the estimate of σ_ϵ from the correct estimates. We do this to keep the ϵ_{itj} draws constant when comparing the two model predictions.

4. Using the *correct* estimates, draw $\beta_i^0, \beta_i^1, \beta_i^r$ for each user.

5. Simulate search behavior for thirty tasks for each user under the draws from the correct estimates in Step 4. We assume that at the start of the first task, the prior mean of each position is 40, as in the experiment, and σ_0, the variance of their prior belief is 11, which is the estimate from Study 2 (Table 6). Position-specific beliefs are updated after each task based on sampled (searched) products.

6. Using the *incorrect* estimates, draw $\beta_i^0, \beta_i^1, \beta_i^r$ for each user.

7. Simulate search behavior for all thirty tasks for each user under the draws from the incorrect estimates in Step 6. We simulate under the typical assumption that there are no position-specific beliefs ($E[b_r] = 40 \forall r$).

8. Calculate the consumer's surplus for each task for the search behavior generated under the correct and incorrect estimates. This is defined as the bonus A + bonus B of the selected product minus the incurred search costs. With the incorrect estimates, we compute costs by inverting the full search propensity, including any residual effects of beliefs: $c_{itj} = \sigma_b \zeta^{-1}(\frac{\delta_{itj} - \epsilon_{itj}}{\sigma_b})$

9. Take the average of consumer over consumers for each task.

10. Repeat Steps 3–9 200 times.

11. Take the average surplus per task across all 200 iterations. The confidence intervals are the 2.5 and 97.5 percentiles.

F.2 Auxiliary Model Estimates

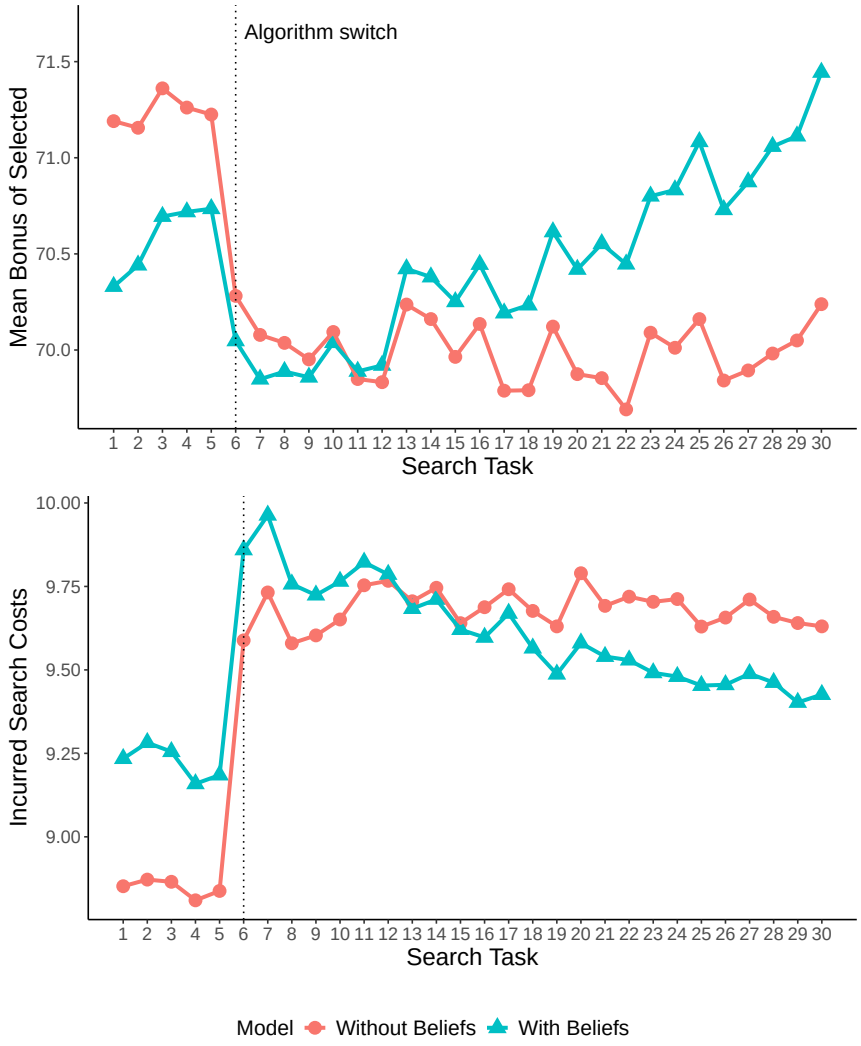
Table F.1: Estimation Results—Auxiliary Models Estimated on Tasks 6-7 (Study 2)

		Estimate (SD)		
Variable	Coefficient	Random Informed	S2R5	S2R5 + Rat. Exp.
Rank Coefficient β^r				
Intercept	Δ_0^r	-0.11 (0.059)*	-0.21 (0.063)***	1.5 (0.073)***
Comp	Δ_{comp}^r	-0.059 (0.062)	-0.05 (0.064)	-0.051 (0.081)
ShopFreq	Δ_{shop}^r	-0.023 (0.047)	-0.052 (0.061)	-0.052 (0.077)
Pos 1 Coefficient β^1				
Intercept	Δ_0^2	2.3 (1)***	3.8 (0.94)***	-0.088 (0.83)
Comp	Δ_{comp}^2	0.34 (0.99)	0.43 (0.94)	-0.61 (0.61)
ShopFreq	Δ_{shop}^2	-0.12 (0.81)	-0.48 (0.95)	0.17 (0.48)
Baseline Search Propensity β^0				
Intercept	Δ_0^0	-23 (1.5)***	-23 (1.5)***	-39 (1.6)***
Comp	Δ_{comp}^0	8.9 (1.6)***	10 (1.7)***	10 (1.8)***
ShopFreq	Δ_{shop}^0	-2.7 (1.4)**	-0.85 (1.5)	-0.73 (1.6)
Heterogeneity				
Rank Coefficient	$\sqrt{V}_{\beta^r}$	0.41 (0.033)	0.5 (0.037)	0.63 (0.052)
Pos1 Coefficient	$\sqrt{V}_{\beta^1}$	4.1 (6.9)	5.3 (9.2)	0.8 (0.77)
Mean Search Propensity	$\sqrt{V}_{\beta^0}$	15 (18)	16 (21)	17 (23)
Reservation Utility	σ_ϵ	11 (8.2)	12 (8.2)	13 (11)

Notes: Table reports posterior means and standard deviations for three models, all estimated only on tasks 6 and 7. Random-Informed refers to estimates using only Random-Informed participants. "S2R5" refers to estimates which use only Strong-to-Random5 participants. These two columns assume that there is no belief component to search. "S2R5 + Strong Beliefs" uses the same Strong-to-Random5 participants and assumes that participants have correct (in expectation) pre-switch beliefs based on the *Strong* algorithm from tasks 1–5. Posterior means and standard deviations are based on the thinned chain which drops the first 1,000 draws of the chain and keeps every tenth draw thereafter. We omit variance of heterogeneity results for brevity. Signif. Codes: $***: 0.01, **: 0.05, *: 0.1$

F.3 Counterfactual Search Behavior

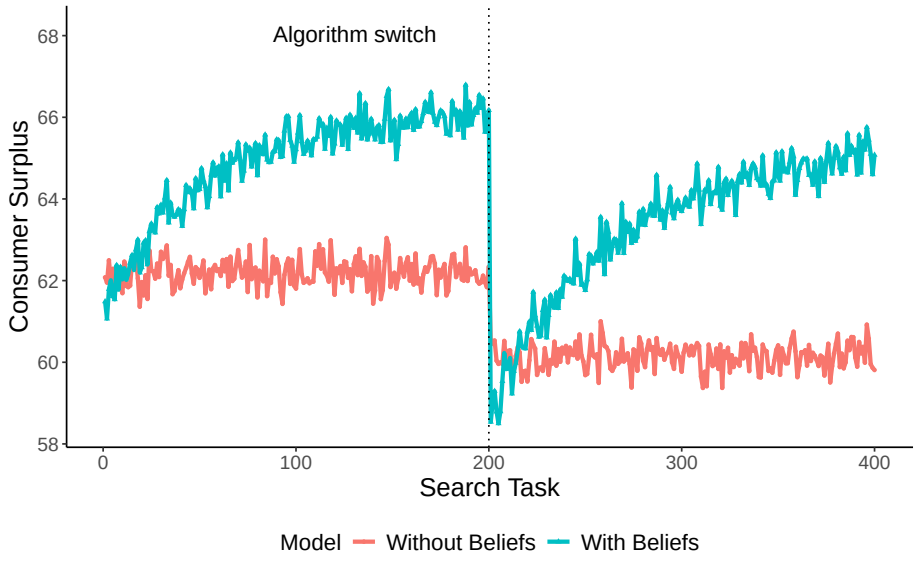
Figure F.1: Bonus and Costs with and without Accounting for Beliefs



Notes: Figure plots the simulated bonus earned (top panel) and total search costs incurred (bottom panel) per consumer for each search task. We calculate the average surplus (bonus chosen - incurred search costs) per consumer per task, given a draw of consumer parameters. We repeat this 200 times. The error bars represent the 95% confidence intervals, which are taken over these iterations. The algorithm switches from *Strong* to *Pref* after task 5.

F.4 Equilibrium Surplus Comparison

Figure F.2: Counterfactuals: Consumer Surplus Effects in Equilibrium



Notes: Figure plots the simulated average consumer surplus per consumer for each search task. We calculate the average surplus (bonus chosen - incurred search costs) per consumer per task, given a draw of consumer parameters. We repeat this 200 times. The ranking algorithm switches from *Strong* to *Pref* after task 200. "Without Beliefs" refers to the model estimated on the Strong-to-Random5 data after the switch to *Random* rankings. "With Beliefs" refers to the model estimated using the learning parameter in the main model and search costs recovered from Random-Informed.

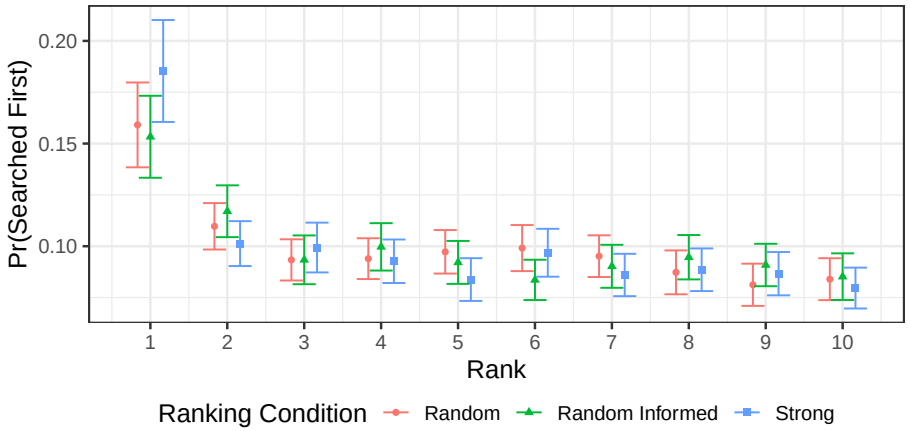
G Additional Figures and Tables

G.1 Reduced-Form Position Effects

G.1.1 Study 1

The main paper reports the position effects as the probability that a participant searches an item. Position effects are starker when we examine the first product searched. The top-ranked product is significantly more likely to be searched in all conditions (Figure G.1). For participants in all conditions, the position effects on the first click *get larger* over the ten tasks. This is not driven by large changes in the average search depth over tasks (Figure G.2).

Figure G.1: Pr(Search First) by Rank by Condition (Study 1)



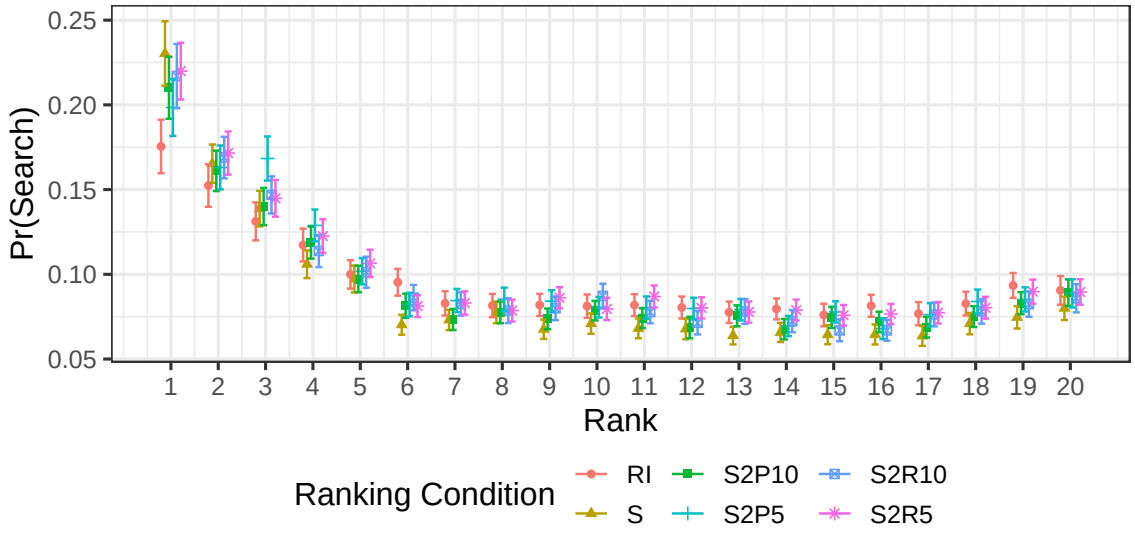
Notes: Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level.

Figure G.2: Depth of Search across Tasks (Study 1)



G.1.2 Study 2

Figure G.3: Search Probability by Rank by Condition (Study 2): All Ranks



Notes: Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level. "RI" refers to Random-Informed, "S" refers to Strong, "S2P10" and "S2P5" refer to Strong-to-Pref10 and Strong-to-Pref5, respectively, and "S2R10" and "S2R5" refer to Strong-toRandom10 and Strong-to-Random5, respectively.

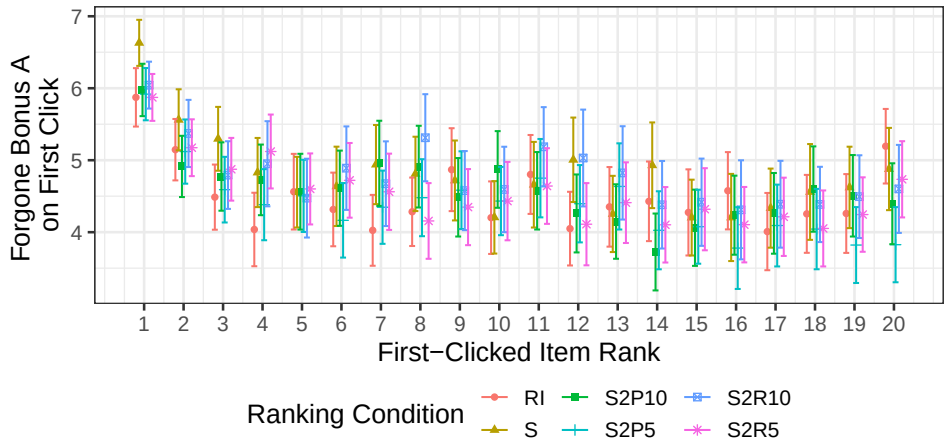
Table G.1: Effect of Rank on Search by Condition (Study 2)

Dependent Variable:	1(Product Searched)					
Model:	(1)	(2)	(3)	(4)	(5)	(6)
Variables						
$r \times$ Condition = Random Informed	-0.0032***	-0.0032***	-0.0032***	-0.0032***	-0.0032***	-0.0032***
	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)
$r \times$ Condition = Strong	-0.0049***	-0.0049***	-0.0049***	-0.0049***	-0.0049***	-0.0049***
	(0.0002)	(0.0002)	(0.0002)	(0.0002)	(0.0002)	(0.0002)
$r \times$ Condition = Strong2Pref10	-0.0043***	-0.0042***	-0.0043***	-0.0042***	-0.0043***	-0.0042***
	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)
$r \times$ Condition = Strong2Pref5	-0.0045***	-0.0044***	-0.0045***	-0.0044***	-0.0045***	-0.0044***
	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)
$r \times$ Condition = Strong2R10	-0.0046***	-0.0046***	-0.0046***	-0.0046***	-0.0046***	-0.0046***
	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)
$r \times$ Condition = Strong2R5	-0.0044***	-0.0044***	-0.0044***	-0.0044***	-0.0044***	-0.0044***
	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)	(0.0003)
a_j (demeaned)		0.0137***		0.0137***		0.0138***
		(0.0003)		(0.0003)		(0.0003)
Fixed-effects						
Condition (6)	Yes	Yes	Yes	Yes		
Search Task (15)			Yes	Yes	Yes	Yes
Participant (3,533)					Yes	Yes
Fit statistics						
Observations	1,059,900	1,059,900	1,059,900	1,059,900	1,059,900	1,059,900
R^2	0.00742	0.04259	0.00762	0.04278	0.03544	0.07080
Within R^2	0.00728	0.04245	0.00728	0.04245	0.00749	0.04388

Clustered (Participant) standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Notes. Column 1 reports the OLS estimates of the following regression: $\mathbb{1}\{ProductSearched_{itj}\} = \alpha_{cond} + \beta \cdot r_{itj} + \delta(r_{itj} \cdot Cond_i) + \epsilon_{itj}$, where the dependent variable is an indicator for whether participant i searches product j in task t. The independent variables are fixed effects for i's experimental condition, the rank of product j, and the interaction of rank and the participant's experimental condition. In columns 2, 4, and 6, we also include product j's demeaned bonus A. We add search task fixed effects in columns 2–4 and participant fixed effects in columns 5–6.

Figure G.4: Forgone Bonus A by Rank of First-Clicked Item (Study 2)



Notes: Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level. Condition and rank-specific values average over all tasks (including changing ranking algorithms). "RI" refers to Random-Informed, "S" refers to Strong, "S2P10" and "S2P5" refer to Strong-to-Pref10 and Strong-to-Pref5, respectively, and "S2R10" and "S2R5" refer to Strong-toRandom10 and Strong-to-Random5, respectively.

G.2 Heterogeneity by Participant Attributes

Tables G.3 and G.4 report for Study 1 and Study 2, respectively, how the total number of products searched (column 1 in both tables) and whether the participant searches a particular product (column 2 in both tables) vary by whether the user passed all comprehension checks on the first attempt and whether the user self-reports that they shop online at least once a week. There is significant heterogeneity in search behavior by the former attribute. Participants who passed all comprehension checks on the first attempt search eight more products (or 44% overall over ten tasks) in Study 1 and five more products (or 20% more over fifteen tasks) in Study 2 than those who did not.[49] They also exhibit stronger position effects, as demonstrated by the statistically significant negative effect of the interaction between rank and whether all the comprehension checks were answered correctly in Column 2 of both tables. One might expect that frequent online shoppers might demonstrate different position effects compared to an infrequent shopper due to greater familiarity with an online retail setting. Here, the interaction between whether the participant is a high-frequency shopper and the product's rank is negative, indicating that these participants have stronger position effects, but it is not statistically significant in either study.

These results demonstrate that participants' search behavior is correlated with their observable attributes, especially whether they pass the comprehension checks on the first try. Allowing the model to account for heterogeneity among these dimensions can improve the model's fit by better explaining across-participant differences in search costs and position effects.

[49]Fourty-four percent of Study 1 participants and sixty-seven percent of Study 2 participants answered all comprehension checks correctly on the first try.

Table G.2: Searching the Third Product First (Study 2)

Dependent Variable:	Searched 3rd First
Model:	(1)
Variables	
Condition = Strong	0.0116*
	(0.0065)
Condition = Strong2Pref10	0.0048
	(0.0064)
Condition = Strong2Pref5	0.0205***
	(0.0067)
Condition = Strong2R10	-0.0008
	(0.0063)
Condition = Strong2R5	0.0042
	(0.0065)
Fixed-effects	
Search Task (6)	Yes
Fit statistics	
Observations	21,198
R^2	0.00124
Within R^2	0.00081

Clustered (Participant) standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Notes. OLS estimates on the subsample of tasks 10–15. Each observation is at the participant-task level. The dependent variable is an indicator for whether the participant searched the third-ranked product first. The third-ranked product yields the highest bonus B in the Strong2Pref5 and Strong2Pref10 conditions.

Table G.3: Search Intensity and Position Effects by Participant Attributes (Study 1)

Dependent Variables:	N Products Searched	1(Product Searched)
Model:	(1)	(2)
Variables		
Comp Checks Correct	7.749***	0.1083***
	(0.8681)	(0.0128)
Comp Checks Correct × r		-0.0058***
		(0.0013)
ShopFreq	0.7155	0.0150
	(0.9183)	(0.0135)
ShopFreq × r		-0.0013
		(0.0014)
Condition = Random	1.230	0.0121
	(1.057)	(0.0112)
Condition = Strong	-0.1996	-0.0019
	(1.068)	(0.0097)
Bonus A (demeaned)		0.0235***
		(0.0008)
r		-0.0055***
		(0.0013)
Constant	17.13***	0.2017***
	(1.040)	(0.0127)
Fit statistics		
Observations	961	96,100
R^2	0.07849	0.06599
Adjusted R^2	0.07464	0.06591

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

Notes: Table reports OLS estimates. The data in column 1 is at participant level, and at the participant-product-task level in column 2. In columns 1 and 2, the dependent variables are the number of products that the participant searches across all ten tasks and an indicator for whether a product is searched, respectively. Comp Checks Correct is a dummy variable which equals one if the participant passed comprehension checks on the first attempt. ShopFreq is a dummy variable which equals one if the user shops online at least once a week (self-reported). The variable b_A (demeaned) is the demeaned value of the product's bonus A. Standard errors are clustered at the participant level in Column 2.

Table G.4: Search Intensity and Position Effects by Participant Attributes (Study 2)

Dependent Variables:	N Products Searched	1(Product Searched)
Model:	(1)	(2)
Variables		
Comp Checks Correct	5.306***	0.0231***
	(0.5199)	(0.0034)
Comp Checks Correct × r		-0.0005**
		(0.0002)
ShopFreq	-0.6814	-0.0015
	(0.5236)	(0.0036)
ShopFreq × r		-8.12×10^{-5}
		(0.0002)
Condition = Strong	-2.285***	-0.0073***
	(0.8423)	(0.0028)
Condition = Strong2Pref10	-0.9617	-0.0030
	(0.8503)	(0.0029)
Condition = Strong2Pref5	0.4313	0.0017
	(0.8489)	(0.0029)
Condition = Strong2R10	-0.1903	-0.0003
	(0.8701)	(0.0031)
Condition = Strong2R5	0.7038	0.0024
	(0.8383)	(0.0029)
b_A (demeaned)		0.0137***
		(0.0003)
r		-0.0039***
		(0.0002)
Constant	25.86***	0.1270***
	(0.7760)	(0.0040)
Fit statistics		
Observations	3,533	1,059,900
R^2	0.03397	0.04332
Adjusted R^2	0.03205	0.04331

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*

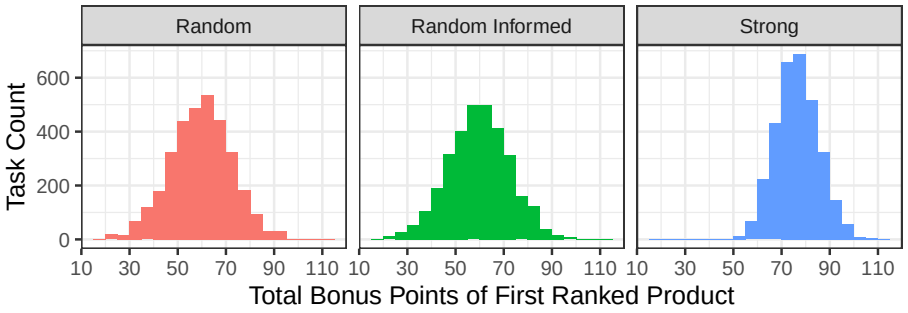
Notes: Table reports OLS estimates. The data in column 1 is at participant level, and at the participant-product-task level in column 2. In columns 1 and 2, the dependent variables are the number of products that the participant searches across all ten tasks and an indicator for whether a product is searched, respectively. Comp Checks Correct is a dummy variable which equals one if the participant passed comprehension checks on the first attempt. ShopFreq is a dummy variable which equals one if the user shops online at least once a week (self-reported). The variable b_A (demeaned) is the demeaned value of the product's bonus A. Standard errors are clustered at the participant level in Column 2.

G.3 Randomization and Manipulation Checks

G.3.1 Study 1

Figure G.5 presents a manipulation check: the first ranked product on average has a higher total bonus in the Strong condition compared to Random and Random-Informed. Figure G.6 shows that the average bonus A does not vary significantly by rank or condition, and the product with the highest bonus A can be found at any rank with similar probability.

Figure G.5: Total Bonus of First-Ranked Products (Study 1)



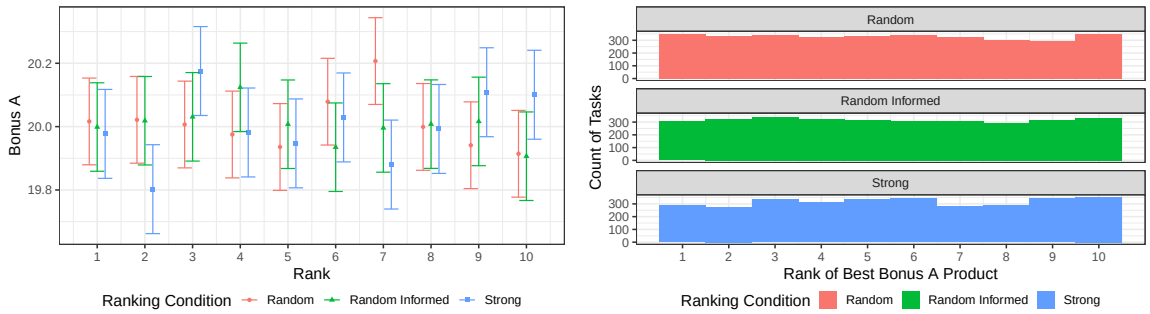
Notes: This figure plots the histograms of the total bonus (bonus A + bonus B) of the first-ranked product for all search tasks and for all participants, for each condition.

Table G.5: Randomization Check—Stimuli Selection (Study 1)

Dependent Variable:	Stimuli Row Number
Model:	(1)
Variables	
Constant	497.3***
	(15.73)
Condition = Random Informed	-21.91
	(22.47)
Condition = Strong	-13.41
	(22.52)
Fit statistics	
Observations	961
R^2	0.00101
Adjusted R^2	-0.00107

IID standard-errors in parentheses
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*
Base condition is Random

Figure G.6: Randomization Check of Bonus A (Study 1)



(a) Average Bonus A by Rank

(b) Rank of Best Bonus A

G.3.2 Study 2

Table G.6: Randomization Check—Stimuli Selection (Study 2)

Dependent Variable:	Stimuli Row Number
Model:	(1)
Variables	
Constant	243.6***
	(6.027)
Condition = Strong	10.92
	(8.383)
Condition = Strong2P10	10.02
	(8.466)
Condition = Strong2P5	-0.8277
	(8.448)
Condition = Strong2R10	7.361
	(8.661)
Condition = Strong2R5	1.182
	(8.341)
Fit statistics	
Observations	3,533
R^2	0.00113
Adjusted R^2	-0.00029
IID standard-errors in parentheses	
*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1*	

Notes: Base condition is Random Informed.

H Study Instructions

This section provides the instructions given to participants in the Random Informed conditions of Studies 1 and 2. The instructions for the other conditions are the same, except they do not include the last sentence "The products are displayed in a random order."

H.1 Study 1 Instructions

On the following pages, you will complete 1 practice round and 10 bonus-eligible rounds of a product search task.

In each round, you will be presented with a product listing page with 10 product options. You'll need to click on and eventually pick one product. Each product has two bonus values: Bonus A and Bonus B. Your performance bonus will depend on Bonus A and Bonus B of the product that you ultimately pick and how many products you click on.

You can see each product's Bonus A before you click. However, Bonus B is hidden until you click on the product.

Clicking on each new product to reveal Bonus B costs 1 point (clicking to open a box you've already opened previously is free and doesn't cost any points).

Bonus B has an average value of 40 points, and a standard deviation of 12. Put otherwise, 95% of products have their Bonus B between 16 and 64.

The value of Bonus A is not informative of the value of Bonus B. In other words, Bonus A and B are not correlated. Products with high Bonus A can have low Bonus B, and vice versa.

Your earnings towards your bonus for each round will be Bonus A plus Bonus B from the product you selected minus the points you lost from clicking. After all 10 rounds, your total points will be added up over all rounds. Your points will convert directly to your performance bonus. 400 points equals $1.00.

The products are displayed in a random order.

H.2 Study 2 Instructions

On the following pages, you will complete 1 practice round and 15 bonus-eligible rounds of a product search task.

In each round, you will be presented with a product listing page with 20 product options. You'll need to click on one or more products and eventually pick one product. Each product has two bonus values: Bonus A and Bonus B. Your performance bonus will depend on Bonus A and Bonus B of the product that you ultimately pick.

You can see each product's Bonus A before you click. However, Bonus B is hidden until you click on the product.

Clicking on each new product to reveal Bonus B costs 1 point (clicking to open a box you've already opened previously doesn't cost any points).

The value of Bonus A is not informative of the value of Bonus B. In other words, Bonus A and B are not correlated. Products with high Bonus A can have low Bonus B, and vice versa.

Bonus B has an average value of 40 points, and a standard deviation of 12. In most rounds, the highest Bonus B is at least 61.

Because Bonus B has a higher average and standard deviation than Bonus A, Bonus B contributes to a larger share of your payment than Bonus A.

Your earnings towards your bonus for each round will be Bonus A plus Bonus B from the product you selected minus the points you lost from clicking. We will randomly select 8 rounds to generate your additional bonus. Your total points will be added up over these 8 selected rounds and will convert directly to your performance bonus. 500 points equals $1.00.

The products are displayed in a random order.