Online Appendix to Consumer Inferences from Product Rankings: The Role of Beliefs in Search Behavior

Jessica Fong, Olivia R. Natan, and Ranmit Pantle

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). 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.[1]

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

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

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 means search is less reward-directed. Additionally, participants search more products relative to field settings, and are more likely to search all products relative to Study 2.[2] 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 the Weitzman search model. We describe the details of this pilot study with zero marginal search costs 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

[2]Participants search on average between 4.5 and 7.4 products across conditions, 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 the 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.

Allowing for an Outside Option In our study, we require participants to search for and select one product. Participants cannot choose the outside option of not selecting any

product. We chose this approach to ensure we observe at least one search per task, since base pay is a substantial portion of participant payment. As such, we do not model the outside option in our search framework.[3] This differs from the real world setting, as it is common for consumers to exit the search environment without purchase or further product examination. Many empirical papers use only data where consumers search at least one product. However, we believe that not having an outside option does not substantially impact our model estimates or conclusions for the following reason. Participants choosing the outside option would receive no bonus payment. Across both studies, for all products, the minimum payoff for a given product is 7 points, and the fifth percentile is 38 points, or \$0.08 cents. Therefore, a rational participant would never choose the outside option, conditional on search. However, if a participant's search cost is large enough such that their overall utility (payoff minus the search cost) is negative, it may be rational for the participant to choose the outside option, which is costless to search. We expect this to be true for few participants, as the average cost for searching the first product is \$0.05. Given the likely infrequency of the outside option impacting search behavior, the lack of the outside option is unlikely to impact our conclusions.

A.2 Model Robustness Checks

[3] See Morozov et al. (2021) for another example of modeling search with no outside option.

Table OA1: 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 27. 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

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

$$= \beta_i' w_{itj} + \epsilon_{itj}. \quad (2)$$

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

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

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

3. **Draw** σ_ϵ.

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

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

4. **Draw** Δ, V_β.

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

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

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:

$$z_{itj} = a_{itj} + (b^0_{ij}\frac{m_{itj}}{1+m_{itj}} + \bar{b}_{itj}\frac{1}{1+m_{itj}}) \cdot \mathbb{1}\{Cond_i \neq RI\}$$
$$+ 40 \cdot \mathbb{1}\{Cond_i = RI\} + \overbrace{\beta^0_i + \beta^1_i \mathbb{1}\{r_{itj} = 1\} + \beta^r_i \Big(\mathbb{1}\{r_{itj} \neq 1\} \cdot (r_{itj} - 2)\Big) + \epsilon_{itj}}^{\delta_{itj}} \quad (28)$$

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 28. 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 OA2. We base these priors on a now-omitted set of estimates on Study 1 data.

Table OA2: 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 OA4. 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 OA5.

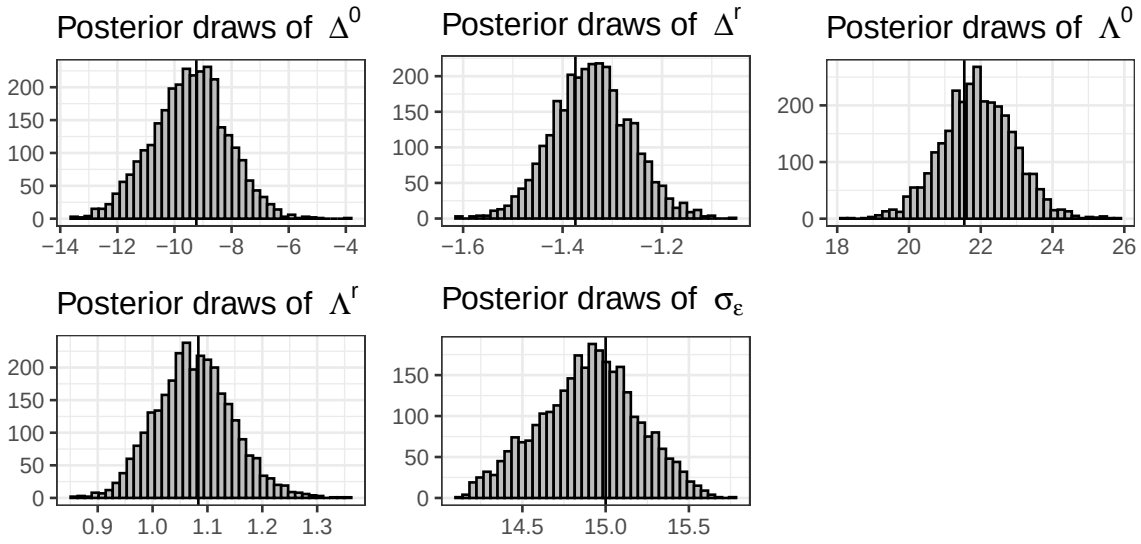
We estimate the model using 5,000 draws from the Gibbs sampler. We apply diffuse priors, and start at random draws for each value. In practice, the starting draws do not impact the chain. We discard the first 2,000 draws, and we thin the subsequent chain by using every fifth draw. Both simulations have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

We report the plotted posterior distributions in Figures OA1, OA2 and OA3. 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 OA3: 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

Figure OA1: Simulation 1a Posterior Draws—Unobservable Heterogeneity Only

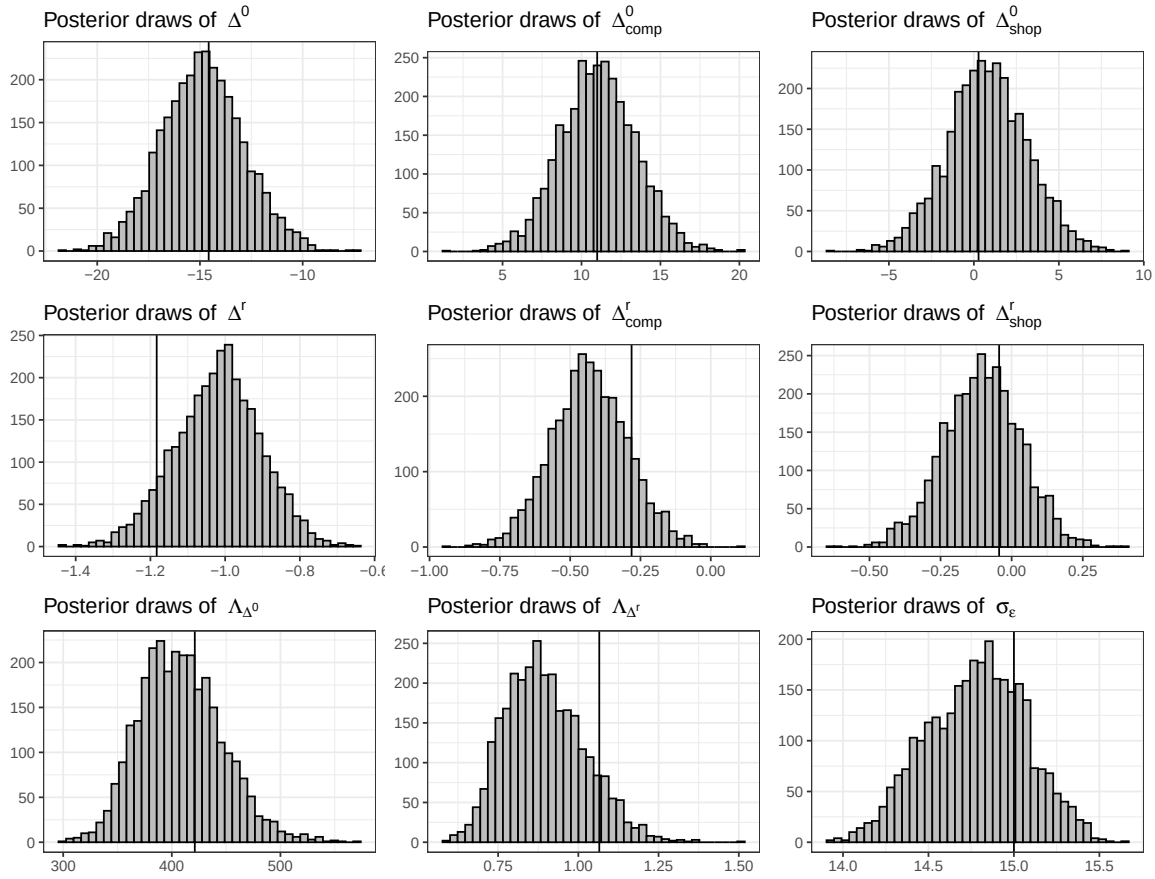


Notes: Each plot provides a histogram of posterior draws for mean search propensity (Δ^0), mean rank effect (Δ^1), the variance in search propensities (Λ^0), the variance in position effects (Λ^1), and the noise term on search decisions (σ^ϵ). The vertical black line corresponds to true value of the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

Table OA4: Simulation 1b Parameter Estimates—Observable and Unobservable Heterogeneity

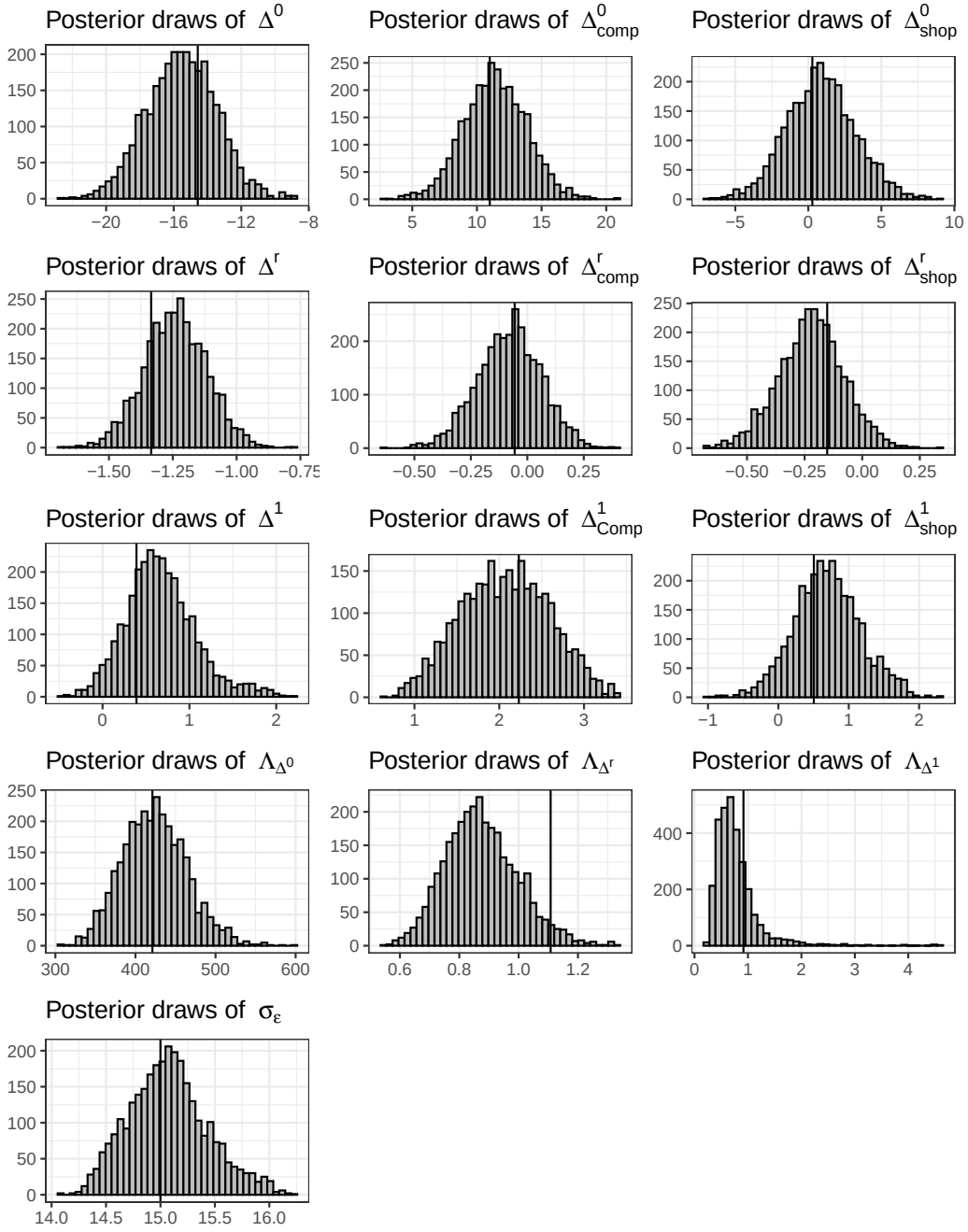
Parameter	Coef	Truth	Posterior Mean	Posterior SD
Baseline search propensity	Δ^0	-14.575	-14.634	1.948
Baseline search propensity Comp FE	Δ^0_{comp}	10.993	10.865	2.287
Baseline search propensity Shop FE	Δ^0_{shop}	0.276	0.689	2.36
Rank Coefficient	Δ^r	-1.183	-1.022	0.114
Rank Coefficient Comp FE	Δ^r_{comp}	-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

Figure OA2: 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 OA3: Simulation 1c Posterior Draws—Separate Position 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 OA5: Simulation 1c Parameter Estimates—Separate Position 1 Effect

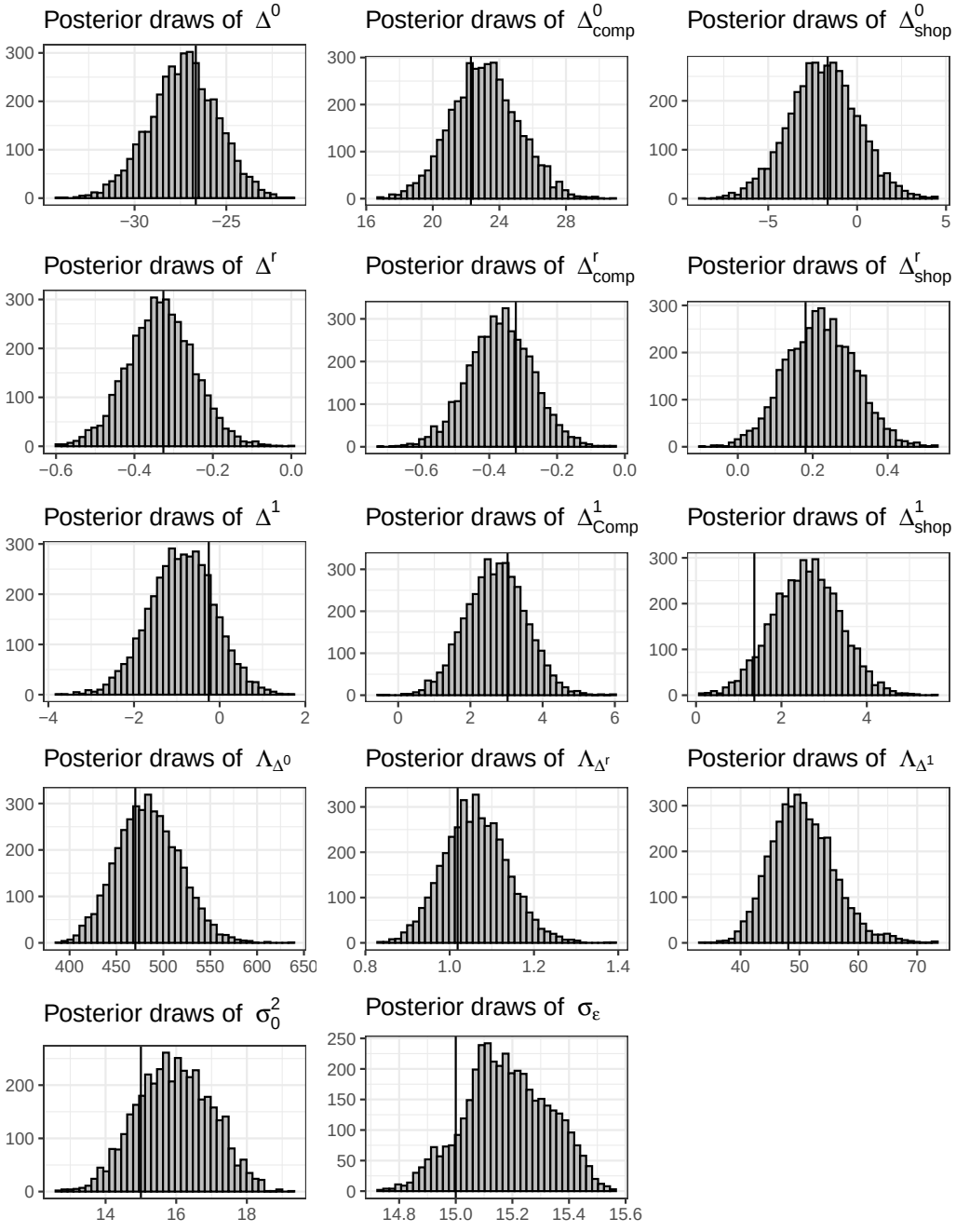
Parameter	Coef	Truth	Posterior Mean	Posterior SD
Baseline search propensity	Δ^0	-14.575	-15.759	2.253
Baseline search propensity Comp FE	Δ^0_{comp}	10.993	11.569	2.47
Baseline search propensity Shop FE	Δ^0_{shop}	0.276	0.961	2.375
Rank Coefficient	Δ^r	-1.335	-1.24	0.125
Rank Coefficient Comp FE	Δ^r_{comp}	-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

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 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 OA4 and the estimated mean (SD) of the parameters are reported in Table OA6.

Figure OA4: 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.

Table OA6: 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

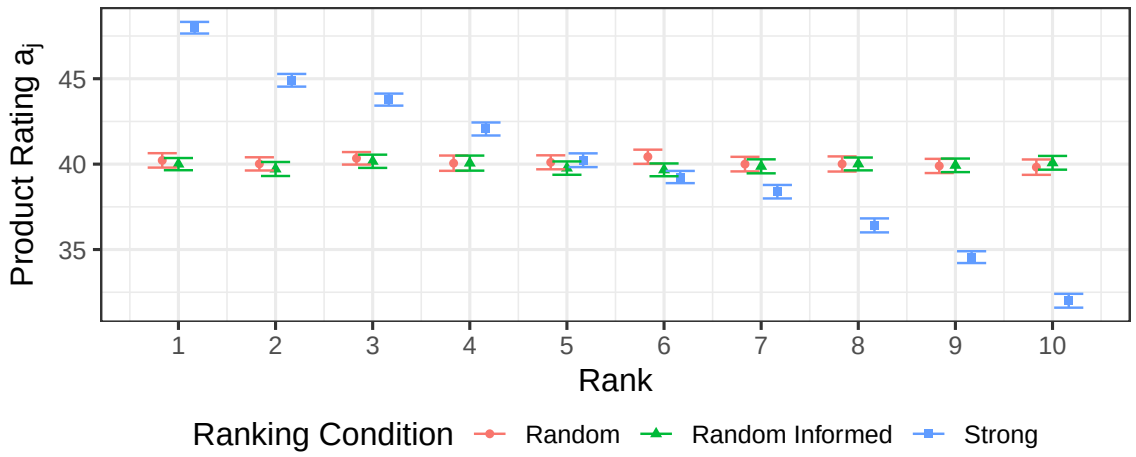
C Additional Studies

C.1 Study 1B: Correlated Attributes

C.1.1 Design

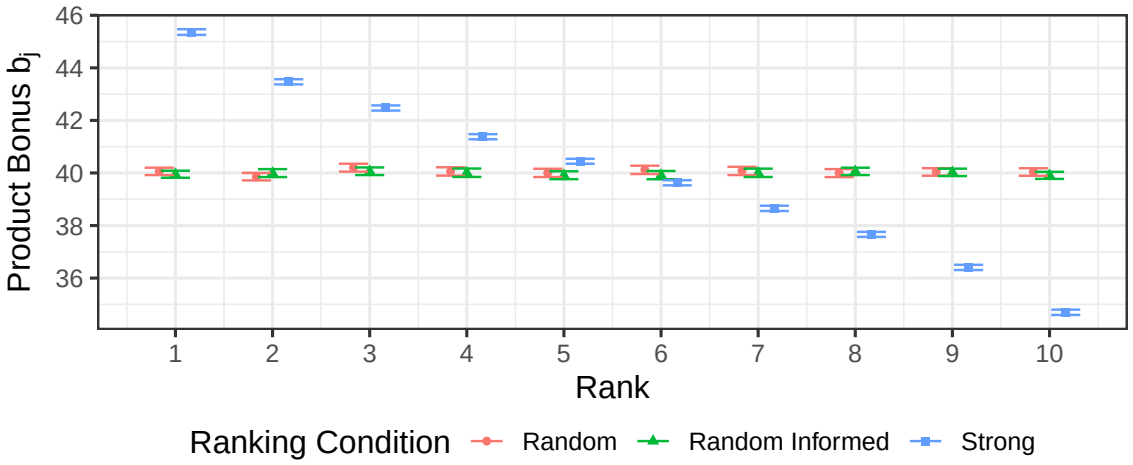
We constructed an alternative version of our first search task paradigm—Study 1B—which incorporates an additional feature of many search markets: informative pre-search signals. We redesign the attribute space to achieve this. We draw bonus b from a Normal distribution with mean 40 and standard deviation 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 OA5 and OA6.

Figure OA5: 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 OA6: 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 OA7: 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 OA8). We conclude from these results

that the primitive rank-specific search costs are likely robust to alternative information structures.

Table OA8: 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 OA9 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 OA9: Summary Statistics across Conditions (Study 2, Zero Marginal Cost)

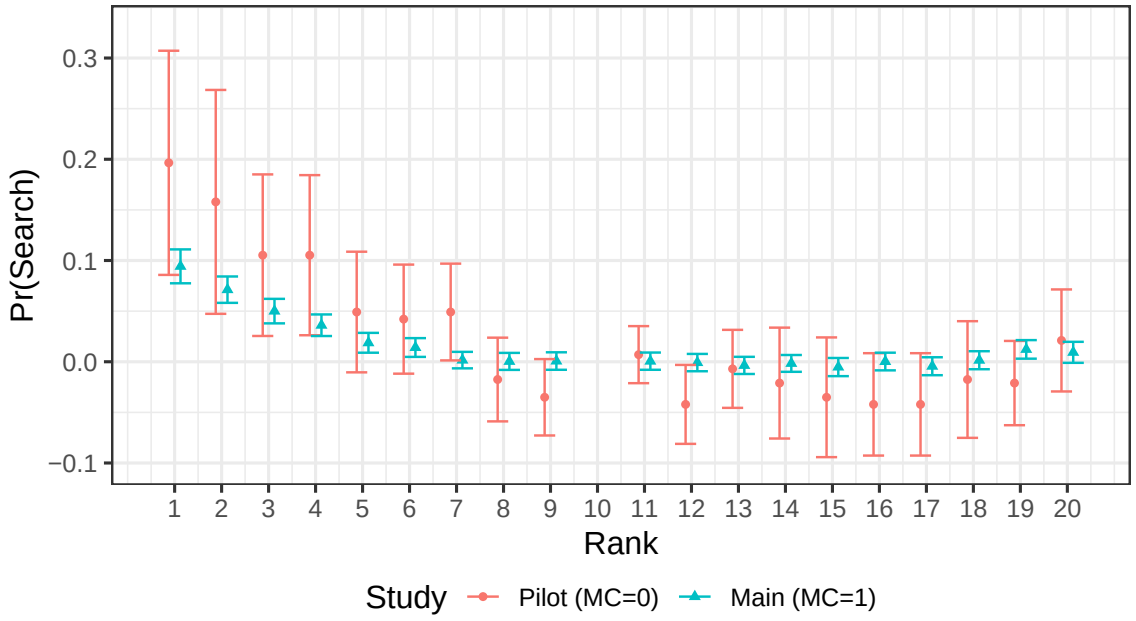
	Condition Mean (SD)			
	RI	Str	Str to P (5)	Str to R (5)
Num Searches	7.4 (7.92)	4.55 (5.15)	6.34 (6.84)	5.07 (5.58)
Bonus Earned	67.2 (10.52)	73.15 (10.43)	70.47 (12.03)	69.8 (10.43)
Click Max(a) First	0.2 (0.4)	0.23 (0.42)	0.18 (0.39)	0.22 (0.41)
Time Spent per Task	24.25 (24.57)	15.92 (16.8)	21.79 (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 OA7 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 OA7: 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:

$$y_{it} = \alpha_t + \beta_1 \mathbb{1}\{HigherStakes_i\} + \beta_2(\mathbb{1}\{HigherStakes_i\} \times Condition_i) + \epsilon_{it}. \quad (7)$$

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 OA10. 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.[4]

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.

Table OA10: 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)
HigherStakes × Condition = Strong	-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 7.

[4]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.

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

$$\mathbb{1}\{Searched_{itj}\} = \alpha_t + \beta(r_{itj} \times Condition_i \times \mathbb{1}\{HigherStakes_i\}) + \epsilon_{itj}. \tag{8}$$

This regression has a three-way interaction between the product j's rank r_{itj}, the participant's condition, and whether the participant is in the higher stakes study. We report the estimates in Table OA11. 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 OA11: 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)
Condition = Strong	0.0083
	(0.0058)
HigherStakes	0.0174**
	(0.0073)
r × Condition = Strong	-0.0013***
	(0.0004)
r × HigherStakes	-0.0004
	(0.0004)
Condition = Strong × HigherStakes	5.5×10^{-5}
	(0.0101)
r × Condition = Strong × HigherStakes	-0.0002
	(0.0006)
Fixed-effects	
Search Task (10)	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 8.

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:

$$\begin{aligned}\mathbb{1}\{\text{Searched Product } r\}_{i,t} = \alpha_t + \alpha_r + \beta_1 \mathbb{1}\{\text{Searched Product } r\}_{i,t-1} + \\ \beta_2 \mathbb{1}\{\text{Searched Product } r\}_{i,t-1} \times b_{i,t-1,r} + \beta_3 b_{i,t-1,r} + \\ \beta_4 a_{i,t,r} + \beta_5 NSearched_{i,t} + \epsilon_{i,t} \quad (9)\end{aligned}$$

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 $t-1$.[5] 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 OA12 displays the OLS estimates of Equation 9. 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 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.

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

Table OA12: 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*

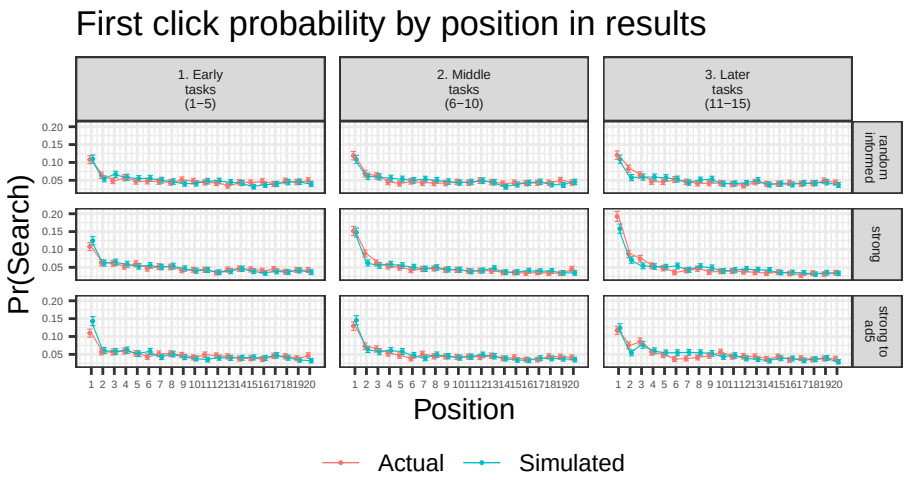
This can easily be accommodated in the model, where the term $E[b_{itj}|r_{itj}]$ in Equation 26 can be parameterized as $\gamma \cdot r_{itj}$, and consumers update beliefs about γ. Our general conclusion—that beliefs can confound the identification of search costs—are not specific to settings without 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 OA8 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.[6] 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 OA9.

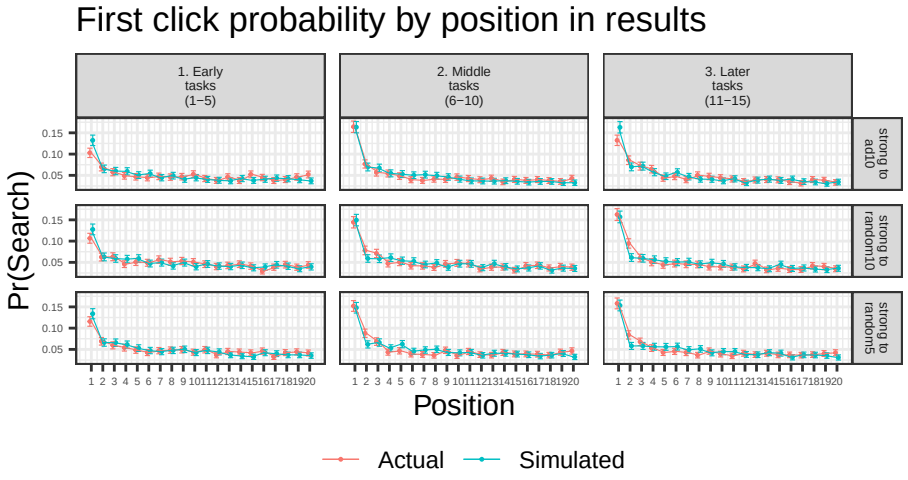
[6]The probability of searching rank 1 in tasks 11–15 in *Strong* is slightly understated by the model (middle right panel in Figure OA8). This suggests that there may be some form of learning that occurs which is not captured by our model. However, given the trade-off between accuracy and model tractability, we feel that our model reasonably captures the essential elements of the consumer search process.

Figure OA8: 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 OA9: 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

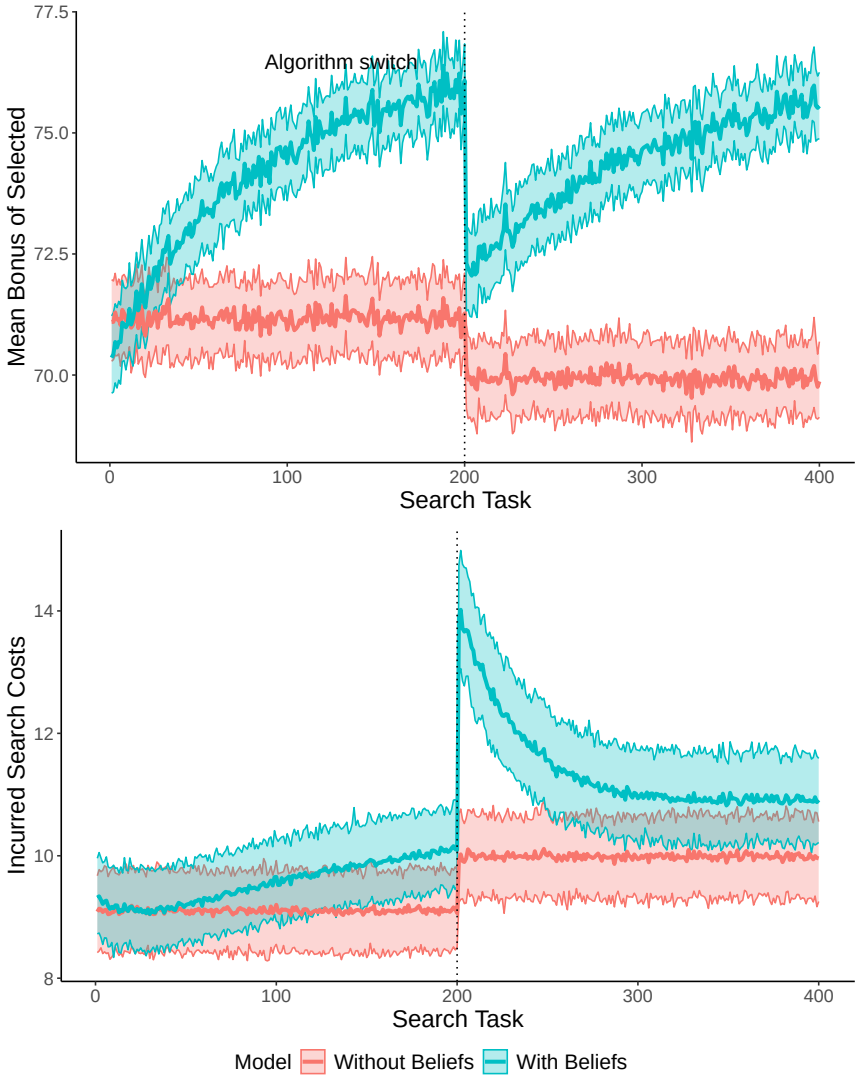
Table OA13: Estimation Results—Auxiliary Models Estimated on Tasks 6-7 (Study 2)

		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 (0.077)***
Comp	Δ_{comp}^r	-0.059 (0.062)	-0.05 (0.064)	-0.022 (0.083)
ShopFreq	Δ_{shop}^r	-0.023 (0.047)	-0.052 (0.061)	-0.053 (0.081)
Pos 1 Coefficient β^1				
Intercept	Δ_0^1	2.3 (1)***	3.8 (0.94)***	-2 (0.96)***
Comp	Δ_{comp}^1	0.34 (0.99)	0.43 (0.94)	0.43 (1)
ShopFreq	Δ_{shop}^1	-0.12 (0.81)	-0.48 (0.95)	-0.12 (0.84)
Baseline Search Propensity β^0				
Intercept	Δ_0^0	-23 (1.5)***	-23 (1.5)***	-38 (1.9)***
Comp	Δ_{comp}^0	8.9 (1.6)***	10 (1.7)***	9.9 (1.7)***
ShopFreq	Δ_{shop}^0	-2.7 (1.4)**	-0.85 (1.5)	-0.71 (1.7)
Heterogeneity				
Rank Coefficient	$\sqrt{V}_{\beta^r}$	0.41 (0.033)	0.5 (0.037)	0.61 (0.053)
Pos1 Coefficient	$\sqrt{V}_{\beta^1}$	4.1 (6.9)	5.3 (9.2)	2.6 (9.3)
Mean Search Propensity	$\sqrt{V}_{\beta^0}$	15 (18)	16 (21)	17 (27)
Reservation Utility	σ_ϵ	11 (8.2)	12 (8.2)	13 (13)

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 OA10: 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 200.

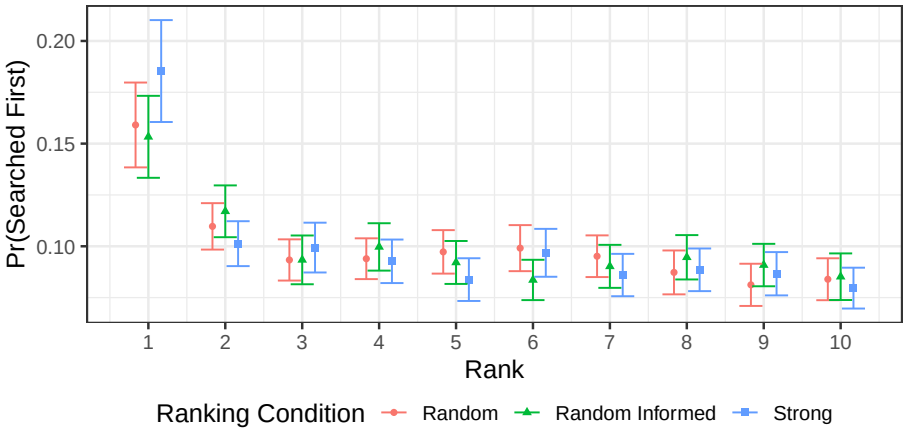
G Additional Figures and Tables

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 OA11). 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 OA12).

Figure OA11: Pr(Search First) by Rank by Condition (Study 1)

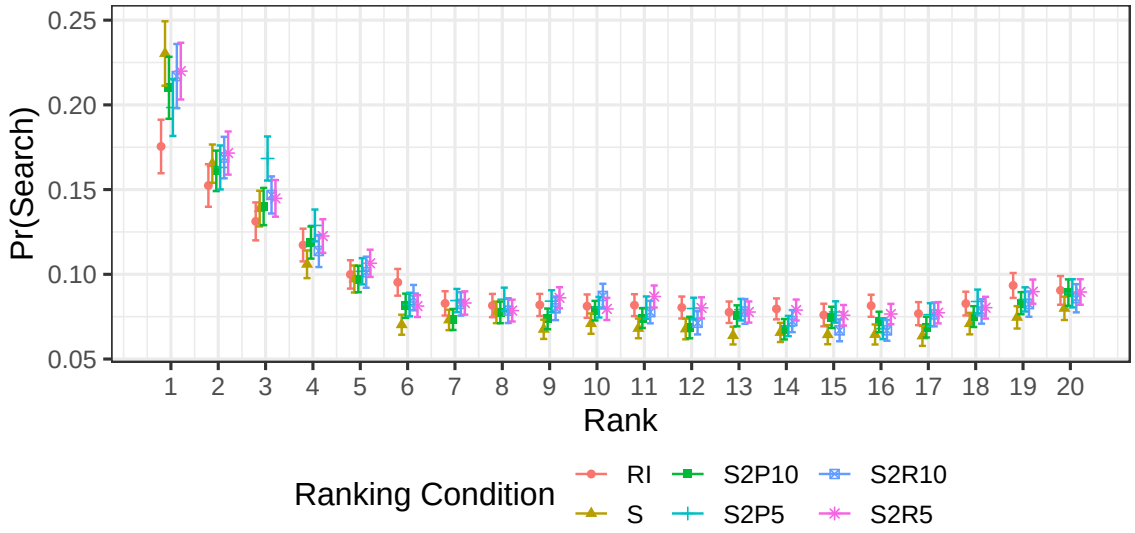


Notes: Error bars represent 95% confidence intervals, which are based on standard errors clustered at the participant level.

Figure OA12: Depth of Search across Tasks (Study 1)

G.1.2 Study 2

Figure OA13: 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-to-Random10 and Strong-to-Random5, respectively.

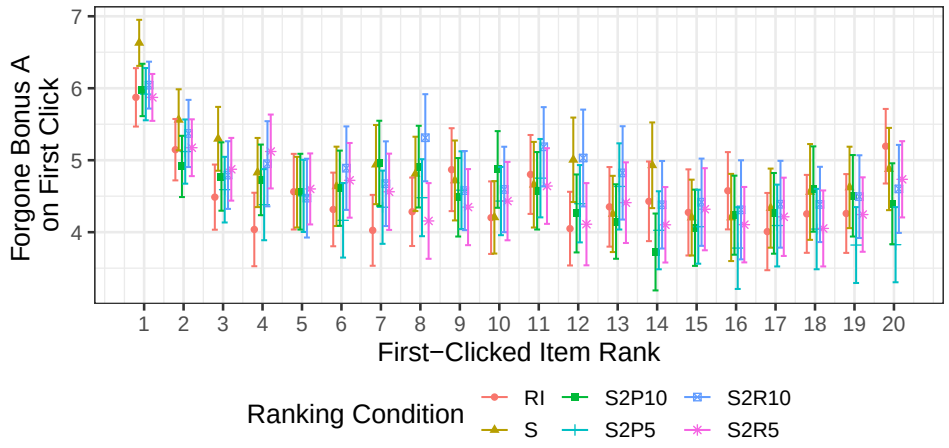
Table OA14: Effect of Rank on Search by Condition (Study 2)

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 OA14: 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-to-Random10 and Strong-to-Random5, respectively.

G.2 Heterogeneity by Participant Attributes

Tables OA16 and OA17 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.[7] 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

[7]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 OA15: 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 OA16: 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 OA17: 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.

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.

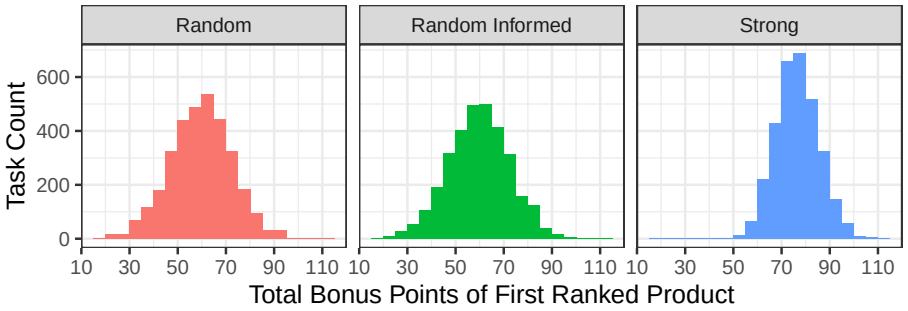
These results demonstrate that participants' search behavior is correlated with their observable attributes, especially whether they pass the comprehension checks on the first try. Allowing the model to account for heterogeneity among these dimensions can improve the model's fit by better explaining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

G.3 Randomization and Manipulation Checks

G.3.1 Study 1

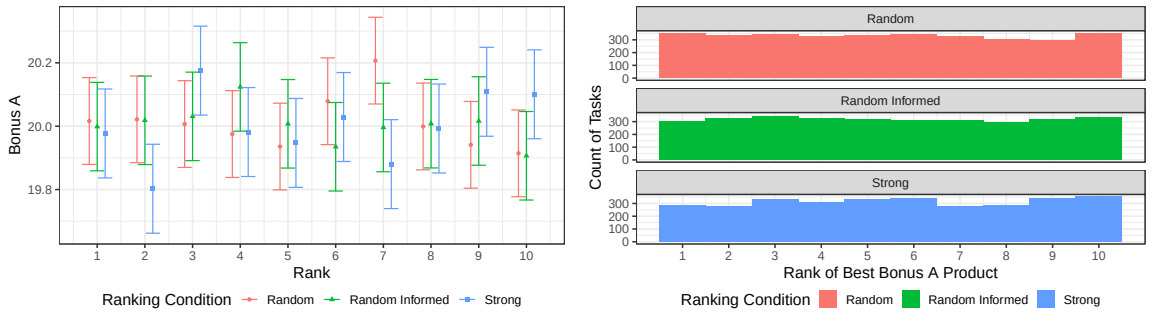
Figure OA15 presents a manipulation check: the first ranked product on average has a higher total bonus in the Strong condition compared to Random and Random-Informed. Figure OA16 shows that the average bonus A does not vary significantly by rank or condition, and the product with the highest bonus A can be found at any rank with similar probability.

Figure OA15: 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.

Figure OA16: Randomization Check of Bonus A (Study 1)



(a) Average Bonus A by Rank

(b) Rank of Best Bonus A

Table OA18: 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

G.3.2 Study 2

Table OA19: 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*
Base condition is Random Informed

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.

References

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.

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

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

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.