Optimal Selection Using Algorithmic Rankings with Side Information

Anonymous Author(s)

Abstract

Motivated by online platforms such as freelance job markets, we study an agent choosing from a list of candidates, each with a hidden quality that determines match value. The agent observes only a noisy ranking of the candidates plus a binary signal that indicates whether each candidate is "free" or "busy." Being busy is positively correlated with higher quality, but can also reduce value due to decreased availability. We study the agent's optimal selection problem in the presence of ranking noise and free-busy signals and ask how the accuracy of the ranking tool impacts outcomes. In a setting with one high-valued candidate and an arbitrary number of low-valued candidates, we show that increased accuracy can result in reduced social welfare. This can occur for two reasons: agents may be more likely to make offers to busy candidates, and (paradoxically) may be more likely to select lower-ranked candidates when rankings are more indicative of quality. We further discuss conditions under which these results extend to more general settings.

CCS Concepts

• **Do Not Use This Code → Generate the Correct Terms for Your Paper**; *Generate the Correct Terms for Your Paper*; Generate the Correct Terms for Your Paper; Generate the Correct Terms for Your Paper.

Keywords

Do, Not, Use, This, Code, Put, the, Correct, Terms, for, Your, Paper

1 Introduction

Consider the following setting: you are a hiring manager, trying to find a candidate to interview for a hiring slot. You know that some candidates are better suited for your role than others - ideally, you would like to interview the best candidate available. To assist you with this goal, you might obtain a (noisy) ranking of the candidates from a commercial employment site, such as LinkedIn, Indeed, or other ranking tools. If the ranking is "good" (if the expected utility of candidates decreases as you navigate further down the list), the best action for the employer clear: always pick the top-ranked candidate. Also, as the ranking tool becomes more accurate, the employer is only more happy with their choice, as the expected value of the top-ranked candidate only increases.

However, in many cases, employers have access to side information, besides the ranking. For example, a hiring manager may know that some candidates are less likely to accept any eventual offer because they is already employed (see [7] for an empirical example of such a setting). We call such a candidate "busy", as compared to a "free" candidate without these competing sources of employment. If a busy candidate would *never* accept an offer, then the choice is clear: the best option is to select the highest-ranked free candidate. However, if a busy candidate still has some chance of accepting an offer, the possible strategy space changes. In many markets there is an opposing force: it may be the case that high-value candidates are more likely to be busy, and thus being busy is a signal of candidate quality. For example, this could be because high value candidates are more likely to have already been identified and hired by competing firms[1]. In many settings it may not be possible to resolve the uncertainty inherent in a busy signal: for example, the candidate themselves may not know whether or not they would be willing to move without going through the interview process, at which point the firm has already used up its interview slot.

Here, an employer faces a difficult question: given a noisy ranking of candidates, along with information about which ones are free or busy, which candidate should she select? Should she pick the top-ranked busy candidate, following the old adage "if you need something done fast, ask a busy person"? Instead, should she aim to avoid the extra hassle of competing for a busy candidate and instead hire the top-ranked free candidate? Moreover, how should the firm's strategy change if the ranking tool becomes more accurate?

Moreover, a firm's hiring strategy has impacts on the rest of society. For example, consider a firm's choice to interview a busy (already employed) candidate over a free (unemployed) candidate may be more or less valuable to the candidate. An unemployed candidate may have higher utility for an interview than an unemployed candidate. Alternatively, an employed candidate may have a stronger ability to negotiate a better offer, which could increase their welfare more. While we do not directly model such externalities, we will use "probability of picking a busy candidate" and other such quantities as benchmarks to understand the impact of a ranking tool on broader society. Specifically, we will be interested in how these quantities change as the accuracy of the ranking tool increases. Do these measures uniformly improve or degrade as the firm uses a more accurate ranking tool?

Motivation: More broadly, in this paper our goal is to study a stylized model of an agent using an algorithmic tool to make a

---

[1]Throughout this paper, we will make the assumption that some candidates are better fits for the job than others (e.g. higher or lower value). We recognize that this assumption drastically simplifies the multi-faced strengths and weaknesses each candidate may bring, and discuss this (and other) assumptions in more depth in Section 2.

decision, and combining the information encoded by the tool with her own source of information. Given the importance attached to continually increasing the accuracy of algorithmic tools, we wish to show that when these tools are used as part of a broader system involving human discretion and alternative sources of information, the returns to increased algorithmic accuracy are less clear: in particular, relevant metrics of social welfare could *deteriorate* under increased accuracy. We hope that our research will further research in other scenarios where algorithmic tools are used alongside other sources of information, and in particular research into what properties of such algorithmic tools are helpful for societal welfare. While we will use the employer hiring job candidates as our motivating example, we note that this setting is quite broad and could model a range of other scenarios where humans make decisions involving. For example, consider a hungry customer navigating a ranking of restaurants, each "crowded" or "not crowded": a crowded restaurant is more likely to be of good, but also has a longer wait.

Summary: The rest of this paper is as follows: in Section 2, we will present our model and assumptions, as well as a permutation model that we believe may be of broader interest. In Section 3, we discuss the connection of our paper to related work, including the extensive literature on herding and the Pandora's Box problem. Next, Section 4 analyzes the optimal strategy that a firm navigating a ranking might take: should she pick the first busy candidate, the first free candidate, or some other candidate entirely? In Section 5 we study notions of social welfare, like how frequently firms preferentially select job candidates who are already employed (potentially modeling unemployment discrimination) or avoid doing so (potentially modeling collusion in job hiring). We also analyze how often firms select the top ranked candidate, a measure of how incentive compatible the ranking is with potential implications for the welfare of the company creating the ranking tool. Throughout, we consider how increasing the accuracy of the ranking tool can affect the strategy, welfare, and fairness results in previous sections. Surprisingly, we show that increased accuracy can *reduce* measures of social welfare. In Section 6 we discuss extensions of our model and in Section 7 we conclude by discussing implications of our results. All proofs are deferred to the appendix.

2 Model, notation, and assumptions

Model and notation: First, we present our model. There are N candidates, of which the agent (alternatively, firm) wishes to select exactly one: for example, to interview a job candidate. The candidates have different true values to the firm, where v_i denotes the value of candidate i, though the firm *cannot* know the true value of candidates at each ranking (this only holds in the limit if a perfectly noiseless algorithmic ranking tool). The goal of the firm is to maximize their expected utility. In order to help with this decision, the firm has access to a ranking tool which produces noisy permutations over the candidates: $\sigma \sim \mathcal{P}$, where σ_i denotes the index of the candidate ranked ith in permutation σ with value v_{σ_i}. We will require that the expected value of candidates decreases in the ranking ($\mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_i}]$ is decreasing in i).

Permutation distribution: When comparing two distributions $\mathcal{P}$ and $\mathcal{P}'$, we will use notation $P[\sigma]$ and $P[\sigma]'$ to denote the probability $\mathcal{P}$ and $\mathcal{P}'$ place on σ, respectively. In general, we will assume

that the probability of observing each permutation is drawn from a Plackett-Luce distribution [14, 19]). This commonly-used model is where i.i.d. noise is added to the true value of each item to produce $\hat{v}_i = v_i + \epsilon$, for ϵ drawn from a Gumbel distribution: then, the items are sorted in order of the noised values $\{\hat{v}_i\}$. Throughout, when we say "increased accuracy", we will mean a smaller degree of noise (smaller standard deviation) in the Gumbel distribution. However, many of our results rely only weakly on the Plackett-Luce model, and in the relevant appendices we state the sufficient properties on the permutation distribution in order for our results to apply.

Firm strategy: In addition to the ranking, the firm has access to a single bit of side information about each candidate: specifically, a status vector s, where $s_i = 1$ if the candidate ranked in position i is free, and $s_i = 0$ if the candidate is busy. We assume that higher-valued candidates are less likely to be available, in particular, that $v_i \geq v_j$ implies $p_i \leq p_j$, where p_i gives the probability that the ith candidate is free. We also assume that each candidate's probability of being free or busy is independent of every other candidate: specifically, each free/busy realization is drawn from a Bernoulli distribution $s_i \sim Bern(p_i)$ before the firm observes the ranking. Throughout this paper we will take these probabilities as fixed and assume each candidate reports her free-busy status honestly: in the conclusion we will discuss extensions of this model.

Picking a busy candidate incurs a penalty: specifically, if a candidate has value v_i if picked when they are free, then they have value v_i/γ if they are selected when they are busy, for $\gamma \geq 1$. In general we allow γ_i to vary with the candidate i: when we use γ without a subscript, we will consider the simplifying case when γ_i is identical across candidates. This γ parameter reflects the fact that a busy (e.g. already employed) job candidate is less likely to accept an offer. We may also view γ as a firm-specific parameter reflecting, for example, the attractiveness of that firm to candidates[2].

One central assumption is that the firm must select a candidate based solely on the ranking, and cannot re-select another candidate later. This could be motivated by hiring in stages, where "selecting a candidate" corresponds to bringing a candidate on for an intensive onsite which cannot easily be filled with another candidate if one declines an offer after the onsite. Thus, we model strategy involved in this first stage of hiring, e.g., the selection of a candidate for an onsite interview. We also assume that it is not possible to resolve the uncertainty inherent in the free/busy signal - for example, while the employer could ask the candidate about their willingness to move jobs, the candidate themselves may not know perfectly without going through the recruiting process.

Superstar model: In much of this paper, we will focus on the *superstar setting*, with exactly one high-value candidate, and all other candidates with lower value: $v_1 \geq v_2 = v_3 \ldots \geq 0$ and $p_1 \leq p_2 = p_3 \ldots$. This models settings where the distribution of candidate quality is skewed: the highest-value individual may have much higher value than other candidates, who are all roughly comparable. In this setting, the space of permutations becomes much smaller, because the only distinguishing feature is the index of the high-value item. We will use the notation σ^i to denote the permutation where the high-value item is in the ith index, and in general we

---

[2] We do not directly model candidate utilities, but include this solely as interpretation for when γ may be high or low.

will assume that $P[\sigma^i] \geq P[\sigma^{i+1}]$ for all i (that is, the ranking is more likely to rank the high value item higher). In Section 6 we will show that while the general setting is more complex, our core results generalize beyond the superstar model.

3 Related work

Our model has connections to celebrated models of agents strategically picking items, such as the hiring problem and Pandora's box problem [13, 22]. However, our setting has key differences from these. Foremost among them is the signaling mechanism: rather than each item having an independent, known distribution of value, we model the scenario where the free/busy signal is generated by items with different probabilities, given different values. This structure, in conjunction with the ranking, means that the free/busy status of item i can affect the expected value of item j – generally not captured in existing models. Furthermore, in our setting the agent must pick a single item and commit to it without being able to observe its true value. This captures the setting where it is difficult or impossible to evaluate the item without committing to it first – for example, in hiring, when a candidate's true "value" may not become apparent until after weeks or months of work.

Additionally, our work has connections to long literature on herding (also known as information cascades) [1, 2, 23], a pattern where it may be optimal for a decision-maker to follow actions taken by previous people, even ignoring their own information. Such patterns can lead to "cascades" where multiple sequential decision-makers each follow actions taken by previous decision-makers, rather than following their own information. In our setting, when an agent chooses to pick a lower ranked busy candidate over a higher ranked free, this could be seen as herding, since the free/busy status of candidates could be seen as created by actions taken by previous decision-makers, whereas the ranking is assumed to be private (personal) information. To our knowledge, herding has not been extensively studied in ranking settings: our model interestingly allows for herding to be moderated by the strength of private information through the quality of the ranking and the index of the first busy candidate.

Our paper also has connections to the literature on algorithmic monoculture [12], which studies when utility may be *decreased* when two agents rely on the same algorithmic ranking, rather than their own (uncorrelated) rankings. Variants of this setting have been explored, such as how algorithmic monoculture can lead to individuals experiencing correlated outcomes ("outcome homogenization") [4], or the implications of monoculture and of noisy matching in two-sided matching markets [16, 17]. Common features include the model of multiple agents taking strategic actions in relation to imperfect rankings, as well as the notion of a penalty for picking an item that another agent has picked. Key differences in our model include the fact that this penalty is softened – in particular, an agent can derive non-zero utility for picking an item that another has already chosen. This dramatically complicates the strategy space. Multiple works have considered the question of fairness in ranking - [24] offers a helpful survey. In particular, [20] describes a notion of fairness relating to how often items are presented in the top k, given that they have some probability of truly being the best k items (relating to our notion of an item being

"picked"). Issues of fairness and strategy have also been studied in the related area of algorithmic monoculture [3, 6, 9, 11], while [18] studies the tradeoff between diversity and accuracy, showing that consumption constraints (e.g. only top item can be consumed) explains away a tension between these two goals.

Finally, there are multiple papers (empirical and theoretical) exploring models of firms using prediction tools to compete with each other. For example, [7] explicitly studies an empirical version of this where job candidates could pay to send a signal of availability to potential employers. Intriguingly, in this setting it seemed like such a signal did *not* lead employers to preferentially pick busy candidates, with most employers choosing a "first-free" strategy, perhaps indicating a parameter regime where this strategy was typically dominant. [8] explores this setting in a dynamical pricing model, giving conditions where pricing levels for such a signal induce a truthful response. Additionally, [10] explores a setting where improved data representation (as modeled by reduced Bayes risk) can paradoxically increase the error that users experience when they choose between two competing firms. [5] studies statistical discrimination in stable matchings where candidates have preferences over firms, who only have noisy access to signals about candidate quality.

4 Superstar: Optimal Selection Strategies

In this section, we begin by evaluating the firm's strategy: given access to an algorithmically-generated ranking over candidates and each candidate's free-busy status, what is the optimal strategy? (All proofs will be deferred to Appendix A.) First, we will show that (given a status vector s) there are only two strategies we need to consider within the superstar setting: picking the first free candidate, or the first busy candidate. This dramatically reduces the strategy space firms need to consider.

Lemma 1. *In the superstar setting, given a realized status vector s, it is always optimal to either pick the first (top-ranked) free item or the first busy item.*

Next, the question turns to *which* of these two candidates (top-ranked free or top-ranked busy) we should pick, and the more general strategy the firm follows. Algorithm 1 describes such a strategy. In particular, this strategy is made up of two decisions: 1) Whether the firm seeks to preferentially select free or busy candidates, and 2) How far down the ranking the firm is willing to look for a candidate of their preferred free/busy status (their "window size"). Specifically, it shows that the strategy space depends on two terms. First, free-busy ratio $R = \frac{p_2/(1-p_2)}{p_1/(1-p_1)} \cdot r$ measures how strongly status is correlated with value - effectively, it shows how strongly observing whether a candidate is free or busy should influence our belief about the (unobserved) candidate value. Secondly, γ measures the penalty for selecting a busy candidate - if this is larger, then the firm is *less* likely to pick a busy candidate. The term $R = \frac{v_1/\gamma_1 - v_2}{v_1 - v_2/\gamma_2} \cdot r$ incorporates both r and γ_1, γ_2. If $R \leq 1$, firms have a preference for free candidates, and if $R \geq 1$, firms have a preference for busy candidates. However, the further down the ranking the firms search, the less likely it is that a given candidate will be high value. Therefore, the optimal strategy is for firms to pick the top-ranked candidate of their preferred status, so long as that candidate doesn't come "too
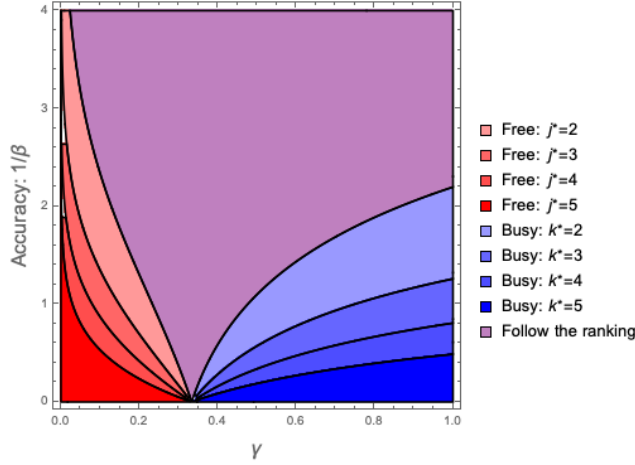
Figure 1: Illustration of strategy in Algorithm 1 for firm, given $N = 5$ candidates with $v_1 > v_2 = 0$ and permutations given by Plackett-Luce. The x axis varies the γ busy penalty, while the y axis varies accuracy as parameterized by the $1/\beta$ Gumbel noise parameter (higher values increase accuracy). Shades of red indicate regions where first-free selection is the optimal strategy and shades of blue indicate where first-busy selection is (darker shades indicate regions where the firm has larger j^* or k^* and thus "hunts" further down the list to find a candidate with their preferred free/busy status. The purple region is where the optimal strategy is "follow the ranking". Note that the choice of first-free or first-busy depends only on γ, while the window size depends on the accuracy of the ranking tool.

far" down the ranking, where "too far" is measured in terms of the accuracy of the ranking tool.

Algorithm 1: Given candidate values $v_1 > v_2$, probabilities of being free $p_1 < p_2$, accuracy of ranking tool $\{P[\sigma^i]\}$, $s \in \{0, 1\}$ for status of top-ranked candidate, and index j of first candidate with different free-busy status from s.

1 Define $R = \frac{v_1/\gamma_1 - v_2}{v_1 - v_2/\gamma_2} \cdot r$.

2 **if** $R \leq 1$ **then**

3   Calculate $j^* = \max_{j \in N} P[\sigma^1]/P[\sigma^j] \leq 1/R$, and set $j^* = 1$ if no such j exists.

4   Calculate $k = \min_{i \in [j^*]} s_k = 1$, and $k = 1$ if no such index exists.

5 **else**

6   Calculate $j^* = \max_{j \in N} P[\sigma^1]/P[\sigma^j] \leq R$, and set $j^* = 1$ if no such j exists.

7   Calculate $k = \min_{i \in [j^*]} s_k = 0$, and $k = 1$ if no such index exists.

8 **end**

9 **return** k

For simplicity, we will refer to the three different strategy families that Algorithm 1 implies as: k-**free** if the firm is preferentially

picking a free candidate within a "window" of size k, k-**busy** if the firm is preferentially picking a busy candidate within a "window" of size k, and **follow the ranking** if the window is of size 1, implying that the firm always ignores its own free/busy information and simply follows the ranking tool. Figure 9 includes a visual illustration of this strategy space as a function both of the busy penalty γ and the accuracy of the ranking tool.

First, we will note one important property of the algorithm in Algorithm 1: as accuracy increases (e.g. noise in the RUM decreases), the optimal window size of the algorithm only shrinks. This should make intuitive sense: firms know that the high value candidate is more likely to be at the top of the list, and thus search less far for candidates with their preferred free/busy status.

Lemma 2. *As accuracy increases, the window size j^* in Algorithm 1 stays constant or shrinks.*

Finally, we will analyze the performance of the Algorithm in 1. As a warm-up, we will show that the there exist cases where the "optimal" strategy differs from that in Algorithm 1.

Lemma 3. *In the limit of an extremely uninformative ranking, there exist scenarios where the optimal strategy differs from Algorithm 1.*

PROOF SKETCH. Consider $N = 3$ candidates with a completely uninformative ranking: $P[\sigma^1] = P[\sigma^2] = P[\sigma^3]$, so the only source of information is the free busy status. Next, consider two status vectors: one is $[0, 0, 1]$ (candidates are busy, busy, free), and the second is $[1, 1, 0]$ (free, free, busy). We show that there exist parameters such that it is optimal to pick the *last* candidate for both of these status vectors: this shows that, while for a fixed status vector the optimal strategy is always to pick the top-ranked free or top-ranked busy candidate, the optimal strategy cannot be described as in Algorithm 1 ("first-free" or "first-busy") because firms may preferentially pick free or busy candidates based on the situation. □

However, this result should not be very surprising: in the limit of an extremely uninformative ranking, we should not expect the optimal strategy to be well-described in terms of a "window" of top j items in the ranking. Instead, we will show in Theorem 1 that Algorithm 1 has low additive error when the ranking tool is relatively accurate.

THEOREM 1. *Algorithm 1 as applied to a setting where j is the first index that differs from 1 has error that is optimal up to an additive factor of ϵ, for $\epsilon \leq \frac{p_2}{p_1} \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot (1 - P[\sigma^1] - P[\sigma^j])$.*

At a high level, the bound in Theorem 1 has high error when a) there is a relatively low probability of the high-valued candidate being in index 1 or j (the only two indices that Algorithm 1 considers choosing), or b) when the low value candidate has relatively high value. Both of these are scenarios where we expect preferentially picking candidates based on free-busy status to be a poor strategy. Figure 4 in the appendix includes numerical simulations studying the sub-optimality of Algorithm 1, showing that often in practice the error is extremely small. Throughout the superstar analysis, we will assume that firms are acting according to Algorithm 1.

5 Superstar: societal welfare

Section 4 described the strategies that firms select under different parameter regimes: either preferentially looking for free candidates,

for busy candidates, or simply following the ranking (picking the top ranked candidate). In this section, we will discuss how the firm's strategy impacts the broader system, such as the welfare of firms and candidates. Throughout, we will focus specifically on the impact of changing accuracy in the ranking tool on both strategy and welfare.

5.1 Welfare of firm

First, we will note that welfare of the firm making hiring decisions always increases with increased accuracy of the ranking tool:

THEOREM 2. *Increasing the accuracy always increases welfare of firms using a first-free or first-busy strategy as described in Section 4.*

This should be relatively intuitive: as the accuracy of the tool increases, the utility of a best-responding firm always increases.

5.2 Welfare for candidates:

Next, we will consider impacts that the firm's hiring strategy has on candidate welfare. For example, a firm using the first-busy strategy will select busy candidates much more frequently than a firm using the first-free strategy. Picking a busy candidate could involve negative impacts: for example, it could be viewed as firms "poaching" candidates from other firms rather than hiring unemployed candidates, which may lead to a higher unemployment rate. However, picking a busy candidate could also be viewed positively: candidates who receive competing offers could potentially use them to increase their compensation.

What is the impact of increasing accuracy on how often busy candidates are picked? Theorem 3 shows that this can *increase* or *decrease*. For intuition, increasing the accuracy of the ranking tool has multiple opposing effects: first, it directly changes the distribution of candidates over indices, which in general means candidates ranked towards the top are more likely to be higher value. Secondly, because it changes the distribution of candidates, it also changes the frequency of status vectors: specifically, top-ranked candidates are more likely to be busy (because they are more likely to be high value candidates). Finally, it also changes the strategy of firms: because Algorithm 1 relies on the accuracy of the ranking tool, as the accuracy of this tool changes, the firm (hiring manager) will also update her strategy.

Theorem 3 considers all three of these simultaneous changes. Specifically, Theorem 3 shows that, when accuracy increases, if a firm is using the strategy first-free, the probability that the picked candidate is busy always decreases. Conversely, if the firm is using the strategy first-busy, then the probability that the picked candidate is busy *may increase* (if the firm's strategy window size stays constant) or *may decrease* (if the window size shrinks).

THEOREM 3. *Increasing the accuracy of the ranking tool* increases *the probability that the picked candidate is busy with among firms using first-free and* may decrease *the probability that the picked candidate is busy among firms using first-busy.*

Figure 2 illustrates this phenomenon with the RUM with Gumbel noise model. Specifically, both figures include colored, dashed lines showing the probability of a firm picking a busy candidate for every possible window length. Note that for both plots, as the accuracy increases, the colored lines increase: holding window length constant,
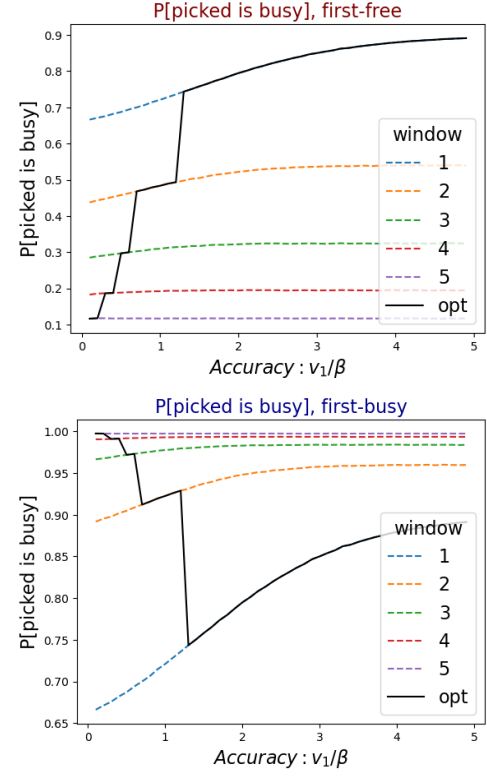

Figure 2: Figures showing how changing the accuracy of the tool changes how frequently the picked candidate is busy. The top shows a firm using "first free" strategy: note that increasing accuracy uniformly increases the probability that the picked candidate is busy. The bottom figure shows a firm using first-busy strategy: note that increasing accuracy could increase or decrease the probability that the picked candidate is busy. Parameters: $v_1 = 1, v_2 = 0, p_1 = 0.1, p_2 = 0.4$, each point run with 10^6 simulations. First-free strategy run with $\gamma = 10$, first-busy run with $\gamma = 1.6$.

increasing accuracy increases how frequently busy candidates are picked. For the top figure (first free), as the window width increases, the probability of picking a busy candidate decreases, but for the bottom figure (first-busy), the probability decreases with window width: this is simply because shrinking the window size makes it less likely that each firm will find their preferred free/busy type within the window. The solid black lines in Figure 2 shows how the firm's strategies change in respond to the ranking tool's increase in accuracy. Recall from Lemma 2 that increased accuracy always shrinks window size. If a firm is using first-free, then increased accuracy causes the firm to move up and to the right along the colored lines, always increasing how often busy candidates are picked. However, if a firm is using first-busy, then increasing accuracy causes the firm's strategy to move down and to the right - and ultimately could increase or decrease the probability that a busy candidate is picked, depending on the degree to which the accuracy changes.

5.3 Strategic use of ranking tool (welfare of ranking company):

Finally, we discuss the impact of increased accuracy on how often firms end up using the ranking tool as it was designed - specifically, on how often firms select the top-ranked candidate. In light of Theorem 3, one might be tempted to assume that increased accuracy leads to higher "congestion at the top," in the form of firms being more likely to select highly-ranked candidates even if they are busy. While it is true that this can occur, the relationship between accuracy and firm selection can also be more subtle: there are instances in which increasing the accuracy of the ranking tool actually *decreases* the likelihood of picking the top-ranked candidate.

Such a non-monotonic relationship between tool accuracy and usage has potentially important implications for tool design. For example, a firm that uses metrics like the click rate of top-ranked candidates might naturally (but incorrectly) infer from a reduced click rate that a supposed tool improvement led to reduced performance and welfare. Additionally, picking the top ranked item could be viewed as a direct positive for the ranking company, if, for example, they obtain higher advertising revenue when this occurs. The importance of studying how often the top candidate is picked is especially salient when increasing the accuracy of the ranking tool involves a heavy investment in increased data collection, increased algorithmic sophistication, or increased engineering effort. However, as we now show, an improved ranking tool can provide improved welfare for the firm (as guaranteed by Theorem 2) while simultaneously causing users to select top-ranked candidates less frequently. Specifically, Theorem 4 shows that increasing the accuracy of the ranking tool can either *increase* or *decrease* the probability that a firm selects the top ranked candidate.

THEOREM 4. *Increasing the accuracy of the ranking tool could* increase *or* decrease *how frequently the firm picks the top-ranked candidate.*

The full proof gives an exact condition for when increasing accuracy increases the chance that the top candidate is picked. For intuition, a smaller window size always *increases* the chances that the top candidate is picked: this should make intuitive sense, as the firm is considering fewer alternatives. However, holding window size constant, increasing accuracy could *increase* or *decrease* the probability that the top candidate is picked. Generally, if the firm is using a first-busy strategy, increasing accuracy increases the probability that the top candidate is busy, and then increases the chance that that candidate is picked. Conversely, if the firm is using a first-free strategy, increasing accuracy generally decreases the probability that the firm picks the top ranked candidate because the firm is actively avoiding busy candidates[3]. Figure 3 in the Appendix gives a corresponding figure to Figure 2 illustrating how the probability of the top-ranked item being picked could increase or decrease with accuracy. At a high level, Theorem 4 illustrates how, at the most basic level, increased accuracy of a ranking tool could lead to counterintuitive effects around how often firms end up picking the candidate suggested by the ranking.

---

[3]Interestingly, the full story is somewhat more nuanced than this: there exist cases where increasing accuracy can *decrease* the probability the top candidate is picked with first-busy, and *increase* the probability with first-free, typically with extreme values of p_2 and very inaccurate rankings: see Appendix B for more analysis.

6 Beyond superstar

In this section, we further relax the model and consider cases where the N items can come in more than two types of values and probabilities of being available. First, we show that very similar welfare properties hold even in the beyond-the-superstar setting, assuming that firms use a strategy similar to "first-free/first-busy" where they create a window size (shrinking with accuracy) and preferentially pick either the top-ranked free or busy candidate within that window. Secondly, we analyze *when* such a strategy is approximately optimal. While we show that the strategy space is much more complex in the beyond-the-superstar setting, we show that under reasonable noise models, a similar strategy achieves low expected error.

6.1 Welfare

First, we will discuss the impacts of increased accuracy on measures of societal welfare, focusing on extending results from Section 5. Throughout, we will assume that firms are using first-free or first-busy: specifically, that firms are preferentially picking either free or busy candidates within some window, and that the size of this window only shrinks with increased accuracy. Note that this is a very general condition, and we believe that this encompasses a wide range of natural strategies by the firm.

First, Theorem 5 generalizes Theorem 3, giving conditions for when the probability of the picked candidate being busy will increase or decrease as accuracy of the permutation increases:

THEOREM 5. *For the beyond-the-superstar setting, if the firm is using a first-free or first-busy strategy with a window size that weakly shrinks with increased accuracy, the following properties hold: 1) If the firm is using first-free, then increased accuracy increases the probability that the picked candidate is busy. 2) If the firm is using first free, then increased accuracy* may decrease *the probability that the picked candidate is busy.*

Similarly, Theorem 6 generalizes Theorem 4, showing that there exists cases where increasing the accuracy of the tool increases or decreases the probability that the top candidate is picked[4].

THEOREM 6. *In the beyond the superstar setting, increasing the accuracy of the ranking tool could* increase *or* decrease *how frequently the firm picks the top-ranked candidate.*

At a high level, we view Theorems 5 and 6 as showing that increased accuracy could have counterintuitive or negative impacts on welfare, even in settings more general than the superstar model.

6.2 Strategy

Finally, we will turn to analyzing the firm's strategy in the beyond-the-superstar setting. As mentioned previously, this setting can be substantially more complex, but we will show that a similar first-free/first-busy strategy can achieve relatively low expected error.

---

[4]This is directly implied by Theorem 4 because the superstar case is a subset of the beyond-the-superstar case, but the proof of Theorem 6 gives more general conditions for when we expect the probability to increase or decrease.

6.2.1 Restricting choice to either top-free or top-busy candidate.
First, recall that Lemma 1 showed that in the superstar setting, it is
always optimal to pick either the first free or first busy candidate
(e.g., it is never optimal to pick the second free candidate). Theorem
7 generalizes this to the beyond the superstar setting, again showing
that for reasonable noise models it is always optimal to pick either
the first free or first busy candidate: this dramatically reduces the
strategy space for the firm. While this result is natural, proving it
is surprisingly non-trivial.

THEOREM 7. *In the beyond the superstar setting, there exists prob-
ability distributions $\mathcal{P}$ such that for some realized status vector s,
the optimal choice is to pick neither the first free nor the first busy
candidate (e.g., instead to pick the 2nd busy candidate). However,
for commonly-used noise models (e.g. Mallows model and Random
Utility Model), it is always optimal to pick the first free or first busy
candidate.*

Specifically, Theorem 7 can be proven through a series of sub-
lemmas. First, Lemma 11 gives a specific example in the beyond-
the-superstar setting where the optimal choice is to pick the *second*
free item.

Lemma 4. *There exists a probability distribution $\mathcal{P}$ such that for
some realized status vector s, picking the* second *free item maximizes
expected utility, even if $\mathcal{P}$ is descending in expected value.*

However, the example in Lemma 11 is constructed by creating a
somewhat "unnatural" permutation distribution with 0 probability
on many permutations. Definition 3 defines a natural property on
permutations where, for every pair of permutations $\sigma, \tilde{\sigma}$ that differ
solely in a pairwise inversion, σ is at least as likely as $\tilde{\sigma}$.

Definition 1. *A probability distribution over permutations $\mathcal{P}$ is
called* inversion-monotone *if it satisfies the following condition: for
any permutation σ with a pair of indices $i < j$ such that $v_{\sigma_i} > v_{\sigma_j}$,
we construct a corresponding permutation $\tilde{\sigma}$ where $\tilde{\sigma} = \sigma$ except for
indices i, j, which we have flipped: $v_{\tilde{\sigma}_i} = v_{\sigma_j}, v_{\tilde{\sigma}_j} = v_{\sigma_i}$. Then, we
require $P[\sigma] \geq P[\tilde{\sigma}]$.*

Lemma 12 shows why this property is important: any permuta-
tion distribution satisfying it must result in the optimal strategy
being to pick either the first free or first busy candidate.

Lemma 5. *Consider any vector of realized status vector s, with $s_i =
s_j$, for some $i < j$. Then, if the probability distribution of permutations
is inversion-monotone as in Definition 3, the optimal solution will
always be to pick the first free item or the first busy item.*

Finally, the proof of Theorem 7 concludes by showing that
commonly-used permutation distributions are inversion monotonic.
While we believe that this property may be of independent interest,
to our knowledge, we are the first paper to prove such a property
for the Random Utility model with Gaussian noise.

Lemma 6. *The following permutation distributions are inversion
monotone: 1) Mallows, 2) RUM with Gumbel noise, and 3) RUM with
noise from any identical, symmetric, noise distribution (e.g. Gaussian).*

6.2.2 Algorithm for picking candidate. Having established that the
optimal choice is either the top-ranked free candidate or top-ranked
busy candidate, we now turn to the question of how to pick between

them. Algorithm 2 generalizes Algorithm 1, providing an algorithm
for how to pick candidates in the beyond-the-superstar setting.

Algorithm 2: Given candidate values $\{v_i\}$, probabilities
of being free $\{p_i\}$, accuracy of ranking tool $\{P[\sigma]\}$, $s \in
\{0, 1\}$ for status of top-ranked candidate, and index j of first
candidate with different free-busy status from s.

---

1 for $v_i, v_k \mid v_i > v_k$ /* Go over all ordered pairs */
2 do
3   Define $r_{i,k} = p_k \cdot (1 - p_i)/((1 - p_k) \cdot p_i)$.
4   Define $\sigma_{i,k,j}$ as permutation with every item ordered
      correctly, but with v_i ranked 1st and v_k is ranked jth.
5   Define $\tilde{\sigma}_{i,k,j} = \sigma_{i,k,j}$ except with indices 1, j flipped.
6   **if** $s = 0$ /* If top candidate is busy. */
7   **then**
8 | $G_{ik} = r_{i,k} \cdot P[\sigma] \cdot (v_k - v_i/\gamma_i) + P[\tilde{\sigma}] \cdot (v_i - v_k/\gamma_k)$
9   **else**
                   /* If top candidate is free */
10 | $G_{ik} = r_{i,k} \cdot P[\sigma] \cdot (v_k/\gamma_k - v_i) + P[\tilde{\sigma}] \cdot (v_i/\gamma_i - v_k)$
11   **end**
12 end
13 $\mathcal{G}_1 = \{G_{ik} \mid G_{ik} \geq 0\}, \mathcal{G}_j = \{G_{ik} \mid G_{ik} < 0\}$
14 if $|G_j| > |G_1|$ /* Vote on index to pick */
15 then
16   **return** j
17 else
18   **return** 1
19 end

At a high level, Algorithm 2 works by iterating over all pairs of
candidates and allowing them to "vote" about whether it would be
better to pick the candidate ranked in index 1 or in index j. The
ultimate choice comes down to what the majority of pairs decide.
Next, we will prove several properties about Algorithm 2. First,
Lemma 7 shows that Algorithm 2 reduces to Algorithm 1 in the
superstar setting, as desired.

Lemma 7. *In the superstar setting, Algorithm 2 becomes Algorithm
1.*

Next, Lemma 8 shows that Algorithm 2 is actually also a k-free/k-
busy strategy with a window that shrinks with increased accuracy,
exactly as Algorithm 1 is. This result is especially important, be-
cause it implies that Theorem 5 applies, meaning that any firm
using the strategy in Algorithm 2 will result in welfare implications
that match those in the superstar setting.

Lemma 8. *Given a constant γ, Algorithm 2 is a k-free/k-busy strat-
egy: it can be reduced to picking a window size and preferentially
picking the top-ranked free or busy candidate within that window. The
window size shrinks with increased accuracy, and if the firm is prefer-
entially picking busy candidates, the window shrinks with increased
γ (increasing if the firm is preferentially picking free candidates).*

Finally, Lemma 9 analyzes the sub-optimality of Algorithm 2.
While Algorithm 2 does satisfy several nice properties (as in Lemma

8), it is an approximation to directly calculating the optimal posterior choice, and as such as some degree of error. In particular, two sources of error are 1) making a decision based on majority "vote" across pairs of candidates, which leads to error from candidates who "voted" the opposite direction, and 2) the approximation in G_{ik}, which uses permutations $\sigma, \tilde{\sigma}$ that make it as easy as possible to select the top candidate.

Lemma 9. *For a particular index* $1, j$, *denote* $\mathcal{G}_+$ *as* $\{v_i, v_j \mid G_{i,k} > 0\}$, $\mathcal{G}_-$ *as* $\{v_i, v_j \mid G_{i,k} < 0\}$, $\mathcal{I}_{1j}(i,k)$ *as the event that items* $\{i, k\}$ *are in indices* $\{1, j\}$, *and* $\tilde{G}_{i,k}$ *as* G_{ik} *calculated in Algorithm 2, but with* $\sigma'_{i,k,j}, \tilde{\sigma}'_{i,k,j}$ *instead defined as permutations where every item is* incorrectly *ordered. Then, if Algorithm 2 picks a candidate in index* j, *the error is upper bounded by:*

$$\sum_{i,k \in \mathcal{G}_-} P[\mathcal{I}_{1j}(i,k)] \cdot (v_i - v_k)$$

and if Algorithm 2 picks a candidate in index 1, *the error is upper bounded by:*

$$\sum_{i,k \in \mathcal{G}_+} P[\mathcal{I}_{1j}(i,k)] \cdot (v_i - v_k) + \sum_{i,k \in \mathcal{G}_-} P[\mathcal{I}_{1j}(i,k)] \cdot \max(-\tilde{G}_{i,k}, 0)$$

Note that the error of this algorithm is high when there is a large chance of observing candidates in indices $1, j$ where the relative benefits of being busy versus free are the *opposite* of candidates in general. However, this error bound is very natural: in such a setting, there is no consistent signal in value related to a candidate being free or busy, and thus we would not expect any algorithm preferentially selecting candidates based on status to have good performance. Figure 4 (bottom) in the Appendix illustrates the expected error of Algorithm 2 in the beyond-the-superstar setting, showing that in many parameter regimes, Algorithm 2 has low or 0 error.

7 Discussion

In this paper, we have proposed a simple model to study cases where strategic, self-interested agents use algorithmically generated rankings in the presence of side information, specifically a free-busy status vector for different candidates. Our results show that even this relatively simple model can show surprisingly complex phenomena.

Firm's strategy: First, we began by analyzing the superstar model, where a single candidate has higher value to the company than all other candidates. In Section 4 we show that the firm's strategy can admit a surprisingly simple form: decide whether to preferentially seek out free or busy candidates, and then decide how far down the list to look for them based on the accuracy of the ranking tool. As the accuracy of the ranking tool increases, the firm only shrinks its "window size" and the firm's welfare only increases.

Welfare of candidates and ranking company: When firms select preferentially based on the free-busy (e.g. employment) status of candidates, this immediately leads to differences in selection rate based on employment status. Preferentially selecting busy candidates or free candidates could have positive or negative effects (e.g. impacts on unemployment rate or competing offers). In Section 5, we showed that there are scenarios where increasing the accuracy

of the ranking tool could *increase* how frequently the firm selects a busy candidate.

Often the incentives of ranking companies, their clients, and societal welfare are assumed to be at least weakly aligned: producing higher-quality ranking tools should increase the value to both companies and clients. However, our results show that their incentives may be misaligned: in particular, a more accurate ranking tool can reduce measures of social welfare and fairness, and can even reduce the welfare of the ranking company itself by decreasing how frequently the top candidate is selected. Our results suggest that improving overall welfare may require more precise interventions than simply increasing the accuracy of tools.

Beyond superstar: Finally, in Section 6, we discuss how our results generalize beyond the Superstar model. Specifically, we show that reasoning about the strategy of agents using ranking tools becomes substantially more subtle, and we prove properties about permutation models that may be of independent interest. However, given reasonable assumptions, we obtain similar results for strategy and welfare as in the Superstar setting, suggesting that our results generalize beyond a specific model.

Implications: While we focus on a specific motivating example of an agent with side information using an algorithmic ranking tool, more broadly we are interested in how algorithmic tools are used as part of broader systems. In general, many algorithmic tools are used not in isolation, but in conjunction with other systems or sources of information. Our motivation in this paper is to show that even in a relatively simple human-algorithmic system, increased algorithmic accuracy can have non-intuitive impacts. In general, we wish to illustrate the importance of understanding such impacts and motivate wider study of such settings.

Extensions: There are multiple fascinating extensions to our model. For example, in this work we assumed that job candidates are honest about their free-busy status, but if firms preferentially select candidates based on this status, job candidates may be incentivized to mis-represent whether or not they are free. An interesting direction for future work would be to explore the extent to which such settings would impact welfare of firms and candidates. Separately, one policy intervention may be to deliberately hide their free/busy status from companies: this could potentially be in pursuit of increasing fairness in selection rates among candidates with different free/busy status. If firms are aware that busy candidates are less likely to accept offers, but are unaware of which candidates are busy, the optimal strategy of how firms select candidates changes. Additionally, these results generally assume that firms perfectly know certain parameters such as the r, γ (which could be viewed as measuring the efficiency of the market and the attractiveness of any given firm, respectively). In reality, firms only have access to estimates of these values, and in particular they may systematically mis-estimate these values, for example, assuming that their firm is more attractive than it really is. In this case, firms may use sub-optimal strategies, which may lead to decreased welfare for multiple agents. Studying how each of these extensions change the overall relationship between increased accuracy and welfare would be interesting avenues for future work.

References

[1] Abhijit V Banerjee. 1992. A simple model of herd behavior. *The quarterly journal of economics* 107, 3 (1992), 797–817.

[2] Sushil Bikhchandani, David Hirshleifer, and Ivo Welch. 1992. A theory of fads, fashion, custom, and cultural change as informational cascades. *Journal of political Economy* 100, 5 (1992), 992–1026.

[3] Emily Black, Manish Raghavan, and Solon Barocas. 2022. Model Multiplicity: Opportunities, Concerns, and Solutions. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency* (Seoul, Republic of Korea) *(FAccT '22)*. Association for Computing Machinery, New York, NY, USA, 850–863. doi:10.1145/3531146.3533149

[4] Rishi Bommasani, Kathleen A. Creel, Ananya Kumar, Dan Jurafsky, and Percy S Liang. 2022. Picking on the Same Person: Does Algorithmic Monoculture lead to Outcome Homogenization?. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 3663–3678. https://proceedings.neurips.cc/paper_files/paper/2022/file/17a234c91f746d9625a75cf8a8731ee2-Paper-Conference.pdf

[5] Rémi Castera, Patrick Loiseau, and Bary S.R. Pradelski. 2022. Statistical Discrimination in Stable Matchings. In *Proceedings of the 23rd ACM Conference on Economics and Computation* (Boulder, CO, USA) *(EC '22)*. Association for Computing Machinery, New York, NY, USA, 373–374. doi:10.1145/3490486.3538364

[6] A Feder Cooper, Katherine Lee, Solon Barocas, Christopher De Sa, Siddhartha Sen, and Baobao Zhang. 2023. Is My Prediction Arbitrary? Measuring Self-Consistency in Fair Classification. *arXiv preprint arXiv:2301.11562* (2023).

[7] Apostolos Filippas, John J Horton, and Diego Urraca. 2023. Advertising as Coordination: Evidence from a Field Experiment. (2023).

[8] John Horton, Nicole Immorlica, Brendan Lucier, and Michelle Si. 2023. Costly Signaling in Online Labor Markets.

[9] Hsiang Hsu and Flavio Calmon. 2022. Rashomon Capacity: A Metric for Predictive Multiplicity in Classification. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 28988–29000. https://proceedings.neurips.cc/paper_files/paper/2022/file/ba4caa85ecdcafbf9102ab8ec384182d-Paper-Conference.pdf

[10] Meena Jagadeesan, Michael I. Jordan, Jacob Steinhardt, and Nika Haghtalab. 2023. Improved Bayes Risk Can Yield Reduced Social Welfare Under Competition. arXiv:2306.14670 [cs.GT]

[11] Shomik Jain, Vinith Suriyakumar, Kathleen Creel, and Ashia Wilson. 2023. Algorithmic Pluralism: A Structural Approach To Equal Opportunity. arXiv:2305.08157 [cs.CY]

[12] Jon Kleinberg and Manish Raghavan. 2021. Algorithmic monoculture and social welfare. *Proceedings of the National Academy of Sciences* 118, 22 (2021), e2018340118.

[13] Ulrich Krengel and Louis Sucheston. 1977. Semiamarts and finite values. (1977).

[14] R Duncan Luce et al. 1959. *Individual choice behavior*. Vol. 4. Wiley New York.

[15] Colin L Mallows. 1957. Non-null ranking models. I. *Biometrika* 44, 1/2 (1957), 114–130.

[16] Kenny Peng and Nikhil Garg. 2023. Monoculture in Matching Markets. arXiv:2312.09841 [cs.GT]

[17] Kenny Peng and Nikhil Garg. 2024. Wisdom and foolishness of noisy matching markets. *arXiv preprint arXiv:2402.16771* (2024).

[18] Kenny Peng, Manish Raghavan, Emma Pierson, Jon Kleinberg, and Nikhil Garg. 2023. Reconciling the accuracy-diversity trade-off in recommendations. arXiv:2307.15142 [cs.IR]

[19] Robin L Plackett. 1975. The analysis of permutations. *Journal of the Royal Statistical Society Series C: Applied Statistics* 24, 2 (1975), 193–202.

[20] Ashudeep Singh, David Kempe, and Thorsten Joachims. 2021. Fairness in Ranking under Uncertainty. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 11896–11908. https://proceedings.neurips.cc/paper_files/paper/2021/file/63c3ddcc7b23daa1e42dc41f9a44a873-Paper.pdf

[21] Louis L Thurstone. 1994. A law of comparative judgment. *Psychological review* 101, 2 (1994), 266.

[22] Martin Weitzman. 1978. *Optimal search for the best alternative*. Vol. 78. Department of Energy.

[23] Ivo Welch. 1992. Sequential sales, learning, and cascades. *The Journal of finance* 47, 2 (1992), 695–732.

[24] Meike Zehlike, Ke Yang, and Julia Stoyanovich. 2021. Fairness in ranking: A survey. *arXiv preprint arXiv:2103.14000* (2021).
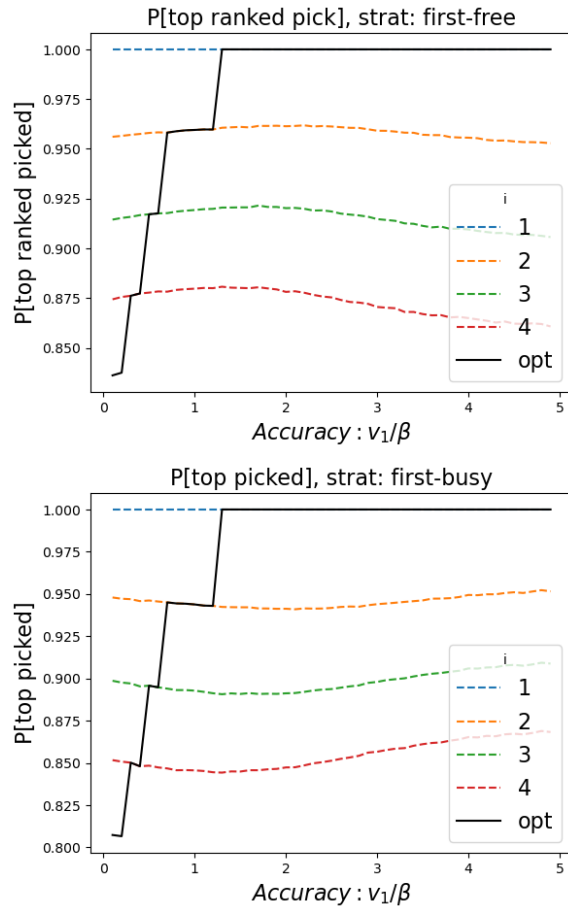
Figure 3: Figure showing examples where increasing accuracy could lead to *increased* probability of picking the top ranked candidate for first-free strategies (top figure) or decreased for first-busy (bottom). In both cases, phenomeona occurs for very low accuracy (far left part of plot). Both figures show $N = 10$ with only lines shown for $i \in [1, 4]$ to show detail, with RUM with Gumbel noise. Top has $p_1 = 0.01 < p_2 = 0.05$, bottom has $p_1 = 0.9, p_2 = 0.95$. Numerical simulations with 10^6 simulations per point.

The full paper, including all proofs, can be viewed at https://anonymous.4open.science/r/www-hiring-1CBF/hiring_WWW.pdf.
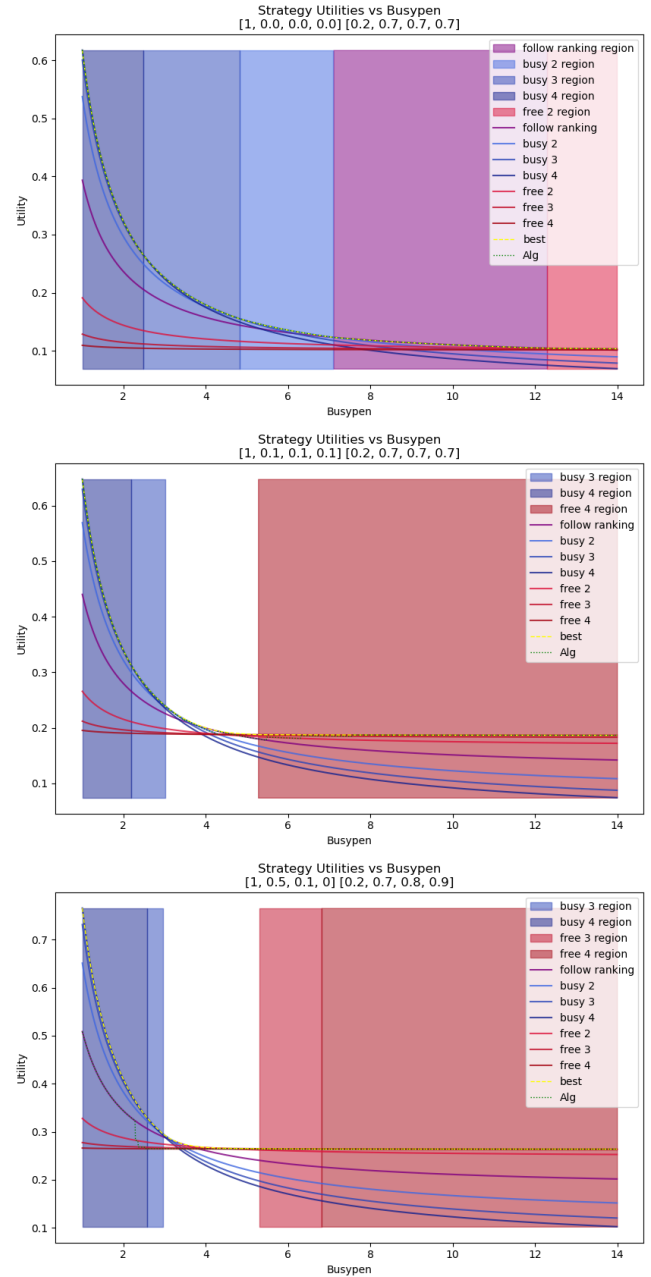

Figure 4: Simulation of performance of strategies with $N = 4$. Each line gives the expected utility of different strategies, where "best" gives the Bayes-optimal posterior strategy and Alg gives the performance of Algorithm 2 (which reduces to Algorithm 1 in the superstar setting). The shaded regions illustrate where different named strategies happen to also be optimal: for example, the purple region is one where "follow the ranking" is exactly optimal. Unshaded regions are those where the optimal strategy cannot be described as k-free or k-busy. However, note that the error (gap between yellow and green lines) is often extremely small. The top and middle are both superstar settings, while the bottom is beyond-the-superstar. Note that the top has 0 error always because $v_2 = 0$ (Theorem 1), while the middle has nonzero error. The bottom (beyond-the-superstar) has regions with non-zero error, but largely follows the optimal strategy.

A Proofs for Section 4

First, we will present general results on properties of distributions over rankings (Appendix A.1) and then prove results given in Section 4.

A.1 Properties of ranking permutations

Definition 2. *[Accuracy: superstar] A ranking $\{P[\sigma^i]\}'$ is more accurate than $\{P[\sigma^i]\}$ if both conditions hold:*

(1) *Ratio of probabilities: The relative probability of the high value candidate being ranked more highly always increases: $\frac{P[\sigma^i]'}{P[\sigma^j]'} \geq \frac{P[\sigma^i]}{P[\sigma^j]} \ \forall i <$
j with the inequality strict for at least one i, j pair.*

(2) *Monotonic increase: if the probability of the high value item being found at index k decreases, then it also decreases for all $j > k$,
or: if $P[\sigma^k]' < P[\sigma^k]$, then $P[\sigma^j]' < P[\sigma^j]$ for all $j > k$. Note that an immediate corollary of this property is majorization:
$\sum_{i=1}^{j} P[\sigma^i]' \geq \sum_{i=1}^{j} P[\sigma^i]$ for all j.*

THEOREM 8. *[From [19], [14]] Consider sets of items with values v_i, and a Random Utility Model with i.i.d. Gumbel noise $G(\mu, \beta)$ is added to
each value. Then, the probability of observing the ordering σ is given by:*

$$\prod_{j=1}^{N} \frac{\exp(v_{\sigma_j}/\beta)}{\sum_{\ell=j}^{N} \exp(v_{\sigma_\ell}/\beta)}$$

Proposition 1. *Consider the superstar setting (1 item of value $v_1 > v_2 \geq 0$, and $n - 1$ elements of item v_2. If i.i.d. Gumbel noise (with scale
parameter β) is added to each value, then the probability of the high value item being in index i is given by:*

$$P[\sigma^i] = \frac{\exp((v_1 - v_2)/\beta)}{\exp((v_1 - v_2)/\beta) + (n-1)} \cdot \prod_{k=1}^{i-1} \frac{1}{1 + \frac{\exp((v_1 - v_2)/\beta) - 1}{n-k}}$$

PROOF. We can prove this by relying on the Plackett-Luce model (Theorem 8), which says that the probability of observing any permutation
σ is given by:

$$\prod_{j=1}^{N} \frac{\exp(v_{\sigma_j}/\beta)}{\sum_{\ell=j}^{N} \exp(v_{\sigma_\ell}/\beta)}$$

Denote σ^k as a permutation with the high-value candidate in index k. Note that there are $(N-1)!$ such permutations, because there are
$(N-1)!$ different ways to arrange the $N-1$ low-value candidates. Applying Theorem 8 in the superstar setting gives:

$$\prod_{j=1}^{k-1} \frac{\exp(v_2/\beta)}{\exp(v_1/\beta) + (N-j) \cdot \exp(v_2/\beta)} \cdot \frac{\exp(v_1/\beta)}{\exp(v_1/\beta) + (N-k) \cdot \exp(v_2/\beta)} \cdot \prod_{j=k}^{N} \frac{\exp(v_2/\beta)}{\exp(v_2/\beta) \cdot (N-j)}$$

$$= \prod_{j=1}^{k-1} \frac{1}{\exp((v_1 - v_2)/\beta) + (N-j)} \cdot \frac{\exp((v_1 - v_2)/\beta)}{\exp((v_1 - v_2)/\beta) + (N-k)} \cdot \frac{1}{\prod_{j=k}^{N}(N-j)}$$

$$= \frac{\exp((v_1 - v_2)/\beta)}{\exp((v_1 - v_2)/\beta) + N - 1} \cdot \prod_{j=2}^{k} \frac{1}{\exp((v_1 - v_2)/\beta) + N - j} \cdot \frac{1}{\prod_{j=k}^{N}(N-j)}$$

Reindexing:

$$= \frac{\exp((v_1 - v_2)/\beta)}{\exp((v_1 - v_2)/\beta) + N - 1} \cdot \prod_{j=1}^{k-1} \frac{1}{\exp((v_1 - v_2)/\beta) + N - j - 1} \cdot \frac{1}{\prod_{j=k}^{N}(N-j)}$$

Next, we multiply this by $(N-1)!$ to get the total probability $P[\sigma^j]$ (probability of any permutation with high value item in index j). We
note that:

$$\frac{(N-1)!}{\prod_{j=k}^{N}(N-j)} = \prod_{j=1}^{k-1}(N-j)$$

which means that the total probability can be written as

$$\frac{\exp((v_1 - v_2)/\beta)}{\exp((v_1 - v_2)/\beta) + N - 1} \cdot \prod_{j=1}^{k-1} \frac{N - j}{\exp((v_1 - v_2)/\beta) + N - j - 1}$$

$$= \frac{\exp((v_1 - v_2)/\beta)}{\exp((v_1 - v_2)/\beta) + N - 1} \cdot \prod_{j=1}^{k-1} \frac{1}{1 + \frac{\exp((v_1 - v_2)/\beta) - 1}{N - j}}$$
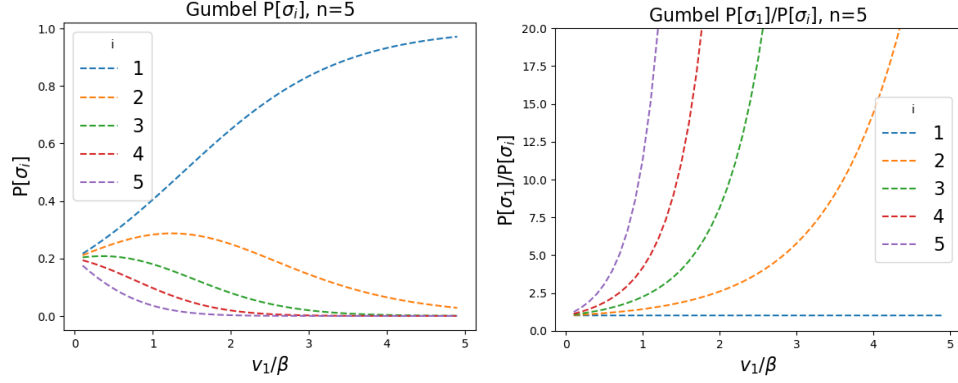
as desired. □

Figure 5: Left: Figure showing the probability of the high-value item being in index i, for $i \in [1, 5]$ and $n = 5$, as a function of accuracy with the RUM with Gumbel noise. Note that if the line decreases for i, it also decreases for $j > i$ (second part of Definition 2). Right: Figure showing ratio of probability of high value item being ranked first over probability of it being in index i. Note that this ratio always increases (first part of Definition 2).

From Proposition 1 we can immediately observe that there is a multiplicative relationship between each of the permutations:

Observation 1. *In the RUM with Gumbel noise and the superstar setting, for any $i < j$, we have:*

$$\frac{P[\sigma^i]}{P[\sigma^j]} = \prod_{k=i}^{j-1} \left(1 + \frac{\exp((v_1 - v_2)/\beta) - 1}{n - k}\right)$$

Corollary 1. *For the RUM with Gumbel noise, decreasing β induces a distribution $\{P[\sigma^i]'\}$ that is more accurate than $\{P[\sigma^i]\}$ according to Definition 2.*

Proof. First, we will prove the first property: that the relative probability of the high value candidate being ranked more highly always increases: $\frac{P[\sigma^i]'}{P[\sigma^j]'} \geq \frac{P[\sigma^i]}{P[\sigma^j]} \ \forall i < j$ with the inequality strict for at least one i, j pair. This comes almost immediately from Observation 1. Note that increasing $(v_1 - v_2)/\beta$ causes the term within Observation 1 to increase, thus increasing the ratio of probabilities $P[\sigma^i]/P[\sigma^j]$.

Second, we will prove the second property: if the probability of the high value item being found at index k decreases, then it also decreases for all $j > k$, or: if $P[\sigma^k]' < P[\sigma^k]$, then $P[\sigma^j]' < P[\sigma^j]$ for all $j > k$.

First, we note that from Proposition 1 we can write:

$$P[\sigma^i] = P[\sigma^1] \cdot \prod_{k=1}^{i-1} q_k$$

for $q_k < 1$, with q_k shrinking with increased accuracy $((v_1 - v_2)/\beta)$. We can write:

$$P[\sigma^i]' - P[\sigma^i] = P[\sigma^1]' \cdot \prod_{k=1}^{i-1} q_k' - P[\sigma^1] \cdot \prod_{k=1}^{i-1} q_k$$

Suppose that $P[\sigma^i]' - P[\sigma^i] < 0$ for some i: we will aim to show that this must also hold for $i + 1$.

We can observe that the model in Proposition 1 satisfies $P[\sigma^1]' > P[\sigma^1]'$, so it must be that $\prod_{k=1}^{i-1} q_k' < \prod_{k=1}^{i-1} q_k$. Because $q_i' \leq q_i < 1$, we must have that $\prod_{k=1}^{i} q_k' < \prod_{k=1}^{i} q_k$, which implies that

$$P[\sigma^1]' \cdot \prod_{k=1}^{i-1} q_k' - P[\sigma^1] \cdot \prod_{k=1}^{i-1} q_k = P[\sigma^i]' - P[\sigma^i] < 0$$

as desired. □

The most common noise distribution with the Random Utility Model is Gaussian noise, while our theoretical results (Proposition 1) uses Gumbel noise as in the Placket-Luce distribution ([14, 19]). Here, we show that empirical results for Gaussian noise closely mimic theoretical results for Gumbel noise. Figure 6 shows PDFS for Gumbel and Gaussian distributions, Figure 7 shows the $P[\sigma^i]$ values for the superstar model, and Figure 8 shows the ratio $P[\sigma^1]/P[\sigma^i]$, both for the Gumbel and Gaussian noise distribution. In all settings, the β and standard deviation values are set to be equal.
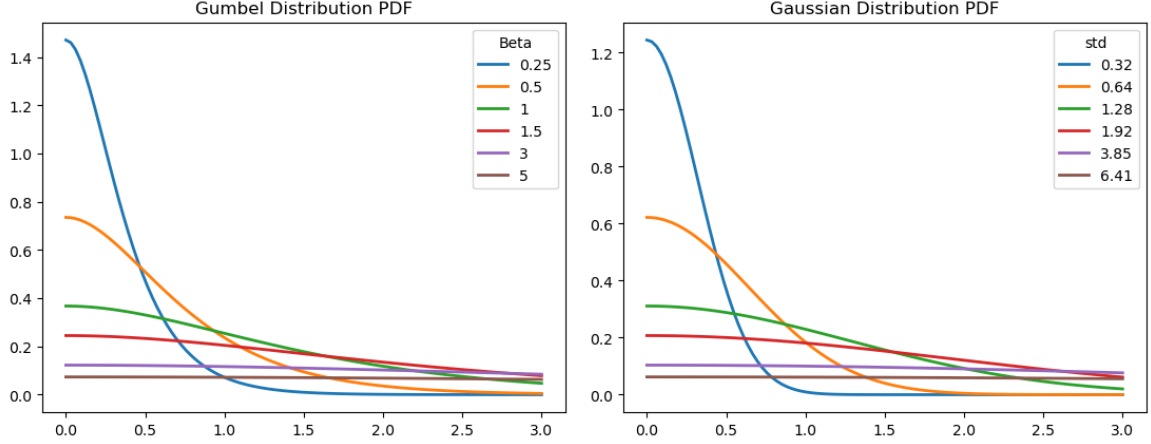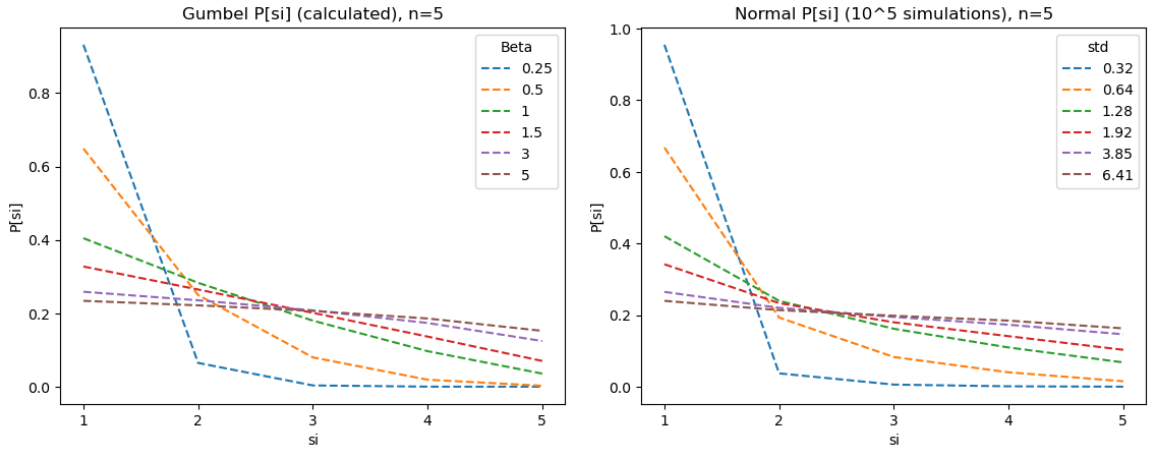
Figure 6: Figure of PDFs of Gumbel and Gaussian noise, respectively.

Figure 7: Figure of $P[\sigma^i]$ for RUM with Gumbel and Gaussian noise, respectively. Gumbel is calculated exactly while Gaussian is simulated numerically.

A.2 General proofs for Section 4

Lemma 1. *In the superstar setting, given a realized status vector s, it is always optimal to either pick the first (top-ranked) free item or the first busy item.*

Proof. In order to prove this, we will prove that the expected value of items given the status vector s is descending, or:

$$\mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_i} \mid s] \geq \mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_i} \mid s] \quad i < j$$

If this condition holds, then for any two i, j given $s_i = s_j$, we know that the lower ranked of the two has higher utility, and thus no optimal strategy could every pick the higher ranked item.

Next, we will show that the superstar setting always satisfies this property. The expected value of the ith entry is given by:

$$\mathbb{E}[v_{\sigma_i} \mid s] = \sum_{\sigma \sim \mathcal{P}} P[\sigma \mid s] \cdot v_{\sigma_i} = \sum_{k=1}^{N} \frac{P[s \mid \sigma^k] \cdot P[\sigma^k]}{P[s]} \cdot v_{\sigma_i^k}$$

where we have used that in the superstar setting, each of the permutations can be identified by the index of the high-value item. By identical reasoning, the expected value of the jth entry is given by:

$$\mathbb{E}[v_{\sigma_j} \mid s] = \sum_{\sigma \sim \mathcal{P}} P[\sigma \mid s] \cdot v_{\sigma_j} = \sum_{k=1}^{N} \frac{P[s \mid \sigma^k] \cdot P[\sigma^k]}{P[s]} \cdot v_{\sigma_j^k}$$
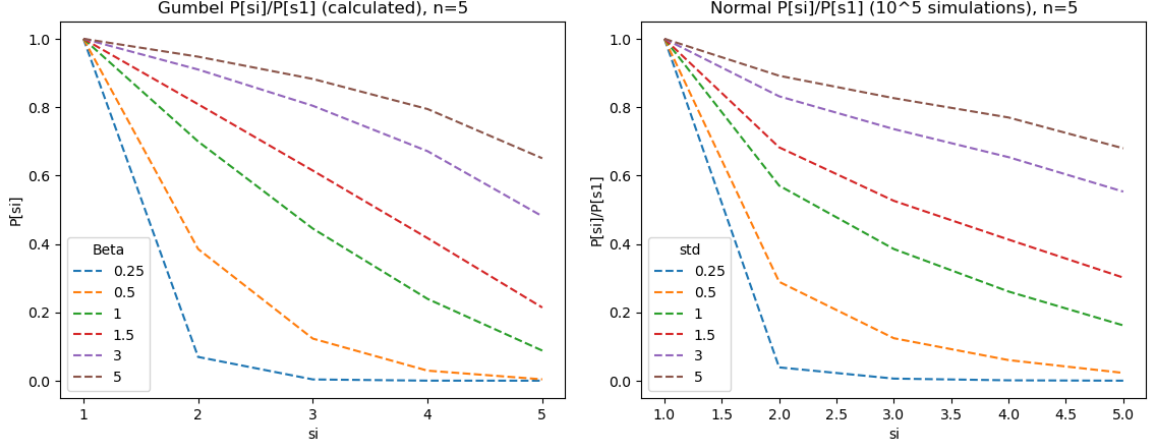
13

Figure 8: Figure of $\frac{P[\sigma^1]}{P[\sigma^i]}$ for RUM with Gumbel and Gaussian noise, respectively. Gumbel is calculated exactly while Gaussian is simulated numerically. Note that $P[\sigma^1]/P[\sigma^i]$ is strictly increasing as noise decreases for both Gumbel and Gaussian noise.

The condition we wish to show is:

$$\sum_{k=1}^{N} \frac{P[s \mid \sigma_k] \cdot P[\sigma_k]}{P[s]} \cdot v_{\sigma_i^k} \geq \sum_{k=1}^{N} \frac{P[s \mid \sigma_k] \cdot P[\sigma_k]}{P[s]} \cdot v_{\sigma_j^k}$$

Or:

$$\sum_{k=1}^{N} P[s \mid \sigma_k] \cdot P[\sigma_k] \cdot \left(v_{\sigma_i^k} - v_{\sigma_j^k} \right) \geq 0$$

Because we are in the superstar setting, we know that $v_{\sigma_i^k} = v_{\sigma_j^k}$ (both items are low value) unless $k \in \{i, j\}$. Dropping all but these values of k and using $v_1 \geq v_2$ for the high and low values respectively gives:

$$P[s \mid \sigma^i] \cdot P[\sigma^i] \cdot (v_1 - v_2) + P[s \mid \sigma^j] \cdot P[\sigma^j] \cdot (v_2 - v_1) \geq 0$$

which is positive whenever:

$$(v_1 - v_2) \cdot (P[s \mid \sigma^i] \cdot P[\sigma^i] - P[s \mid \sigma^j] \cdot P[\sigma^j]) \geq 0$$

By assumption, $v_1 \geq v_2$, and $s_i = s_j$. This latter assumption tells us that $P[s \mid \sigma^i] = P[s \mid \sigma^j]$: because the status in index i, j are identical, they are equally likely to occur if the high-value item is in index i or j. Finally, we require that the expected value of the distribution $\mathbb{E}\sigma \sim \mathcal{P}[v_{\sigma_i}]$ is always descending in i, which for the superstar setting implies that the high-value item is always (weakly) more likely to be at lower indices ($P[\sigma^i] \geq P[\sigma^j]$). Taken together, this implies that the above condition reduces to:

$$(v_1 - v_2) \cdot P[s \mid \sigma^i] \cdot (P[\sigma^i] - P[\sigma^j]) \geq 0$$

which always holds. □

Lemma 3. *In the limit of an extremely uninformative ranking, there exist scenarios where the optimal strategy differs from Algorithm 1.*

PROOF. Consider a setting where $N = 3$ and we have a completely uninformative ranking: $P[\sigma^1] = P[\sigma^2] = P[\sigma^3]$. Then, our posterior will be completely unrelated to the ranking and depend solely on the status vector.

We will consider our posterior after two different status vectors, and shown that the optimal strategy described by both of these shows that neither first-free nor first-busy can be optimal, because there exist settings where the firm would preferentially pick either free or busy candidates.

Case 1: [0, 0, 1]

From our analysis of Theorem 1, picking the bottom ranked (free) candidate is optimal when Equation 1 is satisfied:

$$P[\sigma^1] \cdot r \cdot (v_1/\gamma_1 - v_2) \leq P[\sigma^j] \cdot (v_1 - v_2/\gamma_2) + r \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot P[\sigma^{2 \leq i < j}] + v_2 \cdot (1 - 1/\gamma_2) \cdot \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i]$$

which in our setting with $N = 3$ and $P[\sigma^1] = P[\sigma^2] = P[\sigma^3]$ simplifies to:

$$r \cdot (v_1/\gamma_1 - v_2) \leq (v_1 - v_2/\gamma_2) + r \cdot v_2 \cdot (1 - 1/\gamma_2)$$
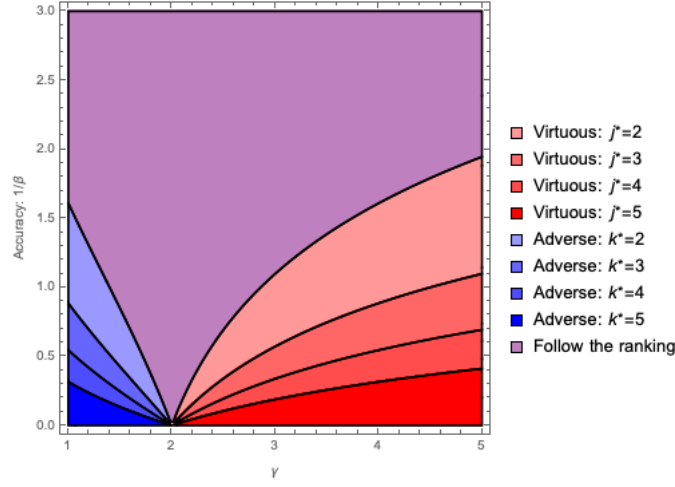
Figure 9: Illustration of optimal strategy for firm, given $N = 5$ candidates with $v_2 = 0$ and permutations given by the RUM with Gumbel noise model). The x axis varies the γ busy penalty, while the y axis varies accuracy as parameterized by the $1/\beta$ Gumbel noise parameter (higher values increase accuracy). Shades of red indicate regions where first-free selection is the optimal strategy and shades of blue indicate where first-busy selection is (darker shades indicate regions where the firm has larger j^* or k^* and thus "hunts" further down the list to find a candidate with their preferred free/busy status. The purple region is where the optimal strategy is "follow the ranking". This figure uses $r = 2$, so $\gamma = 2$ is the critical point on the x-axis where firms switch from using first-free selection to first-busy selection.

Simplifying gives:
$$r \cdot (v_1/\gamma_1 - v_2 - v_2 \cdot (1 - 1/\gamma_2)) \leq v_1 - v_2/\gamma_2$$
$$r \cdot (v_1/\gamma_1 - v_2 \cdot (2 - 1/\gamma_2)) \leq v_1 - v_2/\gamma_2$$

Given $v_1/\gamma_1 - v_2 \cdot (2 - 1/\gamma_2) > 0$, this implies:
$$r \leq \frac{v_1 - v_2/\gamma_2}{v_1/\gamma_1 - v_2 \cdot (2 - 1/\gamma_2)}$$

Case 2: [1, 1, 0]
From our analysis of Theorem 1, picking the last ranked (busy) candidate is optimal when Equation 4 is satisfied:

$$P[\sigma^1] \cdot (v_1 - v_2/\gamma_2) + v_2 \cdot (1 - 1/\gamma_2) \cdot P[\sigma^{2 \leq i < j}] + (1 - 1/\gamma_2) \cdot v_2 \cdot \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \leq P[\sigma^j] \cdot r \cdot (v_1/\gamma_1 - v_2)$$

which in our setting with $N = 3$ and $P[\sigma^1] = P[\sigma^2] = P[\sigma^3]$ simplifies to:

$$(v_1 - v_2/\gamma_2) + v_2 \cdot (1 - 1/\gamma_2) \leq r \cdot (1/\gamma_1 \cdot v_1 - v_2)$$

Simplifying gives:
$$v_1 + v_2 \cdot (1 - 2/\gamma_2) \leq r \cdot (v_1/\gamma_1 - v_2)$$

Assuming $v_1/\gamma_1 - v_2 > 0$, this implies:
$$\frac{v_1 + v_2 \cdot (1 - 2/\gamma_2)}{v_1/\gamma_1 - v_2} \leq r$$

Analysis: Taken together, Case 1 and Case 2 imply that the optimal strategy given observed status vectors $[0, 0, 1]$ and $[1, 1, 0]$ is always to pick the bottom candidate when the condition below holds:
$$\frac{v_1 + v_2 \cdot (1 - 2/\gamma_2)}{v_1 \cdot /\gamma_1 - v_2} < r < \frac{v_1 - v_2/\gamma_2}{v_1/\gamma_1 - v_2 \cdot (2 - 1/\gamma_2)}$$

Note that if $v_2 = 0$ this evaluates to:
$$\gamma_1 < r < \gamma_1$$

which is never satisfied. However, with $v_2 > 0$, it is possible to set parameters such that this evaluates to a positive term: consider $v_1 = 1, v_2 = 0.3, \gamma_1 = \gamma_2 = 2$, where the bounds evaluate to:
$$7.25 \leq r \leq 17$$

In this setting, we have shown that the optimal strategy can *never* be approximated by "first-free, first-busy". However, this is also a setting where the ranking tool is extremely uninformative, and so we shouldn't have expected to have this be a reasonable strategy. Effectively, this shows that if $v_2 > 0$, we can only hope for approximate optimality of strategies like this. □

THEOREM 1. *Algorithm 1 as applied to a setting where j is the first index that differs from 1 has error that is optimal up to an additive factor of ϵ, for $\epsilon \leq \frac{p_2}{p_1} \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot (1 - P[\sigma^1] - P[\sigma^j])$.*

PROOF. We will begin with the setting where $s_{i<j} = 0, s_j = 1$ (the **highest-ranked candidate is busy**). We pick the first candidate whenever:

$$\mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_1} \mid s]/\gamma \geq \mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_j} \mid s]$$

Using similar analysis to the proof of Lemma 1, we can apply Bayes rule to rewrite this as:

$$\sum_{\ell=1}^{N} \frac{P[s \mid \sigma^\ell] \cdot P[\sigma^\ell]}{P[s]} \cdot v_{\sigma_1^\ell}/\gamma_{\sigma_1^\ell} \geq \sum_{\ell=1}^{N} \frac{P[s \mid \sigma^\ell] \cdot P[\sigma^\ell]}{P[s]} \cdot v_{\sigma_j^\ell}$$

$$\sum_{\ell=1}^{N} P[s \mid \sigma^\ell] \cdot P[\sigma^\ell] \cdot v_{\sigma_1^\ell}/\gamma_{\sigma_1^\ell} \geq \sum_{\ell=1}^{N} P[s \mid \sigma^\ell] \cdot P[\sigma^\ell] \cdot v_{\sigma_j^\ell}$$

Because we are in the superstar setting, we know that $v_{\sigma_i^\ell} = v_2$ unless $i = \ell$, so we can rewrite the condition as:

$$P[s \mid \sigma^1]/\gamma_1 \cdot P[\sigma^1] \cdot v_1 + 1/\gamma_2 \sum_{i=2}^{N} P[s \mid \sigma^i] \cdot P[\sigma^i] \cdot v_2 \geq P[s \mid \sigma^j] \cdot P[\sigma^1] \cdot v_1 + \sum_{i \neq j} P[s \mid \sigma^i] \cdot P[\sigma^i] \cdot v_2$$

Again using similar analysis to Lemma 1, we can reason about $P[s \mid \sigma^i]$, for a general i. Specifically, we can show that:

$$\begin{cases} P[s \mid \sigma^i] = P[s \mid \sigma^j] & s_i = s_j = 1 \\ P[s \mid \sigma^i] = r \cdot P[s \mid \sigma^j] & s_i = 0 \neq s_j = 1 \end{cases}$$

where the probability of observing s is *higher* if the high value candidate is in an index i with $s_i = 0$, because the high value candidate has $p_1 < p_2$ and is thus less likely to be free. Using this and the fact that $s_{i<j} = 0$, we can rewrite the conditions as:

$$P[s \mid \sigma^j]\left(r \cdot P[\sigma^1] \cdot v_1/\gamma_1 + r \cdot P[\sigma^{2 \leq i < j}] \cdot v_2/\gamma_2 + P[\sigma^j] \cdot v_2/\gamma_2 + \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \cdot v_2/\gamma_2\right)$$

$$\geq P[s \mid \sigma^j]\left(P[\sigma^j] \cdot v_1 + r \cdot P[\sigma^{1 \leq i < j}] \cdot v_2 + \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \cdot v_2\right)$$

Note the common $P[s \mid \sigma^j]$ term we can drop. Next, we can strategically rearrange and collect common terms.

$$P[\sigma^1] \cdot r \cdot (\gamma_1 \cdot v_1 - v_2) \geq P[\sigma^j] \cdot (v_1 - v_2/\gamma_2) + r \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot P[\sigma^{2 \leq i < j}] + v_2 \cdot (1 - 1/\gamma_2) \cdot \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \quad (1)$$

Next, we will apply the condition in the statement of this theorem, which tells us when we will pick the first candidate. Note that this depends on whether R is less than or greater than 1. If $R < 1$, the firm is using the first-free strategy, and will pick the jth candidate if and only if:

$$\frac{P[\sigma^1]}{P[\sigma^j]} \leq \frac{1}{R} = \frac{v_1 - v_2/\gamma_2}{(v_1/\gamma_1 - v_2) \cdot r}$$

which can be rewritten as:

$$P[\sigma^1] \cdot (v_1/\gamma_1 - v_2) \cdot r \leq P[\sigma^j] \cdot (v_1 - v_2/\gamma_2) \quad (2)$$

Conversely, if $R > 1$, the firm is using the first-busy strategy, and will pick the jth candidate if and only if:

$$\frac{P[\sigma^1]}{P[\sigma^j]} \leq R = \frac{(v_1/\gamma_1 - v_2) \cdot r}{v_1 - v_2/\gamma_2}$$

or rewritten, whenever:

$$P[\sigma^1] \cdot (v_1 - v_2/\gamma_2) \leq P[\sigma^j] \cdot (v_1/\gamma_1 - v_2) \cdot r \quad (3)$$

First, we note that if a firm is using first-free and Equation 2 is satisfied, then Equation 1 *never* satisfied, and so it is always optimal to pick the jth (free) candidate.

Conversely, if Equation 2 is *not* satisfied, then Equation 1 is satisfied with error up to:

$$\frac{P[s \mid \sigma^j]}{P[s]}\left(r \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot P[\sigma^{2 \le i < j}] + v_2 \cdot (1 - 1/\gamma_2) \cdot \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \right)$$

$$\le \frac{p_2}{p_1} \cdot r \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot (1 - P[\sigma^1] - P[\sigma^j])$$

$$= \frac{p_2}{p_1} \cdot \frac{p_2/(1 - p_2)}{p_1/(1 - p_1)} \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot (1 - P[\sigma^1] - P[\sigma^j])$$

$$= \frac{p_2}{p_1} \cdot (1 - p_2)^{N-2} \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot (1 - P[\sigma^1] - P[\sigma^j])$$

$$\le 0.25 \cdot \frac{p_2}{p_1} \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot (1 - P[\sigma^1] - P[\sigma^j])$$

where in the last step we have used that $x \cdot (1 - x) \le 0.25$ and $1 - p_1 < 1$, and in the first step we have used that:

$$\frac{P[s \mid \sigma^j]}{P[s]} = \frac{P[s \mid \sigma^j]}{P[s \mid \sigma^j]} \sum_{i=1}^{N} P[s \mid \sigma^i] \cdot P[\sigma^i] \le \max_{i \in [N]} \frac{P[s \mid \sigma^j]}{P[s \mid \sigma^i]} = \frac{p_2}{p_1}$$

where the last step comes from the fact that σ_i, σ_j differ only in a single index.

Next, we note that if a firm is using first-busy and the top candidate is busy, the firm will always pick it. Specifically, this means that Equation 3 is *not* satisfied, which means:

$$P[\sigma^1] \cdot (v_1 - v_2/\gamma_2) \ge P[\sigma^j] \cdot (v_1/\gamma_1 - v_2) \cdot r \quad \Rightarrow \quad P[\sigma^1] \cdot r \cdot (v_1/\gamma_1 - v_2) \ge P[\sigma^j] \cdot (v_1 - 1/\gamma_2 \cdot v_2)$$

because in the setting where a firm is using first-busy, we know that $R > 1$ or $r \cdot (v_1/\gamma_1 - v_2) \ge v_1 - v_2/\gamma_2$. Thus, Equation 1 is satisfied up to error ϵ^*, by similar reasoning as above.

Next, we move to the setting where $s_{i<j} = 1, s_j = 0$ (the **highest-ranking candidate is free**). We pick the first candidate whenever:

$$\mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_1} \mid s] \ge \mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_j} \mid s]/\gamma$$

By our prior analysis, we know this can be rewritten as:

$$P[s \mid \sigma^1] \cdot P[\sigma^1] \cdot v_1 + \sum_{i=2}^{N} P[s \mid \sigma^i] \cdot P[\sigma^i] \cdot v_2 \ge P[s \mid \sigma^j] \cdot P[\sigma^j] \cdot v_1/\gamma_1 + \sum_{i \ne j} P[s \mid \sigma^i] \cdot P[\sigma^i] \cdot v_2/\gamma_2$$

Next, we again reason about $P[s \mid \sigma^i]$ for an arbitrary i. Note that the s status vector is different than in the previous part of this proof, so we will have a different correspondence:

$$\begin{cases} P[s \mid \sigma^i] = P[s \mid \sigma^1] & s_i = s_1 = 1 \\ P[s \mid \sigma^i] = r \cdot P[s \mid \sigma^1] & s_i = 0 \ne s_1 = 1 \end{cases}$$

This enables us to rewrite the condition as:

$$P[s \mid \sigma^1] \cdot \left(v_1 \cdot P[\sigma^1] + P[\sigma^{2 \le i < j}] \cdot v_2 + r \cdot P[\sigma^j] \cdot v_2 + \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \cdot v_2 \right)$$

$$\ge P[s \mid \sigma^1] \cdot \left(\gamma_1 \cdot r \cdot P[\sigma^j] \cdot v_1 + v_2/\gamma_2 \cdot P[\sigma^{1 \le i < j}] + \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \cdot v_2/\gamma_2 \right)$$

Dropping a common multiplicative term and strategically grouping related terms gives:

$$P[\sigma^1] \cdot (v_1 - 1/\gamma_2 \cdot v_2) + v_2 \cdot (1 - 1/\gamma_2) \cdot P[\sigma^{2 \le i < j}] + (1 - 1/\gamma_2) \cdot v_2 \cdot \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \ge P[\sigma^j] \cdot r \cdot (v_1/\gamma_1 - v_2) \tag{4}$$

We will again apply the conditions within Equations 2, 3 to give us guarantees for which candidates we should pick. First, we note that if a firm is using first-free and the top-ranked candidate is free, the firm will always pick them. In particular, this means that Equation 2 is satisfied, then

$$P[\sigma^1] \cdot (v_1/\gamma_1 - v_2) \cdot r \le P[\sigma^j] \cdot (v_1 - v_2/\gamma_2) \quad \Rightarrow \quad P[\sigma^j] \cdot (v_1/\gamma_1 - v_2) \cdot r \le P[\sigma^1] \cdot (v_1 - v_2/\gamma_2)$$

because $P[\sigma^1] \geq P[\sigma^j]$, which implies that Equation 4 is satisfied and it is always optimal to pick the first (free) candidate.

Next, we will turn to the case where a firm is using first-busy. If Equation 3 is satisfied, then Equation 4 is satisfied up to error:

$$\leq \frac{P[s \mid \sigma^1]}{P[s]} \cdot \left(v_2 \cdot (1 - 1/\gamma_2) \cdot P[\sigma^{2 \leq i < j}] + (1 - 1/\gamma_2) \cdot v_2 \cdot \sum_{i=j+1}^{N} r^{1-s_i} \cdot P[\sigma^i] \right)$$

$$\leq \frac{p_2}{p_1} \cdot \left(v_2 \cdot (1 - 1/\gamma_2) \cdot P[\sigma^{2 \leq i < j}] + (1 - 1/\gamma_2) \cdot v_2 \cdot r \cdot P[\sigma^{i > j}] \right)$$

$$< \frac{p_2}{p_1} \cdot v_2 \cdot (1 - 1/\gamma_2) \cdot (1 - P[\sigma^1] - P[\sigma^j]) = \epsilon^*$$

as desired. Finally, we consider the case where the firm is using first-busy, but Equation 3 is *not* satisfied, or:

$$P[\sigma^1] \cdot (v_1 - v_2/\gamma_2) \geq P[\sigma^j] \cdot (v_1/\gamma_1 - v_2) \cdot r$$

which we note implies that Equation 4 is always satisfied, and thus has 0 error. $\square$

B Proofs for Section 5

Note: all proofs in this section only rely on accuracy properties as stated in Definition 2:

Definition 2. *[Accuracy: superstar] A ranking* $\{P[\sigma^i]\}'$ *is more accurate than* $\{P[\sigma^i]\}$ *if both conditions hold:*

(1) *Ratio of probabilities: The relative probability of the high value candidate being ranked more highly always increases:* $\frac{P[\sigma^i]'}{P[\sigma^j]'} \geq \frac{P[\sigma^i]}{P[\sigma^j]}$ $\forall i <$ *j with the inequality strict for at least one i, j pair.*

(2) *Monotonic increase: if the probability of the high value item being found at index k decreases, then it also decreases for all* $j > k$, *or: if* $P[\sigma^k]' < P[\sigma^k]$, *then* $P[\sigma^j]' < P[\sigma^j]$ *for all* $j > k$. *Note that an immediate corollary of this property is majorization:* $\sum_{i=1}^{j} P[\sigma^i]' \geq \sum_{i=1}^{j} P[\sigma^i]$ *for all j.*

Theorem 2. *Increasing the accuracy always increases welfare of firms using a first-free or first-busy strategy as described in Section 4.*

Proof. First, Lemma 10 shows that it is always possible to take some status vector s generated by a more accurate permutation distribution $\{P[\rho^i]\}$ and convert it into a status vector s' generated by a less accurate permutation distribution $\{P[\sigma^i]\}$ by iteratively swapping pairs of candidates. We will also use Lemma 1, which showed that the top-ranked busy candidate always has higher expected value than any other busy candidate (and similarly the top-ranked free candidate always has higher expected value than any other free candidate).

Consider the candidate picked from perturbed status vector s' using the optimal strategy for the less-accurate distribution $\{P[\sigma^i]\}$. If it is the same as the candidate picked from the original status vector s with the more accurate ranking tool $\{P[\rho^i]\}$, then the utility is exactly the same as with the less accurate ranking.

However, if a different candidate is picked under s', then we know that this candidate corresponds to an element in s lower expected value than the candidate picked under s with the more accurate distribution $\{P[\rho^i]\}$: as a result, the firm responding optimally with less accurate distribution $\{P[\rho^i]\}$ only has lower utility, as desired. $\square$

Lemma 10. *Given any ranking distributions* $\{P[\rho^i]\}$, $\{P[\sigma^i]\}$ *such that* $\{P[\rho^i]\}$ *is more accurate according to Definition 2 and a status vector s generated by* $\{P[\rho^i]\}$, *it is always possible to permute s such that it is equivalent to a sample from the less-accurate distribution* $\{P[\sigma^i]\}$.

Proof. First, by Definition 2, we know that increasing accuracy of a permutation causes majorization, or $\sum_{i=1}^{j} P[\rho^i] \geq \sum_{i=1}^{j} P[\sigma^i]$ for all j. We can view this as saying that the cumulative density function for the more accuracy permutation is always higher: we will use this to guide how we transform the more accurate distribution into the less accurate one.

First, we know that $\sum_{i=1}^{N} P[\rho^i] = \sum_{i=1}^{N} P[\sigma^i] = 1$ (the total probability sums up to 1). Consider the smallest index k such that $\sum_{i=1}^{k} P[\rho^i] > \sum_{i=1}^{k} P[\sigma^i]$: this is the smallest index such that the more accurate distribution has a greater CDF than the less accurate distribution. This implies that the probability of the high value candidate being in index $k + 1$ is larger in the less accurate distribution: $P[\rho^{k+1}] < P[\sigma^{k+1}]$.

Similarly, define ℓ to be the smallest index such that the CDF is strictly higher with the more accurate distribution: $P[\rho^\ell] > P[\sigma^\ell]$.

Then, we can flip indices ℓ and $k+1$ with a fixed probability so as to create a new distribution with $P[\rho^{k+1}]' = P[\sigma^{k+1}]$ and $P[\rho^\ell]' \geq P[\sigma^\ell]$. Specifically, we define α_{k+1} such that:

$$P[\sigma^{k+1}] = P[\rho^{k+1}] \cdot (1 - \alpha_{k+1}) + P[\rho^\ell] \cdot \alpha_{k+1}$$

Note that flipping indices $\ell, k + 1$ also induces a change in $P[\rho^\ell]$: if doing so would cause $P[\rho^\ell] < P[\sigma^\ell]$, we instead set α_{k+1} to be the value such that $P[\rho^\ell]' = P[\sigma^\ell]$:

$$P[\sigma^\ell] = P[\rho^\ell] \cdot (1 - \alpha_{k+1}) + P[\rho^{k+1}] \cdot \alpha_{k+1}$$

If necessary, we then then repeat the process with the new minimum index ℓ' such that $P[\rho^{\ell'}] > P[\sigma^{\ell'}]$ until we arrive at $P[\rho^{k+1}]' = P[\sigma^{k+1}]$.

After this is completed, we move on to the next $j < k$ such that $\sum_{i=1}^{j} P[\rho^i] > \sum_{i=1}^{j} P[\sigma^i]$, swapping permutations until we have $P[\rho^i]' = P[\sigma^i]'$. $\square$

18

THEOREM 3. *Increasing the accuracy of the ranking tool* increases *the probability that the picked candidate is busy with among firms using first-free and* may decrease *the probability that the picked candidate is busy among firms using first-busy.*

PROOF. First, we will consider the case where a firm is using the "first-free" strategy, for some window $k \geq 1$.
If this is the case, that the probability that a firm picks a busy candidate is given by the probability that every candidate in the top k is busy. This occurs with probability:

$$P[\text{top } k \text{ busy} \mid \text{High value in top } k] \cdot P[\text{High value in top } k]$$

$$+P[\text{top } k \text{ are busy} \mid \text{High value NOT in top } k] \cdot P[\text{High value NOT in top } k]$$

$$= P[\sigma^{1 \leq i \leq k}] \cdot (1 - p_1) \cdot (1 - p_2)^{k-1} + (1 - P[\sigma^{1 \leq i \leq k}]) \cdot (1 - p_2)^k$$

$$= (1 - p_2)^{k-1} \cdot \left(P[\sigma^{1 \leq i \leq k}] \cdot (p_2 - p_1) + (1 - p_2) \right)$$

We know that $p_2 > p_1$ and increased accuracy increases $P[\sigma^{1 \leq i \leq k}]$ (by Definition 2). Thus, if the ranking becomes more accurate and the firm's strategy (k) is held fixed, the probability of the firm picking a busy candidate increases.

What happens if accuracy increases and the firm's strategy changes to some top $j < k$? We will establish that the probability of picking a busy candidate is always higher when the firm is using first-free with window size j than window size k for $j < k$ (holding accuracy constant). What we want to show is:

$$(1 - p_2)^{k-1} \cdot \left(P[\sigma^{1 \leq i \leq k}] \cdot (p_2 - p_1) + (1 - p_2) \right) < (1 - p_2)^{j-1} \cdot \left(P[\sigma^{1 \leq i \leq j}] \cdot (p_2 - p_1) + (1 - p_2) \right)$$

It suffices to show this for $j = k - 1$ (because by induction we can show this for all $j < k$).

$$(1 - p_2)^{k-1} \cdot \left(P[\sigma^{1 \leq i \leq k}] \cdot (p_2 - p_1) + (1 - p_2) \right) < (1 - p_2)^{k-2} \cdot \left(P[\sigma^{1 \leq i \leq k-1}] \cdot (p_2 - p_1) + (1 - p_2) \right)$$

$$(1 - p_2) \cdot \left(P[\sigma^{1 \leq i \leq k}] \cdot (p_2 - p_1) + (1 - p_2) \right) < P[\sigma^{1 \leq i \leq k-1}] \cdot (p_2 - p_1) + (1 - p_2)$$

Pulling over terms gives:

$$(p_2 - p_1) \cdot \left((1 - p_2) \cdot P[\sigma^{1 \leq i \leq k}] - P[\sigma^{1 \leq i \leq k-1}] \right) < (1 - p_2) \cdot (1 - 1 + p_2)$$

$$(p_2 - p_1) \cdot \left((1 - p_2) \cdot P[\sigma^{1 \leq i \leq k}] - P[\sigma^{1 \leq i \leq k-1}] \right) < (1 - p_2) \cdot p_2$$

Note that if the term inside the parentheses is negative, then this is automatically satisfied because the righthand side is positive. If not, then we can upper bound the lefthand side by using $p_2 - p_1 < p_2$, which cancels on both sides.

$$(1 - p_2) \cdot P[\sigma^{1 \leq i \leq k}] - P[\sigma^{1 \leq i \leq k-1}] < 1 - p_2$$

Note that $P[\sigma^{1 \leq i \leq k}] \leq 1$ and $P[\sigma^{1 \leq i \leq k-1}] \geq 0$, which suffices to prove the result.

Next, we will consider the case where a firm is using the "k-busy" strategy, for some $k \geq 1$.
We will find it helpful to instead analyze the probability that the firm picks a free candidate (which is exactly the complement of when it picks a busy candidate). A firm using a k-busy strategy picks a free candidate whenever all candidates in the top k are free:

$$P[\sigma^{1 \leq i \leq k}] \cdot p_1 \cdot p_2^{k-1} + (1 - P[\sigma^{1 \leq i \leq k}]) \cdot p_2^k$$

$$= p_2^{k-1} \cdot \left(P[\sigma^{1 \leq i \leq k}] \cdot (p_1 - p_2) + p_2 \right)$$

Note that if the accuracy increases while k stays fixed, this probability decreases because $p_1 < p_2$: so if the accuracy increases, the probability of the firm hiring a busy candidate increases.
What happens if the accuracy increases and the firm's strategy changes? Similarly to the previous analysis, we will show that the probability of picking a free candidate is higher for top $k < k + 1$, so as accuracy increases and the firm decreases k, it will instantaneously increase the probability of picking a free candidate (decreasing the probability of picking a busy candidate).
Specifically, this occurs when:

$$p_2^{k-1} \cdot \left(P[\sigma^{1 \leq i \leq k}] \cdot (p_1 - p_2) + p_2 \right) > p_2^k \cdot \left(P[\sigma^{1 \leq i \leq k+1}] \cdot (p_1 - p_2) + p_2 \right)$$

$$P[\sigma^{1 \leq i \leq k}] \cdot (p_1 - p_2) + p_2 > p_2 \cdot \left(P[\sigma^{1 \leq i \leq k+1}] \cdot (p_1 - p_2) + p_2 \right)$$

$$p_2 \cdot (1 - p_2) > (p_1 - p_2) \cdot (p_2 \cdot P[\sigma^{1 \leq i \leq k+1}] - P[\sigma^{1 \leq i \leq k}])$$

Note that $p_1 - p_2 < 0$, so if the term inside the parentheses is positive, then the entire righthand side is negative, and this quantity automatically holds. If the term inside the parentheses is negative, we can rearrange the righthand side:

$$p_2 \cdot (1 - p_2) > (p_2 - p_1) \cdot (P[\sigma^{1 \leq i \leq k}] - p_2 \cdot P[\sigma^{1 \leq i \leq k+1}])$$

19

We can note that in this setting, we have:

$$P[\sigma^{1 \leq i \leq k}] > p_2 \cdot P[\sigma^{1 \leq i \leq k+1}]$$

even though $P[\sigma^{1 \leq i \leq k}] < \cdot P[\sigma^{1 \leq i \leq k+1}]$. We can use the fact that:

$$P[\sigma^{1 \leq i \leq k}] - p_2 \cdot P[\sigma^{1 \leq i \leq k+1}] \leq P[\sigma^{1 \leq i \leq k+1}] - p_2 \cdot P[\sigma^{1 \leq i \leq k+1}] = P[\sigma^{1 \leq i \leq k+1}] \cdot (1 - p_2)$$

Applying this to the righthand side of the original equation, we have:

$$(p_2 - p_1) \cdot (P[\sigma^{1 \leq i \leq k}] - p_2 \cdot P[\sigma^{1 \leq i \leq k+1}]) \leq (p_2 - p_1) \cdot P[\sigma^{1 \leq i \leq k+1}] \cdot (1 - p_2) \leq p_2 \cdot (1 - p_2)$$

as desired.   □

THEOREM 4. *Increasing the accuracy of the ranking tool could* increase *or* decrease *how frequently the firm picks the top-ranked candidate.*

PROOF. First, we note that if a firm is using "follow the ranking", then it always picks the top candidate, and as accuracy increases it continues to always pick the top candidate.

Next, we consider the case that a firm is using **first-free** with window size k. A firm picks the top either either when a) the first candidate is free, or b) all top k candidates are busy. What is probability that this happens?

$$P[\text{top candidate free}] + P[\text{all top } k \text{ busy}]$$

$$= p_1 \cdot P[\sigma^1] + p_2 \cdot (1 - P[\sigma^1]) + (1 - p_1) \cdot (1 - p_2)^{k-1} \cdot P[\sigma^{1 < i \leq k}] + (1 - p_2)^k \cdot (1 - P[\sigma^{1 < i \leq k}])$$

Collecting like terms gives us:

$$p_2 + (1 - p_2)^k - P[\sigma^1] \cdot (p_2 - p_1) + (1 - p_2)^{k-1} \cdot P[\sigma^{1 \leq i \leq k}] \cdot (1 - p_1 - 1 + p_2)$$

$$= p_2 + (1 - p_2)^k - P[\sigma^1] \cdot (p_2 - p_1) + (1 - p_2)^{k-1} \cdot P[\sigma^{1 \leq i \leq k}] \cdot (p_2 - p_1)$$

$$= p_2 + (1 - p_2)^k + (p_2 - p_1) \cdot ((1 - p_2)^{k-1} \cdot P[\sigma^{1 \leq i \leq k}] - P[\sigma^1])$$

Note that this is decreasing if and only if a more accurate ranking results in *decreasing* the following quantity:

$$(1 - p_2)^{k-1} \cdot P[\sigma^{1 \leq i \leq k}] - P[\sigma^1] = (1 - p_2)^{k-1} \cdot P[\sigma^{2 \leq i \leq k}] - (1 - (1 - p_2)^{k-1}) \cdot P[\sigma^1]$$

Recall that from Definition 2 increased accuracy increases $\sum_{i=1}^{j} P[\sigma^i]$ for all j: thus, the above quantity could *increase* or *decrease* the above quantity. See Figure 3 for an example where this could lead to increased chances of picking the top candidate.

Next, we will show that if the firm looks at the top $j < k$ then it always has a higher chance of picking the top candidate: this should be intuitive because there is a higher chance that all of the top j candidates are busy than all of the top k candidates are busy. This occurs when:

$$p_2 + (1 - p_2)^k + (p_2 - p_1) \cdot ((1 - p_2)^{k-1} \cdot P[\sigma^{1 \leq i \leq k}] - P[\sigma^1])$$

$$< p_2 + (1 - p_2)^{k-1} + (p_2 - p_1) \cdot ((1 - p_2)^{k-2} \cdot P[\sigma^{1 \leq i \leq k-1}] - P[\sigma^1])$$

$$(1 - p_2)^{k-1} \cdot (1 - (1 - p_2)) > (p_2 - p_1) \cdot ((1 - p_2)^{k-1} \cdot P[\sigma^{1 \leq i \leq k}] - (1 - p_2)^{k-2} \cdot P[\sigma^{1 \leq i \leq k-1}])$$

$$(1 - p_2) \cdot p_2 > (p_2 - p_1) \cdot (1 - p_2) \cdot P[\sigma^{1 \leq i \leq k}] - P[\sigma^{1 \leq i \leq k-1}])$$

This is satisfied because $p_2 > p_2 - 1$ and

$$(1 - p_2) \geq (1 - p_2) \cdot P[\sigma^{1 \leq i \leq k}] \geq (1 - p_2) \cdot P[\sigma^{1 \leq i \leq k}] - P[\sigma^{1 \leq i \leq k-1}])$$

Next, we consider the case where a firm is using **first-busy** with parameter j. The firm picks the top candidate either when a) the top candidate is busy or b) all top j candidates are free. What is the probability that this happens?

$$P[\text{top is busy}] + P[\text{all top } j \text{ are free}]$$

$$(1 - p_1) \cdot P[\sigma^1] + (1 - p_2) \cdot (1 - P[\sigma^1]) + p_1 \cdot p_2^{k-1} \cdot P[\sigma^{1 < i \leq k}] + p_2^k \cdot (1 - P[\sigma^{1 \leq i \leq k}])$$

Collecting like terms:

$$1 - p_2 + p_2^k + P[\sigma^1] \cdot (1 - p_1 - 1 + p_2) + p_2^{k-1} P[\sigma^{1 \leq i \leq k}] \cdot (p_1 - p_2)$$

$$= 1 - p_2 + p_2^k + (p_2 - p_1) \cdot (P[\sigma^1] - p_2^{k-1} \cdot P[\sigma^{1 \leq i \leq k}])$$

Note that this is decreasing (holding k constant) if and only if increasing accuracy causes the below quantity to decrease:

$$P[\sigma^1] \cdot (1 - p_2^{k-1}) - p_2^{k-1} \cdot P[\sigma^{2 \leq i \leq k}]$$

See Figure 3 for an example where this could lead to increased chances of picking the top candidate.

Finally, we will show that this quantity increases for $j > k$, or:

$$1 - p_2 + p_2^k + (p_2 - p_1) \cdot (P[\sigma^1] - p_2^{k-1} \cdot P[\sigma^{1 \leq i \leq k}]) < 1 - p_2 + p_2^{k-1} + (p_2 - p_1) \cdot (P[\sigma^1] - p_2^{k-2} \cdot P[\sigma^{1 \leq i \leq k-2}])$$

$$p_2^k + (p_2 - p_1) \cdot (-p_2^{k-1} \cdot P[\sigma^{1 \leq i \leq k}]) < p_2^{k-1} + (p_2 - p_1) \cdot (-p_2^{k-2} \cdot P[\sigma^{1 \leq i \leq k-2}])$$

$$(p_2 - p_1) \cdot (p_2^{k-2} \cdot P[\sigma^{1 \leq i \leq k-2}] - p_2^{k-1} \cdot P[\sigma^{1 \leq i \leq k}]) < p_2^{k-1}(1 - p_2)$$

$$(p_2 - p_1) \cdot (P[\sigma^{1 \leq i \leq k-2}] - p_2 \cdot P[\sigma^{1 \leq i \leq k}]) < p_2 \cdot (1 - p_2)$$

This is satisfied because $p_2 > p_2 - p_1$ and

$$P[\sigma^{1 \leq i \leq k-2}] - p_2 \cdot P[\sigma^{1 \leq i \leq k}] \leq (P[\sigma^{1 \leq i \leq k-1}] - p_2 \cdot P[\sigma^{1 \leq i \leq k}]) = (P[\sigma^{1 \leq i \leq k-1}](1 - p_2) \leq 1 - p_2$$

$\square$

C   Proofs for Section 6

C.1   Inversion-monotone properties (optimality of picking first free or first busy)

Recall that in Section 4 we proved Lemma 1:

Lemma 1. *In the superstar setting, given a realized status vector s, it is always optimal to either pick the first (top-ranked) free item or the first busy item.*

We also noted that the precondition "In the superstar setting" was necessary. Here, we will show why this is: specifically, Lemma 11 shows that in the non-superstar setting, it may be optimal to pick a strategy other than "first-busy, first-free": for example, to pick the second-ranked free item rather than the top-ranked free item.

Our main theorem is given by:

Theorem 7. *In the beyond the superstar setting, there exists probability distributions $\mathcal{P}$ such that for some realized status vector s, the optimal choice is to pick neither the first free nor the first busy candidate (e.g., instead to pick the 2nd busy candidate). However, for commonly-used noise models (e.g. Mallows model and Random Utility Model), it is always optimal to pick the first free or first busy candidate.*

We will prove this theorem through a series of smaller lemmas.

Lemma 11. *There exists a probability distribution $\mathcal{P}$ such that for some realized status vector s, picking the* second *free item maximizes expected utility, even if $\mathcal{P}$ is descending in expected value.*

Proof. We will set the 3 candidates to have values $[v_1 = 1, v_2, v_3 = 0]$. We will create a permutation distribution with nonzero support on exactly two permutations:

$$\sigma^1 = [v_1, v_2, v_3] \quad \sigma^2 = [v_3, v_2, v_1]$$

where $P[\sigma^1] = 1 - \epsilon, P[\sigma^2] = \epsilon$. Before the candidate status vector is taken into account, the expected utility of picking candidate $i \in [1, 2, 3]$ is given by:

$$\mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_1}] = v_1 \cdot (1 - \epsilon) \quad \mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_2}] = v_2 \quad \mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_3}] = \epsilon \cdot v_1$$

If we want the distribution to have descending expected value, we will set:

$$v_1 \cdot (1 - \epsilon) > v_2 > \epsilon \cdot v_1$$

We will next consider the probabilities of being free. For simplicity, we will set $p_1 < p_2 = p_3$, so candidates 2 and 3 have the same probability of being free, which is greater than for candidate 1.

Next, we will consider the case where we have realized status vector $[1, 1, 0]$. This increases our posterior belief that we are in σ^2, which decreases the expected utility of picking the first candidate.

We will formalize this next. The posterior value for the first candidate is given by:

$$P[v_{\sigma_1} = v_1 \mid s] \cdot v_1 = \frac{P[s \mid \sigma^1] \cdot P[\sigma^1]}{P[s]} = v_1 \cdot \frac{p_1 \cdot p_2 \cdot (1 - p_2) \cdot (1 - \epsilon)}{P[s]}$$

The posterior value for the second candidate is always exactly v_2, because both possible permutations involve having v_2 ranked second. However, we will find it useful to strategically rewrite this probability :

$$P[v_{\sigma_2} = v_2 \mid s] \cdot v_2 = v_2 \cdot \frac{P[s]}{P[s]} = v_2 \cdot \frac{P[s \mid \sigma^1] \cdot P[\sigma^1] + P[s \mid \sigma^2] \cdot P[\sigma^2]}{P[s]}$$

$$= v_2 \cdot \frac{p_1 \cdot p_2 \cdot (1 - p_2) \cdot (1 - \epsilon) + p_2^2 \cdot (1 - p_1) \cdot \epsilon}{P[s]}$$

The posterior value for the third candidate is given by:

$$P[v_{\sigma_3} = v_1 \mid s] \cdot v_1 / \gamma = v_1 \cdot \frac{P[s \mid \sigma^2] \cdot P[\sigma^2]}{P[s]} \gamma = v_1 \cdot \frac{p_2^2 \cdot (1 - p_1) \cdot \epsilon}{P[s] \cdot \gamma}$$

Next, we will show that there exists parameters such that the optimal strategy (candidate with highest posterior utility) is the second (free) candidate, rather than the first-free or first-busy strategy. Specifically, we will show that this condition holds for $\epsilon = 0.1, v_1 = 1, v_2 = 2/3, v_3 = 0, p_1 = 0.1, p_2 = 0.4, \gamma \geq 1$. Note that these parameters immediately satisfies $v_2 < (1 - \epsilon) \cdot v_1$, which implies that the prior ranking is descending in expected utility (as desired).

Note that each of these expected utility terms have a common denominator of $P[s]$, which we can drop. We know that picking the second (free) candidate has higher expected utility than picking the first (free) candidate whenever:

$$v_1 \cdot p_1 \cdot p_2 \cdot (1 - p_2) \cdot (1 - \epsilon) < v_2 \cdot (p_1 \cdot p_2 \cdot (1 - p_2) \cdot (1 - \epsilon) + p_2^2 \cdot (1 - p_1) \cdot \epsilon)$$

Dropping a common term of p_2 and distributing v_2:

$$v_1 \cdot p_1 \cdot (1 - p_2) \cdot (1 - \epsilon) < v_2 \cdot p_1 \cdot (1 - p_2) \cdot (1 - \epsilon) + v_2 \cdot p_2 \cdot (1 - v_1) \cdot \epsilon$$

Substituting in for the given values sets:

$$1 \cdot 0.1 \cdot 0.6 \cdot 0.9 < \frac{2}{3} \cdot 0.1 \cdot 0.6 \cdot 0.9 + \frac{2}{3} \cdot 0.4 \cdot 0.9 \cdot 0.1$$

$$0.054 < 0.06$$

as desired.

We will additionally require that picking the third candidate (given s realized) has lower expected utility than picking the second candidate. This means we require:

$$v_2 \cdot (p_1 \cdot p_2 \cdot (1 - p_2) \cdot (1 - \epsilon) + p_2^2 \cdot (1 - p_1) \cdot \epsilon) > \gamma \cdot v_1 \cdot p_2^2 \cdot (1 - p_1) \cdot \epsilon$$

$$v_2 \cdot (p_1 \cdot (1 - p_2) \cdot (1 - \epsilon) + p_2 \cdot (1 - p_1) \cdot \epsilon) > \gamma \cdot v_1 \cdot p_2 \cdot (1 - p_1) \cdot \epsilon$$

Substituting in for given values:

$$v_2 \cdot (p_1 \cdot (1 - p_2) \cdot (1 - \epsilon) + p_2 \cdot (1 - p_1) \cdot \epsilon) > \gamma \cdot v_1 \cdot p_2 \cdot (1 - p_1) \cdot \epsilon$$

$$\frac{2}{3} \cdot 0.1 \cdot 0.6 \cdot 0.9 + \frac{2}{3} \cdot 0.4 \cdot 0.9 \cdot 0.1 > 1 \cdot 0.4 \cdot 0.9 \cdot 0.1/\gamma$$

$$0.06 > 0.036/\gamma$$

which holds for any $\gamma \geq 1$. □

Note that the ranking in Lemma 11 did satisfy descending expected value. However, rankings were somewhat unusual - in particular, there were only two orderings with nonzero probability, even though, given 3 items, there are 6 possible permutations of each of them. A more natural ranking would probably have some nonzero weight over each of these, ideally with greater weight on rankings where more of the items are ordered "correctly" (that is, with $v_{\sigma_i} > v_{\sigma_j}$ for $i < j$).

Definition 3 exactly describes this intuition. Specifically, it defines an "inversion-monotone" probability distribution as one where, for any permutation σ with at least two items inverted (that is, $v_{\sigma_i} < v_{\sigma_j}$ for $i < j$), there exists another permutation $\tilde{\sigma}$ that is identical to σ but with items in indices i, j flipped, and with probability at least as high as σ ($P[\tilde{\sigma}] \geq P[\sigma]$). Lemma 12 proves that, if a probability distribution is inversion-monotone, we regain the same property we had in the superstar setting: the optimal strategy for the firm will always be to pick the first free item or first busy item.

Definition 3. *A probability distribution over permutations $\mathcal{P}$ is called* inversion-monotone *if it satisfies the following condition: for any permutation σ with a pair of indices $i < j$ such that $v_{\sigma_i} > v_{\sigma_j}$, we construct a corresponding permutation $\tilde{\sigma}$ where $\tilde{\sigma} = \sigma$ except for indices i, j, which we have flipped: $v_{\tilde{\sigma}_i} = v_{\sigma_j}, v_{\tilde{\sigma}_j} = v_{\sigma_i}$. Then, we require:*

$$P[\sigma] \geq P[\tilde{\sigma}]$$

Lemma 12. *Consider any vector of realized status vector s, with $s_i = s_j$, for some $i < j$. Then, if the probability distribution of permutations is* inversion-monotone *as in Definition 3, $\mathbb{E}[v_{\sigma_i} \mid s] \geq \mathbb{E}[v_{\sigma_j} \mid s]$, which implies that the optimal solution will always be to pick the first free item or the first busy item.*

While Lemma 12 shows that inversion-monotonicity will guarantee a more straightforward strategy space, it is natural to wonder how reasonable such a requirement is. In fact, it turns out that multiple commonly-used models of permutations already satisfy this property.

First, Lemma 13 shows that the Mallows model is inversion-monotone. While helpful, this is not extremely surprising: the Mallows model [15] is constructed such that the probability of a permutation σ occurring is directly tied to the number of pairs of items that are inverted, so this property follows naturally.

Lemma 13. *The Mallows model is inversion-monotone.*

While the Mallows model is frequently used as a model of permutations, it does have drawbacks. Specifically, one desirable property of rankings is that items with very different true values should be less likely to be inverted. For example, given $[v_1 = 10, v_2 = 9, v_3 = 0]$, we would expect items v_1, v_2 to be more frequently inverted than v_2, v_3, even though their ordinal ranks differ by the same amount. This type of property cannot be expressed by the Mallows model, but it can be captured by the Random Utility Model [21]. As described in Section 2, in the Random Utility Model (RUM), while each item has some true value $\{v_i\}$, it is assumed that they are ranked by noised versions of these values, such as additive noise ($\hat{v}_i \sim \mathcal{D}$). This automatically satisfies the property that items with more similar true values will be more likely to be swapped. Additionally, it seems likely that the Random Utility Model might better capture the performance of rankings produced by

humans or algorithmic tools, which might have some sense of the "true value" of each item, but make small, independent errors in their estimation of these values. However, does the Random Utility Model satisfy inversion-monotonicity?

Theorem 9 answers this question in the affirmative. While we believe that this property may be of independent interest, to our knowledge, we are the first paper to prove such a property for the Random Utility model. The proof of Theorem 9 is largely proven by two sub-lemmas. First, Lemma 14 follows directly from properties of the Gumbel distribution. Next, Lemma 15 generalizes this result to all symmetric noise distribution (e.g., including Gaussian noise, a standard choice in the RUM): the proof of this lemma turns out to be surprisingly subtle.

THEOREM 9. *The RUM with either a) Gumbel noise, or b) any noise from any identical, symmetric, noise distribution is inversion-monotone.*

Lemma 14. *Suppose we have two random variables given by $X_1 = \mu_1 + \epsilon_1, X_2 = \mu_2 + \epsilon_2$, for $\mu_1 > \mu_2$ and $\epsilon_1, \epsilon_2 \sim \mathcal{D}$ from a Gumbel distribution $\mathcal{D}$. Then, if $|X_1 - X_2| = \Delta$,*

$$P[X_1 > X_2 \mid |X_1 - X_2| = \Delta] > P[X_1 < X_2 \mid |X_1 - X_2| = \Delta]$$

Lemma 15. *Suppose we have two random variables given by $X_1 = \mu_1 + \epsilon_1, X_2 = \mu_2 + \epsilon_2$, for $\mu_1 > \mu_2$ and $\epsilon_1, \epsilon_2 \sim \mathcal{D}$ for a symmetric, single-peaked distribution $\mathcal{D}$. Then, if $|X_1 - X_2| = \Delta$,*

$$P[X_1 > X_2 \mid |X_1 - X_2| = \Delta] > P[X_1 < X_2 \mid |X_1 - X_2| = \Delta]$$

These results tell us that, even in settings beyond the superstar model, for cases where permutations are generated by commonly-used models, the optimal strategy will still be to either pick the first free item or the first busy item.

Finally, we present proofs for all above lemmas:

Lemma 12. *Consider any vector of realized status vector s, with $s_i = s_j$, for some $i < j$. Then, if the probability distribution of permutations is inversion-monotone as in Definition 3, $\mathbb{E}[v_{\sigma_i} \mid s] \geq \mathbb{E}[v_{\sigma_j} \mid s]$, which implies that the optimal solution will always be to pick the first free item or the first busy item.*

PROOF. The expected value of the ith entry is given by:

$$\mathbb{E}[v_{\sigma_i} \mid s] = \sum_{\sigma \in \mathcal{P}} P[\sigma \mid s] \cdot v_{\sigma_i} = \sum_{\sigma \in \mathcal{P}} \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \cdot v_{\sigma_i}$$

By identical reasoning, the expected value of the jth entry is given by:

$$\mathbb{E}[v_{\sigma_j} \mid s] = \sum_{\sigma \in \mathcal{P}} P[\sigma \mid s] \cdot v_{\sigma_j} = \sum_{\sigma \in \mathcal{P}} \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \cdot v_{\sigma_j}$$

We wish to show that:

$$\sum_{\sigma \in \mathcal{P}} \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \cdot v_{\sigma_i} \geq \sum_{\sigma \in \mathcal{P}} \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \cdot v_{\sigma_j}$$

Consider some σ such that $v_{\sigma_i} < v_{\sigma_j}$. Then, we consider a unique mapping to some $\tilde{\sigma}$ such that $\sigma_k = \tilde{\sigma}_k$ except for $\tilde{\sigma}_i = \sigma_j, \tilde{\sigma}_j = \sigma_i$ (the ith and jth elements are swapped).

Then, we will show that:

$$P[\sigma \mid s] \cdot v_{\sigma_i} + P[\tilde{\sigma} \mid s] \cdot v_{\tilde{\sigma}_i} > P[\sigma \mid s] \cdot v_{\sigma_j} + P[\tilde{\sigma} \mid s] \cdot v_{\tilde{\sigma}_j}$$

By construction, we have $\tilde{\sigma}_i = \sigma_j, \tilde{\sigma}_j = \sigma_i$, so we can rewrite this as:

$$P[\sigma \mid s] \cdot v_{\sigma_i} + P[\tilde{\sigma} \mid s] \cdot v_{\sigma_j} > P[\sigma \mid s] \cdot v_{\sigma_j} + P[\tilde{\sigma} \mid s] \cdot v_{\sigma_i}$$

$$(P[\tilde{\sigma} \mid s] - P[\sigma \mid s]) \cdot (v_{\sigma_j} - v_{\sigma_i}) > 0$$

Next, we can rewrite using Bayes rule:

$$\left(\frac{P[s \mid \tilde{\sigma}] \cdot P[\tilde{\sigma}]}{P[s]} - \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \right) \cdot (v_{\sigma_j} - v_{\sigma_i}) > 0$$

Dropping the common denominator gives:

$$(P[s \mid \tilde{\sigma}] \cdot P[\tilde{\sigma}] - P[s \mid \sigma] \cdot P[\sigma]) \cdot (v_{\sigma_j} - v_{\sigma_i}) > 0$$

We will argue that $P[s \mid \tilde{\sigma}] = P[s \mid \sigma]$. Recall that $\tilde{\sigma}, \sigma$ are identical except for at entries i, j. Because $s_i = s_j$ by assumption, for the probability of observing s does not change if candidates are swapped between these two entries. Dropping this common term gives:

$$(P[\tilde{\sigma}] - P[\sigma]) \cdot (v_{\sigma_j} - v_{\sigma_i}) > 0$$

The second term is satisfied because $v_{\sigma_j} > v_{\sigma_i}$ by assumption, and the first term holds because $P[\tilde{\sigma}] > P[\sigma]$ by the requirement that the distribution is inversion-monotone as in Definition 3.

23

Finally, we will show that $\mathbb{E}[v_{\sigma_i} \mid s] \geq \mathbb{E}[v_{\sigma_j} \mid s]$ implies that the best strategy is to either pick the top ranked free or top-ranked busy candidate.

Consider any pairs of indices i, j with $i < j$ and $s_i = s_j$ (both candidates free or busy). Then, the condition indicates that:

$$\gamma^{1-s_i} \cdot \mathbb{E}[v_{\sigma_i} \mid s] \geq \gamma^{1-s_j} \cdot \mathbb{E}[v_{\sigma_j} \mid s]$$

Because there is a common term of $\gamma^{1-s_i} = \gamma^{1-s_j}$, this indicates that the firm only gets lower utility by picking the candidate in index j. As a result, the highest-ranked free candidate has the highest expected utility for the firm among candidates that are free, and the highest-ranked busy candidate has the highest expected utility for the firm among candidates that are busy. Thus, the optimal strategy is always to pick either the top ranked free or top ranked busy candidate.   □

Lemma 13. *The Mallows model is inversion-monotone.*

Proof. Consider some permutation σ, with $v_{\sigma_i} > v_{\sigma_j}$ for $i > j$. Recall that the probability of seeing a permutation is inversely proportional to the number of inversions (e.g. $k > l$ such that $v_{\sigma_k} < v_{\sigma_l}$).

If $i = j + 1$ then this is obvious: this only changes the number of inversions for i, j, and increases the number of inversions by exactly 1.

If $i > j + 1$, then there exists some set $S \in [i + 1, j - 1]$. Flipping $v_{\sigma_i}, v_{\sigma_j}$ also changes the number of inversions within this set (but maintains the number of inversions in elements outside of this set). We would like to show that if $v_{\sigma_i} > v_{\sigma_j}$, flipping the elements i, j only increases the number of inversions (and thus makes this alternative permutation less likely).

Within S, we can denote the set of elements that are *greater* and *less* than i and j respectively by:

$$G_i = \{k \in S \mid v_{\sigma_k} > v_{\sigma_i}\} \quad G_j = \{k \in S \mid v_{\sigma_k} > v_{\sigma_j}\}$$

$$L_i = \{k \in S \mid v_{\sigma_k} < v_{\sigma_i}\} \quad L_j = \{k \in S \mid v_{\sigma_k} < v_{\sigma_j}\}$$

Note that $G_i \subseteq G_j$ and $L_j \subseteq$: the set of elements that is greater than i is a subset of the set of elements that is greater than j, and the set of elements that is less than j is a subset of the elements that is smaller than i.

Note that within S in the original permutation π, the number of inversions is given by

$$|G_i| + |L_j|$$

the elements that are greater than v_{σ_i} and less than v_{σ_j}. After swapping the order of elements i, j, the number of inversions within S is given by:

$$|G_j| + |L_i|$$

the elements that are greater than π_j and less than π_i. By our prior reasoning:

$$|G_j| \geq |G_i| \quad |L_i| \geq |L_j|$$

and so:

$$|G_j| + |L_i| \geq |G_i| + |L_j|$$

We note that flipping i, j involves at least one more inversion (since now we have $v_{\sigma_j} < v_{\sigma_i}$) and so the swapping process strictly increased the total number of inversions.   □

Theorem 9. *The RUM with either a) Gumbel noise, or b) any noise from any identical, symmetric, noise distribution is inversion-monotone.*

Proof. Consider any permutation σ with at least one inversion (an instance where $v_{\sigma_i} < v_{\sigma_j}$ but $i < j$). Such a permutation occurs when $\hat{v}_{\sigma_i} > \hat{v}_{\sigma_j}$. By Lemmas 14 and 15, we know that given $v_{\sigma_i} < v_{\sigma_j}$ and pair of noised values $\hat{v}_i, \hat{v}_j$ generated with with noise either a) from a Gumbel distribution or b) any additive symmetric noise distribution, if $|\hat{v}_{\sigma_i} - \hat{v}_{\sigma_j}| = \Delta$, then

$$P[\hat{v}_{\sigma_j} > \hat{v}_{\sigma_i} \mid |\hat{v}_{\sigma_i} - \hat{v}_{\sigma_j}| = \Delta] > P[\hat{v}_{\sigma_j} < \hat{v}_{\sigma_i} \mid |\hat{v}_{\sigma_i} - \hat{v}_{\sigma_j}| = \Delta]$$

This tells us that it is more likely we would have $\hat{v}_i < \hat{v}_j$ (the correct relative ordering of these candidates) than their inversion. Because noise is added to element independently in the RUM, we know that the noised value $\hat{v}_k$ is completely independent of the noised value of $\hat{v}_i, \hat{v}_j$ for $k \neq i, j$. Thus, we know that there must exist a permutation $\tilde{\sigma}$ identical to σ except with the candidates in indices i, j flipped, and we must have $P[\tilde{\sigma}] \geq P[\sigma]$.   □

Lemma 14. *Suppose we have two random variables given by $X_1 = \mu_1 + \epsilon_1, X_2 = \mu_2 + \epsilon_2$, for $\mu_1 > \mu_2$ and $\epsilon_1, \epsilon_2 \sim \mathcal{D}$ from a Gumbel distribution $\mathcal{D}$. Then, if $|X_1 - X_2| = \Delta$,*

$$P[X_1 > X_2 \mid |X_1 - X_2| = \Delta] > P[X_1 < X_2 \mid |X_1 - X_2| = \Delta]$$

Proof. First, note that the mean of a Gumbel distribution is given by $\mu + \beta \cdot \eta$, where η here is Euler's constant. That means that we can model $X_i = \mu_i + \epsilon_i$ as $X_i \sim G(X_i - \beta \cdot \eta)$. Next, we consider the difference distribution induced by $\epsilon_1 - \epsilon_2$: we note that the difference distribution between two Gumbel distributions is given by a Logistic distribution governed by mean $X_1 - \beta \cdot \eta - X_2 + \beta \cdot \eta = X_1 - X_2$ and noise parameter β. Because $X_1 > X_2$, this distribution is right-skewed: thus, for all values x, the probability of $X_1 > X_2$ given $|X_1 - X_2| = x$ is always greater than the probability of $X_1 < X_2$, as desired.   □

Lemma 15. *Suppose we have two random variables given by $X_1 = \mu_1 + \epsilon_1, X_2 = \mu_2 + \epsilon_2$, for $\mu_1 > \mu_2$ and $\epsilon_1, \epsilon_2 \sim \mathcal{D}$ for a symmetric, single-peaked distribution $\mathcal{D}$. Then, if $|X_1 - X_2| = \Delta$,*

$$P[X_1 > X_2 \mid |X_1 - X_2| = \Delta] > P[X_1 < X_2 \mid |X_1 - X_2| = \Delta]$$

PROOF. We will begin by analyzing the distribution $\mathcal{D}'$ induced by $X = X_1 - X_2$.

Symmetric:

First, we will show that it is symmetric around $\mu_1 - \mu_2$. In order to show this, we will show that:

$$P[X = \mu_1 - \mu_2 + \delta] = P[X = \mu_1 - \mu_2 - \delta]$$

for all $\delta > 0$. Suppose that we have some ϵ_1, ϵ_2 such that:

$$X = X_1 - X_2 = \mu_1 + \epsilon_1 - \mu_2 - \epsilon_2 = \mu_1 - \mu_2 + \delta$$

for $\delta = \epsilon_1 - \epsilon_3$. Then, we can show a mapping to an equally-likely event where $X = \mu_1 - \mu_2 - \delta$. Because ϵ_1, ϵ_2 are drawn from the same distribution $\mathcal{D}$, it is equally likely that we would have $\epsilon_1' = \epsilon_2, \epsilon_2' = \epsilon_1$. This would give us:

$$X = X_1 - X_2 = \mu_1 + \epsilon_2 - \mu_2 - \epsilon_1 = \mu_1 - \mu_2 + (\epsilon_2 - \epsilon_1) = \mu_1 - \mu_2 - (\epsilon_1 - \epsilon_2) = \mu_1 - \mu_2 - \delta$$

as desired.

Unimodal:

The next thing we would like to show is that $\mathcal{D}'$ is unimodal (strictly increasing and then decreasing).

We will look at the distribution $g(x)$ of $\mathcal{D}'$ directly. We'd like to show that this is decreasing for $x > \mu_1 - \mu_2$. What's the probability density of $g(\delta)$ for some $\delta = \epsilon_1 - \epsilon_2$? We can obtain this by integrating over all possible values of ϵ_1, ϵ_2, along with the cdf $f(\cdot)$ for the noise distribution:

$$g(\delta) = \int_{-\infty}^{\infty} \mathbb{1}[\delta = \epsilon_1 - \epsilon_2] f(\epsilon_1) \cdot f(\epsilon_2) d\epsilon_1 d\epsilon_2$$

We can rewrite as a single integral over ϵ_1 and use $\epsilon_2 = \epsilon_1 - \delta$.

$$g(\delta) = \int_{-\infty}^{\infty} f(\epsilon_1) \cdot f(\epsilon_1 - \delta) d\epsilon_1$$

We would like to show that this is decreasing in δ for $\delta > 0$. We can take the derivative wrt δ, which gives us:

$$\frac{d}{d\delta} g(\delta) = \frac{d}{d\delta} \int_{-\infty}^{\infty} f(\epsilon_1) \cdot f(\epsilon_1 - \delta) d\epsilon_1$$

Integration and differentiation commutes, so we have:

$$= \int_{-\infty}^{\infty} f(\epsilon_1) \cdot \frac{d}{d\delta} f(\epsilon_1 - \delta) d\epsilon_1$$

by the chain rule:

$$= -\int_{-\infty}^{\infty} f(\epsilon_1) \cdot f'(\epsilon_1 - \delta) d\epsilon_1$$

Which means we want to show that:

$$\int_{-\infty}^{\infty} f(\epsilon_1) \cdot f'(\epsilon_1 - \delta) d\epsilon_1 > 0 \tag{5}$$

Let's consider a closely related term:

$$\int_{-\infty}^{\infty} f(\epsilon_1) \cdot f'(\epsilon_1) d\epsilon_1$$

We know that $f(\cdot)$ is symmetric around 0 and is increasing below 0 and decreasing above 0, which means that this above term equals 0. $f(\epsilon_1 - \delta)$ is shifted to the right: we will show that this suffices to show that Equation 5 is positive.

We can rewrite Equation 5 as:

$$\int_{-\infty}^{\infty} f(\epsilon + \delta) \cdot f'(\epsilon) d\epsilon$$

And what we want to show is

$$\int_{-\infty}^{\infty} f(\epsilon + \delta) \cdot f'(\epsilon) d\epsilon > \int_{-\infty}^{\infty} f(\epsilon) \cdot f'(\epsilon_1) d\epsilon$$

Or rewritten:

$$\int_{-\infty}^{\infty} (f(\epsilon + \delta) - f(\epsilon)) \cdot f'(\epsilon) d\epsilon > 0$$

First, we can divide this into two components based on whether ϵ is positive or negative:

$$\int_{-\infty}^{0} (f(\epsilon + \delta) - f(\epsilon)) \cdot f'(\epsilon) d\epsilon + \int_{0}^{\infty} (f(\epsilon + \delta) - f(\epsilon)) \cdot f'(\epsilon) d\epsilon$$

$$\int_0^\infty (f(-\epsilon + \delta) - f(-\epsilon)) \cdot f'(-\epsilon)d\epsilon + \int_0^\infty (f(\epsilon + \delta) - f(\epsilon)) \cdot f'(\epsilon)d\epsilon$$

Because f is symmetric by our prior reasoning, we have that $f(\epsilon) = f(-\epsilon)$ and $f'(\epsilon) = -f'(-\epsilon)$, which allows us to rewrite:

$$-\int_0^\infty (f(-\epsilon + \delta) - f(\epsilon)) \cdot f'(\epsilon)d\epsilon + \int_0^\infty (f(\epsilon + \delta) - f(\epsilon)) \cdot f'(\epsilon)d\epsilon$$

We can then combine these terms to give:

$$\int_0^\infty f'(\epsilon) \cdot (f(\epsilon + \delta) - f(\epsilon) - f(-\epsilon + \delta) + f(\epsilon))d\epsilon$$

$$= \int_0^\infty f'(\epsilon) \cdot (f(\epsilon + \delta) - f(-\epsilon + \delta))d\epsilon$$

We know that $f'(\epsilon)$ is always negative for positive ϵ (by the assumption that $f(\cdot)$ is unimodal and centered at 0). In order for the total term to be positive, we need to show that the other term is also negative - that is, that

$$f(-\epsilon + \delta) > f(\epsilon + \delta) \quad \forall \epsilon > 0, \delta > 0$$

Again, we use that $f(\cdot)$ is symmetric and unimodal. This means that

$$f(\epsilon + \delta) = f(-\epsilon - \delta) < f(-\epsilon + \delta)$$

as desired.

Overall reasoning: Having shown that $\mathcal{D}'$ is unimodal and symmetric, we will now show that $P[X = \Delta] > P[X = -\Delta]$ for all $\Delta > 0$. We know that $\mathcal{D}'$ has a mean at $\mu_1 - \mu_2 > 0$.

If $0 < \Delta < \mu_1 - \mu_2$, then the event $X = \Delta$ occurs on the lefthand side of the unimodal distribution, as does the event $X = -\Delta$. Because $\mathcal{D}'$ is unimodal and strictly decreasing, this implies that $P[X = -\Delta] < P[X = \Delta]$.

If $0 < \mu_1 - \mu_2 \leq \Delta$, then $P[X = \Delta]$ occurs on the righthand side of the unimodal curve, at a distance $\Delta - (\mu_1 - \mu_2)$ above the peak. Then, reflecting across the axis of symmetry gives the point $P[X = (\mu_1 - \mu_2) - (\Delta - (\mu_1 - \mu_2))] = P[X = 2 \cdot (\mu_1 - \mu_2) - \Delta] = P[X = \Delta]$. This point is on the lefthand side of the unimodal curve, but because $\mu_1 - \mu_2 > 0$, this point also is above the point $P[X = -\Delta]$, and therefore we have:

$$P[X = \Delta] = P[X = 2 \cdot (\mu_1 - \mu_2) - \Delta] > P[X = -\Delta]$$

as desired. □

C.2 Beyond-the-superstar strategy

Definition 4 (Accuracy: beyond superstar). *A ranking distribution $\{P[\sigma]'\}$ is more accurate than $\{P[\sigma]\}$ if both conditions hold:*

(1) *Ratio of probabilities: for every pair of permutations $\sigma, \tilde{\sigma}$ that differ solely by inverting $v_i \geq v_k$ in index $1, j > 2$, the ratio of probabilities of finding $P[\sigma]/P[\tilde{\sigma}]$ a) increases with increased accuracy, and b) increases with increased index j.*

(2) *Majorization: if the probability of a candidate of value v_i being in the top k for any $k \geq 1$ decreases, then the probability of any other candidate with value $v_j < v_i$ being in the top k also decreases.*

Lemma 16. *Plackett-Luce (RUM with Gumbel noise) satisfies the accuracy definitions in Definition 4.*

PROOF. **Ratio of probabilities:** The quantity of interest is:

$$\frac{P[\sigma]}{P[\tilde{\sigma}]}$$

where $\tilde{\sigma}$ is identical to σ, except with elements in $1, j$ flipped. By the definition of the Plackett-Luce model, this is given by:

$$\prod_{j=1}^N \frac{\exp(v_{\sigma_j}/\beta)}{\sum_{\ell=j}^N \exp(v_{\sigma_\ell}/\beta)} \frac{\exp(v_{\sigma_j}/\beta)}{\prod_{j=1}^N \sum_{\ell=j}^N \exp(v_{\sigma_\ell}/\beta)} = \frac{\exp(v_{\sigma_j}/\beta)}{\prod_{j=1}^N \sum_{\ell=j}^N \exp(v_{\sigma_\ell}/\beta)}$$

We can immediately note two points of symmetry:

- The numerator is identical for $\sigma, \tilde{\sigma}$.
- In the denominator, all elements in indices $\ell > j$ the elements within the product are identical.

The ratio of probabilities is given by:

$$\frac{P[\sigma]}{P[\tilde{\sigma}]} = \frac{\prod_{k=1}^j \sum_{\ell=k}^N \exp(v_{\tilde{\sigma}_\ell}/\beta)}{\prod_{k=1}^j \sum_{\ell=k}^N \exp(v_{\sigma_\ell}/\beta)} = \frac{\prod_{k=2}^j \sum_{\ell=k}^N \exp(v_{\tilde{\sigma}_\ell}/\beta)}{\prod_{k=2}^j \sum_{\ell=k}^N \exp(v_{\sigma_\ell}/\beta)}$$

where the equality comes from the fact that the $k = 1$ term is equal on the top and bottom. Noting that $\sigma = \tilde{\sigma}$ except in indices $1, j$, we can rewrite this as:

$$\frac{\prod_{k=2}^{j}\left(\exp(v_{\tilde{\sigma}_j}/\beta) + \sum_{\ell=k,\ell \neq j}^{N} \exp(v_{\tilde{\sigma}_\ell}/\beta)\right)}{\prod_{k=2}^{j}\left(\exp(v_{\sigma_j}/\beta) + \sum_{\ell=k,\ell \neq j}^{N} \exp(v_{\sigma_\ell}/\beta)\right)} = \frac{\prod_{k=2}^{j}\left(\exp(v_{\sigma_1}/\beta) + \sum_{\ell=k,\ell \neq j}^{N} \exp(v_{\sigma_\ell}/\beta)\right)}{\prod_{k=2}^{j}\left(\exp(v_{\sigma_j}/\beta) + \sum_{\ell=k,\ell \neq j}^{N} \exp(v_{\sigma_\ell}/\beta)\right)}$$

Because $v_{\sigma_1} > v_{\sigma_j}$ by assumption, this is satisfied.

Majorization: The property we wish to prove is the following: If the probability of a candidate of value v_i being in the top k for any $k \geq 1$ decreases, then the probability of any other candidate with value $v_j < v_i$ being in the top k also decreases. Instead, we will prove a stronger version: for any set of items S with $|S| = k - 2$ $v_i, v_k \notin S$, if increasing accuracy decreases the probability of v_i being ranked above all elements in $S \cup \{v_k\}$, then it also decreases the probability of v_k being ranked above all elements in $S \cup \{v_i\}$.

What's the probability that v_i is higher than any of these other elements in S?

$$\frac{\exp(v_i/\beta)}{S_\beta + \exp(v_i/\beta) + \exp(v_k/\beta)}$$

Suppose we consider some $\alpha < \beta$ (more accurate distribution). This shrinks the probability of v_i being in the top k if:

$$\frac{\exp(v_i/\beta)}{S_\beta + \exp(v_i/\beta) + \exp(v_k/\beta)} > \frac{\exp(v_i/\alpha)}{S_\alpha + \exp(v_i/\alpha) + \exp(v_k/\alpha)}$$

Cross multiplying gives:

$$\exp(v_i \cdot (1/\beta + 1/\alpha)) + \exp(v_i/\beta) \cdot (S_\alpha + \exp(v_k/\alpha)) > \exp(v_i \cdot (1/\alpha + 1/\beta)) + \exp(v_i/\alpha) \cdot (S_\beta + \exp(v_k/\beta))$$

$$\exp(v_i/\beta) \cdot (S_\alpha + \exp(v_k/\alpha)) > \exp(v_i/\alpha) \cdot (S_\beta + \exp(v_k/\beta))$$

Strategically collecting terms gives:

$$\exp\left(v_i/\beta + v_k/\alpha\right) + \exp(v_i/\beta) \cdot S_\alpha > \exp\left(v_i/\alpha + v_k/\beta\right) + \exp(v_i/\alpha) \cdot S_\beta$$

Because $v_i > v_k$ and $\alpha < \beta$, we know that $\exp\left(v_i/\beta + v_k/\alpha\right) > \exp\left(v_i/\alpha + v_k/\beta\right)$, and so if the above inequality holds, we must have that

$$\exp(v_i/\beta) \cdot S_\alpha > \exp(v_i/\alpha) \cdot S_\beta \quad \Rightarrow \quad S_\alpha/S_\beta > \exp(v_i \cdot (1/\alpha - 1/\beta))$$

When is the same true for v_k? When:

$$\frac{\exp(v_k/\beta)}{S_\beta + \exp(v_i/\beta) + \exp(v_k/\beta)} > \frac{\exp(v_k/\alpha)}{S_\alpha + \exp(v_i/\alpha) + \exp(v_k/\alpha)}$$

Or:

$$\exp\left(v_k/\beta + v_i/\alpha\right) + \exp(v_k/\beta) \cdot S_\alpha > \exp\left(v_k/\alpha + v_i/\beta\right) + \exp(v_k/\alpha) \cdot S_\beta$$

By our previous analysis, we know that $\exp\left(v_i/\beta + v_k/\alpha\right) > \exp\left(v_i/\alpha + v_k/\beta\right)$, which satisfies one pair of inequalities. Because $v_i > v_k$, we know that

$$S_\alpha/S_\beta > \exp(v_i \cdot (1/\alpha - 1/\beta)) \geq \exp(v_k \cdot (1/\alpha - 1/\beta))$$

which satisfies the other pair of terms.   □

THEOREM 5. *For the beyond-the-superstar setting, if the firm is using a first-free or first-busy strategy with a window size that weakly shrinks with increased accuracy, the following properties hold: 1) If the firm is using first-free, then increased accuracy increases the probability that the picked candidate is busy. 2) If the firm is using first free, then increased accuracy* may decrease *the probability that the picked candidate is busy.*

PROOF. First, we will consider the case that a candidate is using first-free. Here, a firm only picks a busy candidate if all in the window size (length k) are busy. The event where this occurs is:

$$\sum_{\sigma \in \mathcal{S}} P[\sigma] \cdot \prod_{i \in \sigma_{1:k}} (1 - p_i)$$

First, we consider the case where the window shrinks from k to $k - 1$: we wish to show that this *increases* the chances that a busy candidate is picked. This event can be written as:

$$\sum_{\sigma \in \mathcal{S}} P[\sigma] \cdot \prod_{i \in \sigma_{1:(k-1)}} (1 - p_i)$$

Because $p_i \in (0, 1)$, we know that the product is always strictly larger:

$$\prod_{i \in \sigma_{1:(k-1)}} (1 - p_i) \geq \prod_{i \in \sigma_{1:k}} (1 - p_i)$$

Next, we will consider the case where the window size is held constant, but accuracy increases (a shift from $\{P[\sigma]\}$ to $\{P[\sigma]'\}$). We wish to show that this increases the probability of the top k candidates being busy (and thus, the probability of a busy candidate being picked). That is, we wish to show that:

$$\sum_{\sigma \in \mathcal{S}} P[\sigma]' \cdot \prod_{i \in \sigma_{1:k}} (1 - p_i) \geq \sum_{\sigma \in \mathcal{S}} P[\sigma] \cdot \prod_{i \in \sigma_{1:k}} (1 - p_i)$$

Note that we can rewrite this as:

$$\prod_{i \in [N] | P[i \in \sigma_{1:k}]' > 0} (1 - p_i) \cdot P[i \in \sigma_{1:k}]' \qquad \prod_{i \in [N] | P[i \in \sigma_{1:k}] > 0} (1 - p_i) \cdot P[i \in \sigma_{1:k}]$$

that is: the probability that the candidate is in the top k, times the probability that it is busy. By Definition 4 we know that if $P[i \in \sigma_{1:k}]' < P[i \in \sigma_{1:k}]$ for some i, then it is also lower for any $j > i$. Because v_i and probability of being free σ_i are inversely related, we know that the quantity on the lefthand side must have lower probability mass on the highest p_i values (lowest $1 - p_i$ values) and higher probability mass on lower p_i values (higher $1 - p_i$ values). This shows that the inequality must hold.

Next, we turn to the setting where the firm is using first-busy as their strategy. Here, a firm selects a busy candidate *unless* all of the top candidates are free. Thus, we will analyze the probability that all of the top candidates are free (and inverting the results for this quantity will tell us results about how often busy candidates are picked).

The event where this occurs is:

$$\sum_{\sigma \in \mathcal{S}} P[\sigma] \cdot \prod_{i \in \sigma_{1:k}} p_i$$

First, we consider the case where the window shrinks from k to $k - 1$: we wish to show that this *increases* the chances that a free candidate is picked (thus, decreasing the probability that a busy candidate is picked). This event can be written as:

$$\sum_{\sigma \in \mathcal{S}} P[\sigma] \cdot \prod_{i \in \sigma_{1:(k-1)}} p_i$$

Because $p_i \in (0, 1)$, we know that the product is always strictly larger:

$$\prod_{i \in \sigma_{1:(k-1)}} p_i \geq \prod_{i \in \sigma_{1:k}} p_i$$

which means that the probability of all candidates in the window being free only *increases*, as desired.

Next, we will consider the case where the window size is held constant, but accuracy increases (a shift from $\{P[\sigma]\}$ to $\{P[\tilde{\sigma}]\}$). We wish to show that this decreases the probability of the top k candidates being free (and thus, increases probability of a busy candidate being picked). That is, we wish to show that:

$$\sum_{\sigma \in \mathcal{S}} P[\sigma]' \cdot \prod_{i \in \sigma_{1:k}} p_i \leq \sum_{\sigma \in \mathcal{S}} P[\sigma] \cdot \prod_{i \in \sigma_{1:k}} p_i$$

Rewriting this gives:

$$\prod_{i \in [N] | P[i \in \sigma_{1:k}]' > 0} p_i \cdot P[i \in \sigma_{1:k}]' \leq \prod_{i \in [N] | P[i \in \sigma_{1:k}] > 0} p_i \cdot P[i \in \sigma_{1:k}]$$

Again, we can prove this by appealing to the accuracy definition in Definition 4: we know that if the probability of candidates i being in the top k decreases, then it must decrease for all candidates $j > i$. Because the total probability of candidates being in the top k must stay constant, this means that the probability of candidates with higher value (lower index) must increase. Because candidates with higher value have lower p_i, this implies that the total probability above must decrease with increased accuracy. □

THEOREM 6. *In the beyond the superstar setting, increasing the accuracy of the ranking tool could* increase *or* decrease *how frequently the firm picks the top-ranked candidate.*

PROOF. Suppose that a firm is using first-busy. Then, the firm picks the top ranked candidate if either a) the top candidate is busy or b) all top k candidates are free. This is given by:

$$\sum_{\sigma \in S} P[\sigma] \cdot (1 - p_{\sigma_1}) + \sum_{\sigma \in S} P[\sigma] \cdot \prod_{i \in \sigma_{1:k}} p_i$$

First, we consider the case where the window size k shrinks. Then this becomes:

$$\sum_{\sigma \in S} P[\sigma] \cdot (1 - p_{\sigma_1}) + \sum_{\sigma \in S} P[\sigma] \cdot \prod_{i \in \sigma_{1:k-1}} p_i$$

which is higher, so the probability of the top candidate being picked only increases. Next, we consider the case where the window stays the same but the accuracy increases. By Definition 4 and our reasoning in the proof of Theorem 5 we know that increasing the accuracy of the ranking tool only increases the total probability of the top k elements being busy. Thurs, this means that the first term increases and the second term increases - whether or not the total term increases or decreases depends on the relative change.

Suppose that a firm is using first-free. Then, the firm picks the top ranked candidate if either a) the top candidate is free or b) all top k candidates are busy. This is given by:

$$\sum_{\sigma \in S} P[\sigma] \cdot p_{\sigma_1} + \sum_{\sigma \in S} P[\sigma] \cdot \prod_{i \in \sigma_{1:k}} (1 - p_i)$$

First, we note that if the window size k shrinks, then the overall term increases and the probability of picking the top candidate increases. Next, we consider the case where the window size stays constant but accuracy increases. Again by Definition 4 and our reasoning in the proof of Theorem 5, the first term increases while the second term decreases: which dominates depends on the relative change in these two. □

Lemma 7. *In the superstar setting, Algorithm 2 becomes Algorithm 1.*

Note that if we are in the superstar setting, there exists only a single pair of unique candidate types to compare, $v_1 > v_2$, so all of the "votes" will go in the same direction. First, we will consider the setting where the first candidate is busy.

The quantity G_{12} can be calculated as:

$$G_{12} = r \cdot P[\sigma] \cdot (v_2 - v_1/\gamma_1) + P[\tilde{\sigma}] \cdot (v_1 - v_2/\gamma_2)$$

This is greater than 0 exactly when:

$$r \cdot P[\sigma] \cdot (v_2 - v_1/\gamma_1) + P[\tilde{\sigma}] \cdot (v_1 - v_2/\gamma_2) > 0$$

$$P[\tilde{\sigma}] \cdot (v_1 - v_2/\gamma_2) > r \cdot P[\sigma] \cdot (v_1/\gamma_1 - v_2)$$

By assumption, $v_1/\gamma_1 - v_2 > 0$ or else it would never be preferable to pick a busy candidate, which means this can be rewritten as:

$$\frac{v_1 - v_2/\gamma_2}{r \cdot (v_1/\gamma_1 - v_2)} > \frac{P[\sigma]}{P[\tilde{\sigma}]}$$

Note that the LHS is equal to $1/R$ as calculated in Algorithm 1. For the *RHS*, σ is defined as a permutation with the high value candidate ranked 1st, and $\tilde{\sigma}$ is a permutation with it ranked in index j. Note that $P[\sigma] \neq P[\sigma^1]$, because the latter is the *total probability* that the high value candidate is ranked 1st, while $P[\sigma]$ is the probability of a single permutation occurring where the high value candidate is ranked first. However, $P[\sigma^1] = N \cdot P[\sigma]$ and $P[\sigma^j] = N \cdot P[\tilde{\sigma}]$, and so $P[\sigma]/P[\tilde{\sigma}] = P[\sigma^1]/P[\sigma^j]$, as desired. Similar analysis holds for the setting where the first candidate is free.

Lemma 8. *Given a constant γ, Algorithm 2 is a k-free/k-busy strategy: it can be reduced to picking a window size and preferentially picking the top-ranked free or busy candidate within that window. The window size shrinks with increased accuracy, and if the firm is preferentially picking busy candidates, the window shrinks with increased γ (increasing if the firm is preferentially picking free candidates).*

PROOF. We can rewrite these conditions as four properties:

(1) Algorithm 2 either preferentially picks free or busy candidates (e.g. if it ever picks a free candidate in index $j > 1$, then it *never* picks a busy candidate in any $k > 1$, and vice versa).
(2) If Algorithm 2 would pick a candidate in index j, then it would also pick a candidate in index $k < j$ (assuming corresponding free/busy status).
(3) Increased accuracy decreases window size.
(4) If a firm is preferentially picking busy candidates, increased γ shrinks the window size (and similarly increases window size if the firm is preferentially picking free candidates).

Note that properties 1 and 2 together tell us that this strategy reduces down to fixing a window size and preferentially picking either free or busy candidates within that window size.

We will prove these properties in sequence. First, we note that 16 gives us a closed-form solution for the relative probability of finding a permutation with two items flipped:

$$\frac{P[\sigma]}{P[\tilde{\sigma}]} = \frac{\prod_{k=2}^{j} \left(\exp(v_{\sigma_1}/\beta) + \sum_{\ell=k,\ell \neq j}^{N} \exp(v_{\sigma_\ell}/\beta) \right)}{\prod_{k=2}^{j} \left(\exp(v_{\sigma_j}/\beta) + \sum_{\ell=k,\ell \neq j}^{N} \exp(v_{\sigma_\ell}/\beta) \right)}$$

We can note that, given fixed $v_{\sigma_1}, v_{\sigma_j}$, this is fixed when all other terms within the sum are minimized. In particular, this means that we must have σ be the ordering with every item besides i, k correctly ordered.

Property 1: Algorithm 2 either preferentially picks free or busy candidates: We will show that if the strategy preferentially picks a candidate who is busy (resp. free) in index $j > 1$, then it will *never* pick a candidate who is free (resp. busy) in any index $k > 1$. WLOG, we will consider a strategy that preferentially picks a free candidate in index j. If this occurs, then $|G_j| > |G_1|$. For a particular pair of candidates i, k, we have G_{ik} given by:

$$G_{ik} = r_{i,k} \cdot P[\sigma] \cdot (v_k - v_i/\gamma) + P[\tilde{\sigma}] \cdot (v_i - v_k/\gamma)$$

Next, we will consider the hypothetical $\tilde{G}_{i,k}$ if a) the first candidate were *free* and b) the choice is picking the candidate ranked 1st and between any possible index ℓ. The relevant quantity for candidates pairs i, k:

$$\tilde{G}_{ik} = r_{i,k} \cdot P[\sigma'] \cdot (v_k/\gamma - v_i) + P[\tilde{\sigma}'] \cdot (v_i/\gamma - v_k)$$

We will show that if $G_{ik} > 0$, we must have $\tilde{G}_{ik} < 0$. By assumption, $P[\sigma] > P[\tilde{\sigma}]$ and $P[\sigma'] > P[\tilde{\sigma}']$ and $v_i > v_k$. Note that $\tilde{G}_{ik}$ has two components, $C_1 = v_k - v_i/\gamma$, and $C_2 = v_i - v_k/\gamma$, and that they are inverted in $\tilde{G}_{ik}$. Analyzing the terms case by case:

- If $C_1, C_2 > 0$, then $\tilde{G}_{ik}$ can be written as $r_{ik} \cdot P[\sigma'] \cdot (-C_1) + P[\tilde{\sigma}'] \cdot (-C_2)$, and thus is negative.
- We cannot have $C_1, C_2 < 0$ because we know $G_{ik} > 0$.
- If $C_1 < 0, C_2 > 0$, then we know because $G_{ik} > 0$ we have $r_{ik} \cdot C_1 < C_2$, which implies that $-C_1 > -C_2$, and thus $-C_1 \cdot r_{ik} > C_2$, which implies that $\tilde{G}_{ik} < 0$
- Note that we cannot have $C_1 > 0, C_2 < 0$, as we must have $C_1 \leq C_2$ because:

$$C_1 \leq C_2$$
$$v_k - v_i/\gamma \leq v_i - v_k/\gamma$$
$$v_k \cdot (1 + 1/\gamma) \leq v_i \cdot (1 + 1/\gamma)$$

which is satisfied by assumption

Property 2: if Algorithm 2 would pick a candidate in index j, then it would also pick a candidate in index $k < j$ In order to prove this, we will show that if Algorithm 2 would pick a candidate in index j, then it would also pick a candidate in index $k < j$ in the same setting (e.g. assuming the first candidate and jth/kth candidate have corresponding statuses).

Note that in Algorithm 2, the only portions that rely on the index of the candidate is the $P[\sigma], P[\tilde{\sigma}]$ calculation. From our analysis at the beginning of this proof, we note that:

$$\frac{P[\sigma_{i,k,j}]}{P[\tilde{\sigma}_{i,k,j}]} = \frac{P[\sigma_{i,k,j-1}]}{P[\tilde{\sigma}_{i,k,j-1}]} \cdot \frac{\exp(v_{\sigma_1}/\beta) + \sum_{\ell>j}^N \exp(v_{\sigma_\ell}/\beta)}{\exp(v_{\sigma_j}/\beta) + \sum_{\ell>j}^N \exp(v_{\sigma_\ell}/\beta)}$$

Because $v_{\sigma_1} > v_{\sigma_j}$ by assumption, the ratio on the RHS is greater than 1. Note that a larger value of $P[\sigma]$ as compared to $P[\tilde{\sigma}]$ only makes it more challenging to pick the candidate ranked jth (lower threshold to pick the candidate ranked 1st).

Property 3: Increased accuracy shrinks window size: Recall that the only component of Algorithm 2 that relies on window size is the ratio $P[\sigma]/P[\tilde{\sigma}]$. Increased accuracy (decreased β) increases this ratio because it increases $\exp(v_{\sigma_1}/\beta)$ more quickly than $\exp(v_{\sigma_j}/\beta)$.

Property 4: Increasing γ shrinks the window size if a firm is preferentially picking busy candidates (and increases it otherwise): Note that the only place that Algorithm 2 relies on γ is in calculating G_{ik}. A firm is preferentially picking busy candidates if it ever picks a busy candidate ranked in index j. Increasing γ decreases $v_k/\gamma, v_i/\gamma$ and thus decreases G_{ik} for when the first candidate is free, either keeping constant or shrinking the window size. Similarly, if the first candidate is busy, increasing γ shrinks $v_i/\gamma, v_k/\gamma$ and increases G_{ik}, either keeping constant or increasing the window size. □

Lemma 9. *For a particular index $1, j$, denote $\mathcal{G}_+$ as $\{v_i, v_j \mid G_{i,k} > 0\}$, $\mathcal{G}_-$ as $\{v_i, v_j \mid G_{i,k} < 0\}$, $\mathcal{I}_{1j}(i,k)$ as the event that items $\{i, k\}$ are in indices $\{1, j\}$, and $\tilde{G}_{i,k}$ as G_{ik} calculated in Algorithm 2, but with $\sigma'_{i,k,j}, \tilde{\sigma}'_{i,k,j}$ instead defined as permutations where every item is incorrectly ordered. Then, if Algorithm 2 picks a candidate in index j, the error is upper bounded by:*

$$\sum_{i,k \in \mathcal{G}_-} P[\mathcal{I}_{1j}(i,k)] \cdot (v_i - v_k)$$

and if Algorithm 2 picks a candidate in index 1, the error is upper bounded by:

$$\sum_{i,k \in \mathcal{G}_+} P[\mathcal{I}_{1j}(i,k)] \cdot (v_i - v_k) + \sum_{i,k \in \mathcal{G}_-} P[\mathcal{I}_{1j}(i,k)] \cdot \max(-\tilde{G}_{i,k}, 0)$$

PROOF. **Exact conditions:** We will find it helpful to begin by exploring the exact conditions for the optimal decision. First, when the **first candidate is free** it is optimal to pick the jth (busy) candidate exactly whenever:

$$\mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_1} \mid s] \leq \mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_j} \mid s]/\gamma$$

Applying Bayes rule tells us that we can rewrite this as:

$$\sum_{\sigma \in \mathcal{P}} \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \cdot v_{\sigma_1} \leq \sum_{\sigma \in \mathcal{P}} \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \cdot v_{\sigma_j}/\gamma$$

Pulling over:

$$0 \leq \frac{1}{P[s]} \sum_{\sigma \in \mathcal{P}} P[s \mid \sigma] \cdot P[\sigma] \cdot (v_{\sigma_j}/\gamma - v_{\sigma_1})$$

Next, we rewrite strategically by grouping together permutations σ that have elements in index $1, j$ ordered correctly, and corresponding permutations $\tilde{\sigma}$ that are identical to σ, but with values $1, j$ exactly inverted:

$$0 \leq \frac{1}{P[s]} \sum_{\sigma \in \mathcal{P}, v_{\sigma_1} > v_{\sigma_j}} P[s \mid \sigma] \cdot P[\sigma] \cdot (v_{\sigma_j}/\gamma - v_{\sigma_1}) + P[s \mid \tilde{\sigma}] \cdot P[\tilde{\sigma}] \cdot (v_{\tilde{\sigma}_j}/\gamma - v_{\tilde{\sigma}_1})$$

which we can rewrite as:

$$0 \leq \frac{1}{P[s]} \sum_{\sigma \in \mathcal{P}, v_{\sigma_1} > v_{\sigma_j}} P[s \mid \sigma] \cdot \left(r_{\sigma_1, \sigma_j} \cdot P[\sigma] \cdot (v_{\sigma_j}/\gamma_{\sigma_j} - v_{\sigma_1}) + P[\tilde{\sigma}] \cdot (v_{\sigma_1}/\gamma_{\sigma_1} - v_{\sigma_j}) \right) \tag{6}$$

Collecting pairs i, k gives us:

$$0 \leq \frac{1}{P[s]} \sum_{i,k \mid v_i > v_k} \left(\left(\sum_{\sigma \mid \sigma_1 = i, \sigma_j = k} P[s \mid \sigma] \cdot P[\sigma] \right) \cdot \left(r_{\sigma_1, \sigma_j} \cdot (v_{\sigma_j}/\gamma_{\sigma_j} - v_{\sigma_1}) \right) + \left(\sum_{\tilde{\sigma} \mid \tilde{\sigma}_1 = k, \tilde{\sigma}_j = i} P[s \mid \sigma] \cdot P[\tilde{\sigma}] \right) \cdot (v_{\sigma_1}/\gamma_{\sigma_1} - v_{\sigma_j}) \right) \tag{7}$$

For any particular pair of elements $v_i > v_k$, the relevant gap in Algorithm 2 is given by:

$$G_{ik} = r_{i,k} \cdot P[\sigma'] \cdot (v_k/\gamma_k - v_i) + P[\tilde{\sigma}'] \cdot (v_i/\gamma_i - v_k)$$

where $\sigma', \tilde{\sigma}'$ are defined specifically where σ is the permutation with every item ordered correctly, except for with item i ranked 1st and item k ranked in index j, and $\tilde{\sigma}$ is identical to σ except with items i, k swapped. By prior analysis in Lemma 8, we know that $P[\sigma]/P[\tilde{\sigma}]$ is maximized with $\sigma', \tilde{\sigma}'$ as defined. Define $\mathcal{P}_{i,k,j}$ as the set of all pairs of permutations $\sigma, \tilde{\sigma}$ such that $\sigma = \tilde{\sigma}$ except with items i, k in indices $1, j$ swapped.

Thus, if $G_{ik} > 0$, then we know that the corresponding sum for every $\sigma, \tilde{\sigma}$ in $\mathcal{P}_{i,k,j}$ is also positive. Thus, we can analyze the error as follows:

If Algorithm 2 picks the candidate in index j, then there is 0 error from pairs i, k with $G_{ik} > 0$, because by our prior reasoning the term within the sum is positive for all permutations in $\mathcal{P}_{i,k,j}$. However, there is error from pairs i, k with $G_{ik} < 0$ (pairs that "voted" to pick the candidate ranked 1st). Using the $\sigma', \tilde{\sigma}'$ as defined in Algorithm 2 are the pair of permutations that makes it *most* easy to "vote" for picking the candidate ranked 1st, and therefore we can upper bound the error associated with picking the candidate ranked jth from this pair i, k as:

$$\frac{P[s \mid \sigma]}{P[s]} \sum_{\sigma, \tilde{\sigma} \in \mathcal{P}_{i,k,j}} - \left(r_{\sigma_1, \sigma_j} \cdot P[\sigma] \cdot (v_{\sigma_j}/\gamma_{\sigma_j} - v_{\sigma_1}) + P[\tilde{\sigma}] \cdot (v_{\sigma_1}/\gamma_{\sigma_1} - v_{\sigma_j}) \right) \leq \max \sigma, \rho \frac{P[s \mid \sigma]}{P[s \mid \rho]} \cdot P[\mathcal{P}_{i,k,j}] \cdot (v_i - v_k/\gamma)$$

For conciseness, in the analysis below we will omit the $\max \sigma, \rho \frac{P[s|\sigma]}{P[s|\rho]}$ term, which is a coefficient on all bounds below.

Next, if Algorithm 2 picks the candidate ranked 1st, then for pairs with $G_{ik} > 0$ the error is upper bounded by:

$$P[\mathcal{P}_{i,k,j}] \cdot (v_i/\gamma - v_k)$$

For pairs with $G_{ik} < 0$ the error comes from the fact that G_{ik} is biased towards picking the top element by its use of $\sigma', \tilde{\sigma}'$. The error is upper bounded by the gap between this and the most conservative permutations, $\sigma'', \tilde{\sigma}''$ where σ'' is created by having every item inverted. Then, the error is bounded by:

$$P[\mathcal{P}_{i,k,j}] \cdot \max(-G'_{ik}, 0)$$

By similar analysis, when the **first candidate is busy** it is optimal to pick the jth (free) candidate exactly whenever:

$$\mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_1} \mid s]/\gamma \leq \mathbb{E}_{\sigma \sim \mathcal{P}}[v_{\sigma_j} \mid s]$$

Applying Bayes rule tells us that we can rewrite this as:

$$\sum_{\sigma \in \mathcal{P}} \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \cdot v_{\sigma_1}/\gamma \leq \sum_{\sigma \in \mathcal{P}} \frac{P[s \mid \sigma] \cdot P[\sigma]}{P[s]} \cdot v_{\sigma_j}$$

Pulling over:

$$0 \leq \frac{1}{P[s]} \sum_{\sigma \in \mathcal{P}} P[s \mid \sigma] \cdot P[\sigma] \cdot (v_{\sigma_j} - v_{\sigma_1}/\gamma)$$

Next, we rewrite strategically by grouping together permutations σ that have elements in index $1, j$ ordered correctly, and corresponding permutations $\tilde{\sigma}$ that are identical to σ, but with values $1, j$ exactly inverted:

$$0 \leq \frac{1}{P[s]} \sum_{\sigma \in \mathcal{P}, v_{\sigma_1} > v_{\sigma_j}} P[s \mid \sigma] \cdot P[\sigma] \cdot (v_{\sigma_j} - v_{\sigma_1}/\gamma) + P[s \mid \tilde{\sigma}] \cdot P[\tilde{\sigma}] \cdot (v_{\tilde{\sigma}_j} - v_{\tilde{\sigma}_1}/\gamma)$$

which we can rewrite as:

$$0 \leq \frac{1}{P[s]} \sum_{\sigma \in \mathcal{P}, v_{\sigma_1} > v_{\sigma_j}} P[s \mid \sigma] \cdot \left(P[\sigma] \cdot (v_{\sigma_j} - v_{\sigma_1}/\gamma_{\sigma_1}) + P[\tilde{\sigma}] \cdot r_{\sigma_1, \sigma_j} \cdot (v_{\sigma_1} - v_{\sigma_j}/\gamma_{\sigma_j}) \right) \tag{8}$$

For any particular pair of elements $v_i > v_k$, the relevant gap in Algorithm 2 is given by:

$$G_{ik} = P[\sigma] \cdot (v_k - v_i/\gamma_i) + P[\tilde{\sigma}] \cdot r_{i,k} \cdot (v_i - v_k/\gamma_k)$$

By using similar analysis to above, we know that when Algorithm 2 picks the candidate in index j, for pairs in $G_{ik} > 0$ there is 0 error, and for pairs with $G_{ik} < 0$ there is error upper bounded by:

$$\max \sigma, \rho \frac{P[s \mid \sigma]}{P[s \mid \rho]} \cdot P[\mathcal{P}_{i,k,j}] \cdot (v_i/\gamma - v_k)$$

Similarly, when Algorithm 2 picks the candidate ranked 1st, for pairs in $G_{ik} > 0$ there is regret upper bounded by:

$$\max \sigma, \rho \frac{P[s \mid \sigma]}{P[s \mid \rho]} \cdot P[\mathcal{P}_{i,k,j}] \cdot (v_i - v_k/\gamma)$$

and for pairs in $G_{ik} < 0$ there is regret upper bounded by:

$$P[\mathcal{P}_{i,k,j}] \cdot \max(-G'_{ik}, 0)$$

$\square$