Fair Ranking with Noisy Protected Attributes

Anonymous Author(s)
Affiliation
Address
email

Abstract

The fair-ranking problem, which asks to rank a given set of items to maximize utility subject to group fairness constraints, has received attention in the fairness, information retrieval, and machine learning literature. Recent works, however, observe that errors in socially-salient (including protected) attributes of items can significantly undermine fairness guarantees of existing fair-ranking algorithms and raise the problem of mitigating the effect of such errors. We study the fair-ranking problem under a model where socially-salient attributes of items are randomly and independently perturbed. We present a fair-ranking framework that incorporates group fairness requirements along with probabilistic information about perturbations in socially-salient attributes. We provide provable guarantees on the fairness and utility attainable by our framework and show that it is information-theoretically impossible to significantly beat these guarantees. Our framework works for multiple non-disjoint attributes and a general class of fairness constraints that includes proportional and equal representation. Empirically, we observe that, compared to baselines, our algorithm outputs rankings with higher fairness, and has a similar or better fairness-utility trade-off compared to baselines.

1 Introduction

Given a query and a set of m items, ranking problems require one to output an ordering of a small subset of items in decreasing order of *relevance* to the query. Such ranking problems have been extensively studied in the information retrieval [40] and the machine learning [39] literature, and algorithms for them are used in applications such as search engines, personalized feed generators, and online recruiting platforms [38, 11, 7] Several studies have observed that when the outputs of ranking algorithms are consumed by end-users, e.g., image results for occupation-related queries, articles with different political leanings, and job applicants in online recruiting, the outputs can mislead or alter their perceptions about socially-salient groups [34], polarize their opinions [21, 43], and affect economic opportunities available to individuals [28]. A reason is that relevance (or utilities) input to ranking algorithms may be influenced by human or societal biases, leading to output rankings that skew representations of socially-salient, and often legally-protected, groups such as women and Black people [48].

A growing number of works aim to make the output of ranking algorithms *fair* with respect to socially-salient attributes [66, 51]. As for notions of fairness, in the case when each item belongs to one of two socially-salient groups (G_1 or G_2), equal representation requires that, for every k, (roughly) $\frac{k}{2}$ items from each of G_1 and G_2 appear in the first k positions of the output ranking. Proportional representation requires that at most $k \cdot \frac{|G_\ell|}{m}$ items from each G_ℓ appear in the first k positions. Fairness criteria that generalize proportional representation and involve $p \geq 2$ groups $G_1, \ldots, G_p$, where each item may belong to multiple groups, have also been considered: Given values $U_{k\ell}$, they require that at most $U_{k\ell}$ items from G_ℓ appear in the first k positions of the output ranking [54, 15]. One set of works in the fair-ranking literature tries to improve fairness in utility-estimation [64, 55, 65, 44]. Such approaches have the benefit that no changes to the existing ranking algorithm are necessary but they may be unable to guarantee that the output ranking satisfies the required fairness criteria

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

[24]. Another set of works use the given utilities as-it-is and change the ranking algorithm to output the ranking with the highest utility subject to satisfying the specified fairness criteria by including them as *fairness constraints* [54, 8, 15, 24, 27]. While these latter approaches can guarantee fairness, they require coming up with new algorithms to solve the arising constrained ranking problems. Both approaches, however, rely on knowledge of the socially-salient attributes of the items [49].

Assuming precise access to socially-salient attributes is reasonable in some contexts and has led to successful deployment of fair-ranking frameworks; see [24]. However, in several contexts, socially-salient attributes can be erroneous, missing, or known only probabilistically. For instance, errors can arise due to misreporting, which is a common concern with self-reported attributes [3]. Attributes can also be missing, as is the case with images in web-search or in settings where it is illegal to collect certain socially-salient attributes [17]. Often attributes are predicted using ML-classifiers, but such prediction has inaccuracies [9]. In such cases, one can calibrate the confidence scores of classifiers to derive (aggregate) probabilistic information about the true attributes [31]. Moreover, probabilistic information about socially-salient (protected) attributes can be sometimes computed from other attributes. For instance, name and location of an individual, combined with aggregate census data may be used to get a conditional distribution of their race [20, 32, 17]. Even accurate attributes may be randomly and independently flipped to preserve user privacy, and the distribution of flipped attributes is determined by public parameters of, e.g., the randomized response mechanism [33, 61].

Several models of inaccuracies in data have been proposed [41, 23]. We consider one such model (due to [4]) to capture inaccuracies in socially-salient attributes. Each item i belongs to the ℓ-th group with a known probability $P_{i\ell}$. For each item i, the distribution corresponding to $P_{i\ell}$s over groups is assumed to be independent of corresponding distributions of other items. This model can be used in cases where these probabilities are available or can be derived, as in some of the aforementioned examples (see Section 4 and Supplementary Material A). In other cases, e.g., when errors are strategic or adversarial, other models are needed. This model and its variants have also been used by works on designing fair algorithms in the presence of inaccuracies, for problems including classification [36, 59, 58, 13], subset selection [42], and clustering [22].

In this noise model, while socially-salient attributes are not explicitly specified, one could still use existing fair-ranking algorithms by first sampling groups for items from the given probabilities. Indeed, [26] evaluate existing fair-ranking algorithms on attributes obtained from the probabilities derived from ML classifiers. They find that "errors in [socially-salient attributes] can dramatically undermine fair-ranking algorithms" and can cause "[non-disadvantaged groups] to become disadvantaged after a 'fair' re-ranking." We confirm this observation on a synthetic dataset when the goal is to finding a ranking that satisfies equal representation (Section 4). We assigned each item the socially-salient group that is most likely and find that when existing fair-ranking algorithms (for equal representation) are run with this group information, they output rankings that significantly violate the equal representation criteria (Figure 1). Further, we mathematically analyze two natural methods to sample groups from probabilities and give examples where taking such information as input, existing fair-ranking algorithms output rankings which provably violate the equal representation criteria (Supplementary Material C). Thus, new ideas are needed to design fair-ranking frameworks that can guarantee given fairness criteria under this noise model.

Our contributions. We present a fair-ranking framework that guarantees given fairness criteria when the socially-salient attributes are assumed to follow the probabilistic noise model mentioned above. In particular, it finds a utility maximizing ranking subject to a class of constraints that only rely on given probability distributions (Program (7)). These constraints relax the given fairness criteria by a carefully chosen factor: for equal representation, the relaxation is by roughly a $1 + \frac{1}{\sqrt{k}}$ multiplicative factor for position k for any k. Moreover, instead of sampling the attribute values and applying constraints on them, these constraints apply the relaxed-fairness criteria to the expected number of items from each group that appear in the first k positions. We show that these constraints ensure that any ranking approximately satisfying the given fairness criteria is feasible for them and any ranking feasible for them approximately satisfies the given fairness criteria (Theorem 3.1). Our fair-ranking framework works for the general class of fairness criteria introduced earlier, which involve multiple overlapping groups $G_1, \ldots, G_p$ and upper bound $U_{k\ell}$ for the ℓ-th group and k-th position (Theorem 3.1), and for their position-weighted versions (Theorem E.1).

We show that our fair-ranking framework, besides nearly satisfying the given fairness criteria, has a provably high utility (Theorem 3.1). Complementing Theorem 3.1, we prove near-tightness of the fairness guarantee (Theorem 3.2): For equal representation fairness criteria, this results shows that that it is information theoretically impossible to output a ranking that violates this criteria by less than a multiplicative factor of $1+\widetilde{O}(\frac{1}{\sqrt{k}})$ at the k-th position for any k. Finally, we give a polynomial-time algorithm to approximately solve Program (7) (Theorem 3.3).

Empirically, we evaluate our framework on both synthetic and real-world data against standard metrics like weighted-risk difference (RD) that measure deviations from specific fairness criteria (Section 4). We compare its performance to key baselines [15, 54, 24, 42] on both single and multiple attributes. In all simulations, we observe that compared to baselines our framework has a higher maximum fairness (2 to 10% for RD; Figures 1 to 3) and a similar or better fairness-utility trade-off (Figures 2, 4 and 6 to 9).

Related work. Work on automated information retrieval dates back to 1940s [37, 18]. Since then the IR literature has devoted a significant effort in measuring relevance of items to specific queries across different tasks: including, web search [6], personalization [30], and product rating [19]; we also refer the reader to [40] and the references therein. In the last three decades, works in the ML literature have also made significant contributions to relevance-estimation [39], by proposing methods that: (1) supplement traditional IR approaches, e.g., by automatically tuning their–previously hard to tune–parameters [57] and by improving their efficiency through clustering-based techniques [56, 2], and (2) substitute traditional IR approaches by neural-network based models to predict item relevance [11, 10, 60, 7].

Fair ranking. Existing works on the fair-ranking problem take diverse approaches: Among works that de-bias utilities, different approaches include, post-processing the utilities so that the post-processed utilities satisfy some fairness requirement [63], introducing a “fairness penalty” in the objective function used to train learning-to-rank models [55, 65], and modifying feature representations generated by up-stream algorithms so that the utilities learned from the modified representations satisfy some fairness requirements [64]. Works that alter the ranking algorithms can also be further categorized into those which satisfy the constraints for each ranking [15, 62, 24, 27] and those that satisfy the constraints in aggregate over multiple rankings [54, 8]. Unlike this work, all aforementioned works need access to the socially-salient attributes of items. When protected attributes are inaccurate, these works can fail to satisfy their fairness and/or utility guarantees [26].

Effect of inaccuracies on fair-ranking algorithms. Some recent works have considered assessing fairness of rankings and ranking algorithms with missing or inaccurate protected attributes. [35] analyze the setting where all protected attributes are missing, but can be purchased at a fixed cost per item. They give statistical-techniques to estimate the fairness-value of a given ranking at a small cost. [26] use ML-classifiers to infer protected attributes from real-world data and study performance of the fair-ranking algorithm by [25] when given inferred attributes as input. While these works underscore the need for fair-ranking algorithms to be robust to inaccuracies in protected attributes, they only assess fairness in the presence of noisy protected attributes.

2 Model of fair ranking with noisy attributes

Ranking problem. In ranking problems, given m items, one has to select a subset of n items and output a permutation of the selected items. This permutation is said to be a *ranking*. There is a large body of work on estimating the relevance of items and personalizing these estimates to specific users/queries [40, 39]. We consider a ranking problem where the relevance of items are known. Abstracting relevance estimation, in this problem, one is given an $m \times n$ matrix W, such that placing the i-th item at the j-th position generates *utility* W_{ij}. The utility of a ranking is the sum of utilities generated by each item in its assigned position. The algorithmic task in the ranking problem is to output a ranking with the highest utility. We denote rankings by assignment matrices $R \in \{0,1\}^{m\times n}$, where $R_{ij} = 1$ indicates that item i appears in position j, and $R_{ij} = 0$ indicates otherwise. In this notation, the utility of a ranking is $\langle R, W\rangle \coloneqq \sum_{i=1}^m \sum_{j=1}^n R_{ij} W_{ij}$. Then this ranking problem is to solve: $\max_{R\in\mathcal{R}} \langle R, W\rangle$. Where $\mathcal{R}$ is the set of all assignment matrices denoting a ranking:

$$\mathcal{R} \coloneqq \Big\{X \in \{0,1\}^{m\times n} : \forall i \in [m], \textstyle\sum_{j=1}^n X_{ij} \le 1,\ \ \forall j \in [n], \sum_{i=1}^m X_{ij} = 1\Big\}. \tag{1}$$

Here, the constraint $\sum_{i=1}^m X_{ij} = 1$ ensures position j has exactly one item and the constraint $\sum_{j=1}^n X_{ij} \le 1$ ensures that item i occupies at most one position.

Fair-ranking problem. There are several versions of the fair-ranking problem. We consider a version with $p \ge 2$ *socially-salient groups* $G_1, G_2, \ldots, G_p \subseteq [m]$ (e.g., the group of all women or all Black people) which are often protected by law. Each of the m items belongs to *one or more* of these socially-salient groups (henceforth referred to as just groups). This fair-ranking problem is to output the ranking with maximum utility subject to satisfying certain fairness criteria with respect to these groups. The appropriate notion of fairness is context dependent, and to capture different fairness criteria numerous *fairness constraints* have been proposed. We consider a class of general fairness constraints.

Definition 2.1 (**Fairness constraints**). Given a matrix $U \in \mathbb{Z}_+^{n\times p}$, a ranking R satisfies the upper bound constraint if $\sum_{i\in G_\ell}\sum_{j=1}^{k} R_{ij} \leq U_{k\ell}$, for all $\ell \in [p]$ and $k \in [n]$.

Existing works consider similar constraints and show that they can encapsulate a variety of fairness criteria [54]. For instance, when groups are disjoint, to capture equal and proportional representation, one can choose $U_{k\ell}:=\left\lceil k\cdot\frac{1}{p}\right\rceil$ and $U_{k\ell}:=\left\lceil k\cdot\frac{|G_\ell|}{m}\right\rceil$ for all k and ℓ respectively. As a running example, we consider the fair-ranking problem with equal representation with two disjoint groups, i.e.,

$$\max_{R\in\mathcal{R}} \langle R, W\rangle \quad \text{s.t.} \quad \forall k\in[n]\ \forall \ell\in[2], \quad \sum_{i\in G_\ell}\sum_{j=1}^{k} R_{ij} \leq \left\lceil \tfrac{k}{2}\right\rceil. \tag{2}$$

To ease readability, we omit ceilings-operators henceforth.

Noise model. If the socially-salient attributes of items are known accurately, then one can solve the fair-ranking problem. However, as discussed, in many contexts, attributes are inaccurate, missing, or only probabilistically known. Several models have been proposed to capture different errors in attributes. Here, we consider a model (due to [4]) which has also appeared in [22, 36, 42].

Definition 2.2 (**Noise model**). Let $P \in [0,1]^{m\times p}$ be a known matrix. The groups $G_1, \ldots, G_p \subseteq [m]$ are random variables, such that, for each $i \in [m]$ and $\ell \in [p]$, $\Pr[G_\ell \ni i] = P_{i\ell}$. Moreover, for different items $i \neq j$ the events $G_\ell \ni i$ and $G_k \ni j$ are *independent* for all $\ell, k \in [p]$.

Definition 2.2 makes two key assumptions: the matrix P is known and for each item i, the events $G_\ell \ni i$ over groups ℓ are independent of the corresponding events for other items. Both of these assumptions hold when attributes are flipped to preserve local differential privacy (Remark A.1). In other settings, P's estimate can be inaccurate and above events may be correlated. These can adversely affect the performance of our framework. We empirically study this in simulations where P is estimated using confidence scores of off-the-shelf classifiers and is *miscalibrated* (Figures 2 and 3).

Fairness constraint with noisy attributes. Most existing fairness constraints assume that the groups are deterministic. Hence, it is not clear how to impose them when groups are random variables, as in Definition 2.2. One definition is to require the constraints to be approximately satisfied with high probability. Consider the instantiation of this definition for equal representation: A ranking R satisfies (ρ,δ)-equal representation, if with probability $1-\delta$, at most $\frac{k}{2}(1+\rho)$ items from G_ℓ appear in the first k positions in R places for all $k \in [n]$ and $\ell \in [2]$. Naturally, one would like to satisfy this definition for small δ, ρ. However, it turns out to be too stringent and is infeasible for any small δ, ρ.

Proposition 2.3. *No ranking satisfies (ρ,δ)-equal representation for $\rho < 1$, $\delta \leq \frac{1}{2}$, and $P = \left[\frac{1}{2}\right]_{m\times p}$.*

The proof of Proposition 2.3 shows that any ranking R violates the equal-representation constraint at the 2nd position by a multiplicative factor of 2 with probability $\frac{1}{2}$. The issue is that the same relaxation parameter ρ is used for each position. Motivated by this observation, we consider the following alternate version of upper bound constraints.

Definition 2.4 (**(ε,δ)-constraint**). For any $\varepsilon \in \mathbb{R}^n_{\geq 0}$ and $\delta \in (0,1]$, a ranking R is said to satisfy (ε,δ)-constraint if with probability at least $1-\delta$ over the draw of $G_1,\ldots,G_p$

$$\forall k\in[n]\ \forall\ell\in[p], \quad \sum_{i\in G_\ell}\sum_{j=1}^{k} R_{ij} \leq U_{k\ell}(1+\varepsilon_k). \tag{3}$$

We would like to output a ranking that satisfies Definition 2.4 for small δ and small $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n$.

Problem 2.5 (**Ranking problem with noisy attributes**). Given matrices $W \in \mathbb{R}^{m\times n}_{\geq 0}$, $U \in \mathbb{R}^{n\times p}_{\geq 0}$, and $P \in [0,1]^{m\times p}$, find the ranking R maximizing utility $\langle R, W\rangle$ subject to satisfying (ε,δ)-constraint for some small ε and δ.

2.1 Challenges in solving Problem 2.5

In this section we discuss potential approaches for solving Problem 2.5. In other words, solving:

$$\max_{R\in\mathcal{R}} \langle R, W\rangle, \ \text{s.t.,}\ R \text{ satisfies } (\varepsilon,\delta)\text{-constraint}. \tag{4}$$

Even for two disjoint groups, given $V \geq 0$, it is **NP**-hard to decide if the value of Program (4) is at least V (Theorem F.5). To bypass this hardness, one can consider approximation algorithms. Program (4) is an integer program because the entries of the matrix R are required to be integers (Equation (1)). A standard approach to (approximately) solve integer programs is to: (1) consider their continuous relaxation that drops the integrality constraints, (2) compute the optimal solution R_c of the relaxed problem, and then (3) "round" R_c to satisfy integrality constraints while "retaining" its utility and fairness properties. To take this approach, we first need an efficient algorithm to find R_c. However, not just Program (4), but even its continuous relaxation is non-convex and, hence, it is unclear how to solve it to find R_c.

Due to the independence assumption in Definition 2.2, the number of items from G_ℓ appearing in the first k positions of a ranking is concentrated around its expectation (for large k). This implies that if, in expectation, less that $U_{k\ell}$ items from G_ℓ appear in the top k positions then, with high probability, the number of items from G_ℓ in the top k positions is not much larger than $U_{k\ell}$. Using this one can show that a ranking satisfying the following constraints

$$\forall k \in [n]\ \forall \ell \in [p], \quad \mathbb{E}[\textstyle\sum_{i\in G_\ell}\sum_{j=1}^{k} R_{ij}] \le U_{k\ell} \tag{5}$$

also satisfies (ε,δ)-constraint for small ε and δ. One idea is to find the ranking maximizing utility subject to satisfying Constraint (5). A feature of Constraint (5) is that it is linear in R as $\mathbb{E}[\sum_{i\in G_\ell}\sum_{j=1}^{k} R_{ij}] = \sum_{i=1}^{m}\sum_{j=1}^{k} P_{i\ell}R_{ij}$ and, hence, one may hope to find the ranking with the maximum utility subject to satisfying Constraint (5). However, the issue is that there are examples where any ranking satisfying Constraint (5) has 0 utility and there are rankings that satisfy (ε,δ)-constraint and have a large positive utility (Lemma F.3). Hence, this approach can output rankings whose utility is significantly smaller than the utility of the solution to Problem 2.5. To overcome this, we relax Constraint (5) by a carefully chosen position-dependent factor, such that, any ranking satisfying the (ε,δ)-constraint (for appropriate ε and δ) is also feasible for our framework.

3 Theoretical results

In this section we present our optimization framework and its fairness and utility guarantees.

Input: Matrices $P\in[0,1]^{m\times p}$, $W\in\mathbb{R}^{m\times n}_{\ge 0}$, $U\in\mathbb{R}^{n\times p}$

Parameters: Constant $c > 1$, failure probability $\delta\in(0,1]$, and $k\in[n]$, relaxation parameter

$$\gamma_k \coloneqq 12\cdot\log\left(\tfrac{2np}{\delta}\right)\cdot \max\nolimits_{\ell\in[p]}\sqrt{\tfrac{1}{U_{k\ell}}}. \tag{6}$$

Our Fair-Ranking Program

$$\max\nolimits_{R\in\mathcal{R}}\ \langle R, W\rangle, \qquad \text{(Noise Resilient)} \tag{7}$$

s.t. $\forall \ell\in[p]\ \ \forall k\in[n]$

$$\textstyle\sum_{i\in[m],\, j\in[k]} P_{i\ell}R_{ij} \le U_{k\ell}\left(1+\left(1-\frac{1}{2\sqrt{c}}\right)\gamma_k\right) \tag{8}$$

The above program is a modification of the program for fair ranking with accurate groups: It has the same objective but different constraints. Instead of sampling the attribute values and applying constraints on the sampled values, Constraint (5) apply upper bounds on the expected number of items in the first k positions from group ℓ (see Section 2.1). Further, Constraint (5) relaxes upper bounds $U_{k\ell}$ by a small position-dependent factor. Like for Constraint (5), one can show that any ranking satisfying Constraint (8) also satisfies (ε,δ)-constraint (for small $\varepsilon_1,\dots,\varepsilon_n$ and δ). But unlike Constraint (5), and somewhat surprisingly, any ranking that satisfies (ε,δ)-constraint (for appropriate $\varepsilon_1,\dots,\varepsilon_n$ and δ) must also satisfy Constraint (8). We use this to prove Theorem 3.1's utility guarantee.

Our first result bounds the fairness and utility of the optimal solution of Program (7).

Theorem 3.1. *Let $\gamma\in\mathbb{R}^n$ be as defined in Equation (6). There is an optimization program (Program (7)), parameterized by a constant c and failure probability δ, such that for any $c>1$ and $\delta\in(0,\frac{1}{2}]$ its optimal solution satisfies $(c\gamma,\delta)$-constraint and has a utility at least as large as the utility of any ranking satisfying $((c-\sqrt{c})\gamma,\delta)$-constraint.*

For equal representation, γ_k is $\widetilde{O}\left(\frac{1}{\sqrt{k}}\right)$. Thus, Theorem 3.1 guarantees that, with high probability, the optimal solution of Program (7) multiplicatively violates equal representation at the k-th position by at most $1+\widetilde{O}\left(\frac{1}{\sqrt{k}}\right)$. Further, this solution's utility is higher than the utility of any ranking satisfying a slight relaxation of this fairness guarantee. Theorem 3.1 can be extended to position-weighted versions of fairness constraints (Theorem E.1), where the fairness constraint is $\sum_{i\in G_\ell}\sum_{j\in[k]} v_j R_{ij} \le U_{k\ell}$ (for all k and ℓ) for specified discount factors $v_1\ge\dots\ge v_n$ such as NDGC [29]. If we are also guaranteed $U_{k\ell}\ge\psi k$ for some constant $\psi>0$ and all k and ℓ, then we can improve γ_k's dependence on δ from $\log\frac{1}{\delta}$ to $\sqrt{\log\frac{1}{\delta}}$ (Supplementary Material D.2). The proof of Theorem 3.1 appears in Section 5.

Since $(c-\sqrt{c})\gamma < c\gamma$, Theorem 3.1 gives a pseudo-optimality guarantee on utility. Does a different constraint $\mathcal{C}$ guarantee optimal utility for the achieved fairness? Let $R_{\mathcal{C}}$ be a ranking maximizing utility subject to satisfying $\mathcal{C}$. Are there small ε and δ, such that $R_{\mathcal{C}}$ satisfies (ε,δ)-constraint and has utility at least as large as any other ranking satisfying the (ε,δ)-constraint? We prove that, for any value of ε and δ, the (ε,δ)-constraint is the unique constraint with this property (Proposition F.1). However, solving the program corresponding to (ε,δ)-constraint (Program (4)) seems intractable (see Section 2.1). Unless Program (4) can be efficiently solve, a pseudo-optimality guarantee is necessary.

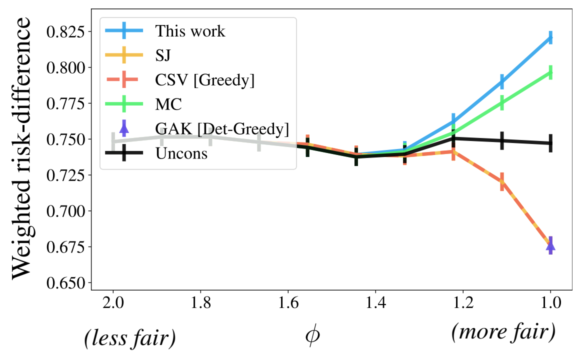


Figure 1: *Synthetic Data: Nonuniform Error Rate.* We consider synthetic data where imputed socially-salient attributes have a higher false-discovery rate on the minority group. We vary the fairness constraint (ϕ) and observe the weighted risk-difference (RD) of algorithms. The y-axis plots RD and x-axis plots ϕ. (*Note that the x-axis decreases toward the right*). We observe that **NResilient** achieves the most fair RD, while obtaining a similar utility for all ϕ (Figure 4). Error-bars denote the error of the mean.

Lower bound on fairness guarantee. Our next result complements Theorem 3.1's fairness guarantee.

Theorem 3.2. *There is a family of matrices $U \in \mathbb{Z}_+^{n\times p}$ such that for any U in the family and any parameters $\delta \in [0,1)$ and $\varepsilon_1,\ldots,\varepsilon_n \geq 0$, if for any position $k \in [n]$ $\varepsilon_k \leq 1$ and $\varepsilon_k < \max_{\ell\in[p]} \sqrt{\frac{1}{2U_{k\ell}} \log \frac{1}{4\delta}}$ then there exists a matrix $P \in [0,1]^{m\times p}$, such that it is information theoretically impossible to output a ranking that satisfies (ε,δ)-constraint. This family contains the matrix U corresponding to equal representation constraints.*

Since γ_k is $O\left(\log(\frac{np}{\delta}) \cdot \max_\ell \sqrt{\frac{1}{U_{k\ell}}}\right)$, Theorem 3.2 shows that Theorem 3.1's fairness guarantee is optimal up to log-factors. Supplementary Material D.3 proves Theorem 3.2.

An efficient algorithm. As for solving our optimization program, it is **NP**-hard to check its feasibility (Theorem D.10). However, because Constraint (8) is linear in R, the continuous relaxation of Program (7) is a standard linear program and can be solved efficiently. Our algorithm (Algorithm 1) solves the standard linear programming relaxation of Program (7) to find a solution R_c and then uses a dependent-rounding algorithm by [16] to convert R_c to a ranking.

Theorem 3.3. *There is a randomized algorithm (Algorithm 1) that given constants $d > 2$, a failure probability $0 < \delta \leq 1$, and matrices $P \in [0,1]^{m\times p}$ and $W \in [0,1]^{m\times n}$, outputs a ranking satisfying $(O(d\gamma),\delta)$-constraint and with probability at least $1-\delta$, and has a utility at least $\left(1-\frac{1}{d}\right)\cdot V - \widetilde{O}(\sqrt{dn})$, where V is the utility of any ranking satisfying $((d-\sqrt{d})\gamma,\delta)$-constraint. The algorithm runs in polynomial time in d and the bit complexity of the input.*

The tension in setting d is that decreasing d improves the fairness guarantee and the second term in the utility guarantee but worsens the first term in the utility guarantee. Under the mild assumption that $V = \Omega(n)$, increasing d improves the utility guarantee because the first term in the utility guarantee dominates the second term. In this case, the utility guarantee improves to $(1-\frac{1}{d}-o(1))\cdot V$. Finally, while Theorem 3.3 requires the utilities (entries of W) to be between 0 and 1, it can be extended to any non-negative and bounded utilities by appropriate scaling. The proof of Theorem 3.3 appears in Supplementary Material D.4.

4 Empirical results

In this section[1] we evaluate our framework's performance synthetic and real-world data.

Baselines and metrics. The correct choice of fairness metric is context-dependent and beyond the scope of this work [53]. To illustrate our results, we arbitrarily fix the fairness metric as weighted risk-difference (RD). This is a position-weighted version of the standard risk-difference metric [12] and measures the extent to which a ranking violates equal representation. The RD of a ranking R is:

$$1 - \frac{1}{Z}\sum_{k=5,10,\ldots} \frac{1}{\log k} \max_{\ell,q\in[p]} \left|\sum_{i\in G_\ell, j\in[k]} R_{ij} - \sum_{i\in G_q, j\in[k]} R_{ij}\right|,$$

Where G denotes the ground-truth protected groups and Z is a constant so that RD has range $[0,1]$. Here, $\mathrm{RD}=1$ is most fair and $\mathrm{RD}=0$ is least fair. We compare our framework, **NResilient**, against state-of-the-art fair-ranking algorithms: **CSV** ("greedy" in [15]), **SJ** [54], and **GAK** ("DetGreedy" in [24]). We also compare against **MC**, which ranks the items, in the subset output by [42]'s algorithm, to maximize utility. Finally, we compare against the baseline, **Uncons**, which outputs the utility maximizing ranking without fairness considerations. We present additional discussion of results, additional plots for RD, and comparisons with weighted selection-lift in Supplementary Material B.

[1] Anonymized code for our simulations is available at `https://github.com/NoisyRanking/FairRankingWithNoisyAttributes`

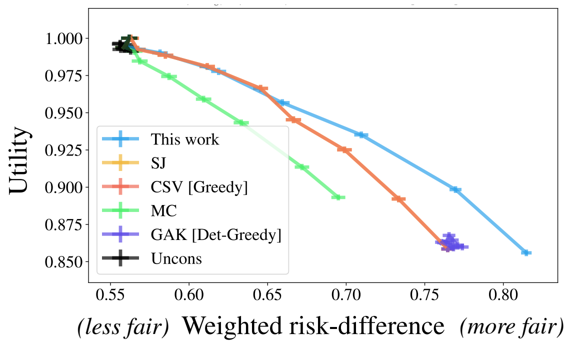


Figure 2: *Real-world image data.* In this simulation, given *non-gender labeled* images and their utilities, our goal is to generate a high-utility gender-balanced ranking. We estimate P using an off-the-shelf ML-classifier and vary ϕ from $p = 2$ (less fair) to 1 (more fair). The y-axis plots the utility of algorithms and the x-axis plots RD. We observe that **NResilient** has the most fair RD and the best fairness-utility trade-off. Error bars show the error of the mean.

Setup. We consider the DCG model of utilities [29] and a relaxation of equal representation constraints: (1) Given an intrinsic value $w_i \geq 0$, for each item i, we set $W_{ij} := w_i \left(\log\left(j+1\right)\right)^{-1} \forall j \in [n]$. (2) Given a parameter $\phi \in [1, p]$, we set upper bounds $U_{k\ell} := \frac{\phi}{p} \cdot k$ for each $k \in [n]$ and $\ell \in [p]$. In simulations, we set $m = 500$, $n = 25$, and vary ϕ from p to 1. For each ϕ, we draw m items uniformly without replacement and compute an estimate $\widehat{P}$ of the matrix P from Definition 2.2; details are given with each simulation. We infer socially-salient groups $\widehat{G}_1, \ldots, \widehat{G}_2$ via $\widehat{P}$ by assigning each item to its most-likely group. Finally, we run all algorithms using $\widehat{P}$ or $\widehat{G}_1, \ldots, \widehat{G}_2$ as discussed next.

Implementation details. NResilient and **MC** take probabilistic information about socially-salient attributes as input and are given $\widehat{P}$. **CSV**, **SJ**, and **GAK** require access to socially-salient groups and are given $\widehat{G}_1, \ldots, \widehat{G}_p$. **NResilient**, **SJ**, and **CSV** use fairness constraints from Definition 2.1 and are given: for each $k \in [n]$ and $\ell \in [p]$, $U_{k\ell} = \frac{\phi}{p} \cdot k$. **MC** requires, for each $\ell \in [p]$, an upper bound on the number of items from G_ℓ that can appear in top-n positions. It is given $\frac{\phi}{p} \cdot n$ for each $\ell \in [p]$. **GAK** requires the desired proportion α_ℓ for each group G_ℓ and, roughly, satisfies the constraint $U_{k\ell} = \alpha_\ell \cdot k$ for each $k \in [n]$ and $\ell \in [p]$. It is given $\alpha_\ell = \frac{1}{p}$ for each $\ell \in [p]$, this corresponds to $\phi = 1$ (hence, the figures only plot the **GAK** at $\phi = 1$). As a heuristic, we set $\gamma_k = \frac{1}{20} \cdot \max_{\ell \in [p]} \sqrt{\frac{1}{U_{k\ell}}}$ in all simulations. We find that this parameter suffices and expect a more refined approach to improve the performance of **NResilient**.

Simulation on synthetic data. We show that on synthetic data, where error-rates of given socially-salient attributes vary over groups, existing fair-ranking algorithms have worse RD than **Uncons**.

Data. We generate w and P for two groups using code by [42] and fix $\widehat{P} = P$. For all items i, w_i is i.i.d. from the uniform distribution over $[0, 1]$. $\widehat{P}$ is constructed such that attributes inferred from $\widehat{P}$ have a higher false-discovery rate for the minority group compared to the majority (40% vs 10%).

Results. See Figure 1 for the observed RD averaged over 500 iterations. We observe that **NResilient** achieves best RD (≈0.81), while not loosing significant utility ($\geq 0.98\%$ of maximum; see Figure 4). **MC** achieves the best RD (≈0.79). In contrast, **CSV**, **SJ**, and **GAK**, which do not account for noise in the socially-salient attributes, achieve a worse RD at $\phi \approx 1$ (≤0.68) than **Uncons** (≈0.75). Thus, we observe that existing fair-ranking algorithms may achieve a worse RD than **Uncons**.

Simulation on real-world image data. In this simulation, given *non-gender labeled* images-search results and their utilities, our goal is to generate a high-utility and gender-balanced ranking.

Data. We use the Occupations dataset [14] which contains the top 100 Google Image results for 96 occupation-related queries. For each image, the data has its position in search results, gender (coded as male/female) of the individual depicted in the image, collected via MTurk. We use the (true) gender labels in the data to compute RD and to estimate $\widehat{P}$, but do not provide them to algorithms.

Setup. For each image i, with rank r_i, we define $w_i := \left(\log\left(1 + r_i\right)\right)^{-1}$. We say an occupation is gender-stereotypical if more than 80% of images for this occupation have the same gender label (41/96 occupations). An image is said to be stereotypical if its in a gender-stereotypical occupation and its gender label is the majority label for its occupation. We define the socially-salient groups as the sets of stereotypical and non-stereotypical images in gender-stereotypical occupations.

Estimating $\widehat{P}$. After pre-processing, we use a CNN-based gender-classifier f [52] to predict the (apparent) gender of the person depicted in each image. We calibrate the confidence scores output by f by binning and use these to estimate $\widehat{P}$ (see Supplementary Material B for more details). We perform this calibration once and on all occupations and, then, use it for gender-stereotypical occupations. Because of this $\widehat{P}$ is miscalibrated (and hence, inaccurate). For instance, among samples i for which $0.25 \leq \widehat{P}_{i2} \leq 0.5$, more than 75% are labeled as 'man' (instead of some percentage between 25% and 50%). *This violates the assumption that P is accurately known.*

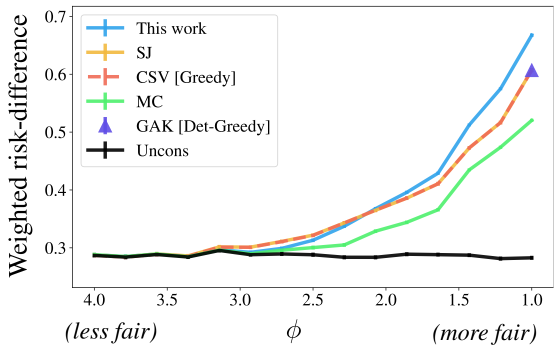


Figure 3: *Real-World Name Data: Multiple Attributes.* In this simulation, the goal is to ensure equal representation across four disjoint groups formed by combinations of two attributes (non-White non-men, White non-men, non-White men, and White men). We estimate P by querying public APIs and libraries with names in the data. The y-axis plots RD and x-axis plots ϕ. (*Note that the values decrease toward the right*). We observe that all algorithms have a better RD than **Uncons** and **NResilient** has the best RD compared to all other baselines. Error bars represent the error of the mean.

Results. See Figure 2 for RD and utilities (NDGC) averaged over 1000 iterations. We observe that **NResilient** achieves the best RD ($\approx$0.81) and has a better RD-utility trade-off than the other baselines. In contrast, **CSV**, **SJ**, and **GAK**, achieve a worse RD ($\leq$0.77). **MC** achieves the worst RD ($\leq$0.70) and a worst RD-utility trade-off. We further evaluate the robustness of **NResilient** to varying levels of noise on the Occupations dataset in Supplementary Material B and observe **NResilient** has a better or similar RD than each baseline at all noise levels.

Simulation on real-world name data. We consider gender and race (encoded as binary) as socially-salient attributes. Our goal is to ensure equal representation across the four disjoint groups formed by combinations of these: non-White non-men, White non-men, non-White men, and White men.
Data. We consider the chess ranking data [26] which has of 3,251 chess players. For each player, among other attributes, the data has their full-name, self-identified gender (coded as male/female), FIDE rating, and race (Asian, Black, Hispanic, White) collected via MTurk. We use the (true) gender and race labels in the data to evaluate RD, but do not provide them to algorithms.
Setup. We partition the races into White (81.66%) and non-White (18.34%). For each player i, we query Genderize and EthniColr[2] with i's full-name to obtain the "probabilities" $p_f(i)$ and $p_{nw}(i)$ that player i is labeled as a women and non-white respectively. We assume that these probabilities are correct and that the gender and race of players are drawn independently. Hence, e.g., we set the probability that i is a non-white women as $\widehat{P}_{i,nw+f}{=}p_{nw}(i)p_f(i)$. Similarly, we set $\widehat{P}_{i,w+f}{=}(1-p_{nw}(i))p_f(i)$, $\widehat{P}_{i,nw+m} = p_{nw}(i)(1-p_f(i))$, and $\widehat{P}_{i,w+m} = (1-p_{nw}(i))(1-p_f(i))$. Notably, we do not calibrate $\widehat{P}$ on this data. We verify that, like the previous simulation, $\widehat{P}$ is miscalibrated in this simulation. E.g., only 31% of the samples i for which $\widehat{P}_{i,nw+m} > 0.75$ are labeled as 'Non-white man' (instead of 75%). *Hence, the assumption that P is accurately known is violated in this simulation.* We expect calibration to improve **NResilient**'s performance.
Results. See Figure 3 for RD averaged over 500 iterations. We observe that all algorithms (**NResilient**, **CSV**, **GAK**, **SJ**, and **MC**) have better RD than **Uncons**. Among these, **NResilient** achieves the best RD ($\approx$0.67), next **CSV**, **GAK**, and **SJ** obtain RD ($\approx$0.61), and **MC** achieves RD ($\leq$ 0.53). Further, in Figure 6, we observe that all algorithms have a similar fairness-utility trade-off.

5 Proof of Theorem 3.1

In this section we prove Theorem 3.1. Some of the details are deferred to Supplementary Material D.2 due to space constraints. The proof is divided into two propositions:

Proposition 5.1. *For any $\delta \in (0,1]$, any ranking feasible for Prog.* (7) *satisfies* $(c\gamma, \delta)$*-constraint.*

Proposition 5.2. *For any $\delta \in (0, \frac{1}{2})$ and $c > 1$, any ranking satisfying the* $((c-\sqrt{c})\gamma, \delta)$*-constraint is feasible for Program* (7).

Proof of Theorem 3.1. Let $R^\star$ be the optimal solution of Program (7). Since $R^\star$ is feasible by definition, Proposition 5.1 implies that $R^\star$ satisfies the $(c\gamma, \delta)$-constraint. Pick any R' that satisfies the $((c-\sqrt{c})\gamma, \delta)$-constraint. Proposition 5.2 implies that R' is feasible for Program (7). Since $R^\star$ is an optimal solution of Program (7), $R^\star$'s utility is at least as large as the utility of R'. □

Notation. For each item i and group ℓ, let $Z_{i\ell} \in \{0,1\}$ be the indicator random variable $Z_i \coloneqq \mathbb{I}[G_\ell \ni i]$. By Definition 2.2, $\Pr[Z_{i\ell}] = P_{i\ell}$ and $Z_{i\ell}$ and $Z_{j\ell}$ are independent for any $i \neq j$. Given ranking $R \in \mathcal{R}$, group $\ell \in [p]$, and position $k \in [n]$, let $Z_\#(R, \ell, k)$ be the number of items from G_ℓ in the top k positions of R and let $P_\#(R, \ell, k) = \mathbb{E}[Z_\#(R, \ell, k)]$. From the above, we get:

$$P_\#(R, \ell, k) = \mathbb{E}\left[Z_\#(R, \ell, k)\right] = \textstyle\sum_{i \in [m]} \sum_{j \in [k]} P_{i\ell} R_{ij}.$$

[2] `gender-api.com` and `github.com/appeler/ethnicolr` respectively

We will use the following concentration result in the proof. It is proved in Supplementary Material D.1.

Lemma 5.3. *For any position $k \in [n]$, group $\ell \in [p]$, parameters $\varepsilon \geq 0$ and $L, U \in \mathbb{R}$, and ranking $R \in \mathcal{R}$, where R is possibly a random variable independent of $\{Z_{i\ell}\}_{i,\ell}$, if $P_\#(R,\ell,k) \leq U$ or $P_\#(R,\ell,k) \geq L$ then the following equations hold respectively* $\Pr\left[Z_\#(R,\ell,k) < (1+\varepsilon)\,U\right] \geq 1 - e^{-\frac{U\varepsilon^2}{2+\varepsilon}}$ *and* $\Pr\left[Z_\#(R,\ell,k) > (1-\varepsilon)\,L\right] \geq 1 - e^{-\frac{L\varepsilon^2}{2(1-\varepsilon)}}$.

Proof of Proposition 5.1. Fix any k and ℓ. Let

$$\phi \coloneqq 1 - \tfrac{1}{2\sqrt{c}}, \quad U' \coloneqq U_{k\ell}\left(1+\phi\gamma_k\right), \quad \text{and} \quad \zeta \coloneqq \tfrac{(1-\phi)\gamma_k}{1+\phi\gamma_k} \tag{9}$$

Here, U' and ζ satisfy that $U'(1+\zeta) = U_{k\ell}(1+c\gamma_k)$. Fix any ranking R that is feasible for Program (7). Since R is feasible, it satisfies that

$$\forall \ell \in [p],\ k \in [n], \quad P_\#(R,\ell,k) \leq U_{\ell k}\left(1+\phi\gamma_k\right). \tag{10}$$

Using that $U'(1+\zeta) = U_{k\ell}(1+c\gamma_k)$, Equation (10), and Lemma 5.3, we get that

$$\Pr\left[Z_\#(R,\ell,k) \geq U'(1+\zeta)\right] \quad \leq \quad e^{-\frac{2U'\zeta^2}{2+\zeta}} \overset{(9)}{=} e^{-\frac{(1-\phi)^2c^2\gamma_k^2U_{k\ell}}{2+(1+\phi)c\gamma_k}} \overset{(\phi\leq 1)}{\equiv} e^{-\frac{(1-\phi)^2c^2\gamma_k^2U_{k\ell}}{2(1+c\gamma_k)}}. \tag{11}$$

Fact 5.4. For all $x, y \geq 0$, if $x \geq y + \sqrt{y}$, then $\frac{x^2}{1+x} \geq y$.

Using Fact 5.4 and Equation (6), we can show that for each k, $\frac{c^2\gamma_k^2}{1+c\gamma_k} \geq \frac{2}{(1-\phi)^2U_{k\ell}} \cdot \log\frac{2np}{\delta}$. (This uses $\delta < \frac{1}{2}$ and $U_{k\ell}, n \geq 1$.) Substituting this in Equation (11) we get:

$$\Pr\left[Z_\#(R,\ell,k) \geq U_{\ell k}(1+c\gamma_k)\right] \leq \tfrac{\delta}{2np}. \tag{12}$$

Taking the union bound over all positions k and ℓ, we get (as desired) that with probability at least $1-\delta$, for all $k \in [n]$ and $\ell \in$, $Z_\#(R,\ell,k) \leq U_{\ell k}(1+c\gamma_k)$. □

Proof of Proposition 5.2. Let $\phi \coloneqq 1 - \frac{1}{2\sqrt{c}}$. Towards a contradiction, suppose that R' satisfies $((c-\sqrt{c})\gamma, \delta)$-constraint but is not feasible for Program (7). Then there exists ℓ and k such that $P_\#(R',k,\ell) > U_{k\ell} \cdot (1+\phi\gamma_k)$. Fix any k and ℓ satisfying this. Let

$$b \coloneqq 1 - \tfrac{1}{\sqrt{c}}, \quad L' \coloneqq U_{k\ell}\left(1+\phi\gamma_k\right) \quad \text{and} \quad \zeta \coloneqq \tfrac{(1+b)\gamma_k}{1+\phi\gamma_k} \tag{13}$$

It holds that $L'(1-\zeta) = U_{k\ell}(1+b\gamma_k)$ and, hence, we get

$$\Pr\left[Z_\#(R',k,\ell) \leq L'(1-\zeta)\right] \overset{(13),\ \text{Lem.5.3}}{\leq} e^{-\frac{L'\zeta^2}{2(1-\zeta)}} \overset{(13)}{=} e^{\frac{-(c-b)^2U_{k\ell}\gamma_k}{2(1+b)}} \leq e^{\frac{-U_{k\ell}c\gamma_k}{4(2\sqrt{c}-1)\sqrt{c}}} \overset{(c\geq 0)}{=} e^{\frac{-U_{k\ell}\gamma_k}{8}}. \tag{14}$$

Since $\gamma_k \geq 8\log\frac{np}{\delta} \cdot \max_\ell \sqrt{\frac{1}{U_{k\ell}}}$, $\delta < \frac{1}{2}$, and $U \geq 1$, we have $\Pr\left[Z_\#(R',k,\ell) \leq U_{k\ell}\right] \leq \frac{\delta}{np} < 1-\delta$. Since R' satisfies $((c-\sqrt{c})\gamma, \delta)$-constraint we have a contradiction, hence R' must be feasible. □

6 Limitations and conclusion

Recent studies find that errors in socially-salient attributes can adversely affect the fairness and utility of existing fair-ranking algorithms [26]. We consider a model of random and independent errors in socially-salient attributes and present a framework that can output rankings with high fairness and utility in this model. This framework works a general class of fairness criteria, which involve multiple overlapping groups and upper bounds on the number of items that appear in the first k positions from each group. We also show near-tightness of the framework's fairness guarantee. Empirically, on both synthetic and real-world datasets, we observe that, compared to baselines, our framework can achieve higher fairness-values and a similar or better fairness-utility trade-off for standard metrics.

Compared to existing fair-ranking frameworks, our framework does not need accurate socially-salient attributes, but assumes that errors in attributes are random and independent. When these assumptions do not hold, our framework may not satisfy its guarantees. Simulations on real-world data suggest that, in contexts represented by this data, our framework can achieve higher fairness than baselines (Section 4). Nevertheless, a careful assessment of this on application-specific data would be important to avoid any (unintended) negative social impact.

Our work only addresses one aspect of how bias may show up in rankings, and more generally, on the web. It is important to take an holistic approach to mitigate bias and incorporate our work as a part of such a broader effort. Finally, our work adds to the line of works that develop fair decision-making algorithms robust to inaccuracies in data [36, 5, 47, 22, 59, 58, 42, 13].

References

[1] OpenCV: Open Source Computer Vision Library. `https://github.com/opencv/opencv_3rdparty/raw/dnn_samples_face_detector_20170830/res10_300x300_ssd_iter_140000.caffemodel`.

[2] Ismail Sengor Altingovde, Engin Demir, Fazli Can, and Özgür Ulusoy. Incremental cluster-based retrieval using compressed cluster-skipping inverted files. *ACM Transactions on Information Systems (TOIS)*, 26(3):1–36, 2008.

[3] McKane Andrus, Elena Spitzer, Jeffrey Brown, and Alice Xiang. What We Can't Measure, We Can't Understand: Challenges to Demographic Data Procurement in the Pursuit of Fairness. In *FAccT*, pages 249–260. ACM, 2021.

[4] Dana Angluin and Philip D. Laird. Learning From Noisy Examples. *Mach. Learn.*, 2(4):343–370, 1987.

[5] Pranjal Awasthi, Matthäus Kleindessner, and Jamie Morgenstern. Equalized Odds Postprocessing under Imperfect Group Information. In *International Conference on Artificial Intelligence and Statistics*, pages 1770–1780. PMLR, 2020.

[6] Ziv Bar-Yossef and Maxim Gurevich. Random Sampling from a Search Engine's Index. *Journal of the ACM (JACM)*, 55(5):1–74, 2008.

[7] Michael Bendersky and Xuanhui Wang. Advances in TF-Ranking, July 2021. `https://ai.googleblog.com/2021/07/advances-in-tf-ranking.html`.

[8] Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. Equity of Attention: Amortizing Individual Fairness in Rankings. In *SIGIR*, pages 405–414. ACM, 2018.

[9] Joy Buolamwini and Timnit Gebru. Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. In *FAT*, volume 81 of *Proceedings of Machine Learning Research*, pages 77–91. PMLR, 2018.

[10] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96, 2005.

[11] Christopher J.C. Burges. From RankNet to LambdaRank to LambdaMART: An Overview. *Learning*, 2010.

[12] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Min. Knowl. Discov.*, 21(2):277–292, 2010.

[13] L. Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K. Vishnoi. Fair Classification with Noisy Protected Attributes. In *ICML*, volume 120 of *Proceedings of Machine Learning Research*. PMLR, 2021.

[14] L. Elisa Celis and Vijay Keswani. Implicit Diversity in Image Summarization. *Proc. ACM Hum. Comput. Interact.*, 4(CSCW2):139:1–139:28, 2020.

[15] L. Elisa Celis, Damian Straszak, and Nisheeth K. Vishnoi. Ranking with Fairness Constraints. In *ICALP*, volume 107 of *LIPIcs*, pages 28:1–28:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018.

[16] Chandra Chekuri, Jan Vondrák, and Rico Zenklusen. Multi-budgeted Matchings and Matroid Intersection via Dependent Rounding. In Dana Randall, editor, *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 1080–1097. SIAM, 2011.

[17] Jiahao Chen, Nathan Kallus, Xiaojie Mao, Geoffry Svacha, and Madeleine Udell. Fairness Under Unawareness: Assessing Disparity When Protected Class Is Unobserved. In *FAT*, pages 339–348. ACM, 2019.

[18] Cyril W Cleverdon. The significance of the cranfield tests on index languages. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 3–12, 1991.

[19] Kushal Dave, Steve Lawrence, and David M Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519–528, 2003.

[20] Marc Elliott, Peter Morrison, Allen Fremont, Daniel Mccaffrey, Philip Pantoja, and Nicole Lurie. Using the Census Bureau's Surname List to Improve Estimates of Race/Ethnicity and Associated Disparities. *Health Services and Outcomes Research Methodology*, 9:252–253, 06 2009.

[21] Robert Epstein and Ronald E Robertson. The Search Engine Manipulation Effect (SEME) And Its Possible Impact on the Outcomes of Elections. *Proceedings of the National Academy of Sciences*, 112(33):E4512–E4521, 2015.

[22] Seyed A. Esmaeili, Brian Brubach, Leonidas Tsepenekas, and John Dickerson. Probabilistic Fair Clustering. In *NeurIPS*, 2020.

[23] Benoît Frénay and Michel Verleysen. Classification in the Presence of Label Noise: A Survey. *IEEE Trans. Neural Networks Learn. Syst.*, 25(5):845–869, 2014.

[24] Sahin Cem Geyik, Stuart Ambler, and Krishnaram Kenthapadi. Fairness-Aware Ranking in Search & Recommendation Systems with Application to LinkedIn Talent Search. In *KDD*, pages 2221–2231. ACM, 2019.

[25] Sahin Cem Geyik and Krishnaram Kenthapadi. Building Representative Talent Search at LinkedIn. *LinkedIn Engineering*, October 2018. `http://bit.ly/2x65HDJ`.

[26] Avijit Ghosh, Ritam Dutt, and Christo Wilson. When Fair Ranking Meets Uncertain Inference. In *SIGIR*, pages 1033–1043. ACM, 2021.

[27] Sruthi Gorantla, Amit Deshpande, and Anand Louis. On the Problem of Underranking in Group-Fair Ranking. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 3777–3787. PMLR, 2021.

[28] Anikó Hannák, Claudia Wagner, David Garcia, Alan Mislove, Markus Strohmaier, and Christo Wilson. Bias in online freelance marketplaces: Evidence from taskrabbit and fiverr. In *Proceedings of the 2017 ACM conference on computer supported cooperative work and social computing*, pages 1914–1933, 2017.

[29] Kalervo Järvelin and Jaana Kekäläinen. Cumulated Gain-Based Evaluation of IR Techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, oct 2002.

[30] Glen Jeh and Jennifer Widom. Scaling personalized web search. In *Proceedings of the 12th international conference on World Wide Web*, pages 271–279. ACM, 2003.

[31] Christopher Jung, Changhwa Lee, Mallesh Pai, Aaron Roth, and Rakesh Vohra. Moment multicalibration for uncertainty estimation. In Mikhail Belkin and Samory Kpotufe, editors, *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 2634–2678. PMLR, 15–19 Aug 2021.

[32] Nathan Kallus, Xiaojie Mao, and Angela Zhou. Assessing Algorithmic Fairness with Unobserved Protected Class Using Data Combination. In *FAT**, page 110. ACM, 2020.

[33] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. What Can We Learn Privately? *SIAM J. Comput.*, 40(3):793–826, 2011.

[34] Matthew Kay, Cynthia Matuszek, and Sean A. Munson. Unequal Representation and Gender Stereotypes in Image Search Results for Occupations. In *CHI*, pages 3819–3828. ACM, 2015.

[35] Ömer Kirnap, Fernando Diaz, Asia Biega, Michael D. Ekstrand, Ben Carterette, and Emine Yilmaz. Estimation of Fair Ranking Metrics with Incomplete Judgments. In *WWW*, pages 1065–1075. ACM / IW3C2, 2021.

[36] Alexandre Louis Lamy and Ziyuan Zhong. Noise-Tolerant Fair Classification. In *NeurIPS*, pages 294–305, 2019.

[37] Elizabeth D. Liddy. Automatic document retrieval. In *Encyclopedia of Language and Linguistics*. Elsevier, 2005.

[38] Jiahui Liu, Peter Dolan, and Elin Rønby Pedersen. Personalized news recommendation based on click behavior. In *Proceedings of the 15th international conference on Intelligent user interfaces*, pages 31–40. ACM, 2010.

[39] Tie-Yan Liu. Learning to Rank for Information Retrieval. 3(3):225–331, mar 2009.

[40] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. Introduction to Information Retrieval. *Natural Language Engineering*, 16(1):100–103, 2010.

[41] Naresh Manwani and P. S. Sastry. Noise Tolerance Under Risk Minimization. *IEEE Trans. Cybern.*, 43(3):1146–1151, 2013.

[42] Anay Mehrotra and L. Elisa Celis. Mitigating Bias in Set Selection with Noisy Protected Attributes. In *FAccT*, pages 237–248. ACM, 2021.

[43] Christopher Mims. Why social media is so good at polarizing us, October 2020. `https://www.wsj.com/articles/why-social-media-is-so-good-at-polarizing-us-11603105204`.

[44] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. Controlling Fairness and Bias in Dynamic Learning-to-Rank. In *SIGIR*, pages 429–438. ACM, 2020.

[45] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge university press, 1995.

[46] Nima Mousavi. How tight is Chernoff bound, 2010.

[47] Hussein Mozannar, Mesrob I. Ohannessian, and Nathan Srebro. Fair Learning with Private Demographic Data. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 7066–7075. PMLR, 2020.

[48] Safiya Umoja Noble. *Algorithms of Oppression: How Search Engines Reinforce Racism*. NYU Press, 2018.

[49] Gourab K. Patro, Lorenzo Porcaro, Laura Mitchell, Qiuyue Zhang, Meike Zehlike, and Nikhil Garg. Fair ranking: a critical review, challenges, and future directions. In *FAccT*, page To appear. ACM, 2022.

[50] Ulrich Pferschy, Joachim Schauer, and Clemens Thielen. The product knapsack problem: Approximation and complexity. *CoRR*, abs/1901.00695, 2019.

[51] Evaggelia Pitoura, Kostas Stefanidis, and Georgia Koutrika. Fairness in Rankings and Recommendations: An Overview. *The VLDB Journal*, 2021.

[52] Rasmus Rothe, Radu Timofte, and Luc Van Gool. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 126(2-4):144–157, 2018.

[53] Andrew D. Selbst, Danah Boyd, Sorelle A. Friedler, Suresh Venkatasubramanian, and Janet Vertesi. Fairness and abstraction in sociotechnical systems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, pages 59–68, New York, NY, USA, 2019. ACM.

[54] Ashudeep Singh and Thorsten Joachims. Fairness of Exposure in Rankings. In *KDD*, pages 2219–2228. ACM, 2018.

[55] Ashudeep Singh and Thorsten Joachims. Policy Learning for Fairness in Ranking. In *NeurIPS*, pages 5427–5437, 2019.

[56] Pavan Kumar C Singitham, Mahathi S Mahabhashyam, and Prabhakar Raghavan. Efficiency-quality tradeoffs for vector score aggregation. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 624–635, 2004.

[57] Michael Taylor, Hugo Zaragoza, Nick Craswell, Stephen Robertson, and Chris Burges. Optimisation methods for ranking functions with multiple parameters. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 585–593, 2006.

[58] Jialu Wang, Yang Liu, and Caleb Levy. Fair Classification with Group-Dependent Label Noise. In *FAccT*, pages 526–536. ACM, 2021.

[59] Serena Wang, Wenshuo Guo, Harikrishna Narasimhan, Andrew Cotter, Maya R. Gupta, and Michael I. Jordan. Robust Optimization for Fairness with Noisy Protected Groups. In *NeurIPS*, 2020.

[60] Jason Weston, Samy Bengio, and Nicolas Usunier. Large scale image annotation: learning to rank with joint word-image embeddings. *Machine learning*, 81(1):21–35, 2010.

[61] Grace Hui Yang and Sicong Zhang. Differential Privacy for Information Retrieval. In *WSDM*, pages 777–778. ACM, 2018.

[62] Ke Yang, Vasilis Gkatzelis, and Julia Stoyanovich. Balanced Ranking with Diversity Constraints. In *IJCAI*, pages 6035–6042. ijcai.org, 2019.

[63] Ke Yang, Joshua R. Loftus, and Julia Stoyanovich. Causal Intersectionality and Fair Ranking. In *FORC*, volume 192 of *LIPIcs*, pages 7:1–7:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[64] Ke Yang and Julia Stoyanovich. Measuring Fairness in Ranked Outputs. In *SSDBM*, pages 22:1–22:6. ACM, 2017.

[65] Meike Zehlike and Carlos Castillo. Reducing Disparate Exposure in Ranking: A Learning To Rank Approach. In *ReducingDisparateExposureZehlike*, pages 2849–2855. ACM / IW3C2, 2020.

[66] Meike Zehlike, Ke Yang, and Julia Stoyanovich. Fairness in Ranking: A Survey. *CoRR*, abs/2103.14000, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See the theorems in Section 3 and Figures 1 to 3
   (b) Did you describe the limitations of your work? [Yes] See Section 6
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Section 6 discusses the importance of assessing the performance of our algorithm on application-specific data and using it as a part of a larger framework for mitigating discrimination.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See, e.g., Theorems 3.1 to 3.3 and E.1
   (b) Did you include complete proofs of all theoretical results? [Yes] See Supplementary Materials D to F
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Anonymized code for our simulations is available at `https://github.com/NoisyRanking/FairRankingWithNoisyAttributes`
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Supplementary Material B
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Please see, e.g., Figures 1 to 3
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Supplementary Material B
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4
   (b) Did you mention the license of the assets? [N/A] We use existing code by [42] and data by [14, 26]. To the best of our knowledge these assets are not licensed.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Contents

1 Introduction **1**

2 Model of fair ranking with noisy attributes **3**
2.1 Challenges in solving Problem 2.5 4

3 Theoretical results **5**

4 Empirical results **6**

5 Proof of Theorem 3.1 **8**

6 Limitations and conclusion **9**

A Additional remarks on the noise model **17**

B Additional empirical results and implementation details **17**
Code. 17
B.1 Implementation details 17
Computational resources used. 18
B.2 Pre-processing details of the simulation with image data 18
Estimating $\widehat{P}$. 18
B.3 Additional discussion and plots for simulations 18
B.4 Additional empirical results with weighted selection-lift 19
B.5 Additional empirical results varying noise 21

C Using existing fair-ranking algorithms with rounding is insufficient **21**
C.1 Proofs of Proposition C.1 and Proposition C.2 22
C.1.1 Proof of Proposition C.1 22
C.1.2 Proof of Proposition C.2 23

D Proofs of theoretical results **24**
D.1 Proof of Lemma 5.3 24
D.2 Improved dependence of γ on δ 26
Proof of Proposition 5.1 26
Proof of Proposition 5.2 27
D.3 Proof of Theorem 3.2 28
D.4 Proof of Theorem 3.3 29
D.5 Proof of Theorem D.10 33

E Extension of Theorem 3.1 to position-weighted constraints **33**
Proof of Proposition 5.1 35
Proof of Proposition 5.2 35

F Proofs of additional theoretical results **35**
F.1 Proof of Proposition 2.3 35
F.2 Proof of Proposition F.1 36
F.3 Proof of Lemma F.3 37
F.4 Proof of Proposition F.4 38
F.5 Proof of Theorem F.5 39
F.5.1 Step 1: Reduction from (61) to (59) 39
F.5.2 Step 2: Reduction from product partition problem to (61) 40
Assumptions on CPPP instances without loss of generality. 40

A Additional remarks on the noise model

Applicability of the noise model in applications. The noise in Definition 2.2, arises in real-world settings where local differential privacy is ensured e.g., using the randomized response mechanism.

Remark A.1 (**Model's assumptions hold if attributes are perturbed by randomized response**). The randomized response mechanism flips each item's protected attribute to an incorrect value with some (public) probability $0 < \eta < \frac{1}{2}$, independent of all other items. Here, the independence assumption holds (by design) and P's entries can be deduced from η. To see the latter concretely, consider two protected groups G_1 and G_2 ($p = 2$), and their noisy versions N_1 and N_2 corresponding to the "flipped" attributes. For any item $i \in N_1$,

$$P_{i1} = (1 - \eta) \cdot {}^{|G_1|}/_{|N_1|} \quad \text{and} \quad P_{i2} = 1 - P_{i1}.$$

For items in N_2, replace P_{i1}, P_{i2}, G_1, and N_1 with P_{i2}, P_{i1}, G_2, and N_2. When there are more than two groups ($p > 2$), then the randomized response mechanism publically specifies the probability $\eta_{a,b}$ with which it flips protected attribute value $\ell = a$ to another value $\ell = b$ (for any $a, b \in [p]$). As in the binary case above, P's entries can be deduced from parameters $\{\eta_{a,b} \colon a, b \in [p]\}$.

Further, in other real-world settings such as image search and online recruiting, the entries of P can be estimated using the confidence scores of classifiers or using auxiliary attributes. In more detail:

- If the protected attribute is skin tone, then a classifier C can be used to predict if image i contains a person with a dark skin tone. If C has a calibrated confidence score $0 \leq c(i) \leq 1$ in this prediction, then $P_{i,\text{darkskin-tone}} = c(i)$. See Figure 2 in Section 4 for results from a simulation that estimates P in this fashion.
- If the protected attribute is race and individuals are uniformly drawn from the population, then for an individual i with surname S and zip-code Z, $P_{i,L} = f(Z, S)$, where $f(Z, S)$ is the fraction of individuals with surname S in zip-code Z who have the L-th race; which can be estimated using census data [20] (see Figure 3 in Section 4).

Discussion on the noise model with disjoint groups vs. overlapping groups. For each item i and group G_ℓ ($\ell \in [p]$), the noise model specifies the marginal probability that i belongs to G_ℓ: $P_{i\ell} \coloneqq \Pr[G_\ell \ni i]$. For any i, the model allows for any joint probability distribution over the events $(G_1 \ni i), (G_2 \ni i), \ldots, (G_p \ni i)$ that is consistent with the above marginal probabilities. This allows the model to capture the setting where all groups are disjoint – by requiring the events $(G_1 \ni i), \ldots, (G_p \ni i)$ to be mutually exclusive. It also allows the model to capture the cases where all or only some of the groups can overlap. For instance, the case where G_1 can overlap with G_2 but both G_1 and G_2 are disjoint from G_3 can be captured by requiring the events $(G_3 \ni i)$ to be mutually exclusive of the events $(G_1 \ni i)$ and $(G_2 \ni i)$. Importantly, we do not need additional information to capture these settings–it suffices to know the marginal probabilities specified by P.

B Additional empirical results and implementation details

In this section, we present the implementation details of our simulations (Supplementary Materials B.1 and B.2), give additional plots for the simulation in Section 4 (Supplementary Material B.3), and additional simulations that use weighted-selection risk as the fairness metric or vary the amount of noise in the data (Supplementary Materials B.4 and B.5)

Code. The anonymized code for all simulations is available at `https://github.com/NoisyRanking/FairRankingWithNoisyAttributes`.

B.1 Implementation details

In this section, we give implementation details of our algorithm and baselines.

- **NResilient**: We implement **NResilient** in Python 3 and use the Gurobi optimization library to solve the linear program in Step 1 of Algorithm 1.
- **SJ**: This is [54]'s algorithm. **SJ** (1) solves a linear program specified by the protected groups $G_1, \ldots, G_p$, upper bounds $U_1, \ldots, U_p$, and utilities, (2) decomposes the solution as a convex combination of the rankings, and uses this convex combination to generate rankings (see [54, Section 3.4]). [54] do not provide an implementation of **SJ**, we implement **SJ** in Python3: We

use the Gurobi optimization library to solve the linear program constructed by [54] and use the code available at `https://github.com/jfinkels/birkhoff` to compute the Birkhoff-von Neumann decomposition of the solution ([54] also use the same code to compute the decomposition, see [54, Section 3.4]).

- **CSV**: This is the greedy algorithm from [15, Theorem 3.3]. [15] do not provide an implementation of **CSV**, we implement their algorithm in Python3 with NumPy.
- **GAK**: This is the Det-Greedy algorithm of [24]. [24] do not provide an implementation of **GAK**, we implement **GAK** in Python3 with NumPy.
- **MC** : This first uses the algorithm of [42] to compute a subset S and then selects a ranking of these items that maximize the utility (in the simulations this amounts to sorting items by w_i). We used the implementation of [42]'s algorithm available at `https://github.com/AnayMehrotra/Noisy-Fair-Subset-Selection` and use Python3's in-built sorting function to generate the ranking. [42]'s algorithm takes P and parameters U specifying upper bound constraints as input.
- **Uncons**: This is the baseline that outputs the ranking with the maximum utility. In the simulation, this amounts to sorting all items in decreasing order of w_i and outputting the ranking with the first n items (in that order). We implement **Uncons** in Python3 with NumPy.

Computational resources used. All simulations were run on a `t3.xlarge` instance with 4 vCPUs and 16Gb RAM, on Amazon's Elastic Compute Cloud (EC2).

B.2 Pre-processing details of the simulation with image data

In this section, we present additional preprocessing details to estimate $\widehat{P}$ in the simulation with the Occupations dataset presented in Section 4.

Estimating $\widehat{P}$. We begin by removing all images with gender label NA; this leaves 5,825 images (out of 9600). On the remaining images, we use an off-the-shelf face-detector [1] to extract the faces of the people from the images and remove all images where the face-detector did not detect a face; this leaves 4,494 the images. We use a CNN-based gender classifier [52] on the detected faces to predict the apparent gender of the depicted individuals. For each image i, the classifier outputs a gender (coded as male and female) and an uncalibrated confidence score $c_i \in [0,1]$. We take the set of uncalibrated confidence scores $\{c_i \in [0,1]\}_i$ and calibrate them by first binning them, then computing the distribution of gender labels (provided in the dataset) for each bin. For each image i, we set $\widehat{P}_{i1}$ (respectively $\widehat{P}_{i2}$) equal to the fraction of images in the same bin as i whose gender label is female (respectively male). We perform this calibration once and on all occupations and, then, use it for a subset of occupations.

B.3 Additional discussion and plots for simulations

Illustrating the fairness vs. utility trade-off. In our empirical results, we use fairness metrics such as weighted risk-difference (Section 4) and weighted selection-lift (Supplementary Material B.4) to measure the algorithms' *achieved* fairness. We do not use the parameter ϕ to measure fairness because the output of algorithms may have lower fairness than specified by ϕ. Figures 2, 4 and 6 plot utility vs. weighted risk-difference and Figures 7(b), 8(b) and 9(b) plot utility vs. weighted selection-lift (SL) for the simulations in Section 4. They show that **NResilient** better or similar (up to standard errors) achieved fairness vs utility trade-off compared to baselines. For example, in Figure 8(b), to achieve SL$=0.55$ use Figure 8(a) to choose $\phi = 1.19$ for **NResilient** and $\phi = 1.15$ for **CSV** or **SJ**. For these values of ϕ, **NResilient** has 2% higher utility than **CSV** and **SJ**.

Comparison to baseline which has access to *accurate* protected attributes. Let **Clean-Fair** be the algorithm that, given utilities and accurate protected attributes, outputs the ranking with the maximum utility subject to satisfying equal representation constraint. Note that **Clean-Fair** can only be run in the ideal scenario where one has access to accurate protected attributes. We repeated the simulations in Section 4 and, for each of them, also measured the utility and fairness of **Clean-Fair**. We observe that the rankings output by **Clean-Fair** have a RDclose to 1 (>0.99), this is expected because **Clean-Fair** has access to the clean protected attributes. We observe that the ranking output by **NResilient** (for any parameter $0 \leq \phi \leq 1$, specifying the fairness constraints for **NResilient**) has a utility that is at most 2%, 10%, and 4% smaller than that the ranking output by **Clean-Fair**.

RD **of Uncons.** **Uncons**'s RDand utility does not vary with ϕ because it does not take ϕ as input. Note that, **Uncons** also does not take the protected groups or P as input.

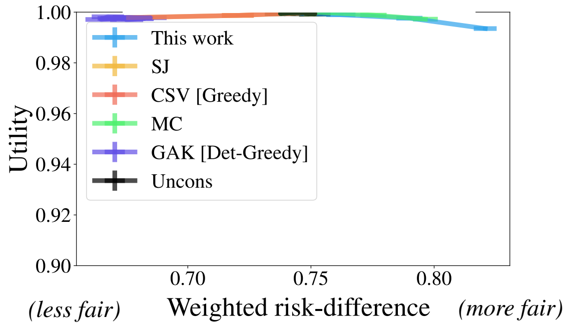


Figure 4: *Synthetic Data: Nonuniform Error Rate.* This simulation considers synthetic data where imputed socially-salient attributes have a higher false-discovery rate for one group compared to the other. We vary the fairness constraint from ϕ from 2 (less fair) to 1 (more fair) and observe the weighted risk-difference (weighted risk-difference) of different algorithms. The y-axis plots utility and x-axis shows weighted risk-difference (*Note that the values decrease toward the right*). Error-bars denote the error of the mean.

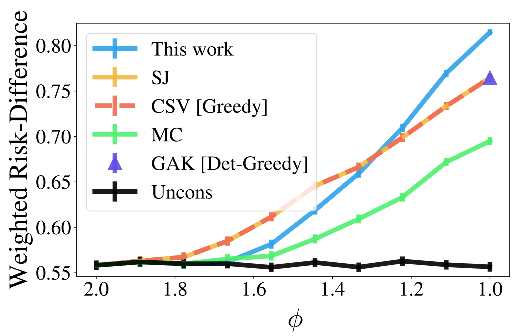


Figure 5: *Real-world image data.* This simulation considers images-search results which are known to overrepresent the stereotypical gender [34]. Given relevant *non-gender labeled* images and their utilities, our goal is to generate a high-utility gender-balanced ranking. We estimate P using an off-the-shelf ML-classifier and vary ϕ from $p = 2$ (less fair) to 1 (more fair). In the first subfigure, the y-axis plots weighted risk-difference and x-axis shows ϕ (*Note that the values decrease toward the right*). Error bars show the error of the mean.

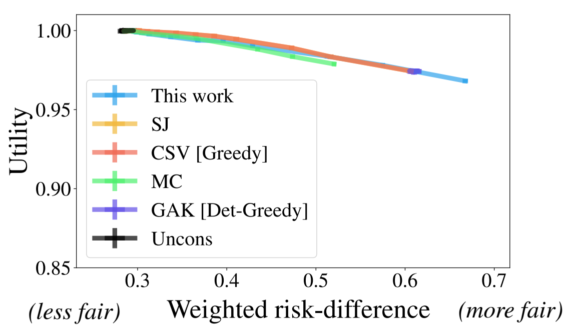


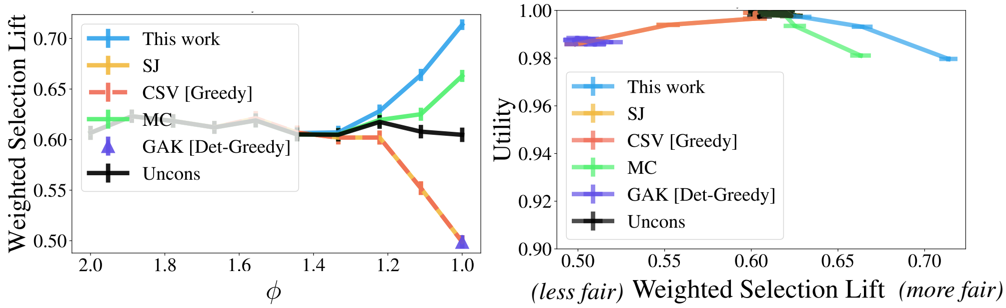
Figure 6: *Real-World Name Data: Intersectional Attributes.* This simulation considers two socially-salient attributes, gender and race, and our goal is to ensure equal representation across the four *intersectional* socially-salient groups (non-White non-men, White non-men, non-White men, and White men). We estimate P from the full names using public APIs and libraries. We vary ϕ from $p = 4$ (less fair) to 1 (more fair) and observe weighted risk-difference of all algorithms. The y-axis plots utility and x-axis shows weighted risk-difference (*Note that the values decrease toward the right*). Error bars represent the error of the mean.

B.4 Additional empirical results with weighted selection-lift

In this section, we present our empirical results with the weighted selection-lift fairness metric (Figures 7 to 9). Weighted selection-lift is a position-weighted version of the standard selection-difference metric. Like weighted risk-difference, it also measures the extent to which a ranking violates equal representation. The weighted selection-lift of a ranking R is:

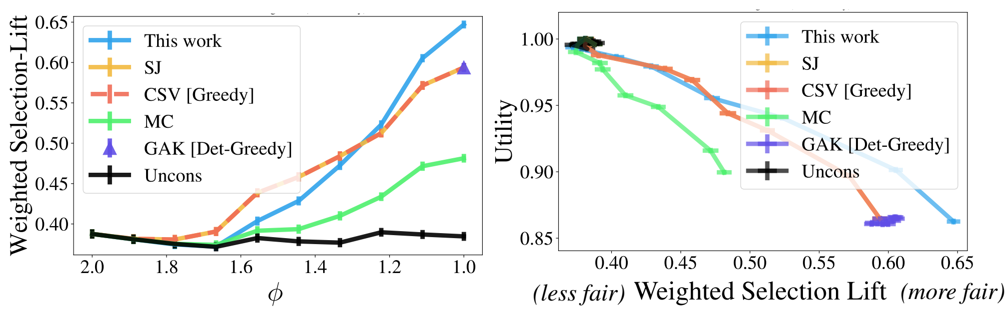
$$\frac{1}{Z}\sum_{k=5,10,\ldots}\frac{1}{\log k}\min_{\ell,q\in[p]}\left|\frac{\sum_{i\in G_\ell,\, j\in[k]} R_{ij}}{\sum_{i\in G_q,\, j\in[k]} R_{ij}}\right|,$$

Where G denotes the ground-truth protected groups and Z is a constant so that RD has range $[0,1]$. Here, a value of 1 is most fair and 0 is least fair.



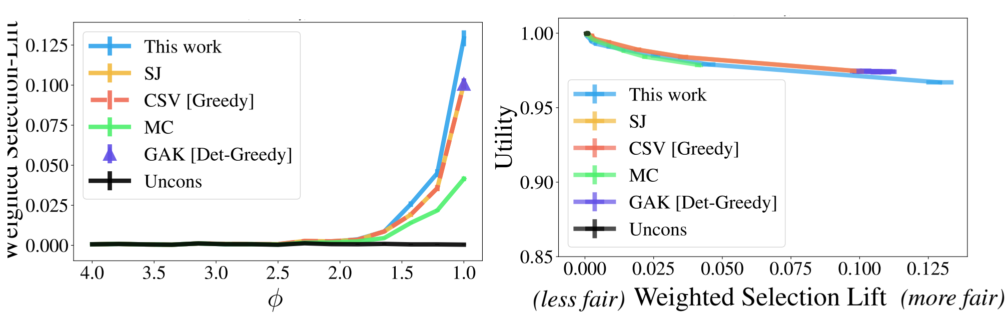
(a) (b)

Figure 7: *Synthetic Data (Weighted Selection Lift): Nonuniform Error Rate.* This simulation considers synthetic data where imputed socially-salient attributes have a higher false-discovery rate for one group compared to the other. We vary the fairness constraint from ϕ from 2 (less fair) to 1 (more fair) and observe the weighted risk-difference (weighted risk-difference) of different algorithms. In the first sub-figure, the y-axis plots weighted selection-lift and x-axis shows ϕ. In the second sub-figure, the y-axis plots utility and x-axis shows weighted selection-lift. Error bars represent the error of the mean.



(a) (b)

Figure 8: *Real-world image data.* This simulation considers images-search results which are known to overrepresent the stereotypical gender [34]. Given relevant *non-gender labeled* images and their utilities, our goal is to generate a high-utility gender-balanced ranking. We estimate P using an off-the-shelf ML-classifier and vary ϕ from $p = 2$ (less fair) to 1 (more fair). In the first sub-figure, the y-axis plots weighted selection-lift and x-axis shows ϕ. In the second sub-figure, the y-axis plots utility and x-axis shows weighted selection-lift. Error bars represent the error of the mean.



(a) (b)

Figure 9: *Real-World Name Data: Intersectional Attributes.* This simulation considers two socially-salient attributes, gender and race, and our goal is to ensure equal representation across the four *intersectional* socially-salient groups (non-White non-men, White non-men, non-White men, and White men). We estimate P from the full names using public APIs and libraries. We vary ϕ from $p = 4$ (less fair) to 1 (more fair) and observe RD of all algorithms. In the first sub-figure, the y-axis plots weighted selection-lift and x-axis shows ϕ. In the second sub-figure, the y-axis plots utility and x-axis shows weighted selection-lift. Error bars represent the error of the mean.

B.5 Additional empirical results varying noise

In this section, we present a simulation which uses the randomized response mechanism to generate noisy protected attributes and compares the performance of algorithms at varying noise levels.

Data. We use the Occupation images data [14]. We refer the reader to Section 4 for a discussion of the data.

Setup. We fix equal representation constraints ($\phi = 1$) and consider the same protected groups as the simulation with the same data in Section 4. We vary the noise level $0 \leq \eta \leq \frac{1}{2}$. For each η, we construct noisy attributes by mislabeling true protected attribute with probability η. Here, P is specified by η as explained in Remark A.1. Specifically, if N_1 and N_2 be the noisy versions of true protected groups G_1 and G_2 (corresponding to the "flipped" protected attributes), then we set: For each item $i \in N_1$,

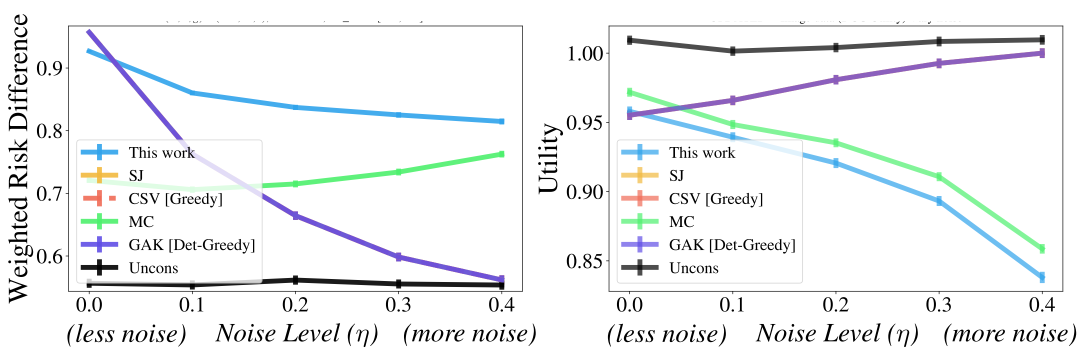
$$\widehat{P}_{i1} = (1-\eta) \cdot \frac{|G_1|}{|N_1|} \quad \text{and} \quad \widehat{P}_{i2} = 1 - \widehat{P}_{i1}.$$

For items in N_2, replace $\widehat{P}_{i1}$, $\widehat{P}_{i2}$, G_1, and N_1 with $\widehat{P}_{i2}$, $\widehat{P}_{i1}$, G_2, and N_2. We do not have access to G_1 (and, hence, $|G_1|$), and in the above expression we estimate $|G_1|$ by $\alpha_1 := \frac{(1-\eta)\cdot}{1-2\eta} \cdot ((1-\eta)\,|N_1| - \eta\,|N_2|)$. This is because α_1 can be shown to be concentrated around $|G_1|$.

Like the simulations in Section 4, **CSV**, **GAK**, and **SJ** are given the noisy attributes (as they require) and **NResilient** and **MC** are given $\widehat{P}$ (computed above).

Observations. See Figure 10 for RD and utilities (NDGC) averaged over 100 iterations. We observe that for each $\eta \geq 0.1$, **NResilient** RD is >6.8% better than any baseline (Figure 10(a)) and its utility is <3% smaller than the baseline (**CSV**) with best RD (Figure 10(b)). At $\eta = 0$, **NResilient** 3.3% lower RD than **CSV**, **GAK**, and **SJ** and the same utility as them.

Note that in Figures 10(a) and 10(b) the plots of **CSV**, **GAK**, and **SJ** overlap. This is consistent with the other simulations where **CSV**, **GAK**, and **SJ** have the same RD and utility at $\phi = 1$.



(a) (b)

Figure 10: *Simulation varying the amount of noise.* In this simulation, we use the Occupation's images data [14] and generate noisy protected attributes using the randomized response mechanism, with parameter η. We vary the amount of noise added from $\eta = 0$ (no noise) to $\eta = 0.4$ (large noise) and compare the performance of different algorithms. The y-axis plots RD and x-axis plots η. We present the key observations in the paragraph above the figure. Error-bars denote the error of the mean.

C Using existing fair-ranking algorithms with rounding is insufficient

Since existing fair-ranking algorithms require access to protected attributes, one way to use them under the above model is to imputed groups $\widehat{G}_1, \ldots, \widehat{G}_p$ using the specified probabilities. Then run these algorithms w.r.t. the imputed groups. To see an illustration, consider two groups G_1 and G_2. A natural imputation strategy is to use the Bayes optimal classifier, which assigns item i to $\widehat{G}_1$ iff $P_{i1} > 0.5$ and has the lowest expected imputation error. This may be reasonable when the imputation error is negligible. However, on exploring this strategy with non-negligible imputation error, we find that the output rankings can violate equal representation significantly (see Proposition C.1). To gain some intuition consider an extreme case where all items in some set S, of size n, have $P_{i1} = 0.51$. The Bayes classifier assigns all items in S to $\widehat{G}_1$, i.e., $|S \cap \widehat{G}_1| = |S|$. However, with high probability,

$$|S \cap G_1| \approx 0.51\,|S|\,.$$

Since $|S \cap G_1|$ and $|S \cap \widehat{G}_1|$ are far, a ranking that selects n items from S and satisfies the constraints for $\widehat{G}_1$ and $\widehat{G}_2$ but violate constraints with respect to the true groups. Proposition C.1 gives an example where this occurs.

Another imputation strategy, is independent rounding: it assigns each item i to $\widehat{G}_1$ with probability P_{i1} and otherwise to $\widehat{G}_1$. This addresses the issue with Bayes imputation, because, it has property that for any set T of size n, $|T \cap G_1|$ are $|T \cap \widehat{G}_1|$ close with probability $1 - e^{\Theta(n)}$. However, when $m \gg n$, there are

$$\binom{m}{n} \gg e^n$$

sets of size n, and hence, with high probability, there exists a set S of size n for which $|S \cap \widehat{G}_1|$ and $|S \cap G_1|$ are arbitrarily far. In this case also, existing fair-ranking algorithms can output rankings which violate equal representation significantly. Proposition C.2 gives an example where this occurs.

Proposition C.1 (**Imputing protected groups using the Bayes optimal classifier is not sufficient**). *Let R be any optimal solution to* (2) *with protected groups imputed using the Bayes optimal classifier for given p. There exists a matrix $P \in [0,1]^{m \times 2}$ such that R does not satisfy the (ε, δ)-equal representation constraint*

$$\text{for any} \quad \delta < \frac{1}{2} \quad \text{and} \quad \varepsilon \text{ s.t. } \varepsilon_k < \frac{1}{20} \text{ for some } k \geq 2.$$

Proposition C.2. *Let R be a random variable denoting the optimal solution to the fair-ranking problem (Program* (2)*) for protected groups imputed using independent rounding with given $P \in [0,1]^{m \times 2}$. For every $\beta > 0$, there exists sufficiently large n and m and a matrix $P \in [0,1]^{m \times 2}$, such that, with probability at least $1 - \beta$ R does not satisfy the (ε, δ)-equal representation constraint*

$$\text{for any} \quad \delta < 1 - \beta \quad \text{and} \quad \varepsilon \in (0,1)^n.$$

C.1 Proofs of Proposition C.1 and Proposition C.2

C.1.1 Proof of Proposition C.1

Proof of Proposition C.1. Pick any even $n \in \mathbb{N}$. Let $m \coloneqq \frac{3n}{2}$. Let $\beta > 0$ be a small constant that we will fix later. We will divide the items into the following three types:

- Type A: For each $1 \leq i \leq \frac{n}{2}$ and $1 \leq j \leq n$,

$$P_{i1} \coloneqq 0 = 1 - P_{i2} \text{ and } W_{ij} \coloneqq 1.$$

- Type B: For each $\frac{n}{2} + 1 \leq i \leq n$ and $1 \leq j \leq n$,

$$P_{i1} \coloneqq \frac{1}{2} + \beta = 1 - P_{i2} \text{ and } W_{ij} \coloneqq 1.$$

- Type C: For each $n + 1 \leq i \leq \frac{3n}{2}$ and $1 \leq j \leq n$,

$$P_{i1} \coloneqq 1 = 1 - P_{i2} \text{ and } W_{ij} \coloneqq 0.$$

Let $\widehat{G}_1$ and $\widehat{G}_2$ be the groups imputed using maximum likelihood rounding. By construction, $\widehat{G}_1$ contains all items of Types A and B and no items of Type C, whereas $\widehat{G}_2$ contains all items of Type C and no items of Types A and B.

Let R be an optimal solution of Program (2) with parameters $G_1 = \widehat{G}_1$ and $G_2 = \widehat{G}_2$. Since $W_{ij} \le 1$ for all $i \in [m], j \in [n]$, $\langle R, W \rangle \le n$. Because R satisfies the equal representation constraints for two disjoint groups, for any even $k \in [n]$, R places exactly $\frac{k}{2}$ items of Type A and $\frac{k}{2}$ items of Type B in the top k positions. From $\widehat{G}_1$, R only places items of Type A: If R picks no items of Type C, then $\langle R, W \rangle = n$, whereas, if R picks one or more items of Type C, then $\langle R, W \rangle \le n-1$, which is a contradiction since there is a ranking with utility n that satisfies equal representation constraints (e.g., a ranking which places items of Type A and B in alternate positions).

Since all items of Type A are (always) in $\widehat{G}_2$, R places at least $\frac{k}{2}$ items from $\widehat{G}_2$ in the first k positions. We will show that with probability larger than $\frac{1}{2}$, at least $\frac{k}{20}$ of the $\frac{k}{2}$ items of Type B are in $\widehat{G}_2$. Thus, with probability larger than $\frac{1}{2}$, R places more than $\frac{k}{2} \cdot \frac{11}{10}$ items from $\widehat{G}_2$ in the top-k positions, and hence, R does not satisfy the (ε, δ)-equal representation constraint for any $\delta < \frac{1}{2}$ and $\varepsilon \in \left(0, \frac{1}{10}\right)^n$.

It remains to prove our claim. Select any $k \in \{2, 4, \ldots, n\}$. Let $i_1, i_2, \ldots, i_{k/2} \in [m]$ be the n items of Type B that R places in the first k positions. Let $Z_{i_j} \in \{0, 1\}$ be the indicator random variable that $i_j \in \widehat{G_2}$. Thus, $Z_{i_1}, \ldots, Z_{i_{k/2}}$ are independent random variables, such that, for $j \in [k]$, $\Pr[Z_{i_j}] = 1 - P_{i_j} = \frac{1}{2} - \beta$. It follows that $\mathbb{E}[\sum_{j=1}^{k/2} Z_{i_j}] = \frac{k}{2}\left(\frac{1}{2} - \beta\right)$ and $\mathrm{Var}[\sum_{j=1}^{k/2} Z_{i_j}] = \frac{k}{2}\left(\frac{1}{4} - \beta^2\right)$. Thus, using the Chebyshev's inequality on $\sum_{j=1}^{k/2} Z_{i_j}$,

$$\Pr\left[\left|\sum_{j=1}^{k/2} Z_{i_j} - \frac{k}{4}(1-2\beta)\right| > \frac{k}{8}\left(1 - 4\beta^2\right) \cdot \sqrt{2+\beta}\right] \le \frac{1}{2+\beta}.$$

Thus,

$$\Pr\left[\sum_{j=1}^{k/2} Z_{i_j} < \frac{k}{4}(1-2\beta) - \frac{k}{8}\left(1 - 4\beta^2\right) \cdot \sqrt{2+\beta}\right] \le \frac{1}{2+\beta}.$$

Since $\frac{k}{4}(1-2\beta) - \frac{k}{8}\left(1-4\beta^2\right) \cdot \sqrt{2+\beta} = k\left(\frac{1}{4} - \frac{\sqrt{2}}{8}\right) + k \cdot O(\beta)$, for a sufficiently small $\beta > 0$,

$$\frac{k}{4}(1-2\beta) - \frac{k}{8}\left(1-4\beta^2\right) \cdot \sqrt{2+\beta} > \frac{k}{20}.$$

Hence,

$$\Pr\left[\sum_{j=1}^{k/2} Z_{i_j} < \frac{k}{20}\right] \le \frac{1}{2+\beta} \overset{(\beta>0)}{<} \frac{1}{2}. \tag{15}$$

□

C.1.2 Proof of Proposition C.2

Proof of Proposition C.2. Let $\phi > 0$ be a small constant that we will fix later. We will divide the items into the following two types:

- Type A: For each item i of Type A

$$P_{i1} := \phi,\ P_{i2} := 1 - \phi \text{ and } W_{ij} := 1 \text{ for all } j \in [n].$$

- Type B: For each item i of Type B

$$P_{i1} := 1,\ P_{i2} := 0 \text{ and } W_{ij} := 0 \text{ for all } j \in [n].$$

- Type B: For each item i of Type C

$$P_{i1} := 0,\ P_{i2} := 1 \text{ and } W_{ij} := 0 \text{ for all } j \in [n].$$

Let there be $m_A := O\left(\log\left(\frac{n}{\beta}\right) \cdot \frac{n}{\log(\frac{1}{1-\phi})}\right)$ items of Type A, $m_B := n$ items of Type B, and $m_C := n$ items of Type C.

Note that a ranking which ranks items of Type B and Type C alternately, satisfies the equal representation constraints with probability 1. So in this instance, there exists a ranking which satisfies (δ, ε)-equal representation. However, we will show that R does not satisfy (δ, ε)-equal representation with probability at least $1 - \beta$.

Let $\widehat{G}_1$ and $\widehat{G}_2$ be the groups imputed by independent rounding. Let $\mathscr{E}$ be the event that $\widehat{G}_1$ contains at least n items of Type A and $\mathscr{F}$ be the event that $\widehat{G}_2$ contains at least n items of Type A. Both $\mathscr{E}$ and $\mathscr{F}$ occur with probability at most $O(\beta)$. To see this, divide the items of Type A into n groups of equal size. From each group, at least one item is selected in $\widehat{G}_1$ and $\widehat{G}_2$ with probabilities at least $1-(1-\phi)^{\frac{m_A}{n}}$ and $1-(\phi)^{\frac{m_A}{n}}$ respectively. Taking a union bound over all groups and substituting m_A, we get

$$\Pr[\mathscr{E}] \geq 1-\beta \quad \text{and} \Pr[\mathscr{F}] \geq 1-\beta.$$

Since only items of Type A have a nonzero contribution to the utility of a ranking and because there are at least n items of Type A in each imputed group, it follows that R only selects items of Type A. Now, the claim follows because, for small ϕ, most items of Type A belong to G_1.

Suppose $\mathscr{E}$ and $\mathscr{F}$ happen and, hence, R only selects items of Type A. Let Z_j be the indicator random variable that the item in the j-th position of R is in G_1. We have that $\Pr[Z_j] = \phi$. Therefore, $\mathrm{Var}[\sum_{j=1}^n Z_j] = n\phi(1-\phi)$. Thus, using the Chebyshev's inequality we have

$$\Pr\left[\left|\sum_{j=1}^n Z_j - n\phi\right| \geq \frac{n\varepsilon_n}{4}\right] \leq \frac{4n\phi(1-\phi)}{n^2\varepsilon_n^2}.$$

Hence, for $\phi = \Theta(\varepsilon_n^2\beta)$, we have that

$$\Pr\left[\sum_{j=1}^n Z_j \leq \frac{n\varepsilon_n}{2}\right] \geq 1-\beta.$$

The result follows since whenever $\sum_{j=1}^n Z_j \leq \frac{n\varepsilon}{2}$, R violates the equal representation constraint at the n-th position by a multiplicative factor larger than $1+\varepsilon_n$.

□

D Proofs of theoretical results

D.1 Proof of Lemma 5.3

In this section, we prove certain concentration inequalities which are used in the proof of Theorem 3.1. We divide the proof of Lemma 5.3 into two parts: Lemmas D.1 and D.6

For each item $i \in [m]$ and protected attribute $\ell \in [p]$, let $Z_{i\ell} \in \{0,1\}$ be the indicator random variable that the i-th item is in the ℓ-th protected group, i.e., if $i \in G_\ell$, then $Z_i = 1$, and other $Z_i = 0$. Using Definition 2.2, it follows that:

$$\forall i \in [m],\ \ell \in [p], \quad \Pr[Z_{i\ell}] = P_{i\ell}, \tag{16}$$
$$\forall i, j \in [m],\ \ell \in [p], \quad \text{s.t.}, i \neq j, \quad Z_{i\ell} \text{ and } Z_{j\ell} \text{ are independent.} \tag{17}$$

To simplify the notation, given a ranking $R \in \mathcal{R}$, a protected attribute $\ell \in [p]$, and a position $k \in [n]$, let $Z_\#(R,\ell,k) \in \mathbb{Z}$ be the random variable equal to the number of items from G_ℓ in the top k positions of R and let $P_\#(R,\ell,k) \in \mathbb{R}$ be the expectation of $Z_\#(R,\ell,k)$, i.e.,

$$Z_\#(R,\ell,k) := \sum_{i\in[m]}\sum_{j\in[k]} Z_{i\ell}R_{ij} \quad \text{and} \quad P_\#(R,\ell,k) := \mathbb{E}\left[Z_\#(R,\ell,k)\right].$$

Using Equation (16) and linearity of expectation it follows that

$$P_{\#}(R,\ell,k)=\sum_{i\in[m]}\sum_{j\in[k]}P_{i\ell}R_{ij}.$$

Lemma D.1. *For any position $k\in[n]$, attribute $\ell\in[p]$, parameters $\varepsilon\geq 0$ and $L\in\mathbb{R}$, and ranking $R\in\mathcal{R}$, where R is possibly a random variable and is independent of $\{Z_{i\ell}\}_{i,\ell}$, if $P_{\#}(R,\ell,k)\geq L$ then with probability at least $1-\exp\left(-\frac{L\varepsilon^2}{2(1-\varepsilon)}\right)$, it holds that $Z_{\#}(R,\ell,k)>L\,(1-\varepsilon)$.*

Proof. Since ℓ, k, and R are fixed, we use $Z_{\#}$ and $P_{\#}$ to denote $Z_{\#}(R,\ell,k)$ and $P_{\#}(R,\ell,k)$ respectively.

Since R and $\{Z_{i\ell}\}_{i,\ell}$ are independent, we can bound the required probability as follows

$$\begin{aligned}\Pr\left[Z_{\#}\leq L(1-\varepsilon)\right] &= \Pr\left[Z_{\#}\leq P_{\#}\cdot\left(1-\frac{P_{\#}-L(1-\varepsilon)}{P_{\#}}\right)\right]\\ &\leq \exp\left(-\frac{P_{\#}}{2}\cdot\left(\frac{P_{\#}-L(1-\varepsilon)}{P_{\#}}\right)^2\right) \qquad \text{(Chernoff's bound, see [45])}\\ &= \exp\left(-\frac{1}{2}\cdot\frac{\left(P_{\#}-L(1-\varepsilon)\right)^2}{P_{\#}}\right). \qquad (18)\end{aligned}$$

To bound the right-hand side of Equation (18), we will use the following fact.

Fact D.2. For all $L,\varepsilon>0$, $\frac{(x-L(1-\varepsilon))^2}{x}$ attains its minima at L over the domain $[L,\infty)$.

Since $P_{\#}\geq L$, from Fact D.2 it follows that the right-hand side of Equation (18) attains its maxima at $P_{\#}=L$. Substituting $P_{\#}=L$ in Equation (18), we get:

$$\Pr\left[Z_{\#}\leq L(1-\varepsilon)\right]\leq\exp\left(-\frac{1}{2}\cdot\frac{(L\varepsilon)^2}{L(1-\varepsilon)}\right)=\exp\left(\frac{-L\varepsilon^2}{2(1-\varepsilon)}\right).$$

□

Lemma D.3. *For any position $k\in[n]$, attribute $\ell\in[p]$, parameters $\varepsilon\geq 0$ and $U\in\mathbb{R}$, and ranking $R\in\mathcal{R}$, where R is possibly a random variable and is independent of $\{Z_{i\ell}\}_{i,\ell}$, if R satisfies that $P_{\#}(R,\ell,k)\leq U$ then with probability at least $1-\exp\left(-\frac{U\varepsilon^2}{2+\varepsilon}\right)$, it holds that $Z_{\#}(R,\ell,k)<(1+\varepsilon)\cdot U$.*

Proof. Since ℓ, k, and R are fixed, we use $Z_{\#}$ and $P_{\#}$ to denote $Z_{\#}(R,\ell,k)$ and $P_{\#}(R,\ell,k)$ respectively. Since R and $\{Z_{i\ell}\}_{i,\ell}$ are independent, we can bound the required probability as follows

$$\begin{aligned}\Pr\left[Z_{\#}\geq U(1+\varepsilon)\right] &= \Pr\left[Z_{\#}\leq P_{\#}\cdot\left(1+\frac{U(1+\varepsilon)-P_{\#}}{P_{\#}}\right)\right]\\ &\leq \exp\left(P_{\#}\cdot\left(\frac{U(1+\varepsilon)-P_{\#}}{P_{\#}}\right)^2\cdot\frac{1}{2+\frac{U(1+\varepsilon)-P_{\#}}{P_{\#}}}\right).\end{aligned}$$

Where we used the fact that: For any $\delta>0$ and independent 0/1 random variables $Y_1,Y_2,\ldots,Y_n$, $\Pr\left[\sum_i Y_i>(1+\delta)\mu\right]<\exp\left(\frac{\mu\delta^2}{2+\delta}\right)$, where $\mu:=\mathbb{E}[\sum_i Y_i]$ (see[45]). Simplifying the right-hand side of the above equation, we get:

$$\Pr\left[Z_{\#}\geq U(1+\varepsilon)\right] = \exp\left(-\frac{\left(U(1+\varepsilon)-P_{\#}\right)^2}{U(1+\varepsilon)+P_{\#}}\right). \qquad (19)$$

To bound the right-hand side of Equation (19), we will use the following fact.

Fact D.4. For all $U, \varepsilon > 0$, $\frac{(U(1+\varepsilon)-x)^2}{U(1+\varepsilon)+x}$ attains its minima at U over the domain $[0, U]$.

Since $P_\# \leq U$, from Fact D.4 it follows that the right-hand side of Equation (19) attains its maxima at $P_\# = U$. Substituting $P_\# = U$ in Equation (19), we get:

$$\Pr\left[Z_\# \geq U(1+\varepsilon)\right] \leq \exp\left(\frac{-U\varepsilon^2}{2+\varepsilon}\right). \tag{20}$$

□

D.2 Improved dependence of γ on δ

In this section, we show that given a constant $\psi > 0$, if U satisfies that

$$\forall \ell \in [p], \forall k \in [n], \quad U_{k\ell} \geq \psi k,$$

then we can improve the dependence of γ (from Equation (6)) on $\log \frac{2np}{\delta}$ and α. Concretely, Theorem 3.1 holds for the following γ:

$$\forall k \in [n], \quad \gamma_k \coloneqq \max_{\ell \in [p]} \sqrt{\frac{1}{2\psi} \cdot \log\left(\frac{2np}{\delta}\right) \cdot \frac{1}{U_{k\ell}}}. \tag{21}$$

The proof of this relies on analogous of Lemmas D.1 and D.3: Lemmas D.5 and D.6.

Lemma D.5. *For any position $k \in [n]$, attribute $\ell \in [p]$, parameter $\varepsilon \geq 0$, and lower bound constraint $L \in \mathbb{Z}_{\geq 0}^{n \times p}$, and ranking $x \in \mathcal{R}$, if x satisfies that $P_\#(R, \ell, k) \geq L$ then with probability at least $1 - \exp\left(-2L^2\varepsilon^2 k^{-1}\right)$, it holds that $Z_\#(R, \ell, k) > L(1-\varepsilon)$.*

Lemma D.6. *For any position $k \in [n]$, attribute $\ell \in [p]$, parameters $\varepsilon \geq 0$ and $U \in \mathbb{R}$, and ranking $R \in \mathcal{R}$, where R is possibly a random variable and is independent of $\{Z_{i\ell}\}_{i,\ell}$, if R satisfies that $P_\#(R, \ell, k) \leq U$ then with probability at least $1 - \exp\left(-\frac{2U^2\varepsilon^2}{k}\right)$, it holds that $Z_\#(R, \ell, k) < U(1+\varepsilon)$.*

To prove the improved dependence of γ, it suffices to prove Propositions 5.1 and 5.2. For the new value of γ, their proofs change as follows:

Proof of Proposition 5.1 The parameters in Equation (9) remain the same. Hence, following the same argument, Equation (10) holds. Now, we can prove Equation (12) as follows:

$$\begin{aligned}
\Pr\left[Z_\#(R, \ell, k) \geq U_{\ell k}(1+\phi\gamma_k)\right] &= \Pr\left[Z_\#(R, \ell, k) \geq U'(1+\zeta)\right] \\
&\qquad \text{(Using that } U'(1+\zeta) = U_{k\ell}(1+\phi\gamma_k)\text{)} \\
&\leq \exp\left(-\frac{2\left(U'\right)^2\zeta^2}{k}\right) \qquad \text{(Using Lemma D.6)} \\
&= \exp\left(-\frac{2(1-\phi)^2 U_{\ell k}^2 \gamma_k^2}{k}\right) \qquad \text{(Using Equation (9))} \\
&\leq \exp\left(-2\psi(1-\phi)^2 U_{\ell k}\gamma_k^2\right) \qquad \text{(Using that } U_{k\ell} \geq \psi k\text{)} \\
&\leq \frac{\delta}{2np}. \qquad \text{(Using Equation (21))}
\end{aligned} \tag{22}$$

Proposition 5.1 follows by replacing Equation (12) by Equation (22) in the rest of its proof.

Proof of Proposition 5.2 The parameters in Equation (13) remain the same. Now, we can prove $\Pr\left[Z_{\#}(R',k,\ell) \le U_{k\ell}\right] < 1-\delta$ as follows:

$$\begin{aligned}
\Pr\left[Z_{\#}(R',k,\ell) \le U_{k\ell}\right] &= \Pr\left[Z_{\#}(R',k,\ell) \le L' \cdot (1-\zeta)\right] \\
&\qquad \text{(Using that } L'(1-\zeta) = U_{k\ell}(1+b\gamma_k)\text{)} \\
&\le \exp\left(-\frac{2\,(L')^2\,\zeta^2}{k}\right) && \text{(Using Lemma D.5)} \\
&= \exp\left(-\frac{2(\phi-b)^2\gamma_k^2 U_{k\ell}^2}{k}\right) && \text{(Using Equation (13))} \\
&\le \exp\left(-2\psi(\phi-b)^2\gamma_k^2 U_{k\ell}\right) && \text{(Using that } U_{k\ell} \ge \psi k\text{)} \\
&< \frac{\delta}{2np} && \text{(Using Equation (21) and Equation (13))} \quad (23) \\
&< 1-\delta. && \text{(Using that } \delta < \tfrac{1}{2} \text{ and } n \ge 1\text{)} \quad (24)
\end{aligned}$$

The rest of the proof is identical.

Proof of Lemma D.5. First, note that since x is not a function of the outcomes of the random variables $Z_{i\ell}$, x is independent of the random variables $\{Z_{i\ell}\}_{i,\ell}$. Since ℓ, k, and x are fixed, we use $Z_{\#}$ and $P_{\#}$ to denote $Z_{\#}(R,\ell,k)$ and $P_{\#}(R,\ell,k)$ respectively. Now, we can bound the required probability as follows

$$\begin{aligned}
\Pr\left[Z_{\#} \le L(1-\varepsilon)\right] = &\ \Pr\left[Z_{\#} \le P_{\#} \cdot \left(1 - \frac{P_{\#} - L(1-\varepsilon)}{P_{\#}}\right)\right] \\
\le &\ \exp\left(-\frac{2}{k} \cdot P_{\#}^2 \cdot \left(\frac{P_{\#} - L(1-\varepsilon)}{P_{\#}}\right)^2\right)
\end{aligned}$$

(Where we used the fact that: For any $\delta > 0$ and bounded random variables $Y_1, Y_2, \ldots, Y_n \in [0,1]$, $\Pr\left[\sum_i Y_i < (1-\delta)\mu\right] < \exp\left(-2\mu^2\delta^2 n^{-1}\right)$, where $\mu \coloneqq \mathbb{E}[\sum_i Y_i]$)

$$\begin{aligned}
&= \exp\left(-\frac{2}{k} \cdot \left(P_{\#} - L(1-\varepsilon)\right)^2\right) \\
&\le \exp\left(-2L^2\varepsilon^2 k^{-1}\right).
\end{aligned}$$

□

Proof of Lemma D.6. Since ℓ, k, and R are fixed, we use $Z_{\#}$ and $P_{\#}$ to denote $Z_{\#}(R,\ell,k)$ and $P_{\#}(R,\ell,k)$ respectively. Since R and $\{Z_{i\ell}\}_{i,\ell}$ are independent, we can bound the required probability as follows

$$\begin{aligned}
\Pr\left[Z_{\#} \ge U(1+\varepsilon)\right] &= \Pr\left[Z_{\#} \le P_{\#} \cdot \left(1 + \frac{U(1+\varepsilon) - P_{\#}}{P_{\#}}\right)\right] \\
&\le \exp\left(-\frac{2}{k} \cdot P_{\#}^2 \cdot \left(\frac{U(1+\varepsilon) - P_{\#}}{P_{\#}}\right)^2\right).
\end{aligned}$$

Where we used the fact that: For any $\delta > 0$ and bounded random variables $Y_1, Y_2, \ldots, Y_n \in [0,1]$, $\Pr\left[\sum_i Y_i > (1+\delta)\mu\right] < \exp\left(-2\mu^2\delta^2 n^{-1}\right)$, where $\mu \coloneqq \mathbb{E}[\sum_i Y_i]$ ([45]). Simplifying the right-hand side of the above equation, we get

$$\begin{aligned}
\Pr\left[Z_{\#} \ge U(1+\varepsilon)\right] &\le \exp\left(-\frac{2}{k}\left(U(1+\varepsilon) - P_{\#}\right)^2\right) \\
&\le \exp\left(-\frac{2U^2\varepsilon^2}{k}\right). && \text{(Using that } P_{\#} \le U\text{)}
\end{aligned}$$

□

D.3 Proof of Theorem 3.2

We consider the family of matrices $U \in \mathbb{R}^{n\times p}$ that satisfy the following condition: For each position $k \in [n]$, there exists an attribute ℓ such that

$$U_{k\ell} \le \frac{k}{4}.$$

Notably, equal representation constraints satisfy this condition for any $p \ge 4$. We will use Fact D.7 to prove Theorem 3.2.

Fact D.7 (Theorem 2 in [46]). For all $p \in (0, \frac{1}{4}]$, $0 \le \varepsilon \le \frac{1}{p}(1-p)$, and $s \in \mathbb{N}$ independent 0/1 random variables $Z_1, Z_2, \ldots, Z_s \in \{0,1\}$, such that for all $i \in [s]$, $\Pr[Z_i = 1] = p$,

$$\Pr\left[\sum\nolimits_{i\in[s]} Z_i \ge (1+\varepsilon)ps\right] \ge \frac{1}{4}\exp\left(-2\varepsilon^2 ps\right).$$

Proof of Theorem 3.2. Fix the k to the value specified in the theorem. Let $\ell \in [n]$, be any attribute such that $U_{k\ell} \le \frac{k}{4}$. Such a ℓ exists because of the family of constraints we chose. Without loss of generality suppose $\ell \neq 1$. Fix any $n, m \ge k$. For each item $i \in [m]$, set

$$P_{i\ell} \coloneqq \frac{U_{k\ell}}{k} \quad \text{and} \quad P_{i1} \coloneqq 1 - \frac{U_{k1}}{k} \tag{25}$$

Further, for all $k \in [p]$, $k \neq p$ and $k \neq 1$, let $P_{ik} \coloneqq 0$.

Suppose, toward a contradiction, that there is a ranking $R \in \mathcal{R}$ that satisfies the (ε, δ)-constraint. R must satisfy the following equation:

$$\Pr\left[Z_{\#}(R,k,\ell) \le U_{k\ell}\cdot(1+\varepsilon_k)\right] \ge 1-\delta. \tag{26}$$

For each position $j \in [n]$, let $Z_j \in \{0,1\}$ be the indicator random variable that the item placed in the j-th place in the ranking R is in the protected group G_ℓ. From Equation (25) and Definition 2.2, it follows that:

$$\forall j \in [n], \quad \Pr[Z_j] = \frac{U_{k\ell}}{k} \tag{27}$$

$$\forall u, v \in [n], \quad \text{s.t.}, u \neq v, \quad Z_u \text{ and } Z_v \text{ are independent.} \tag{28}$$

Using linearity of expectation and Equation (27), we get that:

$$\Pr\left[Z_{\#}(R,k,\ell) \le (1+\varepsilon_k)\cdot U_{k\ell}\right] \;=\; \Pr\left[\sum\nolimits_{j\in[k]} Z_j \ge (1+\varepsilon_k)\cdot \mathbb{E}\left[\sum\nolimits_{j=1}^{k} Z_j\right]\right]. \tag{29}$$

Since $0 \le \varepsilon_k \le 1$ and $\frac{1}{k}\mathbb{E}\left[\sum_{j=1}^{k} Z_j\right] \le \frac{1}{4}$, we can use Fact D.7 with $\varepsilon \coloneqq \varepsilon_k$, $p \coloneqq \frac{1}{k}\mathbb{E}\left[\sum_{j=1}^{k} Z_j\right] \le \frac{1}{4}$, $s \coloneqq k$, and for all $j \in [n]$, $Z_j = Z_j$. Using this, we get that

$$\begin{aligned}\Pr\left[\sum\nolimits_{j\in[k]} Z_j \ge (1+\varepsilon_k)\cdot \mathbb{E}\left[\sum\nolimits_{j=1}^{k} Z_j\right]\right] \;&\le\; 1 - \frac{1}{4}\exp\left(-2\varepsilon_k^2\cdot \mathbb{E}\left[\sum\nolimits_{j=1}^{k} Z_j\right]\right)\\ &\le\; 1-\frac{1}{4}\exp\left(-2\varepsilon_k^2 U_{k\ell}\right)\end{aligned}$$

(Using Equation (27)) (30)

Chaining Equations (26), (29), and (30), we get that

$$1 - \frac{1}{4}\exp\left(-2\varepsilon_k^2 U_{k\ell}\right) \ge 1-\delta.$$

Hence,

$$\varepsilon_k \ge \sqrt{\frac{1}{2U_{k\ell}}\log\frac{1}{4\delta}}.$$

This is a contradiction since ε_k is specified to be less than $\sqrt{\frac{1}{2U_{k\ell}}\log\frac{1}{4\delta}}$. Thus, no ranking R satisfies the (ε, δ)-constraint for any U in the chosen family chosen. □

D.4 Proof of Theorem 3.3

In this section, we prove Theorem 3.3. Our algorithm uses the rounding algorithm of [16] as a subroutine. [16]'s algorithm satisfies the following guarantees.

Theorem D.8 (**Theorem 1.1 from [16]**). *Let $P \subseteq [0,1]^N$ be either a matroid intersection polytope or a (non-bipartite graph) matching polytope. For any fixed $0 < \alpha \leq \frac{1}{2}$, there is an efficient randomized rounding procedure, such that given a (fractional) point $R_F \in P$, it outputs a random feasible solution R corresponding to a (integer) vertex of P such that $\mathbb{E}[1_R] = (1-\alpha)\cdot R_F$. In addition, for any linear function $w(R) \coloneqq \sum_{i\in R} w_i$, where $w_i \in [0,1]$ it holds that*

1. *for any $\delta \in [0,1]$ and $\mu \leq \mathbb{E}[1_R]$,* $\Pr[w(R) \leq (1-\delta)\mu] \leq \exp\left(-\frac{1}{20}\cdot \mu\alpha\delta^2\right)$,
2. *for any $\delta \in [0,1]$ and $\mu \geq \mathbb{E}[1_R]$,* $\Pr[w(R) \geq (1-\delta)\mu] \leq \exp\left(-\frac{1}{20}\cdot \mu\alpha\delta^2\right)$,
3. *for any $\Delta \geq 1$ and $\mu \geq \mathbb{E}[1_R]$,* $\Pr[w(R) \geq \mu(1+\Delta)] \leq \exp\left(-\frac{1}{20}\cdot \mu\alpha(2\Delta-1)\right)$.

The algorithm runs in time polynomial in the size of the ground set, N, and $\frac{1}{\alpha}$, and makes at most $\mathrm{poly}(N,d)$ *calls to the independence oracles for the underlying matroids.*

We claim that the following algorithm satisfies the claim in Theorem 3.3

Algorithm 1 Algorithm from Theorem 3.3

Input: Matrices $P \in [0,1]^{m\times p}$, $W \in \mathbb{R}_{\geq 0}^{m\times n}$, $U \in \mathbb{R}^{n\times p}$

Parameters: Constant $d > 2$ and $c > 1$, a failure probability $\delta \in (0,1]$, and for each $k \in [n]$, a relaxation parameter

$$\gamma_k \coloneqq 12\cdot\log\left(\frac{2np}{\delta}\right)\cdot \max_{\ell\in[p]}\sqrt{\frac{1}{U_{k\ell}}}.$$

1. **Initialize** $R_F \leftarrow$ Solve the linear-programming relaxation of Program (7) with the specified inputs
2. **Round** $R \leftarrow$ Run [16]'s rounding algorithm with input $\alpha \coloneqq \frac{1}{d}$ and $P \coloneqq \mathrm{conv}(\mathcal{R})$
3. **Return** R

For each item $i \in [m]$ and protected attribute $\ell \in [p]$, let $Z_{i\ell} \in \{0,1\}$ be the indicator random variable that the i-th item is in the ℓ-th protected group, i.e., if $i \in G_\ell$, then $Z_i = 1$, and other $Z_i = 0$. Using Definition 2.2, it follows that:

$$\forall i \in [m],\ \ell \in [p], \quad \Pr[Z_{i\ell}] = P_{i\ell}, \tag{31}$$
$$\forall i,j \in [m],\ \ell \in [p], \quad \text{s.t.}, i \neq j, \quad Z_{i\ell} \text{ and } Z_{j\ell} \text{ are independent.} \tag{32}$$

To simplify the notation, given a ranking $R \in \mathcal{R}$, a protected attribute $\ell \in [p]$, and a position $k \in [n]$, let $Z_\#(R,\ell,k) \in \mathbb{Z}$ be the random variable equal to the number of items from G_ℓ in the top k positions of R and let $P_\#(R,\ell,k) \in \mathbb{R}$ be the expectation of $Z_\#(R,\ell,k)$, i.e.,

$$Z_\#(R,\ell,k) \coloneqq \sum_{i\in[m]}\sum_{j\in[k]} Z_{i\ell}R_{ij} \quad \text{and} \quad P_\#(R,\ell,k) \coloneqq \mathbb{E}\left[Z_\#(R,\ell,k)\right].$$

Using Equation (31) and linearity of expectation it follows that

$$P_\#(R,\ell,k) = \sum_{i\in[m]}\sum_{j\in[k]} P_{i\ell}R_{ij}.$$

Proof.

Running time. The Step 1 of Algorithm 1 runs in polynomial time when implemented with any polynomial-time linear programming solver. Observe that $\mathcal{R}$ corresponds to the bipartite matching polytope, whose bi-partitions have size n and m respectively. Since the bipartite matching polytope is a matroid intersection polytope, we can use Theorem D.8. The independence oracle for this polytope can be implemented in poly(m) time, e.g., using the Birkhoff–von Neumann theorem. Finally, since $\alpha = \frac{1}{d}$ and $N = O(m^2)$, it follows that Step 2 of Algorithm 1 runs in polynomial time in d and the bit complexity of the input (which is at least m).

Let
$$\phi \coloneqq \frac{2\sqrt{c}-1}{2\sqrt{c}}.$$
Let R_F and R be the rankings from Steps 1 and 2 of Algorithm 1. From Theorem D.8, we have that $\mathbb{E}[1_R] = (1-\alpha)\cdot R_F$. Hence, for any weights $V \in \mathbb{R}^{n\times m}$, it holds that
$$\mathbb{E}\left[\langle R, V\rangle\right] = (1-\alpha)\cdot \langle R_F, V\rangle\,. \tag{33}$$
Fix any position $k \in [n]$ and group $\ell \in [p]$. Since ℓ, k, and R are fixed, we use $Z_{\#}(R)$ and $Z_{\#}(R')$ and $P_{\#}$ to denote $Z_{\#}(R,\ell,k)$ and $P_{\#}(R,\ell,k)$ respectively.

Utility guarantee. Let $R^\star$ be the solution of Program (7) for $c = d$. Let $V \coloneqq \langle W, R^\star\rangle$. Let $0 \le \Delta \le V$ be a parameter. Since R_F is a solution of the LP-relaxation of Program (7) and $R^\star$ is a solution of Program (7), R_F's utility is at least as large as the utility of $R^\star$. From this it follows that
$$\Pr\left[\langle W, R\rangle \le \langle W, R^\star\rangle\cdot(1-\alpha) - \Delta\right] \le \Pr\left[\langle W, R\rangle \le \langle W, R_F\rangle\cdot(1-\alpha) - \Delta\right]. \tag{34}$$
Since $W \in [0,1]^{m\times n}$, we can use Theorem D.8 with $a = W$. Using this we get can upper bound the RHS of the above equation.
$$\begin{aligned}\Pr\left[\langle W, R\rangle \le \langle W, R_F\rangle\cdot(1-\alpha) - \Delta\right] &= \Pr\left[\langle W, R\rangle \le \mathbb{E}\left[\langle W, R\rangle\right] - \Delta\right] \quad \text{(Using Equation (33))}\\ &\le \exp\left(-\frac{\alpha}{20}\cdot\frac{\Delta^2}{\langle W, R_F\rangle\cdot(1-\alpha)}\right)\end{aligned}$$
Let $\Delta \coloneqq \sqrt{\frac{20}{\alpha}\cdot\langle W, R_F\rangle\cdot(1-\alpha)\cdot\log\left(\frac{2np}{\delta}\right)}$. Substituting the value of Δ in the above equation, we have:
$$\Pr\left[\langle W, R\rangle \le \mathbb{E}\left[\langle W, R\rangle\right] - \Delta\right] \le \frac{\delta}{2np}. \tag{35}$$
Chaining the inequalities in Equations (34) and (35)
$$\Pr\left[\langle W, R\rangle \le \langle W, R^\star\rangle\cdot(1-\alpha) - \Delta\right] \le \frac{\delta}{2n}.$$
Since each entry of W is at most 1 and $\sum_{i,j}(R_F)_{ij} = n$, it follows that $\langle W, R_F\rangle \le n$. Using this and that $\alpha = \frac{1}{d}$,
$$\Delta = O\left(\sqrt{dn\cdot\log\frac{2np}{\delta}}\right).$$
Thus, the utility guarantee follows.

Fairness guarantee. Since R_F is feasible for the LP-relaxation of Program (7), it holds that
$$P_{\#}(R_F) \le U_{k\ell}(1+\phi\gamma_k). \tag{36}$$
Let $\varepsilon > 0$ be some constant such that
$$\varepsilon \ge \phi\gamma_k. \tag{37}$$
We divide the analysis into two cases depending on the value of ε.

Case A ($P_{\#}(R) \ge \frac{1}{2}U_{k\ell}(1+\varepsilon)$): Since $P_{\#}(R) \ge \frac{1}{2}\cdot U_{k\ell}(1+\varepsilon)$, we have that
$$\frac{U(1+\varepsilon) - P_{\#}(R)}{P_{\#}(R)} \le 1. \tag{38}$$

We have that

$$\Pr\left[Z_{\#}(R) > U_{k\ell}(1+\varepsilon)\right] = \Pr\left[Z_{\#}(R) > P_{\#}(R) \cdot \left(1 + \frac{U_{k\ell}(1+\varepsilon) - P_{\#}(R)}{P_{\#}(R)}\right)\right]$$

From Equation (33) it follows that $P_{\#}(R) = P_{\#}(R_F)(1-\alpha)$. Then from Equations (36) and (37) we have that $P_{\#}(R) \leq U_{k\ell}(1+\varepsilon)$. Hence, $\frac{U_{k\ell}(1+\varepsilon)-P_{\#}(R)}{P_{\#}(R)} \geq 0$. Further, from Equation (38) $\frac{U_{k\ell}(1+\varepsilon)-P_{\#}(R)}{P_{\#}(R)} \leq 0$. Hence, we can use the second statement of Theorem D.8. Using this we get

$$\begin{aligned}
&\leq \exp\left(-\frac{\alpha}{20} \cdot P_{\#}(R) \cdot \left(\frac{U_{k\ell}(1+\varepsilon) - P_{\#}(R)}{P_{\#}(R)}\right)^2\right) \\
&\leq \exp\left(-\frac{\alpha}{20} \cdot P_{\#}(R_F) \cdot \left(\frac{U_{k\ell}(1+\varepsilon) - P_{\#}(R_F)}{P_{\#}(R_F)}\right)^2\right) \\
&\qquad\qquad \text{(Fact D.2 and that } P_{\#}(R) \leq P_{\#}(R_F)\text{)} \\
&\leq \exp\left(-\frac{\alpha}{20} \cdot U_{k\ell} \cdot \frac{(\varepsilon - \phi\gamma_k)^2}{1+\phi\gamma_k}\right) \\
&\qquad\qquad \text{(Fact D.2 and Equation (36))} \quad (39)
\end{aligned}$$

Case B ($P_{\#}(R) < \frac{1}{2}U_{k\ell}(1+\varepsilon)$): Since $P_{\#}(R) < \frac{1}{2} \cdot U_{k\ell}(1+\varepsilon)$, we have that

$$\frac{U_{k\ell}(1+\varepsilon) - P_{\#}(R)}{P_{\#}(R)} \geq 1. \tag{40}$$

We have that

$$\begin{aligned}
\Pr\left[Z_{\#}(R) > U_{k\ell}(1+\varepsilon)\right] &= \Pr\left[Z_{\#}(R) > P_{\#}(R) \cdot \left(1 + \frac{U_{k\ell}(1+\varepsilon) - P_{\#}(R)}{P_{\#}(R)}\right)\right] \\
&\leq \exp\left(-\frac{\alpha}{20} \cdot P_{\#}(R) \cdot \left(2 \cdot \frac{U_{k\ell}(1+\varepsilon) - P_{\#}(R)}{P_{\#}(R)} - 1\right)\right) \\
&\qquad \text{(Using third statement in Theorem D.8 and that Equation (40))} \\
&= \exp\left(-\frac{\alpha}{20} \cdot \left(2U_{k\ell}(1+\varepsilon) - 3P_{\#}(R)\right)\right) \\
&\leq \exp\left(-\frac{\alpha}{40} \cdot U_{k\ell}(1+\varepsilon)\right). \\
&\qquad \text{(Using that } P_{\#}(R) < \tfrac{1}{2} \cdot U_{k\ell}(1+\varepsilon)\text{)} \quad (41)
\end{aligned}$$

Combining Equations (39) and (41) we get that

$$\Pr\left[Z_{\#}(R) > U(1+\varepsilon)\right] \leq \max\left\{\exp\left(-\frac{\alpha}{20} \cdot U_{k\ell}\frac{(\varepsilon - \phi\gamma_k)^2}{1+\phi\gamma_k}\right), \exp\left(-\frac{\alpha}{40} \cdot U_{k\ell}(1+\varepsilon)\right)\right\} \tag{42}$$

Let

$$\varepsilon \coloneqq \frac{40}{\alpha} \cdot \gamma_k. \tag{43}$$

We claim that for this value of ε, it holds that

$$\Pr\left[Z_{\#}(R) > U_{k\ell}(1+\varepsilon)\right] \leq \frac{\delta}{2n}. \tag{44}$$

Now by taking a union bound over bound over all $\ell \in [n]$ and using that $\alpha \coloneqq \frac{1}{d}$, it follows that R satisfies the fairness guarantee with probability at least $\frac{\delta}{2n}$.

We can upper bound the second term in Equation (42), as follows

$$\begin{aligned}\exp\left(-\frac{\alpha}{40}\cdot U_{k\ell}(1+\varepsilon)\right) &\leq \exp\left(-\frac{\alpha}{40}\cdot U_{k\ell}\cdot\varepsilon\right)\\ &\leq \exp\left(-U_{k\ell}\cdot\gamma_k\right)\\ &\leq \frac{\delta}{np}.\end{aligned}$$

(Using that $\gamma_k \geq \frac{1}{U_{k\ell}}\cdot\log\frac{2np}{\delta}$; which follows from Equation (6), $U_{k\ell}\geq 1$, and $\log\frac{2np}{\delta}\geq 1$)

To upper bound the first term in Equation (42), we use Fact D.9.

Fact D.9. For all $x, y \geq 0$, if $x \geq y + \sqrt{y}$, then $\frac{x^2}{1+x} \geq y$.

Proof. Since $1 + x > 0$, $\frac{x^2}{1+x} \geq y$ holds if and only if $x^2 - xy - y \geq 0$. The roots of the quadratic $f(x) \coloneqq x^2 - xy - y$ are

$$\frac{y}{2} - \sqrt{\frac{y^2}{4} + y} \quad \text{and} \quad \frac{y}{2} + \sqrt{\frac{y^2}{4} + y}.$$

If x is larger than both roots, then $f(x) \geq 0$ and, hence, $\frac{x^2}{1+x} \geq y$. It follows that $x \geq \frac{y}{2} + \sqrt{\frac{y^2}{4} + y}$ suffices. Then using that for all $a, b \geq 0$, $\sqrt{a} + \sqrt{b} \geq \sqrt{a+b}$, we get that

$$y + \sqrt{y} \geq \frac{y}{2} + \sqrt{\frac{y^2}{4} + y}.$$

Thus, it suffices $x \geq y + \sqrt{y}$ implies that $\frac{x^2}{1+x} \geq y$. □

We have

$$\begin{aligned}\frac{(\varepsilon - \phi\gamma_k)^2}{1+\phi\gamma_k} &\geq \left(\frac{39}{\alpha}\right)^2\cdot\frac{{\gamma_k}^2}{1+\phi\gamma_k} && \text{(Using that } 0\leq\phi\leq 1,\ \alpha\leq\tfrac{1}{2}\text{, and Equation (43))}\\ &\geq \left(\frac{39}{\alpha}\right)^2\cdot\frac{{\gamma_k}^2}{1+\gamma_k}. && \text{(Using that } 0<\phi\leq 1\text{)}\end{aligned}$$

To proof Equation (44), it suffices to prove that

$$\frac{{\gamma_k}^2}{1+\gamma_k} \geq \frac{1}{U_{k\ell}}\cdot\log\left(\frac{n+2}{\delta}\right). \tag{45}$$

Further, Fact D.9 implies that to prove Equation (45) it suffices to prove that

$$\gamma_k \geq y + \sqrt{y},$$

where $y \coloneqq \frac{1}{U_{k\ell}}\cdot\log\frac{n+2}{\delta}$. To prove this, observe that

$$\begin{aligned}\log\frac{np}{\delta}\cdot\frac{1}{U_{k\ell}} &\leq \log\frac{np}{\delta}\cdot\sqrt{\frac{1}{U_{k\ell}}}, && \text{(Using that } U_{k\ell}\geq 1\text{)}\\ \sqrt{\log\frac{np}{\delta}\cdot\frac{1}{U_{k\ell}}} &\leq \log\frac{np}{\delta}\cdot\sqrt{\frac{1}{U_{k\ell}}}. && \text{(Using that } \log\tfrac{np}{\delta}\geq\tfrac{1}{2}\text{ as } n\geq 1\text{ and } \delta\leq\tfrac{1}{2}\text{)}\end{aligned}$$

Hence, Equation (45) follows from Equation (6). □

D.5 Proof of Theorem D.10

Theorem D.10. *Given constants $c > 1$ and vector $\gamma \in \mathbb{R}^n_{\geq 0}$, , and matrices $P \in [0,1]^{m\times p}$, $W \in \mathbb{R}^{m\times n}_{\geq 0}$, $U \in \mathbb{R}^{n\times p}$, it is* **NP**-*hard to decide if Program* (7) *is feasible.*

Theorem D.10 follows from Theorem 5.2 of [42], which proves that checking the feasibility of the following program is **NP**-hard.[3]

$$\max_{x\in\{0,1\}^m} \quad \sum\nolimits_{i=1}^{m} w_i^\circ x_i \tag{46}$$

$$\text{s.t.,} \quad \forall\, \ell \in [p^\circ], \quad \sum\nolimits_{i=1}^{m^\circ} q_{i\ell}^\circ x_i \leq U_\ell^\circ, \tag{47}$$

$$\sum\nolimits_{i=1}^{m^\circ} x_i = n^\circ. \tag{48}$$

Where we used a superscript "$\circ$" on the variables of [42], to differentiate between ours and [42]'s variables. Theorem D.10 follows from Theorem 5.2 of [42] by observing that Program (46) is a special case of Program (7), when:

$$\begin{aligned} & n \coloneqq n^\circ, m \coloneqq m^\circ, p \coloneqq p^\circ, \gamma \coloneqq 1_n, P = q^\circ, \\ \forall k \in [n], \quad & \gamma_k = 1, \\ & U_{n\ell} = U_\ell^\circ, \\ \forall k \in [n] \setminus \{1\}, \quad & U_{k\ell} = n, \\ \forall i \in [m], j \in [n], \quad & W_{ij} = w_i^\circ. \end{aligned}$$

Finally, we can choose any $c > 1$.

E Extension of Theorem 3.1 to position-weighted constraints

In this section, we extend Theorem 3.1 to position-weighted version of fairness constraints. In particular, given position-discounts

$$v_1 \geq v_2 \geq \cdots \geq v_n$$

and a matrix $U \in \mathbb{Z}_+^{n\times p}$ the position-weighted fairness constraint requires a ranking R to satisfy:

$$\forall k \in [n], \ell \in [p], \quad \sum_{i\in G_\ell} \sum_{j\in[k]} v_j R_{ij} \leq U_{k\ell}$$

for all k and ℓ. For these constraints, we consider the following analogue of (ε, δ)-constraints: A ranking R is said to satisfy (ε, δ, v)-constraint if with probability at least $1 - \delta$ over the draw of $G_1, \ldots, G_p$

$$\forall k \in [n]\ \forall \ell \in [p], \quad \sum\nolimits_{i\in G_\ell} \sum\nolimits_{j=1}^{k} v_j R_{ij} \leq U_{k\ell}(1 + \varepsilon_k). \tag{49}$$

For these position-dependent constraints, our framework largely remains the same and is stated in Program (52). Compared to Program (7), the main difference is in the left-hand side of Program (51). We can prove the guarantees on the fairness and accuracy of the optimal solution of Program (52), under the additional assumption that, for a constant $\psi > 0$, U satisfies that

$$\forall \ell \in [p], \forall k \in [n], \quad U_{k\ell} \geq \psi k. \tag{50}$$

[3] Theorem 5.2 of [42] states an **NP**-hardness result holds for a generalization of Program (46). However, in their proof they only consider the special case of Program (46). Thus, their proof also implies **NP**-hardness of Program (46).

The parameter ψ shows up in Equation (51).

Our Fair-Ranking Framework for Position-Dependent Constraints

Input: Matrices $P \in [0,1]^{m\times p}$, $W \in \mathbb{R}_{\geq 0}^{m\times n}$, $U \in \mathbb{R}^{n\times p}$
Parameters: A constant $c > 1$, a failure probability $\delta \in (0,1]$, and for each $k \in [n]$, a relaxation parameter

$$\gamma_k \coloneqq \frac{1}{\psi} \cdot \log\left(\frac{2np}{\delta}\right) \cdot \max_{\ell\in[p]} \sqrt{\frac{1}{U_{k\ell}}}. \tag{51}$$

Program:

$$\max\nolimits_{R\in\mathcal{R}} \langle R, W\rangle, \tag{52}$$

$$\text{s.t. } \forall \ell \in [p] \ \ \forall k \in [n]$$

$$\sum\nolimits_{i\in[m], j\in[k]} v_j P_{i\ell} R_{ij} \leq U_{k\ell}\left(1 + \frac{2\sqrt{c}-1}{2\sqrt{c}} \cdot \gamma_k\right) \tag{53}$$

We prove the following guarantees on the fairness and accuracy of the optimal solution of Program (52).

Theorem E.1. *Let $\gamma \in \mathbb{R}^n$ be as defined in Equation* (51)*. If the matrix $U \in \mathbb{Z}_+^{n\times p}$ satisfies that for all $\ell \in [p]$ and $k \in [n]$, $U_{k\ell} \geq \psi k$, then is an optimization program Program* (52)*, parameterized by a constant c and failure probability δ, such that for any $c > 1$ and $\delta \in (0, \frac{1}{2}]$ its optimal solution satisfies $(c\gamma, \delta, v)$-constraint and has a utility at least as large as the utility of any ranking satisfying $((c - \sqrt{c})\gamma, \delta, v)$-constraint.*

The proof of Theorem E.1 is analogous to the proof of Theorem 3.1. Here, we highlight the differences.

Notation. Recall that for each item $i \in [m]$ and group $\ell \in [p]$, let $Z_{i\ell} \in \{0,1\}$ is indicator random variable that $Z_i \coloneqq \mathbb{I}[G_\ell \ni i]$.

The first change is in the definition of $Z_\#(R,\ell,k)$. In particular, we need to define

$$Z_\#(R,\ell,k) = \sum\nolimits_{i\in G_\ell} \sum\nolimits_{j=1}^{k} v_j R_{ij}.$$

For the new definition of $Z_\#$, we have following concentration result.

Lemma E.2. *For any position $k \in [n]$, group $\ell \in [p]$, parameters $\varepsilon \geq 0$ and $L, U \in \mathbb{R}$, and ranking $R \in \mathcal{R}$, where R is possibly a random variable independent of $\{Z_{i\ell}\}_{i,\ell}$, if $P_\#(R,\ell,k) \leq U$ or $P_\#(R,\ell,k) \geq L$ then the following equations hold respectively*

$$\Pr\left[Z_\#(R,\ell,k) < (1+\varepsilon)\, U\right] \geq 1 - e^{-\frac{2U^2\varepsilon^2}{k}},$$
$$\Pr\left[Z_\#(R,\ell,k) > (1-\varepsilon)\, L\right] \geq 1 - e^{-\frac{2L^2\varepsilon^2}{k}}.$$

The proof of Lemma E.2 is identical to the proofs of Lemmas D.5 and D.6; the only change is the new definition of $Z_\#$.

To prove Theorem E.1, it suffices to prove analogues of Propositions 5.1 and 5.2 for the new definition of $Z_\#$. Their proofs change as follows:

Proof of Proposition 5.1 The parameters in Equation (9) remain the same. Hence, following the same argument, Equation (10) holds. Now, we can prove Equation (12) as follows:

$$
\begin{aligned}
\Pr\left[Z_{\#}(R,\ell,k) \geq U_{\ell k}(1+\phi\gamma_k)\right] &= \Pr\left[Z_{\#}(R,\ell,k) \geq U'(1+\zeta)\right] \\
&\qquad \text{(Using that } U'(1+\zeta) = U_{k\ell}(1+\phi\gamma_k)\text{)} \\
&\leq \exp\left(-\frac{2\left(U'\right)^2\zeta^2}{k}\right) \qquad \text{(Using Lemma E.2)} \\
&= \exp\left(-\frac{2(1-\phi)^2 U_{\ell k}^2\gamma_k^2}{k}\right) \qquad \text{(Using Equation (9))} \\
&\leq \exp\left(-2\psi(1-\phi)^2 U_{\ell k}\gamma_k^2\right) \qquad \text{(Using that } U_{k\ell} \geq \psi k\text{)} \\
&\leq \frac{\delta}{2np}. \qquad \text{(Using Equation (51))} \qquad (54)
\end{aligned}
$$

Proposition 5.1 follows by replacing Equation (12) by Equation (54) in the rest of its proof.

Proof of Proposition 5.2 The parameters in Equation (13) remain the same. Now, we can prove $\Pr\left[Z_{\#}(R',k,\ell) \leq U_{k\ell}\right] < 1-\delta$ as follows:

$$
\begin{aligned}
\Pr\left[Z_{\#}(R',k,\ell) \leq U_{k\ell}\right] &= \Pr\left[Z_{\#}(R',k,\ell) \leq L' \cdot (1-\zeta)\right] \\
&\qquad \text{(Using that } L'(1-\zeta) = U_{k\ell}(1+b\gamma_k)\text{)} \\
&\leq \exp\left(-\frac{2\left(L'\right)^2\zeta^2}{k}\right) \qquad \text{(Using Lemma E.2)} \\
&= \exp\left(-\frac{2(\phi-b)^2\gamma_k^2 U_{k\ell}^2}{k}\right) \qquad \text{(Using Equation (13))} \\
&\leq \exp\left(-2\psi(\phi-b)^2\gamma_k^2 U_{k\ell}\right) \qquad \text{(Using that } U_{k\ell} \geq \psi k\text{)} \\
&< \frac{\delta}{2np} \qquad \text{(Using Equation (51) and Equation (13))} \qquad (55) \\
&< 1-\delta. \qquad \text{(Using that } \delta < \tfrac{1}{2} \text{ and } n \geq 1\text{)} \qquad (56)
\end{aligned}
$$

The rest of the proof is identical.

F Proofs of additional theoretical results

F.1 Proof of Proposition 2.3

Proof of Proposition 2.3. Suppose R is deterministic. Suppose it places items $i, j \in [m]$ on the first and second position respectively. With probability $p_i \cdot p_j = \frac{1}{4}$, both i and j belong to G_1, and with probability $p_i \cdot p_j = \frac{1}{4}$ both i and j belong to G_2. Thus, at least one of these events occurs with probability $\frac{1}{2}$. If either of these events hold, then R violates the equal representation constraint on the top-2 positions by a multiplicative factor of 2. The last two statements imply that R violates (ρ,δ)-equal representation for any $\rho < 1$ and $\delta < \frac{1}{2}$.

If R is a random variable, then any draw R' of R is a deterministic ranking, and hence, by the above argument R' violates the equal representation constraint on the top-2 positions by a multiplicative factor of 2 with a probability $\frac{1}{2}$ (over the randomness in G_1 and G_2). Since this holds for all draws of R and R is independent of G_1 and G_2, it follows that R violates the equal representation constraint on the top-2 positions by a multiplicative factor of 2 with a probability $\frac{1}{2}$ (over the randomness in G_1 and G_2, and R). Thus, R does not satisfy (ρ,δ)-equal representation for any $\rho < 1$ and $\delta < \frac{1}{2}$. ☐

F.2 Proof of Proposition F.1

Given a non-empty subset $\mathcal{C} \subseteq \mathcal{R}$ denoting a constraint, let $R_{\mathcal{C}}$ be the ranking with the highest utility in $\mathcal{C}$, i.e.,

$$R_{\mathcal{C}} \coloneqq \mathrm{argmax}_{R \in \mathcal{C}} \langle R, W \rangle .$$

In other words, $R_{\mathcal{C}}$ is the utility maximizing ranking subject to satisfying the "constraint" $\mathcal{C}$.

Proposition F.1. *Let $\mathcal{C}^\star$ be the set of all rankings that satisfy (ε, δ)-constraint. For any subset $\mathcal{C} \subseteq \mathcal{R}$, such that $\mathcal{C} \neq \mathcal{C}^\star$, at least one of the following holds:*

- *there exists a matrix $W \in \mathbb{R}_{\geq 0}^{m \times n}$ such that, $R_{\mathcal{C}}$ does not satisfy (ε, δ)-equal representation,*
- *there exists a matrix $W \in \mathbb{R}_{\geq 0}^{m \times n}$ such that, $\langle R_{\mathcal{C}}, W \rangle \leq \langle R_{\mathcal{C}^\star}, W \rangle \cdot \left(1 - \frac{1}{n}\right)$.*

We will use the following lemma in the proof of Proposition F.1.

Lemma F.2. *For all rankings $R \in \mathcal{R}$, there exists a matrix $W \in \mathbb{R}_{\geq 0}^{m \times n}$ such that for all other rankings $R' \in \mathcal{R}$, $R \neq R'$, it holds that $\langle R', W \rangle \leq \langle R, W \rangle \cdot \left(1 - \frac{1}{n}\right)$.*

Proof. Suppose R ranks items $i_1, i_2, \ldots, i_n$, in that order, in the first n positions. Pick $W \in [0,1]^{n \times m}$ such that $W_{ij} = 1$ if $i = i_j$ and 0 otherwise. R has utility $\langle W, R \rangle = \sum_{j=1}^{n} (W)_{i_j j} = n$. We claim that $\langle W, R' \rangle \leq n - 1$. If this is true, then the lemma follows.

Since $R \neq R'$, there exists a position $k \in [n]$ such that $(x_{\mathcal{C}})_{i_k k} = 0$. We can upper bound $\langle W, R' \rangle$ as follows:

$$\begin{aligned}
\langle W, R' \rangle &= \sum_{j=1}^{n} \sum_{i=1}^{m} \mathbb{I}[i = i_j] \, (R')_{ij} && \text{(By the choice of } W\text{)} \\
&= \sum_{j=1}^{n} (R')_{i_j j} \\
&= \sum_{j=1}^{k-1} (R')_{i_j j} + 0 + \sum_{j=k+1}^{n} (R')_{i_j j} && \text{(Using that } (R')_{i_k k} = 0\text{)} \\
&\leq n - 1. && \text{(Using that for all } i \in [m] \text{ and } j \in [n],\ (W)_{ij} \leq 1\text{)}
\end{aligned}$$

□

Proof of Proposition F.1. Since $\mathcal{C} \neq \mathcal{C}^\star$, at least one of the sets $\mathcal{C} \setminus \mathcal{C}^\star$ or $\mathcal{C}^\star \setminus \mathcal{C}$ is nonempty. We divide the proof into two cases.

Case A ($|\mathcal{C} \setminus \mathcal{C}^\star| \neq 0$): In this case, there exists a ranking $R \in \mathcal{C}$ such that $R \notin \mathcal{C}^\star$. Since $\mathcal{C}^\star$ is the set of all rankings that satisfy (ε, δ)-constraint, it follows that R does not satisfy (ε, δ)-constraint. Further, from Lemma F.2 it follows that there exists a matrix W such that $R \coloneqq \mathrm{argmax}_{R' \in \mathcal{R}} \langle R', W \rangle$. Since $\mathcal{C} \subseteq \mathcal{R}$, it follows that $R_{\mathcal{C}} = R$. Therefore, for this W, $R_{\mathcal{C}}$ does not satisfy (ε, δ)-constraint.

Case B ($|\mathcal{C}^\star \setminus \mathcal{C}| \neq 0$): In this case, there exists a ranking $R \in \mathcal{C}^\star$ such that $R \notin \mathcal{C}$. From Lemma F.2 it follows that there exists a matrix W such that, for rankings R' different from R (i.e., $R \neq R'$),

$$\langle R', W \rangle \leq \langle R, W \rangle \cdot \left(1 - \frac{1}{n}\right).$$

Thus, for this W, it follows that

$$\langle R_{\mathcal{C}^\star}, W \rangle \cdot \left(1 - \frac{1}{n}\right) \geq \langle R, W \rangle \cdot \left(1 - \frac{1}{n}\right) \geq \langle R', W \rangle .$$

In particular, for $R' = R_{\mathcal{C}}$, we get $\langle R_{\mathcal{C}^\star}, W \rangle \cdot \left(1 - \frac{1}{n}\right) \geq \langle R', W \rangle$.

□

F.3 Proof of Lemma F.3

Suppose there are two groups G_1 and G_2. Let R_E be the optimal solution to Equation (5) and let $R^\star$ be the ranking with the highest utility subject to satisfying (γ, δ)-equal representation constraints for the following γ:

$$\forall k \in [n], \quad \gamma_k \coloneqq \frac{1}{k} + 2\sqrt{\frac{6}{k} \cdot \log\left(\frac{2n}{\delta}\right)}. \tag{57}$$

Lemma F.3. *There exists a matrices $P \in [0,1]^{m\times 2}$ and $W \in [0,1]^{m\times 2}$ such that*

- *R_{E} satisfies (γ, δ)-equal representation and has utility 0,*
- *$R^\star$ has utility 1.*

Proof. Let P be the matrix with $P_{i1} = P_{i2} = \frac{1}{2}$ for all $i \in \{1, 2, \ldots, m-1\}$ and $P_{m1} = 1$ and $P_{m1} = 0$. Let W be the matrix whose first $m-1$ rows are 0, and the last row has is all 1s. Hence, only the last item, say i_m, has a nonzero contribution to the utility: If a ranking R ranks i_m in the first n positions, then the utility of R is 1, otherwise the utility of R is 0.

Our first claim will follow because the choice of P ensures that any ranking which ranks i_m in the first n positions cannot satisfy Equation (5). To see this, suppose R ranks i_m at the k-th position, then

$$\begin{aligned}
\mathbb{E}\left[\sum\nolimits_{i\in G_1} \sum\nolimits_{j=1}^{k} R_{ij}\right] &= \sum\nolimits_{i\in[m]} \sum\nolimits_{j=1}^{k} P_{i1} R_{ij} \\
&= 1 + \sum\nolimits_{i\in[m]\setminus\{i_m\}} \sum\nolimits_{j=1}^{k-1} P_{i1} R_{ij} && \text{(Using that } P_{i_m,1} = 1\text{)} \\
&= \frac{k+1}{2} && \text{(Using that } P_{i,1} = \tfrac{1}{2} \text{ for all } i \neq i_m\text{)} \\
&> \frac{k+1}{2}.
\end{aligned}$$

Hence, R cannot satisfy Equation (5).

To prove our second claim, we will construct a ranking which has utility 1 and satisfies (γ, δ)-equal representation . It suffices to choose any ranking R which places i_m in the first n position satisfies constraint. By our earlier argument this ranking has a utility 1. Let Z_j be the indicator random variable that the item in the j-th position in R belongs to G_1. This implies that $\sum_{i\in G_1} \sum_{j=1}^{k} R_{ij} = \sum_{j=1}^{k} Z_j$ for all k. Further, by the choice of P, we have

$$\frac{k}{2} \le \mathbb{E}\left[\sum\nolimits_{j=1}^{k} Z_j\right] \le \frac{k+1}{2}. \tag{58}$$

Further, by Definition 2.2, we have that Z_j is independent of Z_k for any $j \neq k$. Let $\varepsilon_k \coloneqq \sqrt{\frac{6}{k} \cdot \log\left(\frac{2n}{\delta}\right)}$. Using the above, we have

$$\begin{aligned}
\Pr\left[\sum\nolimits_{i\in G_1} \sum\nolimits_{j=1}^{k} R_{ij} \ge \frac{k+1}{2} \cdot (1+\varepsilon_k)\right] &= \Pr\left[\sum\nolimits_{j=1}^{k} Z_j \ge \frac{k+1}{2} \cdot (1+\varepsilon_k)\right] \\
&\le \Pr\left[\sum\nolimits_{j=1}^{k} Z_j \ge \mathbb{E}\left[\sum\nolimits_{j=1}^{k} Z_j\right] \cdot (1+\varepsilon_k)\right] \\
& && \text{(Using Equation (58))} \\
&\le \exp\left(-\frac{\varepsilon_k^2}{3} \cdot \mathbb{E}\left[\sum\nolimits_{j=1}^{k} Z_j\right]\right) \\
& && \text{(Using the Chernoff's bound, see [45])} \\
&\le \exp\left(-\frac{\varepsilon_k^2 k}{6}\right) && \text{(Using Equation (58))} \\
&\le \frac{\delta}{2n}. && \text{(Using that } \varepsilon_k \coloneqq \sqrt{\tfrac{6}{k} \cdot \log\left(\tfrac{2n}{\delta}\right)}\text{)}
\end{aligned}$$

Further, as $\gamma_k \geq \frac{k+1}{k} \cdot (1+\varepsilon_k)$, we get

$$\Pr\left[\sum\nolimits_{i\in G_1}\sum\nolimits_{j=1}^{k} R_{ij} \geq \frac{k}{2}\cdot(1+\gamma_k)\right] \leq \Pr\left[\sum\nolimits_{i\in G_1}\sum\nolimits_{j=1}^{k} R_{ij} \geq \frac{k+1}{2}\cdot(1+\varepsilon_k)\right]$$
$$\leq \frac{\delta}{2n}.$$

Further, considering $1 - Z_j$ and repeating a similar argument for G_2, we get

$$\begin{aligned}
\Pr\left[\sum\nolimits_{i\in G_2}\sum\nolimits_{j=1}^{k} R_{ij} \geq \frac{k}{2}\cdot(1+\varepsilon_k)\right] &= \Pr\left[\sum\nolimits_{j=1}^{k}(1-Z_j) \geq \frac{k}{2}\cdot(1+\gamma_k)\right] \\
&\leq \Pr\left[\sum\nolimits_{j=1}^{k}(1-Z_j) \geq \mathbb{E}\left[\sum\nolimits_{j=1}^{k}(1-Z_j)\right]\cdot(1+\gamma_k)\right] \\
&\qquad\text{(Using Equation (58))} \\
&\leq \exp\left(-\frac{\gamma_k^2}{3}\cdot\mathbb{E}\left[\sum\nolimits_{j=1}^{k}(1-Z_j)\right]\right) \\
&\qquad\text{(Using the Chernoff's bound, see [45])} \\
&\leq \exp\left(-\frac{\gamma_k^2(k-1)}{6}\right) \qquad\text{(Using Equation (58))} \\
&\leq \frac{\delta}{2n}. \qquad\text{(Using Equation (57))}
\end{aligned}$$

By taking the union bound over all k, one can show that R satisfies (γ,δ)-equal representation. $\square$

F.4 Proof of Proposition F.4

Proposition F.4. *There exist $p \in [0,1]^m$ such that* (4) *is non-convex in R.*

Proof. It suffices to specify n, m, p, ε, δ, and two rankings R_1 and R_2 such that both R_1 and R_2 satisfy (ε,δ)-equal representation, but $\frac{R_1+R_2}{2}$ does not satisfy (ε,δ)-equal representation.

Define $n := 2$, $m := 4$, and $\varepsilon := \left[\,\frac{1}{3}\ \frac{1}{3}\,\right]^\top$. Fix any $0 < \delta < \frac{1}{2}$. Define

$$p := \begin{bmatrix}1 & 0 & \delta & 1-\delta\end{bmatrix}^\top.$$

Let R_1 be the ranking that places items 1 and 3 in the first and second position, and R_2 be the ranking that places items 2 and 4 in the first and second position, i.e.,

$$R_1 := \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix} \quad\text{and}\quad R_2 := \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}.$$

If $1 \in G_1$ and $3 \in G_2$, then R_1 places an equal number of items from G_1 and G_2 in the first two positions, and hence, satisfies equal representation. This event, happens with probability $p_1(1-p_3) = 1-\delta$. Thus, R_1 satisfies $(0,\delta)$-equal representation, and hence, (ε,δ)-equal representation. Replace item 1 and 3 with 2 and 4 and swap G_1 and G_2 in the above argument, to get that R_2 also satisfies (ε,δ)-equal representation.

However, we claim that $\frac{R_1+R_2}{2}$ does not satisfy (ε,δ)-equal representation. Note that with probability 1, $1 \in G_1$ and $2 \in G_2$. If $3,4 \in G_1$ or $3,4 \in G_2$, then $\frac{R_1+R_2}{2}$ violates the equal representation constraint on the top-2 positions by a multiplicative factor of $\frac{3}{2}$. At least one of these events happens with probability $p_3p_4 + (1-p_3)(1-p_4) = 2\delta(1-\delta) > \delta$, as $\delta < \frac{1}{2}$. Thus, $\frac{R_1+R_2}{2}$ does not satisfy (ε,δ)-equal representation for the specified $\varepsilon := \left[\,\frac{1}{3}\ \frac{1}{3}\,\right]^\top$ and $\delta < \frac{1}{2}$. $\square$

F.5 Proof of Theorem F.5

In this section, we prove the following theorem.

Theorem F.5. *Given $p \in [0,1]^m$, $\delta \in (0,1]$, $W \in \mathbb{R}_{\geq 0}^{m \times n}$, $\varepsilon \in [0,1]^n$, and $V \geq 0$ it is* **NP***-hard to decide if the value of Program* (4) *is at least V.*

Recall that constraint (60) is necessary and sufficient to satisfy (ε, δ)-equal representation, and hence, the value of (59) is the maximum utility of a ranking subject to satisfying (ε, δ)-equal representation.

$$\max_{R \in \mathcal{R}} \ \langle R, W \rangle \tag{59}$$

$$\text{s.t. w.p. at least } 1 - \delta \text{ over draw of } G_1, G_2, \tag{60}$$

$$\forall k \in [n],\ \forall \ell \in [2], \quad \textstyle\sum_{i \in G_\ell} \displaystyle\sum_{j=1}^{k} R_{ij} \leq \frac{k}{2} \cdot (1 + \varepsilon_k).$$

We will show that the decision version of (59) is **NP**-hard:

Theorem F.6. *Given $L \geq 0$, $\delta \in [0,1]$, $\varepsilon \in [0,1]^n$, $P \in [0,1]^{m \times p}$, and $W \in \mathbb{R}_{\geq 0}^{m \times n}$ it is* **NP***-hard to decide if the value of* (59) *is at least L.*

The proof of Theorem F.6 proceeds in two steps. In the first step, we reduce (61) to (59). In the second step, we prove that (61) is **NP**-hard because the **NP**-complete product partition problem reduces to (61). Together, the two steps imply the hardness of (59). The proof of the second step is inspired by the construction of [50] for the product knapsack problem, which is similar to (61).

F.5.1 Step 1: Reduction from (61) to (59)

In this step, we will reduce the following problem to (59).

Input: $L \geq 0$, $n \in [m]$, $\delta \in [0,1]$, $U \in \left[0, \frac{n}{2}\right]$ $v \in \mathbb{R}_{\geq 0}^{m}$, and $P \in [0,1]^{m \times p}$
Decision problem: Is the value of (61) at least L?

$$\max_{S \subseteq [m] \colon |S| = n} \quad \sum_{i \in S} v_i \tag{61}$$

$$\text{s.t.} \quad \text{w.p. at least } 1 - \delta \text{ over draw of } G_1, G_2,$$

$$|S \cap G_1| \leq U + \frac{n}{2} \quad \text{and} \quad |S \cap G_2| \leq U + \frac{n}{2}.$$

Reduction. Given an instance of (61) we construct the following instance of (59):

$$W \coloneqq v 1_n^\top, \tag{62}$$

$$\varepsilon_1 = \varepsilon_2 = \cdots = \varepsilon_{n-1} \coloneqq \frac{2n}{k} - 1, \tag{63}$$

$$\varepsilon_n \coloneqq \frac{2U}{n} - 1, \tag{64}$$

where $1_n \coloneqq (1, \ldots, 1) \in \mathbb{R}^n$.[4] The parameters L, δ, and P are the same as the instance of (61).

The reduction from (61) to (59) is as follows: First solve (59) to obtain a ranking R. Let S be the set of items R places in the top-n positions. Output S. Clearly, this is a polynomial-time reduction. It remains to prove that it is sound and complete.

In our construction, Condition (62) implies that the utility of a ranking only depends on the set of n items it places in the top-n positions, and hence, any two rankings that place the same set of items in the top-n positions have the same utility. Condition (63) ensures that any ranking satisfies the constraints in the first $n - 1$ positions with probability 1. This is because, for all $k \in [n-1]$,

[4] To be precise, we consider $\varepsilon_1 = \varepsilon_2 = \cdots = \varepsilon_{n-1} \coloneqq \min\left\{1, \frac{2n}{k} - 1\right\}$ and $\varepsilon_n \coloneqq \min\left\{1, \frac{2U}{n} - 1\right\}$.

$\frac{k}{2}(1+\varepsilon_k) = n > k$. Thus, a ranking R is feasible for (59) iff it satisfies: With probability at least $1-\delta$ over draw of G_1, G_2,

$$\forall \ell \in [2], \quad \sum_{i \in G_\ell} \sum_{j=1}^{k} R_{ij} \leq \frac{n}{2} \cdot (1+\varepsilon_n) = U + \frac{n}{2}.$$

Soundness and completeness. Fix any $R \in \mathcal{R}$. Let S be the set of items R places in the top-n positions. It holds that

$$\langle R, W \rangle \overset{(62)}{=} \sum_{i \in S} v_i.$$

It remains to show that R is feasible for (59) iff S is feasible for (61). Due to conditions (63) and (64), R is feasible for (59) iff: With probability at least $1-\delta$ over draw of G_1, G_2,

$$\forall \ell \in [2], \quad \sum_{i \in G_\ell} \sum_{j=1}^{k} R_{ij} \leq U + \frac{n}{2}.$$

Since by the definition of S, for all $T \subseteq [m]$, $\sum_{i \in T} \sum_{j=1}^{n} R_{ij} = |S \cap T|$, it follows that with probability 1 $\sum_{i \in G_\ell} \sum_{j=1}^{n} R_{ij} = |S \cap G_\ell|$. Thus, S is feasible for (61) iff R is feasible for (59). Thus, the reduction is sound and complete.

F.5.2 Step 2: Reduction from product partition problem to (61)

We consider the following version of the product partition problem:

*Cardinality constrained product parition problem (*CPPP*)*

Input: $a_1, a_2, \ldots, a_q \in \mathbb{Z}_{\geq 0}$ and $\ell \in \{0, 1, \ldots, q\}$.
Decision problem: Is there a set $S \subseteq [q]$ of size ℓ such that

$$\prod_{i \in S} a_i = \prod_{i \in [q] \setminus S} a_i?$$

The usual product partition problem (PPP) does not require S to have size ℓ and is known to be **NP**-complete. CPPP is clearly in **NP**. To see that CPPP is **NP**-complete, one can reduce PPP to CPPP: To see this, given an instance of PPP, construct $q+1$ instances of CPPP, one for each value of $\ell \in \{0, 1, \ldots, q\}$. Then, PPP is a 'Yes' instance iff at least one of the $q+1$ CPPP instances in a 'Yes' instance. Thus, it follows that CPPP is also **NP**-complete.

Assumptions on CPPP instances without loss of generality. The decision problem for CPPP is simple for instances with $\ell = 0$, or with one or more of $a_1, \ldots, a_q$ as 0. As all inputs are integral, without loss of generality, we assume that $\ell \geq 1$ and $a_1, \ldots, a_q \geq 1$. Note that if in an CPPP $\sqrt{\prod_{i=1}^{q} a_i}$ is non-integral, then it is a 'No' instance. This can be verified in polynomial time, and hence, without loss of generality, we assume that $\sqrt{\prod_{i=1}^{q} a_i}$ is integral.

Reduction from CPPP to (61)**.** Given an instance of CPPP, we construct an instance of (61) with

$$n \coloneqq 2\ell, \quad m \coloneqq q + \ell, \quad U \coloneqq \ell - 1, \quad \text{and} \quad \delta \coloneqq \left(\frac{1}{a_{\max}}\right)^{\ell^2}, \tag{65}$$

where $a_{\max} \coloneqq \max_{i \in [q]} a_i$. Further, define constants

$$M \coloneqq (\ell + 2) \cdot \sqrt{\prod_{i=1}^{q} a_i} \quad \text{and} \quad B \coloneqq q \lceil M \log(a_{\max}) \rceil + 1. \tag{66}$$

We choose v so that the first q items correspond to the q numbers in the CPPP instance, and the next ℓ items have a "high" value:

$$\forall i \in [q], \quad v_i := \lceil M \log(a_i) \rceil, \tag{67}$$
$$\forall i \in [\ell], \quad v_{i+q} := L. \tag{68}$$

Note that each of the last ℓ items has a value larger than the total value of the first q items, i.e.,

$$\forall\, i \in [\ell], \quad v_{i+q} = B > \sum_{j \in [q]} v_j. \tag{69}$$

We choose P so that for the first q items $P_{i,1} \propto a_i^\ell$ and the next ℓ are in G_1 with probability 1:

$$\forall i \in [q], \qquad P_{i,1} := \left(\frac{a_i}{a_{\max}}\right)^\ell \cdot \frac{1}{\sqrt{\prod_{i=1}^q a_i}} \ \text{ and } \qquad P_{i,2} = 1 - P_{i,1}, \tag{70}$$
$$\forall i \in [\ell], \quad P_{i+q,1} := 1 \qquad \text{and } \ P_{i+q,2} = 1 - P_{i+q,1}. \tag{71}$$

Finally, let

$$L := \ell B + \left\lfloor \frac{M}{2} \sum_{i=1}^q \log(a_i) \right\rfloor. \tag{72}$$

The reduction from CPPP to (61) is as follows: First solve the constructed instance of (61) to get S. Then output $S \backslash Q$, where

$$Q := [\ell + q] \setminus [q]$$

is the set of the last ℓ items.

Let $C \in \mathbb{Z}$ be the bit complexity of the input for the given instance of (61). To show that the reduction is polynomial time, it suffices to show that L and $\lceil M \log(a_1) \rceil, \ldots, \lceil M \log(a_q) \rceil$ can be computed in poly(C) time. Note that, $M \leq 2^{O(C)}$, and hence, to compute $\lceil M \log(a_i) \rceil$ it suffices to compute $\log(a_i)$ up to $O(C)$ bits, which can be done in poly(C) time. Similarly, to compute L it suffices to compute $\sum_{i=1}^q \log(a_i)$ up to $O(C)$ bits, which can be done in poly(C) time. Thus, the reduction is polynomial time.

The choice of L and v ensures that the following fact holds.

Fact F.7. If a set $S \subseteq [q]$ satisfies $\sum_{i \in S} v_i \geq L$ and $|S| = n$, then $S \supseteq Q$.

Proof. Suppose toward a contradiction that satisfies $\sum_{i \in S} v_i \geq L$ and $|S| = n$ but S does not contain Q. Since $S = n = 2\ell$ Then,

$$\begin{aligned}
\sum_{i \in S} v_i \quad &= \quad \sum_{i \in S \cap Q} v_i + \sum_{i \in S \setminus Q} v_i \\
&\leq \quad |S \cap Q| \cdot \max_{i \in Q} v_i + \sum_{i \in [q] \setminus Q} v_i && \text{(Using } S \subseteq [q] \text{ and } v_i \geq 0) \\
&\overset{(68),\,(69)}{<} \quad |S \cap Q| \cdot B + B \\
&< \quad |Q| \cdot B && \text{(Using that } |S \cap Q| \leq |Q| - 1 \text{ and } B > 0) \\
&\leq \quad L. && \text{(Using (72), } |Q| = \ell \text{, and } L \geq \ell B)
\end{aligned}$$

□

Soundness. Suppose S is feasible for (61) and satisfies $\sum_{i \in S} v_i \geq L$. Due to (71), with probability 1, $G_1 \supseteq Q$. Hence, $G_2 \cap Q = \emptyset$. Thus,

$$\text{with probability 1,} \quad |S \cap G_2| = |(S \setminus Q) \cap G_2| \leq |S \setminus Q|.$$

Since $\sum_{i \in S} v_i \geq L$ and $|S| = n$ (as S is feasible for (61)), Fact F.7 implies that $S \supseteq Q$, hence $|S \setminus Q| = |S| - \ell$. Combining this with the above equation, we get that

$$\text{with probability } 1, \quad |S \cap G_2| \leq |S| - \ell = \ell. \qquad \text{(Using that } |S| = n = 2\ell\text{)}$$

Since $U \geq 0$,

$$\text{with probability } 1, \quad |S \cap G_2| \leq U + \ell. \tag{73}$$

S is feasible for (61) iff:

$$\begin{aligned}
&\Pr_{G_1,G_2}\left[|S \cap G_1| \leq U + \ell \ \text{ and } \ |S \cap G_2| \leq U + \ell\right] \geq 1 - \delta \\
&\overset{(73)}{\iff} \Pr_{G_1,G_2}\left[|S \cap G_1| \leq U + \ell\right] \geq 1 - \delta \\
&\iff \Pr_{G_1,G_2}\left[|(S \setminus Q) \cap G_1| \leq U + \ell\right] \geq 1 - \delta \\
&\qquad\qquad \text{(Using that with probability 1, } S, G_1 \supseteq Q\text{)} \\
&\iff \Pr_{G_1,G_2}\left[|S' \cap G_1| \leq U\right] \geq 1 - \delta \\
&\iff \Pr_{G_1,G_2}\left[|S' \cap G_1| > U\right] \leq \delta \\
&\iff \Pr_{G_1,G_2}\left[|S' \cap G_1| = n\right] \leq \delta \qquad \text{(Using that } U = n-1 \text{ and } |S'| = \ell\text{)} \\
&\iff \prod_{i \in S'} P_{i1} \leq \delta \\
&\overset{(71),(70),(65)}{\iff} a_{\max}^{-\ell \cdot |S'|} \cdot \left(\prod_{i \in [q]} a_i\right)^{-|S'|/2} \cdot \prod_{i \in S'} a_i^{\ell} \leq \left(\frac{1}{a_{\max}^{\ell}}\right)^{\ell} \\
&\iff \prod_{i \in S'} a_i \leq \sqrt{\prod_{i \in [q]} a_i} \qquad \text{(Using that } \ell > 0,\ a_1, \ldots, a_q > 0, \text{ and } |S'| = \ell\text{)} \qquad (74)
\end{aligned}$$

Since S is feasible for (61), it holds that

$$\prod_{i \in S'} a_i \leq \sqrt{\prod_{i \in [q]} a_i}.$$

To show that S' is feasible for CPPP, it remains to show that the above equation holds with equality.

Suppose toward a contradiction that $\prod_{i \in S'} a_i < \sqrt{\prod_{i \in [q]} a_i}$. Then, because $\sqrt{\prod_{i \in [q]} a_i}$ and $a_1, \ldots, a_q$ are integral

$$\prod_{i \in S'} a_i \leq \sqrt{\prod_{i \in [q]} a_i} - 1.$$

Because $M \geq 0$, taking the logarithm we get

$$M \sum_{i \in S'} \log a_i \leq M \log\left(\sqrt{\prod_{i \in [q]} a_i} - 1\right). \tag{75}$$

To upper bound the RHS, we will use the following fact:

Fact F.8. For all $x \geq 1$, $\log x - \log(x - 1) \geq \frac{1}{x}$.

Using Fact F.8 with $x = \sqrt{\prod_{i \in [q]} a_i}$ (as $a_1, \ldots, a_q \geq 1$), we get

$$\log\left(\sqrt{\prod_{i \in [q]} a_i}\right) - \log\left(\sqrt{\prod_{i \in [q]} a_i} - 1\right) \geq \frac{1}{\sqrt{\prod_{i \in [q]} a_i}}.$$

Hence, by (66)

$$M = (\ell+2)\cdot\sqrt{\prod_{i\in[q]} a_i} \geq \frac{\ell+2}{\log\left(\sqrt{\prod_{i\in[q]} a_i}\right) - \log\left(\sqrt{\prod_{i\in[q]} a_i} - 1\right)}.$$

On rearranging, we get

$$M\log\left(\sqrt{\prod_{i\in[q]} a_i} - 1\right) \leq M\log\left(\sqrt{\prod_{i\in[q]} a_i}\right) - \ell - 2.$$

Substituting this in (75), we get

$$M\sum_{i\in S'}\log a_i \leq M\log\left(\sqrt{\prod_{i\in[q]} a_i}\right) - \ell - 2.$$

Since for all $i \in S'$, $v_i \leq M\log(a_i) + 1$, it follows that

$$\sum_{i\in S'} v_i \leq \frac{M}{2}\log\left(\prod_{i\in[q]} a_i\right) - 2 < \left\lfloor \frac{M}{2}\log\left(\prod_{i\in[q]} a_i\right)\right\rfloor. \tag{76}$$

Thus,

$$\begin{aligned}\sum_{i\in S} v_i &= \sum_{i\in S\cap Q} v_i + \sum_{i\in S\setminus Q} v_i \\ &= \ell B + \sum_{i\in S'} v_i && \text{(Using that } S\supseteq Q \text{ and } S' := S\setminus Q\text{)} \\ &\overset{(76)}{<} \ell B + \left\lfloor M\log\left(\sqrt{\prod_{i\in[q]} a_i}\right)\right\rfloor \\ &= L.\end{aligned}$$

This is a contradiction to $\sum_{i\in S} v_i \geq L$.

Completeness. It suffices to show that if S' is feasible for the given instance of CPPP, then $S := S' \cup Q$ is feasible for (61) and satisfies $\sum_{i\in S} v_i \geq A$.

Due to (71), with probability 1, $G_1 \supseteq Q$. Hence, $G_2 \cap Q = \emptyset$. Thus,

$$\text{with probability 1,}\quad |S\cap G_2| = |(S\setminus Q)\cap G_2| \leq |S\setminus Q| = |S'| = \ell,$$

where the last equality holds as S' is feasible for the given instance of CPPP. This implies that (73) holds. Hence, by following the same arguments, (74) also holds. Thus, $S := S'\cup Q$ is feasible for (61)

It remains to show that $\sum_{i\in S} v_i \geq L$.

$$\begin{aligned}\sum_{i\in S} v_i &= \sum_{i\in Q} v_i + \sum_{i\in S'} v_i && \text{(Using that } S := S'\cup Q\text{)} \\ &\overset{(68)}{=} \ell B + \sum_{i\in S'} v_i \\ &\overset{(67)}{\geq} \ell B + \sum_{i\in S'} M\log a_i \\ &= \ell B + \frac{M}{2}\log\left(\prod_{i\in[q]} a_i\right) && \text{(Using that } \textstyle\prod_{i\in S'} a_i = \prod_{i\in[q]} a_i\text{)} \\ &\overset{(72)}{\geq} A.\end{aligned}$$