On the Within-Group Fairness of Screening Classifiers

Nastaran Okati
MPI for Software Systems
nastaran@mpi-sws.org

Stratis Tsirtsis
MPI for Software Systems

Manuel Gomez-Rodriguez
MPI for Software Systems

ABSTRACT

Screening classifiers are increasingly used to identify qualified candidates in a variety of selection processes. In this context, it has been recently shown that, if a classifier is calibrated, one can identify the smallest set of candidates which contains, in expectation, a desired number of qualified candidates using a threshold decision rule. This lends support to focusing on calibration as the only requirement for screening classifiers. In this paper, we argue that screening policies that use calibrated classifiers may suffer from an understudied type of within-group unfairness—they may unfairly treat qualified members *within* demographic groups of interest. Further, we argue that this type of unfairness can be avoided if classifiers satisfy within-group monotonicity, a natural monotonicity property within each of the groups. Then, we introduce an efficient post-processing algorithm based on dynamic programming to minimally modify a given calibrated classifier so that its probability estimates satisfy within-group monotonicity. We validate our algorithm using US Census survey data and show that within-group monotonicity can be often achieved at a small cost in terms of prediction granularity and shortlist size.

1 INTRODUCTION

As many selection processes receive thousands of applications, it has become increasingly common to rely on automated screening tools to shortlist a tractable set of promising candidates. These shortlisted candidates then move forward in the selection process and are evaluated in detail, possibly multiple times, until one or more qualified candidates are selected.

In the machine learning literature, algorithmic screening has been studied together with other high-stakes decision making problems as a supervised learning problem [1]. Under this view, algorithmic screening consists of designing both a screening classifier, which estimates the probability that a candidate is qualified, and a screening policy, which shortlists candidates using the candidates' probability values estimated by the screening classifier. Only very recently, a line of work has focused specifically on algorithmic screening [2, 3]. Therein, [2] argue that, to increase the efficiency of the selection process without decreasing the quality of the shortlisted candidates, the focus should be on screening policies that find the smallest shortlist of candidates containing a desired average number of qualified candidates with high probability without making any distributional assumptions on the candidates. Further, this work has shown that, if the screening classifier is calibrated [4], such distribution-free guarantees can be achieved using threshold decision rules as screening policies and, the more granular the predictions of the classifier, the smaller the shortlists provided by such policies.

In this work, our starting point is the realization that any threshold decision rule that uses calibrated screening classifiers may be biased against qualified candidates *within* demographic groups of interest. More specifically, it may shortlist one or more candidates from a group who are less likely to be qualified than one or more rejected candidates from the same group. This type of within-group unfairness may result in precluding the *best* candidates within each group—the candidates who are more likely to be qualified—to move forward in the selection process and have a chance to be selected.

Our contributions. We first show that to avoid such within-group unfairness, screening classifiers need to satisfy a monotonicity property within each of the groups of interest, which we refer to as within-group monotonicity. Then, we develop a set partitioning post-processing framework to minimally modify any calibrated classifier such that it satisfies within-group monotonicity. We make the following contributions:

I. We show that the problem is NP-hard using a reduction from a variation of the partition problem [5], which we refer to as the equal average partition problem and prove it is NP-complete. However, we identify a natural class of partitions—contiguous partitions—under which the problem is tractable.
II. We derive a dynamic programming algorithm for contiguous partitions that is guaranteed to find an optimal solution to our problem in polynomial time.
III. We create a simulated screening process using US Census survey data to validate and complement our methodological contributions and theoretical results.

Our results firstly show that the probability that an individual from a minority group suffers from within-group unfairness may be significant, which may lead to perpetuating historical biases against such groups. Secondly, within-group monotonicity can be achieved at a small cost in terms of prediction granularity and shortlist size. Appendix A contains a detailed discussion of the related work.

2 SCREENING, CALIBRATION AND WITHIN-GROUP DISCRIMINATION

Given a candidate with a feature vector $x \in \mathcal{X}$, we assume the candidate belongs to one demographic group of interest $z \in \mathcal{Z}$ and can be qualified ($y = 1$) or unqualified ($y = 0$) for the selection objective[1] Next, let $f : \mathcal{X} \rightarrow \text{Range}(f) \subseteq [0, 1]$ be a screening classifier that maps a candidate's feature vector $x \in \mathcal{X}$ to a quality score $f(x)$, where the higher the quality score $f(x)$, the more the classifier believes the candidate is qualified. Then, given a pool of m candidates, a screening policy $\pi : [0, 1]^m \rightarrow \mathcal{P}(\{0, 1\}^m)$ maps the candidates' quality scores to a probability distribution over shortlisting decisions $\{s_i\}_{i \in [m]}$. Here, each decision s_i specifies whether the corresponding candidate is shortlisted ($s_i = 1$) or is not shortlisted ($s_i = 0$).

[1] We do not require a candidate's group membership z to be included in or be inferable from their feature vector x.

In high-stakes applications, screening classifiers f are usually demanded to provide calibrated quality scores [6], *i.e.*, for every $a \in \text{Range}(f)$, it should hold that $\Pr(Y = 1 \,|\, f(X) = a) = a$. In this context, [2] have recently shown that, if the classifier f is calibrated, the optimal screening policy π^*_f that is guaranteed to shortlist, in expectation, the smallest set of candidates with a desired number of qualified candidates with high probability is given by a simple threshold decision rule that take shortlisting decisions as

$$s_i = \begin{cases} 1 & \text{if } f(x_i) > t_f, \\ \text{Bernoulli}(\theta_f) & \text{if } f(x_i) = t_f \\ 0 & \text{otherwise,} \end{cases} \qquad (1)$$

where t_f and θ_f depend on the classifier and data distribution. These results suggest focusing on calibration as the only requirement for screening classifiers. In this work, we argue that screening policies given by threshold decision rules using calibrated classifiers may suffer from an understudied type of unfairness—they may be biased against qualified members *within* demographic groups. More formally, the following proposition shows that any threshold decision rule may be biased against qualified members within demographic groups[2]:

Proposition 2.1. *Let π be a screening policy given by a threshold decision rule using a calibrated classifier f with threshold t. Assume there exist $a, b \in$ Range(f), with $a < t < b$, and $z \in \mathcal{Z}$ such that $P(Y = 1 \,|\, f(X) = a, Z = z) > P(Y = 1 \,|\, f(X) = b, Z = z)$. Then, it holds that*

$$\mathbb{E}_{Y \sim P_{Y \,|\, X,Z}, S \sim \pi}\left[Y(1-S) \,|\, f(X) = a, Z = z\right] \\ > \mathbb{E}_{Y \sim P_{Y \,|\, X,Z}, S \sim \pi}\left[YS \,|\, f(X) = b, Z = z\right].$$

The above result implies that there exist pools of applicants for which an optimal policy using a calibrated classifier may shortlist a candidate from a group who is less likely to be qualified than a rejected candidate from the same group. Importantly, the assumption under which the above within-group unfairness appears is not just a theoretical construct—it has been observed empirically in multiple real-world domains whenever the group membership Z is a spurious confounding factor that causes both X and Y [7]. The case in which the assumption holds for *every* group $z \in \mathcal{Z}$ and *any* threshold decision rule is known as Simpson's paradox [8].

To avoid the above within-group unfairness, we introduce and study within-group monotonicity:

Definition 2.2. *Given a set of groups $\mathcal{Z}$, a classifier f is within-group monotone if, for any $z \in \mathcal{Z}$ and $a, b \in$ Range(f) such that $a < b$, $\Pr(Z = z \,|\, f(X) = a) > 0$ and $\Pr(Z = z \,|\, f(X) = b) > 0$, it holds that*

$$\Pr(Y = 1 \,|\, f(X) = a, Z = z) \;\leq\; \Pr(Y = 1 \,|\, f(X) = b, Z = z).$$

In what follows, we will design a post-processing framework that, given a calibrated classifier, modifies it minimally so that it is within-group monotone, as shown in Figure 1. As a result, any screening policy given by a threshold decision rule using the modified classifier will not suffer from within-group unfairness. We favor a post-processing approach (rather than an in-processing one) mainly because post-processing approaches can be applied to any black-box classifier without asking for retraining or introducing training overhead [9]. Furthermore, in-processing approaches commonly need access to the feature defining group membership to ensure group-level fairness, which may not be available to the classifier due to privacy reasons.

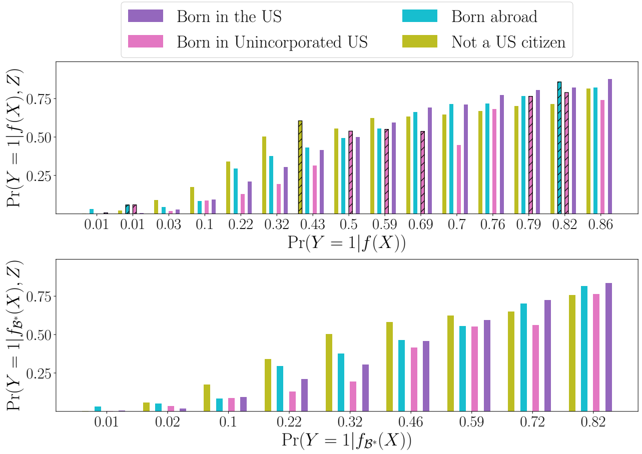


Figure 1: Quality score values $a = P(Y = 1 \,|\, f(X) = a)$ and group conditional quality score values $a_z = P(Y = 1 \,|\, f(X) = a, Z = z)$ of a (approximately) calibrated screening classifier f with finite range trained on US Census survey data and its within-group monotone counterpart $f_{\mathcal{B}^*}$ found by our post-processing framework. The demographic groups of interest $\mathcal{Z}$ are defined using US citizen status and the hatched bars indicate within-group monotonicity violations. Note that there exist no such violations in $f_{\mathcal{B}^*}$ (second row).

3 A SET PARTITIONING POST-PROCESSING FRAMEWORK

Let f be a calibrated classifier with $\text{Range}(f) = \{a_1, \ldots, a_n\}$ and $\rho_i := \Pr(f(X) = a_i)$. Here, note that we focus on calibrated classifiers with finite range, *i.e.*, $|\text{Range}(f)| = n < \infty$, since it is - impossible to find non-atomic calibrated classifiers from data , even asymptotically [10, 11]. Here, assume that $a_i < a_j$ for any $i < j$ without loss of generality. Further, for every demographic group of interest $z \in \mathcal{Z}$, let $a_{i,z} := \Pr(Y = 1 \,|\, f(X) = a_i, Z = z)$ and $\rho_{z \,|\, i} := \Pr(Z = z \,|\, f(X) = a_i)$, and note that, by definition, we have that $a_i = \sum_{z \in \mathcal{Z}} \rho_{z \,|\, i} a_{i,z}$. Then, our goal is to modify f minimally so that it is within-group monotone.

To this end, we note that the classifier f induces a partition of $\mathcal{X}$ into n disjoint regions or bins $\{\mathcal{X}_1, \ldots, \mathcal{X}_n\}$, where each bin $\mathcal{X}_i$ is characterized by a_i and ρ_i. Building upon this fact, we look at the problem from the perspective of set partitioning and seek to *merge* a small number of these induced bins to achieve within-group monotonicity. More formally, let $\mathscr{P}$ be the set of all partitions of the bin indices $\{1, \ldots, n\}$. Every $\mathcal{B} \in \mathscr{P}$ is a partition of the bin indices into a collection of nonempty and disjoint equivalence classes $\{\mathcal{A}_1, \ldots, \mathcal{A}_{|\mathcal{B}|}\}$, which we call cells. For each $x \in \mathcal{X}$, denote the index of the bin it belongs to as $i(x) = \{i \,|\, f(x) = a_i\}$ and represent a cell in $\mathcal{B}$ containing index $i(x)$ by $[i(x)]_{\mathcal{B}}$, where we

[2] All proofs can be found in the Appendix D.

drop the subscript $\mathcal{B}$ whenever it is clear from the context. Further, we know that the equivalence relation $\sim_{\mathcal{B}}$ implies that, for all $i(x') \in [i(x)]$, we have that $i(x) \sim_{\mathcal{B}} i(x')$. Then, we can use the partition $\mathcal{B}$ to define the modified classifier $f_{\mathcal{B}} : \mathcal{X} \rightarrow \text{Range}(f_{\mathcal{B}}) = \{a_{\mathcal{A}}\}_{\mathcal{A} \in \mathcal{B}}$, where

$$a_{\mathcal{A}} := \frac{\sum_{j \in \mathcal{A}} a_j \rho_j}{\sum_{j \in \mathcal{A}} \rho_j} = \quad \text{and} \quad f_{\mathcal{B}}(x) := a_{[i(x)]}.$$

Without loss of generality, we keep the cells induced by the partition $\mathcal{B}$ in increasing order with respect to $a_{\mathcal{A}}$, *i.e.*, $a_{\mathcal{A}_i} \leq a_{\mathcal{A}_j}$ for any $i < j$. By definition, $f_{\mathcal{B}}$ is calibrated, *i.e.*,

$$\Pr\left(Y = 1 \mid f_{\mathcal{B}}(X) = a_{\mathcal{A}}\right) = \frac{\sum_{j \in \mathcal{A}} a_j \rho_j}{\sum_{j \in \mathcal{A}} \rho_j} = a_{\mathcal{A}},$$

and we further define

$$a_{\mathcal{A},z} := \frac{\sum_{j \in \mathcal{A}} \rho_j \rho_{z \mid j} a_{j,z}}{\sum_{j \in \mathcal{A}} \rho_j \rho_{z \mid j}} = \Pr\left(Y = 1 \mid f_{\mathcal{B}}(X) = a_{\mathcal{A}}, Z = z\right).$$

Moreover, the larger the partition size $|\mathcal{B}|$, the more fine-grained the predictions of the classifier $f_{\mathcal{B}}$ [12]. Therefore, we can think of reducing the problem to finding a partition $\mathcal{B}$ of maximum size such that $f_{\mathcal{B}}$ is within-group monotone , *i.e.*,

$$\begin{aligned} \underset{\mathcal{B} \in \mathscr{P}}{\text{maximize}}\ |\mathcal{B}| \quad \text{subject to} \quad & a_{\mathcal{A}_i,z} \leq a_{\mathcal{A}_j,z} \\ & \forall \mathcal{A}_i, \mathcal{A}_j \in \mathcal{B} \text{ such that } a_{\mathcal{A}_i} < a_{\mathcal{A}_j}, \forall z \in \mathcal{Z}. \end{aligned}$$

However, such a problem formulation presents difficulties both in terms of tractability and soundness. First, we cannot expect to find such a partition in polynomial time:

Theorem 3.1. *Given a calibrated classifier f, the problem of finding the partition $\mathcal{B} \in \mathscr{P}$ of maximum size such that $f_{\mathcal{B}}$ is within-group monotone is NP-hard.*

To prove the above result in Appendix D.2, we first show that, by finding the partition $\mathcal{B}$ of maximum size such that $f_{\mathcal{B}}$ is within-group monotone, we can decide whether there exists a partition $\mathcal{B}'$ of size $|\mathcal{B}'| = 2$ such that $f_{\mathcal{B}'}$ is within-group monotone. Then, we show that the latter decision problem is NP-complete by a reduction from a variation of the partition problem [5], which we refer to as the equal average partition problem and prove it is NP-complete.

Second, even if the size of the partition $\mathcal{B}$ is large, the shortlists provided by threshold decision rules using $f_{\mathcal{B}}$ may differ greatly from those using f. The reason is that, in general, we may merge very different bins to ensure monotonicity within groups and, as a consequence, $f_{\mathcal{B}}$ may rank (pairs of) candidates *strictly differently*. More specifically, $f_{\mathcal{B}}$ may *not* satisfy the following monotonicity property with respect to f:

Definition 3.2. *A classifier f' is monotone with respect to f if, for all $f(x_1), f(x_2) \in Range(f)$ such that $f(x_1) < f(x_2)$, it holds that $f'(x_1) \leq f'(x_2)$.*

To guarantee that $f_{\mathcal{B}}$ is monotone with respect to f, we need to restrict our attention to the set of contiguous partitions $\mathscr{B} \subseteq \mathscr{P}$ of $\{1, \ldots, n\}$, *i.e.*, for any $\mathcal{B} \in \mathscr{B}$, if $i(x_1) < i(x_2) < i(x_3)$ and $i(x_1) \sim_{\mathcal{B}} i(x_3)$, then it also holds that $i(x_1) \sim_{\mathcal{B}} i(x_2)$ and $i(x_2) \sim_{\mathcal{B}} i(x_3)$. More formally, we have the following result:

Proposition 3.3. *Given a classifier f with n bins, $f_{\mathcal{B}}$ is monotone with respect to f iff $\mathcal{B}$ is a contiguous partition on $\{1, \ldots, n\}$.*

Algorithm 1 It returns the optimal partition $\mathcal{B}^*$ such that $f_{\mathcal{B}^*}$ is within-group monotone.

```
Input: $\{a_{1,z}, \ldots, a_{n,z}\}_{z \in \mathcal{Z}}$
Initialize: $\mathcal{B}_{l,r} = \{\}\ \forall l, r \in \{2, \ldots, n\}$, $\mathcal{B}_{1,r} = \{1, \ldots, r\}\ \forall r \in \{1, \ldots, n\}$
for $l \in \{2, \ldots, n\}$ do
    for $r \in \{l, \ldots, n\}$ do
        $\mathcal{S}_{l,r} = \left\{k | k < l, a_{\{k,\ldots,l-1\},z} \leq a_{\{l,\ldots,r\},z}\ \forall z \in \mathcal{Z}\right\}$   {Refer to Lemma. 4.1}
        if $\mathcal{S}_{l,r} = \emptyset$ then
            Continue   {In this case $\mathscr{B}_{l,r} = \emptyset$}
        end if
        $k^* = \text{argmax}_{k \in \mathcal{S}_{l,r}} \left|\mathcal{B}_{k,l-1}\right|$
        $\mathcal{B}_{l,r} = \mathcal{B}_{k^*,l-1} \cup \{\{l, \ldots, r\}\}$
    end for
end for
$l^* = \text{argmax}_{i \in \{1,\ldots,n\}} \left|\mathcal{B}_{i,n}\right|$
return $\mathcal{B}_{l^*,n}$
```

Surprisingly, while $|\mathscr{B}| = 2^{n-1}$, we will show in the next section that it is possible to find the optimal contiguous partition $\mathcal{B}^* = \text{argmax}_{\mathcal{B} \in \mathscr{B}} |\mathcal{B}|$ such that $f_{\mathcal{B}^*}$ is within-group monotone in polynomial time using dynamic programming.

4 OPTIMAL SET PARTITIONING VIA DYNAMIC PROGRAMMING

In this section, we derive an efficient algorithm based on dynamic programming that is guaranteed to find the optimal partition.

Our starting point is the following observation, which allows us to break down the problem of finding the optimal partition $\mathcal{B}^*$ into several subproblems. Let $\mathscr{B}_r$ be the set of contiguous partitions of the bin indices $\{1, \ldots, r\}$, with $r \leq n$, and $\mathscr{B}_{l,r} \subseteq \mathscr{B}_r$ be the subset of those partitions such that, for any $\mathcal{B} = \{\mathcal{A}_1, \ldots, \mathcal{A}_{|\mathcal{B}|}\} \in \mathscr{B}_{l,r}$, it holds that $\mathcal{A}_{|\mathcal{B}|} = \{l, \ldots, r\}$ and $f_{\mathcal{B} \cup \mathcal{B}'}$ is within-group monotone on the region of the feature space defined by $\cup_{i \leq r} \mathcal{X}_i$, where $\mathcal{B}'$ is any partition of the bin indices $\{r + 1, \ldots, n\}$[3]. Then, it clearly holds that the optimal partition $\mathcal{B}^* \in \cup_{l=1}^{n} \mathscr{B}_{l,n}$ and thus we can break the problem of finding $\mathcal{B}^*$ into n subproblems, *i.e.*, finding the optimal partition $\mathcal{B}^*_{l,n} = \text{argmax}_{\mathcal{B} \in \mathscr{B}_{l,n}} |\mathcal{B}|$ within each subset $\mathscr{B}_{l,n}$. From now on, with a slight abuse of notation, we will write $f_{\mathcal{B}}$ instead of $f_{\mathcal{B} \cup \mathcal{B}'}$ whenever $\mathcal{B}'$ refers to any partition of the bin indices not in $\mathcal{B}$.

Next, we realize that we can efficiently find the optimal partition $\mathcal{B}^*_{l,n}$ in each subset $\mathscr{B}_{l,n}$ recursively using dynamic programming. The key idea of the recursion is that any partition $\mathcal{B} \in \mathscr{B}_{l,r}$ needs to satisfy the following necessary and sufficient conditions:

Lemma 4.1. *Given any $\mathcal{B} \in \mathscr{B}_r$, it holds that $\mathcal{B} \in \mathscr{B}_{l,r}$ if and only if $\exists k < l$ such that $\mathcal{B} \setminus \{\{l, \ldots, r\}\} \in \mathscr{B}_{k,l-1}$ and $a_{\{k,\ldots,l-1\},z} \leq a_{\{l,\ldots,r\},z}\ \forall z \in \mathcal{Z}$.*

Consequently, we can efficiently find all the partitions in the subsets $\mathscr{B}_{l,r}$ iterating through l using the partitions in the subsets $\mathscr{B}_{k,l-1}$ with $k < l$. Finally, by construction, it clearly holds that, if $\mathcal{B}^*_{l,r} = \mathcal{B}' \cup \{\{l, \ldots, r\}\}$, with $\mathcal{B}' \in \mathscr{B}_{k,l-1}$, is the optimal partition

[3] Note that it may be impossible to satisfy both conditions simultaneously if, for example, the Simpon's paradox [13] holds, *i.e.*, for every group $z \in \mathcal{Z}$ and every pair of indices $i < j$, we have that $a_{i,z} > a_{j,z}$. In those cases, we may have that $\mathscr{B}_{l,r} = \emptyset$ for all $1 < l \leq r$.

in $\mathscr{B}_{l,r}$ then $\mathcal{B}' = \mathcal{B}^*_{k,l-1}$ is the optimal partition in $\mathscr{B}_{k,l-1}$. As a result, at each step of the recursion, we only need to store the optimal partition $\mathcal{B}^*_{l,r}$.

Algorithm 1 summarizes the overall procedure, which has complexity $O(n^3 \times |\mathcal{Z}|)$ and is guaranteed to find the optimal partition $\mathcal{B}^*$, as formalized by the following theorem:

Theorem 4.2. *Algorithm 1 returns* $\mathcal{B}^* = \text{argmax}_{\mathcal{B} \in \mathscr{B}} |\mathcal{B}|$ *such that* $f_{\mathcal{B}^*}$ *is within-group monotone.*

Remark The problem of finding a within-group monotone classifier relates to isotonic regression and within-group calibration. More specifically, since the structure of our problem resembles isotonic regression, one may think of using a simple variation of the Pool Adjacent Violators (PAV) algorithm [14] to find the optimal (contiguous) partition. However, in Appendix B, we show that the PAV algorithm (Algorithm 2) may not find the optimal partition. Furthermore, it is not even guaranteed to find a partition satisfying an intuitive type of local optimality. Within-group calibration requires that the probability that a candidate is qualified is independent of their group membership conditioned on their quality score and hence implies within-group monotonicity. In Appendix C, we first propose an algorithm to find an optimal within-group calibrated classifier (Algorithm 3). Then, we show that finding the optimal within-group calibrated classifier is computationally easier, however, in many cases, such a classifier may not exist and, when it does, the size of its partition may be much smaller than the size of $\mathcal{B}^*$, leading to less fine-grained predictions.

5 EXPERIMENTS USING SURVEY DATA

In this section, we create multiple instances of a simulated screening process using US Census survey data to first investigate how frequently within-group unfairness occurs in a recruiting domain. We then compare the partitions, as well as induced screening classifiers, provided by Algorithms 1, 2 and 3 in Appendix E.

Experimental setup. We use a dataset consisting of ~3.2 million individuals from the US Census [15]. Each individual is represented by sixteen features and one label $y \in \{0, 1\}$ indicating whether the individual is employed ($y = 1$) or not ($y = 0$). We think of employment as a (imperfect) proxy of qualification. The features contain demographic information such as age, gender, etc (Appendix B4, [15]). We run four sets of experiments where, in each of them, we use a different feature (US citizen status, race, gender, or disability record) to define the demographic groups of interest $\mathcal{Z}$[4].

For the experiments, we randomly split the dataset into two equally-sized and disjoint subsets. We use the first subset for training and calibration and the second subset for testing. More specifically, for each experiment, we create the training and calibration sets $\mathcal{D}_{\text{tr}}$ and $\mathcal{D}_{\text{cal}}$ by picking 100,000 and 50,000 individuals at random (without replacement) from the first subset. We use $\mathcal{D}_{\text{tr}}$ to train a logistic regression model f_{LR}[5] and use $\mathcal{D}_{\text{cal}}$ to both (approximately) calibrate f_{LR} using uniform mass binning (UMB) [2, 16], *i.e.*, discretize its outputs to n calibrated quality scores, and estimate the relevant probabilities ρ_i, a_i, $\rho_{z\,|\,i}$ and $a_{i,z}$ needed by Algorithms 1, 2, and 3. The resulting (approximately) calibrated classifier serves as our screening classifier f. For testing, we create a set $\{\mathcal{D}^i_{\text{pool}}\}_{i=1}^{100}$ of 100 pools, each with $m = 100$ individuals picked at random from the second subset, and create (the smallest) shortlists with at least k qualified individuals using the screening classifiers $f_{\mathcal{B}^*}$, $f_{\mathcal{B}_{\text{pav}}}$, and $f_{\mathcal{B}^*_{\text{cal}}}$ induced by the partitions found by Algorithms 1, 2 and 3, respectively. Throughout the experiments, we estimate the average and the standard error of the reported quantities by repeating each experiment 100 times.

Within-group unfairness occurs frequently between individuals from minority groups, especially with fine-grained classifiers. We start by estimating the probability $p_{d\,|\,z}$ that an individual from a demographic group of interest $z \in \mathcal{Z}$ may suffer from within-group unfairness, *i.e.*, $p_{d\,|\,z} = \frac{1}{\Pr(Z=z)} \sum_{i \in \{1,\ldots,n\}} \rho_i \rho_{z\,|\,i} v_i$, where $v_i = \mathbb{I}\left[\exists a_j \in \text{Range}(f) \,|\, a_i < a_j \wedge a_{i,z} > a_{j,z}\right]$. Figure 2a summarizes the results for a screening classifier f with $n = 15$ bins. We find that individuals who belong to minority groups are much more likely to suffer from within-group unfairness than those who belong to a majority group. For example, the probability that an individual who is not a US citizen may suffer from within-group unfairness is $p_{d\,|\,z} > 0.3$ while it is almost impossible that an individual born in the US is treated unfairly within its group. Further, we investigate to what extent the probability $p_d = \sum_{z \in \mathcal{Z}} P(Z = z) p_{d\,|\,z}$ that an individual may suffer from within-group unfairness depends on the number of bins n of f. Figure 2b shows that the more fine-grained a classifier is, the higher the probability that an individual may suffer from within-group unfairness, *e.g.*, for $n \leq 10$, $p_d < 0.05$ while, for $n = 40$, $p_d > 0.12$ across all sets of groups $\mathcal{Z}$. Since the accuracy of a calibrated classifier is related to how fine-grained its predictions are [2], the above finding suggests that high accuracy may have a cost in terms of within-group unfairness.

Our results so far show that the probability that individuals *may* suffer from within-group unfairness is significant. Next, we estimate the probability that, in a test pool of size m, an individual *does* suffer from within-group unfairness, *i.e.*, $p_{d\,|\,\mathcal{D}_{\text{pool}}} = \frac{1}{m} \sum_{x \in \mathcal{D}_{\text{pool}}} v_x$, where $v_x = \mathbb{I}\left[\exists x' \in \mathcal{D}_{\text{pool}} \,|\, a_{i(x)} < a_{i(x')} \wedge a_{i(x),z} > a_{i(x'),z}\right]$. Figure 2c shows that, on average across all test pools, the probability $p_{d\,|\,\mathcal{D}_{\text{pool}}}$ follows the same trend as p_d, however, it is slightly lower in value because each of the test pools is not representative of the entire population. However, note that, as $m \to \infty$, one can readily conclude that $p_{d\,|\,\mathcal{D}_{\text{pool}}} \to p_d$.

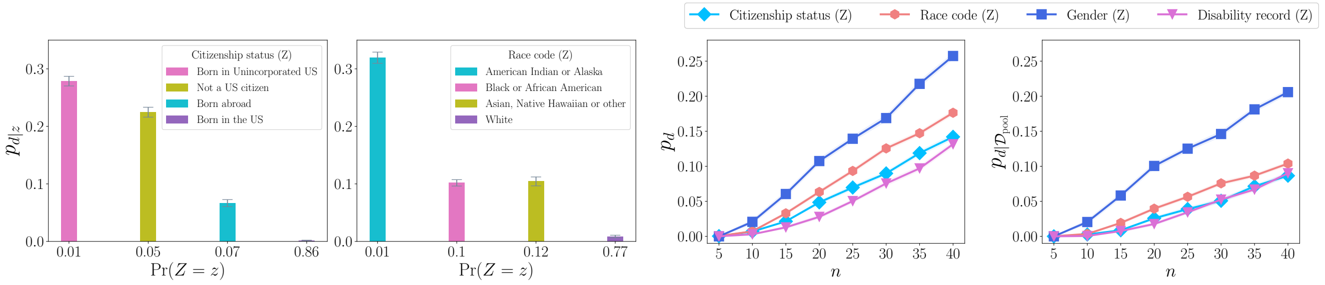
Algorithm 1 consistently provides larger partitions, which result in more fine-grained classifiers and smaller shortlists, than Algorithms 2 and 3. We further compare the partitions, as well as induced screening classifiers, provided by Algorithms 1, 2 and 3. Our results show that, as expected, Algorithm 1 consistently provides larger partitions, which result in more fine-grained classifiers and smaller shortlists, than Algorithms 2 and 3. Refer to Appendix E for a detailed discussion.

6 CONCLUSIONS

In this work, we have first shown that optimal screening policies using calibrated classifiers may suffer from an understudied type of within-group unfairness. Then, we have developed a polynomial time algorithm based on dynamic programming to minimally modify any given calibrated classifier so that it satisfies within-group

[4] In this section, we focus mainly on groups $\mathcal{Z}$ based on US citizenship status and race. However, Appendix E.4 shows similar results for groups $\mathcal{Z}$ defined based on gender and disability record.

[5] The classifier f_{LR} achieves a test accuracy of ~74% at predicting whether an individual is qualified.



(a) $p_{d\,|\,z}$ **vs.** $\Pr(Z = z)$

(b) p_d **vs.** n

(c) $p_{d\,|\,\mathcal{D}_{\textbf{pool}}}$ **vs.** n

Figure 2: Probability that an individual suffers from within-group unfairness. Panel (a) shows the probability $p_{d\,|\,z}$ that an individual from group z may suffer from within-group unfairness against $\Pr(Z = z)$ for $n = 15$. Panel (b) shows the probability p_d that an individual may suffer from within-group unfairness and Panel (c) shows the probability $p_{d\,|\,\mathcal{D}_{\textbf{pool}}}$ that an individual suffers from within-group unfairness in a test pool $\mathcal{D}_{\textbf{pool}}$ of size m, averaged across all test pools, against $n = |\textbf{Range}(f)|$.

monotonicity, a natural monotonicity property that prevents the occurrence of within-group unfairness. Finally, we have shown that within-group monotonicity can be achieved at a small cost in terms of prediction granularity and shortlist size.

Our work opens up many interesting avenues for future work. For example, it would be interesting to design classifiers that are within-group monotone with respect to every group that can be identified within a specified class of computations [17]. Further, it would be important to investigate how within-group monotonicity interacts with group fairness [9, 18]. Finally, it would be interesting to design post-processing algorithms using a sample access model [19] rather than a prediction-only access model and optimize other quality measures different from the partition size.

REFERENCES

[1] Niki Kilbertus, Manuel Gomez-Rodriguez, Bernhard Schölkopf, Krikamol Muandet, and Isabel Valera. Fair decisions despite imperfect predictions. In *International Conference on Artificial Intelligence and Statistics*, pages 277–287, 2020.

[2] Lequn Wang, Thorsten Joachims, and Manuel Gomez-Rodriguez. Improving screening processes via calibrated subset selection. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[3] Ying Jin and Emmanuel J Candès. Selection by prediction with conformal p-values. *arXiv preprint arXiv:2210.01408*, 2022.

[4] A. Philip Dawid. The well-calibrated bayesian. *Journal of the American Statistical Association*, 1982.

[5] Richard M Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations*, pages 85–103. Springer, 1972.

[6] Chirag Gupta, Aleksandr Podkopaev, and Aaditya Ramdas. Distribution-free binary classification: prediction sets, confidence intervals and calibration. *Advances in Neural Information Processing Systems*, 33:3711–3723, 2020.

[7] Judea Pearl. Models, reasoning and inference. *Cambridge, UK: Cambridge University Press*, 19(2), 2000.

[8] Colin R Blyth. On simpson's paradox and the sure-thing principle. *Journal of the American Statistical Association*, 67(338):364–366, 1972.

[9] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016.

[10] Linda C van der Gaag, Hans L Bodlaender, and Ad Feelders. Monotonicity in bayesian networks. *arXiv preprint arXiv:1207.4160*, 2012.

[11] Rina Foygel Barber. Is distribution-free inference possible for binary regression? 2020.

[12] Tilmann Gneiting, Fadoua Balabdaoui, and Adrian Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2007.

[13] English Simpson. The interpretation of interaction in contingency tables. *Journal of the royal statistical society series b-methodological*, 13:238–241, 1951.

[14] Miriam C. Ayer, Hugh D. Brunk, George M. Ewing, W. T. Reid, and Edward Silverman. An empirical distribution function for sampling with incomplete information. *Annals of Mathematical Statistics*, 26:641–647, 1955.

[15] Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. *Advances in Neural Information Processing Systems*, 34:6478–6490, 2021.

[16] Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, volume 1, pages 609–616. Citeseer, 2001.

[17] Ursula Hébert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*, pages 1939–1948. PMLR, 2018.

[18] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial intelligence and statistics*, pages 962–970, 2017.

[19] Jarosław Błasiok, Parikshit Gopalan, Lunjia Hu, and Preetum Nakkiran. A unifying theory of distance from calibration. *arXiv preprint arXiv:2211.16886*, 2022.

[20] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

[21] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. Fa* ir: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1569–1578, 2017.

[22] Till Speicher, Hoda Heidari, Nina Grgic-Hlaca, Krishna P Gummadi, Adish Singla, Adrian Weller, and Muhammad Bilal Zafar. A unified approach to quantifying algorithmic unfairness: Measuring individual &group unfairness via inequality indices. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2239–2248, 2018.

[23] David García-Soriano and Francesco Bonchi. Maxmin-fair ranking: individual fairness under group-fairness constraints. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 436–446, 2021.

[24] Meike Zehlike, Tom Sühr, Ricardo Baeza-Yates, Francesco Bonchi, Carlos Castillo, and Sara Hajian. Fair top-k ranking with multiple protected groups. *Information Processing & Management*, 59(1):102707, 2022.

[25] Jon Kleinberg. Inherent trade-offs in algorithmic fairness. In *Abstracts of the 2018 ACM International Conference on Measurement and Modeling of Computer Systems*, pages 40–40, 2018.

[26] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

[27] Rebecca Roelofs, Nicholas Cain, Jonathon Shlens, and Michael C Mozer. Mitigating bias in calibration error estimation. In *International Conference on Artificial Intelligence and Statistics*, pages 4036–4054. PMLR, 2022.

[28] Richard E Barlow and Hugh D Brunk. The isotonic regression problem and its dual. *Journal of the American Statistical Association*, 67(337):140–147, 1972.

[29] Alexander I Jordan, Anja Mühlemann, and Johanna F Ziegel. Optimal solutions to the isotonic regression problem. *arXiv preprint arXiv:1904.04761*, 2019.

[30] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. *Advances in neural information processing systems*, 30, 2017.

[31] Brian W Collins. Tackling unconscious bias in hiring practices: The plight of the rooney rule. *NYUL Rev.*, 82:870, 2007.

A RELATED WORK.

There is an extensive and rapidly growing line of work addressing bias and discrimination in the machine learning literature (refer to [20] for a detailed survey). This line of work has applications in a variety of important domains, including health care, criminal justice, and recommender systems. However, it has predominantly focused on preventing discrimination *across* demographic groups of interest, *e.g.*, designing machine learning models whose predictive performance is invariant across groups. In contrast, we focus on preventing unfairness *within* groups.

Within the above machine learning literature, there are a few notable exceptions [21–24], which studied similar notions to within-group monotonicity (in the context of ranking) and within-group unfairness. Among them, the notion of in-group monotonicity by [21, 24] is perhaps the most similar to within-group monotonicity. However, it comprises only the top-k ranked candidates in a specific pool of candidates (*i.e.*, in our work, the shortlisted candidates), rather than every candidate in a population of interest, and unconditional quality scores, rather than group conditional quality scores. Moreover, their formulation is fundamentally different and their technical contributions are orthogonal to ours. [22] addresses within-group unfairness as a measure of how unequally members within a group benefit from algorithmic decisions. In contrast, our notion of within-group monotonicity asks for accurately ranking individuals belonging to a group in terms of how worthy they are of receiving a beneficial decision rather than equally benefiting them. In this context, it is also worth highlighting the notion of within-group calibration [25], which implies within-group monotonicity, as discussed previously. Within-group calibration asks for equally well-calibrated probability estimates *across* groups so that a decision maker cannot use group membership to interpret these estimates. However, in the context of screening, our results show that within-group calibration may be an unnecessarily strong requirement.

Our work also relates to a line of work devoted to the study of calibration in supervised learning [16, 26]. Here, the main focus has been the design of classifiers with low calibration error using calibration-aware training or post-hoc re-calibration. However, there have also been efforts to ensure calibration errors are bias-free [27]. Here, we do not aim to minimize calibration error but ensure a calibrated classifier satisfies within-group monotonicity.

Algorithm 2 It returns a partition $\mathcal{B}_{\text{pav}}$ such that $f_{\mathcal{B}_{\text{pav}}}$ is within-group monotone.

```
Input: $\{a_{1,z}, \ldots, a_{n,z}\}_{z \in \mathcal{Z}}$
Initialize: $\mathcal{B}_{\text{pav}} = \{\{1\}, \ldots, \{n\}\}$
while $\exists \mathcal{A}_{i-1}, \mathcal{A}_i \in \mathcal{B}_{\text{pav}}$ and $z \in \mathcal{Z}$ such that $a_{\mathcal{A}_i,z} < a_{\mathcal{A}_{i-1},z}$ do
    $\mathcal{B}_{\text{pav}} = \mathcal{B}_{\text{pav}} \setminus \{\mathcal{A}_{i-1}, \mathcal{A}_i\}$
    $\mathcal{B}_{\text{pav}} = \mathcal{B}_{\text{pav}} \cup \{\mathcal{A}_{i-1} \cup \mathcal{A}_i\}$
end while
return $\mathcal{B}_{\text{pav}}$
```

B POOL ADJACENT VIOLATORS (PAV) ALGORITHM

Since the structure of our problem resembles isotonic regression, one may think of using a simple variation of the many times re-discovered Pool Adjacent Violators (PAV) algorithm [14] to find the optimal (contiguous) partition. However, in what follows, we first show that the PAV algorithm may not find the optimal partition—it is not even guaranteed to find a partition satisfying an intuitive type of local optimality. In comparison with the original PAV algorithm, the only difference is that, in our setting, one needs to check for monotonicity violations across multiple sets of conditional predictors, one per group $z \in \mathcal{Z}$, rather than only one set of predictors. However, the main idea underpinning the PAV algorithm remains the same, *i.e.*, as long as there are monotonicity violations between two adjacent cells, the algorithm merges the corresponding cells into one. Algorithm 2 summarizes the overall procedure, which has complexity $O(n^2 \times |\mathcal{Z}|)$ and is guaranteed to return a partition $\mathcal{B}_{\text{pav}}$ such that $f_{\mathcal{B}_{\text{pav}}}$ is within-group monotone, as formalized by the following Proposition:

Proposition B.1. *Algorithm 2 returns a partition $\mathcal{B}_{pav} \in \mathscr{B}$ such that the classifier $f_{\mathcal{B}_{pav}}$ is within-group monotone.*

Unfortunately, while the original PAV algorithm does enjoy global optimality guarantees for the isotonic regression problem[6] under multiple choices of loss functions [29], this is not true for our problem. There exist many instances for which Algorithm 2 fails to find the optimal partition $\mathcal{B}^*$, with one being the following example:

Example 1. *let Range(f) = $\{a_1, a_2, a_3\}$, $\mathcal{Z} = \{z_1, z_2\}$ and $\rho_i \rho_{z\,|\,i} = \frac{1}{6}$ for all $i \in \{1,2,3\}$ and $z \in \mathcal{Z}$. Further, let $a_{1,z_2} = a_{2,z_1} = a_{3,z_2} = \alpha$, $a_{1,z_1} = 2\alpha$, $a_{2,z_2} = 3\alpha$ and $a_{3,z_1} = 4\alpha$, where $\alpha \in [0, 0.25]$. First, we note that, by construction, it holds that $a_1 = \frac{3}{2}\alpha < a_2 = 2\alpha < a_3 = \frac{5}{2}\alpha$. Now, since $a_{1,z_1} > a_{2,z_1}$, Algorithm 2 first merges these two bins, then, since $a_{\{1,2\},z_2} > a_{\{3\},z_2}$, it merges all the three bins together and finally it terminates, returning $\mathcal{B} = \{\{1,2,3\}\}$. However, since it holds that $a_{1,z_1} < a_{\{2,3\},z_1}$ and $a_{1,z_2} < a_{\{2,3\},z_2}$, it clearly holds that the partition $\mathcal{B}' = \{\{1\}, \{2,3\}\}$ induces a classifier $f_{\mathcal{B}'}$ that is within-group monotone and it readily follows that $f_{\mathcal{B}'}$ dominates $f_{\mathcal{B}}$.*

Also refer to Figure 5 in Appendix E.2. In fact, Algorithm 2 does not even enjoy a type of intuitive local optimality guarantee based on the notion of dominance [2]:

Definition B.2. *Let f and f' be calibrated classifiers. Classifier f dominates f' if, for any $x_1, x_2 \in \mathcal{X}$ such that $f(x_1) = f(x_2)$, it holds that $f'(x_1) = f'(x_2)$.*

More specifically, if $f_{\mathcal{B}}$ dominates $f_{\mathcal{B}'}$, it can be shown that the expected size of the shortlists provided by the optimal screening policies using $f_{\mathcal{B}}$ are not larger than those using $f_{\mathcal{B}'}$ (Corollary 4.3, [2]) and it clearly holds that $|\mathcal{B}| \geq |\mathcal{B}'|$.

The reason why Algorithm 2 may fail to find the optimal partition is that, whenever it tries to fix a monotonicity violation between two adjacent cells $\mathcal{A}_{i-1}$ and $\mathcal{A}_i$, it does so by merging them. However, in our problem, the optimal fix may require merging cells $\mathcal{A}_i$ and $\mathcal{A}_{i+1}$. partition.

[6] In the isotonic regression problem [28], given a set of response variables $\{y_i\}_{i \in [n]}$, the goal is to find a set of predictor values $\{x_i\}_{i \in [n]}$, with $x_i \leq x_{i+1}$ for all $i \in [n]$, such that $\sum_i \ell(x_i, y_i)$ is minimized, where $\ell(x_i, y_i)$ is a loss measuring how well x_i approximates y_i.

Algorithm 3 It returns the optimal partition $\mathcal{B}^*_{\text{cal}}$ such that $f_{\mathcal{B}^*_{\text{cal}}}$ within-group calibrated.

```
**Input:** $\{a_{1,z}, \dots, a_{n,z}\}_{z \in \mathcal{Z}}$
**Initialize:** $\mathcal{B}_{\text{cal},i} = \{\} \ \forall i \in \{1, \dots, n\}$
**if** $a_{1,z} = a_1 \ \forall z \in \mathcal{Z}$ **then**
    $\mathcal{B}_{\text{cal},1} = \{\{a_1\}\}$
**end if**
**for** $r \in \{2, \dots, n\}$ **do**
    $\mathcal{S}_r = \left\{ i \in \{2, \dots, r\} \mid a_{\{i,\dots,r\},z} = a_{\{i,\dots,r\}} \ \forall z \in \mathcal{Z} \right\}$
    $k^* = \text{argmax}_{k \in \mathcal{S}_r} \left| \mathcal{B}_{\text{cal},k-1} \right|$
    **if** $\mathcal{B}_{\text{cal},k^*-1} \neq \emptyset$ **then**
        $\mathcal{B}_{\text{cal},r} = \mathcal{B}_{\text{cal},k^*-1} \cup \{\{k^*, \dots, r\}\}$
    **else if** $a_{\{1,\dots,r\}} = a_{\{1,\dots,r\},z} \ \forall z \in \mathcal{Z}$ **then**
        $\mathcal{B}_{\text{cal},r} = \{\{1, \dots, r\}\}$
    **end if**
**end for**
**return** $\mathcal{B}_{\text{cal},n}$
```

C WITHIN-GROUP MONOTONICITY VS WITHIN-GROUP CALIBRATION

Within-group calibration, or calibration within groups requires that the probability that a candidate is qualified is independent of their group membership conditioned on their quality score. More specifically, it is defined as follows [25, 30]:

Definition C.1. *Given a set of groups $\mathcal{Z}$, a classifier f is within-group calibrated iff, for every $z \in \mathcal{Z}$, $a \in$ Range(f) such that* $\Pr(Z = z \mid f(X) = a) > 0$, *it holds that* $\Pr(Y = 1 \mid f(X) = a, Z = z) = a$.

As discussed previously, within-group calibration implies within-group monotonicity. Then, to minimally modify a calibrated classifier f so that it becomes within-group monotone, one may think of finding the optimal partition $\mathcal{B}^*_{\text{cal}} = \text{argmax}_{\mathcal{B} \in \mathscr{B}} |\mathcal{B}|$ such that $f_{\mathcal{B}}$ is within-group calibrated. In what follows, we will first show that, perhaps surprisingly, finding $\mathcal{B}^*_{\text{cal}}$ is computationally *easier*[7] than finding $\mathcal{B}^*$. However, we will further show that, in many cases, $\mathcal{B}^*_{\text{cal}}$ may not exist and, when it does, the size of $\mathcal{B}^*_{\text{cal}}$ may be much smaller than the size of $\mathcal{B}^*$, leading to less fine-grained predictions.

To find the optimal $\mathcal{B}^*_{\text{cal}}$, we proceed recursively. Let $\mathscr{B}_r$ be the set of contiguous partitions of the bin indices $\{1, \dots, r\}$, with $r \leq n$. Then, iterating through r, we find the optimal partitions $\mathcal{B}^*_{\text{cal},r} = \text{argmax}_{\mathcal{B} \in \mathscr{B}_r} |\mathcal{B}|$ such that $f_{\mathcal{B}^*_{\text{cal},r}}$ is within-group calibrated in $\cup_{i \leq r} \mathcal{X}_i$. In this case, the key idea of the recursion is that any partition $\mathcal{B} \in \mathscr{B}_r$ such that $f_{\mathcal{B}}$ is within-calibrated on $\cup_{i \leq r} \mathcal{X}_i$ needs to satisfy the following necessary and sufficient condition:

Lemma C.2. *Given any $\mathcal{B} \in \mathscr{B}_r$, it holds that $f_{\mathcal{B}}$ is within-calibrated on $\cup_{i \leq r} \mathcal{X}_i$ if and only if $\exists l < r$ such that $\mathcal{B} \backslash \{\{l, \dots, r\}\} \in \mathscr{B}_{l-1}$ and $f_{\mathcal{B} \backslash \{\{l,\dots,r\}\}}$ is within-group calibrated on $\cup_{i \leq l-1} \mathcal{X}_i$ and $a_{\{l,\dots,r\},z} = a_{\{l,\dots,r\}} \ \forall z \in \mathcal{Z}$.*

As a consequence, we can efficiently find all partitions $\mathcal{B}$ in the subsets $\mathscr{B}_r$ such that $f_{\mathcal{B}}$ is within-group calibrated iterating through r using the partitions $\mathcal{B}'$ in the subsets $\mathscr{B}_l$ with $l < r$ such that $f_{\mathcal{B}'}$ is within-group calibrated. Finally, by construction, it clearly holds that if the optimal partition $\mathcal{B}^*_{\text{cal},r} = \mathcal{B}' \cup \{\{l, \dots, r\}\}$, with $\mathcal{B}' \in \mathscr{B}_{l-1}$, is the optimal partition in $\mathscr{B}_r$ then $\mathcal{B}' = \mathcal{B}^*_{\text{cal},l-1}$ is the optimal partition in $\mathscr{B}_{l-1}$. As a result, at each step of the recursion, we only need to store the optimal partition $\mathcal{B}^*_r$, not all partitions $\mathcal{B} \in \mathscr{B}_r$ such that $f_{\mathcal{B}}$ is within-group calibrated, and reuse it to find all $\mathcal{B}^*_{r'}$ with $r' > r$.

Algorithm 3 summarizes the overall procedure, which has complexity $O(n^2 \times |\mathcal{Z}|)$ and is guaranteed to find the optimal partition $\mathcal{B}^*_{\text{cal}}$, if such a partition exists, as formalized below:

Theorem C.3. *Algorithm 3 returns $\mathcal{B}^*_{cal} = \text{argmax}_{\mathcal{B} \in \mathscr{B}} |\mathcal{B}|$ such that $f_{\mathcal{B}^*_{cal}}$ is within-group calibrated if such partition exists or $\emptyset$ otherwise.*

Unfortunately, there are many cases in which $\mathcal{B}^*_{\text{cal}}$ does not exist, *e.g.*, this will happen if f systematically undervalues the probability that individuals from a group are qualified, in comparison with individuals from another group:

Proposition C.4. *Let $\mathcal{Z} = \{z, z'\}$, $\rho_{z \mid i} = \rho_{z' \mid i}$ and $a_{i,z} < a_{i,z'}$ for all $i \in \{1, \dots, n\}$. Then, there exists no $\mathcal{B} \in \mathscr{B}$ such that $f_{\mathcal{B}}$ is within-group calibrated.*

In the above situation, f may actually be within-group monotone and thus $|\mathcal{B}^*| = n$. Even if $\mathcal{B}^*_{\text{cal}}$ exists, there are examples where $|\mathcal{B}^*| - |\mathcal{B}^*_{\text{cal}}| = n - 1$.

[7] Using a similar proof technique as in Theorem 3.1, it can be proven that the problem of finding the partition $\mathcal{B} \in \mathscr{P}$ of maximum size such that $f_{\mathcal{B}}$ is within-group calibrated is NP-hard. Therefore, in general, the computational complexity is not lower.

D PROOFS

D.1 Proof of Proposition 2.1

By definition, the threshold decision rule π outputs $S = 0$ if $f(X) = a$ and $S = 1$ if $f(X) = b$. As a result, it immediately follows that:

$$\begin{aligned}\mathbb{E}_{Y\sim P_{Y\,|\,X,Z},S\sim\pi}\left[Y(1-S)\,|\,f(X)=a, Z=z\right] &= \mathbb{E}_{Y\sim P_{Y\,|\,X,Z}}\left[Y\,|\,f(X)=a, Z=z\right]\\ &> \mathbb{E}_{Y\sim P_{Y\,|\,X,Z}}\left[Y\,|\,f(X)=b, Z=z\right] = \mathbb{E}_{Y\sim P_{Y\,|\,X,Z},S\sim\pi}\left[YS\,|\,f(X)=b, Z=z\right].\end{aligned}$$

D.2 Proof of Theorem 3.1

We call a partition $\mathcal{B} \in \mathscr{P}$ valid if $f_{\mathcal{B}}$ is within-group monotone. We first show that, by finding a valid partition $\mathcal{B}$ of maximum size, we can decide whether there exists a valid partition $\mathcal{B}'$ of size $|\mathcal{B}'| = 2$. Assume the valid partition $\mathcal{B}$ of maximum size has size $|\mathcal{B}| = m$. Then, if $m \geq 2$, we can conclude that such a partition exists using Lemma D.1 and, if $m < 2$, no such partition exists because $\mathcal{B}$ is the valid partition of maximum size. Now, since we prove in Lemma D.2 that this decision problem is NP-complete, we can directly conclude that the problem of finding the valid partition of maximum size is NP-hard.

Lemma D.1. *Assume the valid partition $\mathcal{B}$ of maximum size has size $|\mathcal{B}| = k$. Then for every $k' \in \{1,\ldots,k-1\}$, there exist a valid partition $\mathcal{B}'$ such that $|\mathcal{B}'| = k'$.*

Proof. By Proposition 3.3, we have that any contiguous partition $\mathcal{B}'$ on $\{1,\ldots,|\mathcal{B}|\}$ is monotone with respect to $f_{\mathcal{B}}$. Furthermore, due to the same proposition, $\mathcal{B}'$ is also monotone with respect to the set $\left\{a_{\mathcal{A}_i,z}\right\}_{i\in\{1,\ldots,|\mathcal{B}|\}}$ for all $z \in \mathcal{Z}$. Since $\mathcal{B}$ is valid, we have that $\left\{a_{\mathcal{A}_i,z}\right\}_{i\in\{1,\ldots,|\mathcal{B}|\}}$ is increasing for all $z \in \mathcal{Z}$. As a result, $\mathcal{B}'$ is a valid partition. Thus, for any $k' \in \{1,\ldots,k-1\}$, we have that the contiguous partition $\mathcal{B}' = \left\{\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_{|\mathcal{B}|-k'-1}, \cup_{j\in\{0,\ldots,k'\}}\mathcal{A}_{|\mathcal{B}|-j}\right\}$ is valid and $|\mathcal{B}'| = k'$. This concludes the proof. □

Lemma D.2. *The problem of deciding whether there exists a valid partition $\mathcal{B}$ such that $|\mathcal{B}| = 2$ is NP-complete.*

Proof. First it is easy to see that, given a partition $\mathcal{B}$, we can check whether the partition is valid and has size $|\mathcal{B}| = 2$ in polynomial time. Therefore, the problem belongs to NP.

Now, to show the problem is NP-complete, we perform a reduction from a variation of the classical partition problem [5], which we refer to as the equal average partition problem. The equal average partition problem seeks to decide whether a set of n positive integers $\mathcal{S} = \{s_1,\ldots,s_n\}$ can be partitioned into two subsets of equal average. In Theorem D.3, we prove that the equal average partition problem is NP-complete, a result which may be of independent interest[8].

Without loss of generality, we assume $s_i \in [0,1]$ for all $s_i \in \mathcal{S}$[9] and, $s_i \leq s_j$ if $i < j$. For every $s_i \in \mathcal{S}$, we set $a_{i,z_1} = s_i$, $a_{i,z_2} = 1 - s_i$, $\rho_i = \frac{1}{n}$, $\rho_{z_1\,|\,i} = \alpha$, $\rho_{z_2\,|\,i} = 1-\alpha$ for $\alpha \in (0.5, 0.75]$. Note that we will have that $a_i = \alpha s_i + (1-\alpha)(1-s_i) = (2\alpha-1)s_i + (1-\alpha) \in [0,1]$. Note first that for any $\mathcal{A} \in \mathcal{B}$

$$a_{\mathcal{A},z_1} = \frac{\sum_{j\in\mathcal{A}}\rho_j\rho_{z_1\,|\,j}a_{j,z_1}}{\sum_{j\in\mathcal{A}}\rho_j\rho_{z_1\,|\,j}} = \frac{\sum_{j\in\mathcal{A}}\frac{\alpha}{n}a_{j,z_1}}{\sum_{j\in\mathcal{A}}\frac{\alpha}{n}} = \frac{\sum_{j\in\mathcal{A}}a_{j,z_1}}{|\mathcal{A}|} = 1 - \frac{\sum_{j\in\mathcal{A}}(1-a_{j,z_1})}{|\mathcal{A}|} = 1 - a_{\mathcal{A},z_2}. \quad (2)$$

, and

$$a_{\mathcal{A}} = \frac{\sum_{j\in\mathcal{A}}((2\alpha-1)a_{j,z_1}+1-\alpha)}{|\mathcal{A}|} = (2\alpha-1)\frac{\sum_{j\in\mathcal{A}}a_{j,z_1}}{|\mathcal{A}|} + 1 - \alpha = (2\alpha-1)a_{\mathcal{A},z_1} + 1 - \alpha \quad (3)$$

Note that, whenever we have that $a_{\mathcal{A},z_1} \leq a_{\mathcal{A}',z_1}$, it will also hold that $a_{\mathcal{A}} < a_{\mathcal{A}'}$ as $2\alpha - 1 > 0$.

Now, assume a valid partition $\mathcal{B}$ with $|\mathcal{B}| = 2$ exists and $\mathcal{B} = \{\mathcal{A}_1, \mathcal{A}_2\}$. Without loss of generality, assume $a_{\mathcal{A}_1,z_1} \leq a_{\mathcal{A}_2,z_1}$. Since $\mathcal{B}$ is a valid partition, we should have also that $a_{\mathcal{A}_1,z_2} \leq a_{\mathcal{A}_2,z_2}$, furthermore,

$$a_{\mathcal{A}_1,z_1} \leq a_{\mathcal{A}_2,z_1} \Rightarrow 1 - a_{\mathcal{A}_1,z_1} \geq 1 - a_{\mathcal{A}_2,z_1} \Rightarrow a_{\mathcal{A}_1,z_2} \geq a_{\mathcal{A}_2,z_2} \quad (4)$$

Since it simultaneously holds that $a_{\mathcal{A}_1,z_2} \geq a_{\mathcal{A}_2,z_2}$ and $a_{\mathcal{A}_1,z_2} \leq a_{\mathcal{A}_2,z_2}$, a valid partition $\mathcal{B}$ with $|\mathcal{B}| = 2$ exists if and only if $a_{\mathcal{A}_1,z_2} = a_{\mathcal{A}_2,z_2}$ and hence $a_{\mathcal{A}_1,z_1} = a_{\mathcal{A}_2,z_1}$. As $a_{\mathcal{A}_1,z_1}$ is the average of s_j for $j \in \mathcal{A}_1$ and $a_{\mathcal{A}_2,z_1}$ is the average of s_j for $j \in \mathcal{A}_2$ the partition $\mathcal{B}$ can partition $\mathcal{S}$ into two subsets of equal average.

We now prove that if no valid partition $\mathcal{B}$ with $|\mathcal{B}| = 2$ exists, there is no way of partitioning $\mathcal{S}$ into two subsets of equal average. For the sake of contradiction, assume $\mathcal{S}$ can be partitioned into $\mathcal{S}_1$ and $\mathcal{S}_2$ with equal averages κ. Define $\mathcal{A}_1 = \{i\,|\,s_i \in \mathcal{S}_1\}$ and $\mathcal{A}_2 = \left\{j\,|\,s_j \in \mathcal{S}_2\right\}$. Now if we build an instance of our problem based on $\mathcal{S}$ as described before and set $\mathcal{B} = \{\mathcal{A}_1, \mathcal{A}_2\}$ (clearly we have that $\mathcal{B}$ is a partition of $\{1,\ldots,n\}$) we have that $a_{\mathcal{A}_1,z_1} = a_{\mathcal{A}_2,z_1} = \kappa$, $a_{\mathcal{A}_1,z_2} = a_{\mathcal{A}_2,z_2} = 1-\kappa$ (refer to Eq. 2) and $a_{\mathcal{A}_1} = a_{\mathcal{A}_2} = (2\alpha-1)\kappa + (1-\alpha)$ (refer to Eq. 3). As a result, we have that $\mathcal{B}$ is a valid partition of size 2 which is a contradiction. This concludes the proof. □

Theorem D.3. *Given a set of n positive integers, the problem of deciding whether it can be partitioned into two non-empty subsets of equal average is NP-complete.*

[8] Given the similarity of the equal average partition problem to the classical partition problem, we would have expected to find a proof of NP-completeness elsewhere. However, we failed to find such a proof in previous work.

[9] We can always divide every element in $\mathcal{S}$ by the largest member of $\mathcal{S}$ to ensure elements fall in $[0,1]$.

Proof. First it is easy to see that, given two subsets, we can evaluate in polynomial time their averages and check whether they are equal or not. Therefore, the problem belongs to NP.

In the remainder of the proof, we will perform a reduction from the equal cardinality partition problem, which is known to be NP-complete, to the equal average partition problem. In the original problem, we are given a set of n positive integers $\mathcal{S}$, where n is an even number. The objective is to decide whether there exist two subsets $\mathcal{S}_1, \mathcal{S}_2 \subseteq \mathcal{S}$ such that $\mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}$ and $\mathcal{S}_1 \cap \mathcal{S}_2 = \emptyset$, with $|\mathcal{S}_1| = |\mathcal{S}_2|$ and $\sum_{i \in \mathcal{S}_1} i = \sum_{j \in \mathcal{S}_2} j$.

Now, we will transform an arbitrary instance of that problem into an instance of the equal average partition problem. Let the set of integers be $\mathcal{S}' = \mathcal{S} \cup \{n\sigma, n\sigma\}$, where $\sigma = \sum_{k \in \mathcal{S}} k$. It is easy to see that the average of $\mathcal{S}'$ is equal to $\frac{(2n+1)\sigma}{n+2}$.

We will start by showing that, if we can decide positively about that instance of the equal average partition problem, we can also decide positively about the original instance of the equal cardinality partition problem. Assume there exists a partition of $\mathcal{S}'$ into two sets $\mathcal{S}'_1, \mathcal{S}'_2$, with equal averages. As an intermediate result, we will show that the two copies of the number $n\sigma$ cannot belong to the same set $\mathcal{S}'_1$ or $\mathcal{S}'_2$. For the sake of contradiction, and without loss of generality, assume that both copies belong to $\mathcal{S}'_1$.

In the case where $\mathcal{S}'_1 = \{n\sigma, n\sigma\}$, it holds that $\frac{\sum_{i \in \mathcal{S}'_1} i}{|\mathcal{S}'_1|} = n\sigma$ and $\frac{\sum_{i \in \mathcal{S}'_2} j}{|\mathcal{S}'_2|} = \frac{\sigma}{n}$, which is a contradiction, since the two quantities cannot be equal because of $n \geq 2$. In cases where $\mathcal{S}'_1$ contains at least one more element, since $\mathcal{S}'_2 \neq \emptyset$, we get that $\frac{\sum_{i \in \mathcal{S}'_1} i}{|\mathcal{S}'_1|} = \frac{2n\sigma+\kappa}{2+l}$, with $0 < \kappa < \sigma$ and $1 \leq l \leq n-1$, and $\frac{\sum_{j \in \mathcal{S}'_2} j}{|\mathcal{S}'_2|} = \frac{\sigma-\kappa}{n-l}$. It follows that

$$\frac{1}{n-l} \leq 1 \Rightarrow \frac{\sigma-\kappa}{n-l} \leq \sigma-\kappa \Rightarrow \frac{\sum_{j \in \mathcal{S}'_2} j}{|\mathcal{S}'_2|} < \sigma \overset{(*)}{\Rightarrow} \frac{\sum_{j \in \mathcal{S}'_2} j}{|\mathcal{S}'_2|} < \frac{(2n+1)\sigma}{n+2} \Rightarrow \frac{\sum_{j \in \mathcal{S}'_2} j}{|\mathcal{S}'_2|} < \frac{\sum_{k \in \mathcal{S}'} k}{|\mathcal{S}'|},$$

where $(*)$ holds because $n > 1$. According to Lemma D.4, the last inequality leads to a contradiction. With that, we can conclude that one copy of $n\sigma$ belongs to $\mathcal{S}'_1$ and the other one belongs to $\mathcal{S}'_2$.

Let $\mathcal{S}_1, \mathcal{S}_2$ be such that $\mathcal{S}'_1 = \{n\sigma\} \cup \mathcal{S}_1$ and $\mathcal{S}'_2 = \{n\sigma\} \cup \mathcal{S}_2$. We will now show that $\mathcal{S}_1$ and $\mathcal{S}_2$ are a solution to the original instance of the equal cardinality partition problem, *i.e.*, $|\mathcal{S}_1| = |\mathcal{S}_2|$ and $\sum_{i \in \mathcal{S}_1} i = \sum_{j \in \mathcal{S}_2} j$. It is trivial to see that $\mathcal{S}_1, \mathcal{S}_2$ have to be non-empty, otherwise the averages of $\mathcal{S}'_1$ and $\mathcal{S}'_2$ would differ. Since $\mathcal{S}'_1, \mathcal{S}'_2$ are a partition of $\mathcal{S}'$ with equal averages and because of Lemma D.4, we know that

$$\frac{n\sigma + \sum_{i \in \mathcal{S}_1} i}{1 + |\mathcal{S}_1|} = \frac{n\sigma + \sum_{j \in \mathcal{S}_2} j}{1 + |\mathcal{S}_2|} = \frac{(2n+1)\sigma}{n+2}. \tag{5}$$

For the sake of contradiction, assume that either $|\mathcal{S}_1| \neq |\mathcal{S}_2|$ or $\sum_{i \in \mathcal{S}_1} i \neq \sum_{j \in \mathcal{S}_2} j$. For brevity, we will focus only on the two following cases, as any other case leads easily to a contradiction:

- $|\mathcal{S}_1| < |\mathcal{S}_2|$ and $\sum_{i \in \mathcal{S}_1} i < \sum_{j \in \mathcal{S}_2} j$: Since $\mathcal{S}_1, \mathcal{S}_2$ are such that $\mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}$, it holds that

$$\sum_{j \in \mathcal{S}_2} j - \sum_{i \in \mathcal{S}_1} i < \sigma \overset{(*)}{\Rightarrow} \frac{(2n+1)\sigma}{n+2}(1 + |\mathcal{S}_2|) - n\sigma - \frac{(2n+1)\sigma}{n+2}(1 + |\mathcal{S}_1|) + n\sigma < \sigma \Rightarrow$$
$$\frac{(2n+1)\sigma}{n+2}(|\mathcal{S}_2| - |\mathcal{S}_1|) < \sigma \Rightarrow (2n+1)(|\mathcal{S}_2| - |\mathcal{S}_1|) < (n+2) \overset{(**)}{\Rightarrow} 2n+1 < n+2 \Rightarrow n < 1,$$

  where $(*)$ follows from Equation 5, and $(**)$ holds because $|\mathcal{S}_2| - |\mathcal{S}_1| \geq 1$. The last inequality is clearly a contradiction.
- $|\mathcal{S}_1| > |\mathcal{S}_2|$ and $\sum_{i \in \mathcal{S}_1} i > \sum_{j \in \mathcal{S}_2} j$: The proof is the symmetric version of the proof in the previous case.

Therefore, we can conclude that $\mathcal{S}_1$ and $\mathcal{S}_2$ are a solution to the original problem, *i.e.*, they are a partition of $\mathcal{S}$ with equal cardinality and equal sums.

Lastly, we will show that, if there is no partition of $\mathcal{S}'$ with equal averages, there can be no equal cardinality partition of $\mathcal{S}$ with equal sums. For the sake of contradiction, assume there exist $\mathcal{S}_1$, $\mathcal{S}_2$ with $|\mathcal{S}_1| = |\mathcal{S}_2|$ and $\sum_{i \in \mathcal{S}_1} i = \sum_{j \in \mathcal{S}_2} j$. Then, let $\mathcal{S}'_1 = \{n\sigma\} \cup \mathcal{S}_1$ and $\mathcal{S}'_2 = \{n\sigma\} \cup \mathcal{S}_2$. It is easy to see that

$$\frac{\sum_{i \in \mathcal{S}'_1} i}{|\mathcal{S}'_1|} = \frac{n\sigma + \sum_{i \in \mathcal{S}_1} i}{1 + |\mathcal{S}_1|} = \frac{n\sigma + \sum_{j \in \mathcal{S}_2} j}{1 + |\mathcal{S}_2|} = \frac{\sum_{i \in \mathcal{S}'_2} i}{|\mathcal{S}'_2|}, \tag{6}$$

which is a contradiction, since it means that $\mathcal{S}'_1$ and $\mathcal{S}'_2$ are a partition of $\mathcal{S}'$ with equal averages.

Following the above procedure, we can decide whether the original instance of the equal-cardinality problem has a solution or not. As a consequence, the problem of deciding whether a set of positive integers can be partitioned into two subsets of equal average is NP-complete. □

Lemma D.4. *A set of integers $\mathcal{S}$ can be partitioned into two non-empty sets $\mathcal{S}_1, \mathcal{S}_2$ with equal averages $\frac{\sum_{i \in \mathcal{S}_1} i}{|\mathcal{S}_1|} = \frac{\sum_{j \in \mathcal{S}_2} j}{|\mathcal{S}_2|}$, iff $\frac{\sum_{i \in \mathcal{S}_1} i}{|\mathcal{S}_1|} = \frac{\sum_{k \in \mathcal{S}} k}{|\mathcal{S}|}$, with $|\mathcal{S}_1| \subset |\mathcal{S}|$.*

Proof. First, assume there is such a partition of $\mathcal{S}$ into $\mathcal{S}_1, \mathcal{S}_2$, with equal averages. It holds that

$$\frac{\sum_{i \in \mathcal{S}_1} i}{|\mathcal{S}_1|} = \frac{\sum_{k \in \mathcal{S}} k - \sum_{i \in \mathcal{S}_1} i}{|\mathcal{S}| - |\mathcal{S}_1|} \Rightarrow (|\mathcal{S}| - |\mathcal{S}_1|) \sum_{i \in \mathcal{S}_1} i = |\mathcal{S}_1| \left(\sum_{k \in \mathcal{S}} k - \sum_{i \in \mathcal{S}_1} i \right) \Rightarrow |\mathcal{S}| \sum_{i \in \mathcal{S}_1} i = |\mathcal{S}_1| \sum_{k \in \mathcal{S}} k$$

$$\Rightarrow \frac{\sum_{i \in \mathcal{S}_1} i}{|\mathcal{S}_1|} = \frac{\sum_{k \in \mathcal{S}} k}{|\mathcal{S}|},$$

where $\mathcal{S}_1 \subset \mathcal{S}$ because $\mathcal{S}_2 \neq \emptyset$.

Now, assume there exists a set $\mathcal{S}_1 \subset \mathcal{S}$, such that $\frac{\sum_{i \in \mathcal{S}_1} i}{|\mathcal{S}_1|} = \frac{\sum_{k \in \mathcal{S}} k}{|\mathcal{S}|}$ and let $\mathcal{S}_2 = \mathcal{S} \setminus \mathcal{S}_1$. It is easy to see that

$$\frac{\sum_{j \in \mathcal{S}_2} j}{|\mathcal{S}_2|} = \frac{\sum_{k \in \mathcal{S}} k - \sum_{i \in \mathcal{S}_1} i}{|\mathcal{S}| - |\mathcal{S}_1|} = \frac{\sum_{k \in \mathcal{S}} k - \frac{|\mathcal{S}_1|}{|\mathcal{S}|} \sum_{k \in \mathcal{S}} k}{|\mathcal{S}| \left(1 - \frac{|\mathcal{S}_1|}{|\mathcal{S}|}\right)} = \frac{\sum_{k \in \mathcal{S}} k}{|\mathcal{S}|},$$

and therefore, the sets $\mathcal{S}_1, \mathcal{S}_2$ consist a partition of $\mathcal{S}$ with equal averages. □

D.3 Proof of Proposition 3.3

We first prove the sufficient condition, *i.e.*, we prove that, if $f_{\mathcal{B}}$ is monotone with respect to f, then $\mathcal{B}$ is a contiguous partition on $\{1, \dots n\}$. The proof is by contradiction. Assume $\mathcal{B}$ is not a contiguous partition, *i.e.*, there exists $x_1, x_2, x_3 \in \mathcal{X}$ such that $i(x_1) < i(x_2) < i(x_3)$ and $i(x_1) \sim_{\mathcal{B}} i(x_3)$ while $i(x_1) \nsim_{\mathcal{B}} i(x_2)$. If $a_{[i(x_1)]} > a_{[i(x_2)]}$, then $f_{\mathcal{B}}(x_1) > f_{\mathcal{B}}(x_2)$, however, since $f(x_1) < f(x_2)$, this leads to a contradiction with the monotonicity assumption. On the other hand, if $a_{[i(x_1)]} < a_{[i(x_2)]}$, then $f_{\mathcal{B}}(x_3) < f_{\mathcal{B}}(x_2)$ since $i(x_1) \sim_{\mathcal{B}} i(x_3)$ and thus $a_{[i(x_3)]} < a_{[i(x_2)]}$, however, this leads again to a contradiction with the monotonicity assumption. This proves that $\mathcal{B}$ must be a contiguous partition.

Next, we prove the necessary condition, *i.e.*, we prove that, if $\mathcal{B}$ is a contiguous partition on $\{1, \dots n\}$, then $f_{\mathcal{B}}$ is monotone with respect to f. For any $x_1, x_2 \in \mathcal{X}$ such that $f(x_1) < f(x_2)$, we have that:

$$f_{\mathcal{B}}(x_1) = a_{[i(x_1)]} = \frac{\sum_{l \in [i(x_1)]} a_l \rho_l}{\sum_{l \in [i(x_1)]} \rho_l} \leq \frac{\sum_{l \in [i(x_2)]} a_l \rho_l}{\sum_{l \in [i(x_2)]} \rho_l} = a_{[i(x_2)]} = f_{\mathcal{B}}(x_2).$$

where the inequality is due to Lemma D.5 below and the fact that the weighted average of a set of numbers is lower and upper bounded by the smallest and largest element of the set respectively.

Lemma D.5. *Let f be a classifier with Range(f) $= \{a_1, \dots, a_n\}$, $\mathcal{B}$ be a contiguous partition on $\{1, \dots, n\}$ and $x_1, x_2 \in \mathcal{X}$. If $i(x_1) < i(x_2)$ and $i(x_1) \nsim_{\mathcal{B}} i(x_2)$, then, for every $k \in [i(x_1)]$ and $k' \in [i(x_2)]$, it holds that $k < k'$.*

Proof. To prove the lemma, we just need to prove that the largest index in $[i(x_1)]$ is smaller than the smallest index in $[i(x_2)]$. The proof is by contradiction. Let $l = \max\{k \mid k \in [i(x_1)]\}$ and $s = \min\{k \mid k \in [i(x_2)]\}$ and assume that $l > s$. Then, it cannot simultaneously hold that $i(x_1) = l$ and $i(x_2) = s$ since we have that $i(x_1) < i(x_2)$. Assume first that $i(x_1) \neq l$, and take $x_3, x_4 \in \mathcal{X}$ such that $i(x_3) = s$ and $i(x_4) = l$. If $i(x_3) < i(x_1)$, then it holds that $i(x_3) < i(x_1) < i(x_2)$, however, since $i(x_2) \sim_{\mathcal{B}} i(x_3)$ and $i(x_1) \nsim_{\mathcal{B}} i(x_2)$, this leads to a contradiction with the assumption that $\mathcal{B}$ is contiguous. If $i(x_3) > i(x_1)$, then it holds that $i(x_1) < i(x_3) < i(x_4)$, however, since $i(x_1) \sim_{\mathcal{B}} i(x_4)$ while $i(x_3) \nsim_{\mathcal{B}} i(x_4)$, this also leads to a contradiction with the assumption that $\mathcal{B}$ is contiguous. If one assumes instead that $i(x_1) = l$, a similar reasoning using $i(x_2)$ and $i(x_4)$ leads to a contradiction too. This completes the proof. □

D.4 Proof of Lemma 4.1

We first prove the sufficient condition, *i.e.*, we prove, for any $\mathcal{B} \in \mathscr{B}_{l,r}$, $\exists k < l$ such that $\mathcal{B} \setminus \{\{l, \dots, r\}\} \in \mathscr{B}_{k,l-1}$ and $a_{\{k,\dots,l-1\},z} \leq a_{\{l,\dots,r\},z}$ $\forall z \in \mathcal{Z}$. Let $\mathcal{B}' = \mathcal{B} \setminus \{\{l, \dots, r\}\}$. To this end, we start by proving by contradiction that $\exists k < l$ such that $\mathcal{B}' \in \mathscr{B}_{k,l-1}$. Since the partition $\mathcal{B}$ covers $\{1, \dots, r\}$, we have that the last cell of $\mathcal{B}'$ contains bin $l - 1$. Assume $\mathcal{B}' \notin \cup_{k=1}^{l-1} \mathscr{B}_{k,l-1}$. Then, there must exist $\mathcal{A}, \mathcal{A}' \in \mathcal{B}'$ and $z \in \mathcal{Z}$ such that $a_{\mathcal{A}} < a_{\mathcal{A}'}$ and $a_{\mathcal{A},z} > a_{\mathcal{A},z'}$. However, since $\mathcal{B}' \subset \mathcal{B}$, it also holds that $\mathcal{A}, \mathcal{A}' \in \mathcal{B}$ and $f_{\mathcal{B}}$ cannot be within-group monotone on $\cup_{i \leq r} \mathcal{X}_i$, leading to a contradiction. Therefore, it must hold that $\mathcal{B}' \in \cup_{k=1}^{l-1} \mathscr{B}_{k,l-1}$. Now, to prove that, if $\mathcal{B}' \in \cup_{k=1}^{l-1} \mathscr{B}_{k,l-1}$ and $\mathcal{B} \in \mathscr{B}_{l,r}$, then it must hold that $a_{\{k,\dots,l-1\},z} \leq a_{\{l,\dots,r\},z}$ $\forall z \in \mathcal{Z}$, we resort to Lemma D.6.

We next prove the necessary condition, *i.e.*, we prove that, given any $\mathcal{B} \in \mathscr{B}_r$, if $\exists k < l$ such that $\mathcal{B} \setminus \{\{l, \dots, r\}\} \in \mathscr{B}_{k,l-1}$ and $a_{\{k,\dots,l-1\},z} \leq a_{\{l,\dots,r\},z}$ $\forall z \in \mathcal{Z}$ then $\mathcal{B} \in \mathscr{B}_{l,r}$. Let $\mathcal{B}' = \mathcal{B} \setminus \{\{l, \dots, r\}\}$. Since $\mathcal{B}' \in \mathscr{B}_{k,l-1}$, we know that no violations of within-group monotonicity occurs on $\cup_{i \leq l-1} \mathcal{X}_i$. Now, we prove that there are no violations of within-group monotonicity between $\{l, \dots, r\}$ and any $\mathcal{A} \in \mathcal{B}'$. By assumption, we know that there are not violations of within-group monotonicity between $\{l, \dots, r\}$ and $\{k, \dots, l-1\}$. Then, we prove by contradiction that there are not violations between $\{l, \dots, r\}$ and any $\mathcal{A} \in \mathcal{B}' \setminus \{\{k, \dots, l-1\}\}$. For any $\mathcal{A} \in \mathcal{B}' \setminus \{\{k, \dots, l-1\}\}$, it follows from Proposition 3.3 that $a_{\mathcal{A}} < a_{\{k,\dots,l-1\}}$ and $a_{\mathcal{A}} < a_{\{l,\dots,r\}}$. Now, assume there exists $\mathcal{A} \in \mathcal{B}' \setminus \{\{k, \dots, l-1\}\}$, $z \in \mathcal{Z}$ such that $a_{\mathcal{A},z} > a_{\{l,\dots,r\},z}$. Since, by assumption, we have that $a_{\{k,\dots,l-1\},z} \leq a_{\{l,\dots,r\},z}$, it should hold that $a_{\{k,\dots,l-1\},z} < a_{\mathcal{A},z}$, which contradicts with the assumption that $\mathcal{B}' \in \mathscr{B}_{k,l-1}$, leading to a contradiction. This proves that $\mathcal{B} \in \mathscr{B}_{l,r}$.

Lemma D.6. *Let $\mathcal{B} = \mathcal{B}' \cup \{\{l, \dots, r\}\} \in \mathcal{B}_{l,r}$ and $\mathcal{B}' \in \mathscr{B}_{k,l-1}$ with $k < l$. Then, it must hold that $a_{\{k,\dots,l-1\},z} \leq a_{\{l,\dots,r\},z}$ $\forall z \in \mathcal{Z}$.*

Proof. Since $\mathcal{B}' \in \mathscr{B}_{k,l-1}$, we know that $\{k, \dots, l-1\} \in \mathcal{B}'$. Moreover, it follows from Proposition 3.3 that $f_{\mathcal{B}}$ is monotone with respect to f and hence, since $k < l$ and $k \nsim_{\mathcal{B}} l$, we have that $a_{\{k,\dots,l-1\}} < a_{\{l,\dots,r\}}$. Further, since $\mathcal{B} \in \mathcal{B}_{l,r}$, we have that, for every $\mathcal{A}, \mathcal{A}' \in \mathcal{B}$ such that $a_{\mathcal{A}} < a_{\mathcal{A}'}$, it holds that $a_{\mathcal{A},z} \leq a_{\mathcal{A}',z}$ for all $z \in \mathcal{Z}$. Thus, it also holds that $a_{\{k,\dots,l-1\},z} \leq a_{\{l,\dots,r\},z}$ for all $z \in \mathcal{Z}$.

□

D.5 Proof of Theorem 4.2

To prove that Algorithm 1 returns the optimal partition $\mathcal{B}^*$, we just need to prove that, for each $l, r \in \{1, \dots, n\}$, the partition $\mathcal{B}_{l,r}$ the algorithm finds is optimal, *i.e.*, $\mathcal{B}_{l,r} = \mathcal{B}^*_{l,r}$. In what follows, we prove this by induction.

For the base cases, we have that $\mathcal{B}_{1,r} = \{\{1, \dots, r\}\}$ are clearly optimal since $\mathscr{B}_{1,r}$ only contains $\{\{1, \dots, r\}\}$ for all $r \in \{1, \dots, n\}$. As the induction hypothesis, assume that, for any $l' < l$ and $r' < r$, the partition $\mathcal{B}_{l',r'}$ the algorithm finds is optimal. Moreover, let $\mathcal{S}_{l,r} = \left\{k \mid k < l, a_{\{k,\dots,l-1\},z} \leq a_{\{l,\dots,r\},z}\ \forall z \in \mathcal{Z}\right\}$. Then, for (l, r), we need to show that $\mathcal{B}_{l,r} = \mathcal{B}_{k^*,l-1} \cup \{\{l, \dots, r\}\}$, with $k^* = \operatorname{argmax}_{k \in \mathcal{S}_{l,r}} \left|\mathcal{B}_{k,l-1}\right|$, is optimal.

To this end, we first show that $f_{\mathcal{B}_{l,r}}$ is within-group monotone on $\cup_{i \leq r} \mathcal{X}_i$, *i.e.*, $\mathcal{B}_{l,r} \in \mathscr{B}_{l,r}$. We have that, by the induction hypothesis, $\mathcal{B}_{k^*,l-1} \in \mathscr{B}_{k^*,l-1}$ and, by definition, $k^* \in \mathcal{S}_{l,r}$. Then, it follows directly from Lemma 4.1 that $f_{\mathcal{B}} \in \mathscr{B}_{l,r}$. Next, we show that $\mathcal{B}_{l,r} = \operatorname{argmax}_{\mathcal{B} \in \mathscr{B}_{l,r}} |\mathcal{B}|$. Using again Lemma 4.1, we have that, for any $\mathcal{B} \in \mathscr{B}_{l,r}$, it holds that $\mathcal{B} = \mathcal{B}' \cup \{\{l, \dots, r\}\}$, with $\mathcal{B}' \in \mathscr{B}_{k,l-1}$, for some $k \in \mathcal{S}_{l,r}$. As a result, since $|\mathcal{B}' \cup \{\{l, \dots, r\}\}| = |\mathcal{B}'| + 1$, it suffices to find $\mathcal{B}' = \operatorname{argmax}_{\mathcal{B}'' \in \cup_{k \in \mathcal{S}_{l,r}} \mathscr{B}_{k,l-1}} |\mathcal{B}''|$. Now, by the induction hypothesis, we know that, for each $\mathscr{B}_{k,l-1}$, $\mathcal{B}_{k,l-1}$ is the optimal partition. Then, since $k^* = \operatorname{argmax}_{k \in \mathcal{S}_{l,r}} \left|\mathcal{B}_{k,l-1}\right|$, we can conclude that $\mathcal{B}_{l,r}$ is optimal.

D.6 Proof of Proposition B.1

We prove by contradiction. Assume there exist violations of within-group monotonicity. We first define the nearest violating triplet, (l, r, z), as:

$$(l, r, z) = \operatorname*{argmin}_{\{(i,j,z) \mid i,j \in \text{Range}(f_{\mathcal{B}}), i<j, z \in \mathcal{Z}\}} |j - i| \ \text{ such that } a_{\mathcal{A}_i,z} > a_{\mathcal{A}_j,z}$$

If $r = l + 1$ then it contradicts with the assumption that no monotonicity violations occur between adjacent cells. If $r \neq l + 1$, there exists $i \in \text{Range}(f_{\mathcal{B}})$ such that $l \leq i \leq r$ and it does not happen simultaneously that $i = l$ and $i = r$. Then it should hold that $a_{\mathcal{A}_l,z} \leq a_{\mathcal{A}_i,z} \leq a_{\mathcal{A}_r,z}$ since otherwise either of (l, i, z) or (i, r, z) is the nearest violating triplet. In this case however, $a_{\mathcal{A}_l,z} \leq a_{\mathcal{A}_r,z}$ which is a contradiction with it being a violating triplet. As a result, no such triplet can exist and $f_{\mathcal{B}}$ is within-group monotone.

D.7 Proof of Lemma C.2

We first prove the sufficient condition, *i.e.*, we prove that, given any $\mathcal{B} \in \mathscr{B}_r$, if it holds that $f_{\mathcal{B}}$ is within-group calibrated on $\cup_{i \leq r} \mathcal{X}_i$ then $\exists l < r$ such that $\mathcal{B} \backslash \{\{l, \dots, r\}\} \in \mathscr{B}_{l-1}$ and $f_{\mathcal{B} \backslash \{\{l,\dots,r\}\}}$ is within-group calibrated on $\cup_{i \leq l-1} \mathcal{X}_i$ and $a_{\{l,\dots,r\},z} = a_{\{l,\dots,r\}}$ for all $z \in \mathcal{Z}$. Let $\mathcal{B}' = \mathcal{B} \setminus \{\{l, \dots, r\}\}$. Since $\mathcal{B}$ covers $\{1, \dots, r\}$, then it holds that $\mathcal{B}'$ covers $\{1, \dots, l-1\}$ and hence $\mathcal{B}' \in \mathscr{B}_{l-1}$. Since $\mathcal{B}' \subset \mathcal{B}$ and $f_{\mathcal{B}}$ is within-group calibrated on $\cup_{i \leq r} \mathcal{X}_i$, then it holds that $f_{\mathcal{B}'}$ is within-group calibrated on $\cup_{i \leq l-1} \mathcal{X}_i$. Finally, since $\{l, \dots, r\} \in \mathcal{B}$, it also holds that $a_{\{l,\dots,r\},z} = a_{\{l,\dots,r\}}$.

Next, we prove the necessary condition, *i.e.*, given any $\mathcal{B} \in \mathscr{B}_r$, if $\exists l < r$ such that $\mathcal{B} \backslash \{\{l, \dots, r\}\} \in \mathscr{B}_{l-1}$ and $f_{\mathcal{B} \backslash \{\{l,\dots,r\}\}}$ is within-group calibrated on $\cup_{i \leq l-1} \mathcal{X}_i$ and $a_{\{l,\dots,r\},z} = a_{\{l,\dots,r\}}\ \forall z \in \mathcal{Z}$ then $f_{\mathcal{B}}$ is within-group calibrated on $\cup_{i \leq r} \mathcal{X}_i$. We need to show that, for every $\mathcal{A} \in \mathcal{B}$, it holds that $a_{\mathcal{A},z} = a_{\mathcal{A}}$. Let $\mathcal{B}' = \mathcal{B} \setminus \{\{l, \dots, r\}\}$. For every $z \in \mathcal{Z}$, it holds by assumption that $a_{\mathcal{A},z} = a_{\mathcal{A}}\ \forall \mathcal{A} \in \mathcal{B}'$ and $a_{\{l,\dots,r\},z} = a_{\{l,\dots,r\}}$. As a result, $f_{\mathcal{B}}$ is within-group calibrated on $\cup_{i \leq r} \mathcal{X}_i$.

D.8 Proof of Theorem C.3

To prove that Algorithm 3 returns the optimal $\mathcal{B}^*_{\text{cal}}$, if a solution exists, we just need to prove that, for every $r \in \{1, \dots, n\}$, the partition $\mathcal{B}_{\text{cal},r}$ the algorithm finds is optimal, *i.e.*, $\mathcal{B}_{\text{cal},r} = \mathcal{B}^*_{\text{cal},r}$. In what follows, we prove this by induction.

For the base case ($r = 1$), we have that $\mathcal{B}_{\text{cal},1} = \{\{a_1\}\}$ iff, for all $z \in \mathcal{Z}$ with $\rho_{z \mid 1} > 0$, it holds that $a_{1,z} = a_1$. This is clearly optimal since $\mathscr{B}_1$ only contains $\{\{a_1\}\}$. Otherwise, it holds that $\mathcal{B}_{\text{cal},1} = \emptyset$. As the induction hypothesis, assume that, for any $r' < r$, the partition $\mathcal{B}_{\text{cal},r'}$ the algorithm finds is either the optimal partition or, if there is no solution, an empty partition. Moreover, let $\mathcal{S}_r = \left\{i \in \{2, \dots, r\} \mid a_{\{i,\dots,r\},z} = a_{\{i,\dots,r\}}\ \forall z \in \mathcal{Z}\right\}$. Then, for r, we distinguish between two cases. If $\mathcal{B}_{\text{cal},r'}$ is empty for all $r' < r$, we again distinguish between two cases. If $a_{\{1,\dots,r\}} \neq a_{\{1,\dots,r\},z}\ \forall z \in \mathcal{Z}$, it means that $\mathcal{B}_{\text{cal},r} = \{\{1, \dots, r\}\}$ is the only partition in $\mathscr{B}_r$ that is within-group calibrated and thus it is optimal. Otherwise, we can conclude that no partition $\mathcal{B} \in \mathscr{B}_r$ is within-group calibrated and thus $\mathcal{B}_{\text{cal},r} = \emptyset$. Now, if $\mathcal{B}_{\text{cal},r'}$ is not empty for some $r' < r$, we need to show that $\mathcal{B}_{\text{cal},r} = \mathcal{B}_{\text{cal},k^*-1} \cup \{\{k^*, \dots, r\}\}$, with $k^* = \operatorname{argmax}_{k \in \mathcal{S}_r} |\mathcal{B}_{\text{cal},k-1}|$, is optimal.

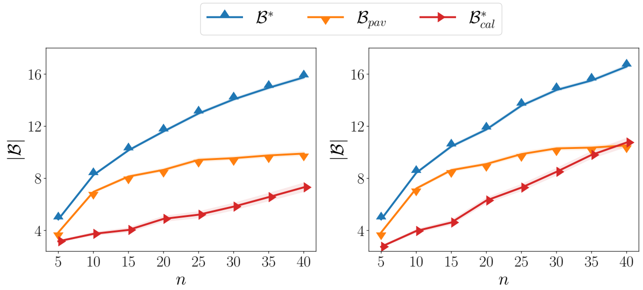
To this end, we first show that $f_{\mathcal{B}_{\text{cal},r}}$ is within-group calibrated on $\cup_{i \leq r} \mathcal{X}_i$. Using the induction hypothesis and the fact that $k^* \leq r$, we have that $\mathcal{B}_{\text{cal},k^*-1}$ is the optimal partition in $\mathscr{B}_{k^*-1}$. As a result, it follows from Lemma C.2 that $f_{\mathcal{B}_{\text{cal},r}}$ is within-group calibrated on $\cup_{i \leq r} \mathcal{X}_i$. Next, we show that $\mathcal{B}_{\text{cal},r} = \operatorname{argmax}_{\mathcal{B} \in \mathscr{B}_r} |\mathcal{B}|$ among those partitions $\mathcal{B}$ such that $f_{\mathcal{B}}$ is within-group calibrated. Using again Lemma C.2, we have that, for any $\mathcal{B}$ such that $f_{\mathcal{B}}$ is within-group calibrated, it holds that $\mathcal{B} = \mathcal{B}' \cup \{\{k, \dots, r\}\}$, with $\mathcal{B}' \in \mathscr{B}_{k-1}$, for some $k \in \mathcal{S}_r$. As a result, since $|\mathcal{B}| = |\mathcal{B}'| + 1$, it suffices to find $\mathcal{B}' = \text{argmax}_{\mathcal{B}'' \in \cup_{k \in \mathcal{S}_r} \mathscr{B}_{k-1}} |\mathcal{B}''|$ such that $f_{\mathcal{B}''}$ is within-group calibrated. Now, by the induction hypothesis, we know that, for each $\mathscr{B}_{k-1}$, $\mathcal{B}_{k-1}$ is the optimal partition. Then, since $k^* = \text{argmax}_{k \in \mathcal{S}_r} \left|\mathcal{B}_{\text{cal},k-1}\right|$, we can conclude that $\mathcal{B}_{\text{cal},r}$ is optimal.

D.9 Proof of Proposition C.4

We prove by contradiction. Assume there exists a $\mathcal{B} \in \mathscr{B}$ such that $f_{\mathcal{B}}$ is within-group calibrated. Then, for every $\mathcal{A} \in \mathcal{B}$, it must hold that $a_{\mathcal{A},z} = a_{\mathcal{A},z'} = a_{\mathcal{A}}$. Consider an arbitrary cell $\mathcal{A} \in \mathcal{B}$. We have that

$$a_{\mathcal{A},z} = \frac{\sum_{j \in \mathcal{A}} \rho_j \rho_{z \mid j} a_{j,z}}{\sum_{j \in \mathcal{A}} \rho_j \rho_{z \mid j}} \overset{(i)}{=} \frac{\sum_{j \in \mathcal{A}} \rho_j \rho_{z' \mid j} a_{j,z}}{\sum_{j \in \mathcal{A}} \rho_j \rho_{z' \mid j}} \overset{(ii)}{<} \frac{\sum_{j \in \mathcal{A}} \rho_j \rho_{z' \mid j} a_{j,z'}}{\sum_{j \in \mathcal{A}} \rho_j \rho_{z' \mid j}} = a_{\mathcal{A},z'}$$

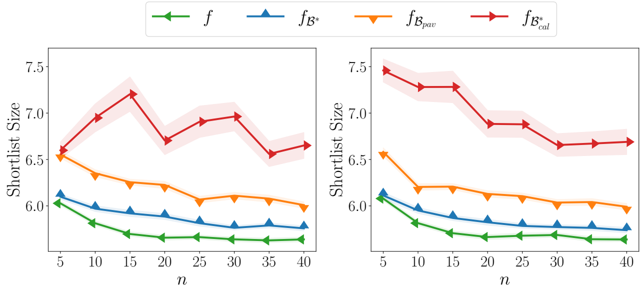
where (i) follows from the fact that $\rho_{z \mid i} = \rho_{z' \mid i}$ for all $i \in \text{Range}(f)$ and (ii) follows from the fact that, by assumption, $a_{i,z} < a_{i,z'}$ for all $i \in \{1, \dots, n\}$. As an immediate consequence, we have that $a_{\mathcal{A},z} < a_{\mathcal{A}} < a_{\mathcal{A},z'}$, contradicting the within-group calibration property.



(a) Citizenship status (Z)

(b) Race code (Z)

Figure 3: Size of the partitions $\mathcal{B}_{\text{pav}}$, $\mathcal{B}^*$ and $\mathcal{B}^*_{\text{cal}}$ returned by Algorithms 2, 1 and 3, respectively (higher is better).



(a) Citizenship status (Z)

(b) Race code (Z)

Figure 4: Size of the shortlists created using both the original classifier f and the modified classifiers $f_{\mathcal{B}^*}$, $f_{\mathcal{B}_{\text{pav}}}$ and $f_{\mathcal{B}^*_{\text{cal}}}$ for $k = 5$ (lower is better).

E ADDITIONAL EXPERIMENTS USING SURVEY DATA

First note that, since we find that, in most experiments, no within-group calibrated classifier exists, we allow $f_{\mathcal{B}^*_{\text{cal}}}$ to be within-group ϵ-calibrated[10] within Algorithm 3 and use binary search to find the smallest $\epsilon \in (0, 1)$ such that $f_{\mathcal{B}^*_{\text{cal}}}$ exists. Refer to Appendix E.3 for additional experiments on within-group ϵ-calibration.

E.1 Comparison of Partitions and Shortlist Sizes of Induced Screening Classifiers Corresponding to Algorithms 1, 2 and 3

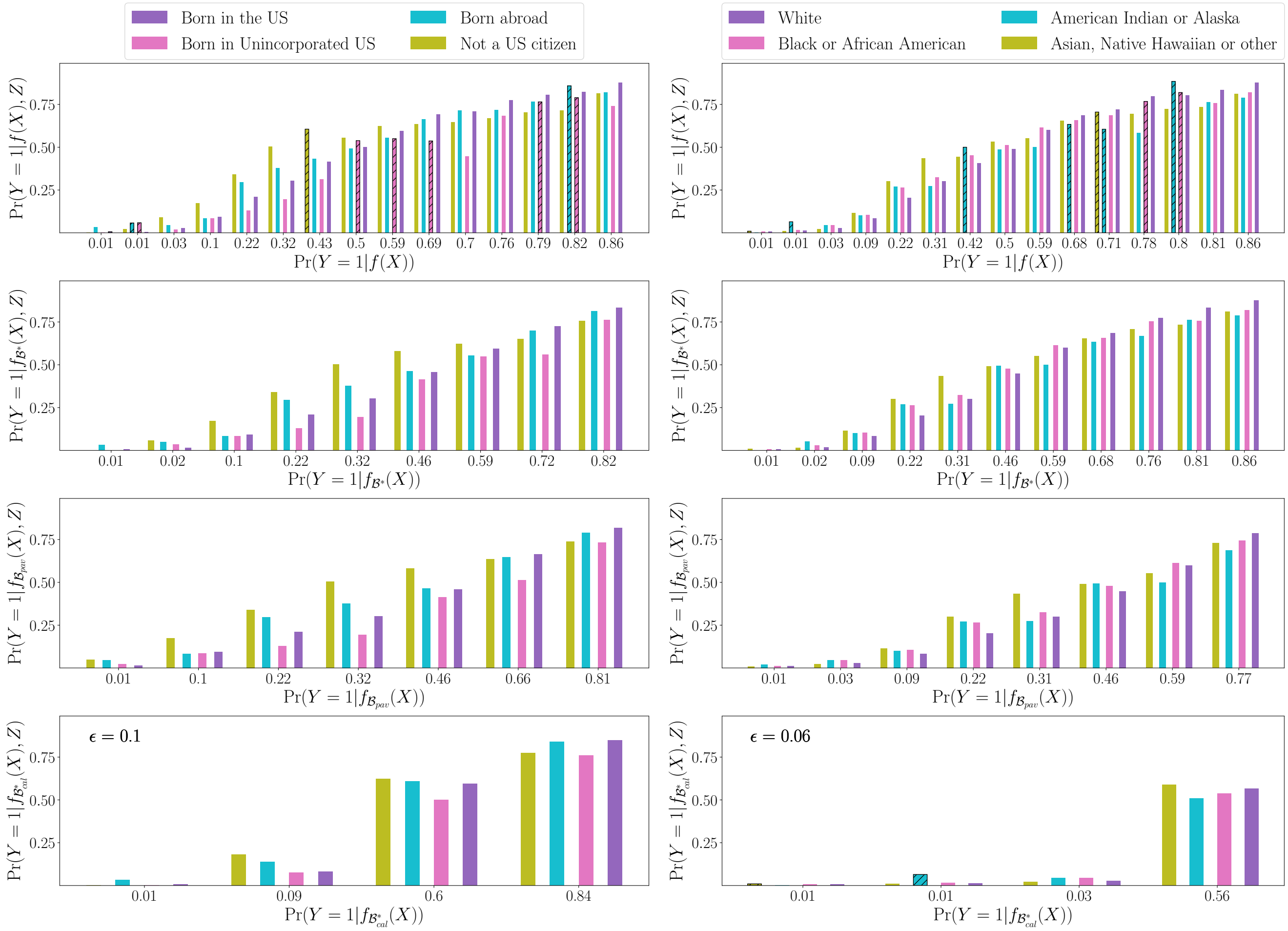
Algorithm 1 consistently provides larger partitions, which result in more fine-grained classifiers and smaller shortlists, than Algorithms 2 and 3. We experiment with several screening classifiers f with a varying number of bins n and compare the size of the partitions $\mathcal{B}$ provided by each of the algorithms, *i.e.*, the number of bins of the modified classifiers $f_{\mathcal{B}}$. Figure 3 shows that the optimal partition $\mathcal{B}^*$ is always greater in size than the partitions $\mathcal{B}^*_{\text{cal}}$ and $\mathcal{B}_{\text{pav}}$. Moreover, it also shows that, as n increases, the growth in the size of the partitions $\mathcal{B}^*$ and $\mathcal{B}_{pav}$ diminishes because the occurrence of within-group unfairness increases, as shown in Figure 2. Further, we use both the original classifier f and the modified classifiers $f_{\mathcal{B}^*}$, $f_{\mathcal{B}_{\text{pav}}}$ and $f_{\mathcal{B}^*_{\text{cal}}}$ to shortlist the minimum number of individuals among those in each of the simulated test pools $\{\mathcal{B}^i_{\text{pool}}\}$ such that, in expectation, there are at least k qualified shortlisted individuals per pool. To this end, for each test pool and classifier, we sort the candidates in decreasing order with respect to the corresponding quality score and, starting from the first, we keep shortlisting individuals in order until the sum of the quality scores reaches k (Appendix, A.3, [2]). Figure 4 shows that the shortlists created using $f_{\mathcal{B}^*}$ are consistently smaller than those created using $f_{\mathcal{B}_{\text{pav}}}$ and $f_{\mathcal{B}^*_{\text{cal}}}$ for $k = 5$. Moreover, it also shows that the price to pay for achieving within-group monotonicity, *i.e.*, the difference in size between the shortlists created using f and $f_{\mathcal{B}^*}$, is small. We found qualitatively similar results for other k values. Appendix E.2 takes a closer look at the (group conditional) score values of f, $f_{\mathcal{B}^*}$, $f_{\mathcal{B}_{\text{pav}}}$ and $f_{\mathcal{B}^*_{\text{cal}}}$.

E.2 Screening Classifiers Induced by the Partitions Found by Algorithms 1, 2 and 3

In this section, we take a closer look at all the quality score values $a = \Pr(Y = 1 \mid f(X) = a)$ and group conditional score values $a_z = \Pr(Y = 1 \mid f(X) = a, Z = z)$ of both the original classifier f and the modified classifiers $f_{\mathcal{B}}$ induced by the partitions $\mathcal{B}$ found by

[10] Given a set of groups $\mathcal{Z}$, a classifier f is within-group ϵ-calibrated iff, for every $z \in \mathcal{Z}$ and $a \in \text{Range}(f)$ such that $\Pr(Z = z \mid f(X) = a) > 0$, it holds that $|\Pr(Y = 1 \mid f(X) = a, Z = z) - a| \leq \epsilon$.

Algorithms 2, 1 and 3. Figure 5 summarizes the results for one experiment with a classifier f with $n = 15$, which reveal several interesting findings.



(a) Citizenship status (Z)

(b) Race code (Z)

Figure 5: Quality score values $a = P(Y = 1 \mid f(X) = a)$ and group conditional quality score values $a_z = P(Y = 1 \mid f(X) = a, Z = z)$ of the screening classifier f and the modified classifiers $f_{\mathcal{B}_{\text{pav}}}$, $f_{\mathcal{B}^*}$, and $f_{\mathcal{B}^*_{\text{cal}}}$ induced by the partitions found by Algorithms 2, 1 and 3, respectively. In the first and last rows, the hatched bars indicate within-group monotonicity violations and, in the last row, we report the smallest ϵ value such that a within-group ϵ-calibrated classifier $f_{\mathcal{B}^*_{\text{cal}}}$ exists.

As expected, $f_{\mathcal{B}^*}$ and $f_{\mathcal{B}_{\text{pav}}}$ are within-group monotone and $f_{\mathcal{B}^*}$ is more fine-grained than $f_{\mathcal{B}_{\text{pav}}}$, *i.e.*, $|\mathcal{B}^*| \geq |\mathcal{B}_{\text{pav}}|$. However, the minimum value of ϵ such that $f_{\mathcal{B}^*_{\text{cal}}}$ exists is not always low enough for $f_{\mathcal{B}^*_{\text{cal}}}$ to be within-group monotone. Moreover, we find that, for f, $f_{\mathcal{B}^*}$ and $f_{\mathcal{B}_{\text{pav}}}$, the difference among group conditional score values a_z for a given quality score values a is often significant. As a result, one should be cautious about comparing candidates from different groups z and instead utilize group-dependent decision thresholds [2] to implement more equitable hiring practices such as the Rooney rule [31], which requires that, when hiring for a given position, at least one (or more) candidate(s) from each minority group should be interviewed. In this context, it is also worth noting that, while using $f_{\mathcal{B}^*_{\text{cal}}}$ would mitigate such differences, our results show that this would reduce dramatically the granularity of the predictions. We found qualitatively similar results for different n values.

E.3 Additional Experiments On Within-Group ϵ-Calibration

In this section, we investigate how the smallest ϵ such that a within-group ϵ-calibrated classifier $f_{\mathcal{B}^*_{\text{cal}}}$ exists varies against the number of bins n of the screening classifier f. Figure 6 shows that, for each set of groups $\mathcal{Z}$, ϵ remains relatively constant with respect to n, however, the greater the difference across group conditional quality scores $a_z = P(Y = 1 \mid f(X) = a, Z = z)$, the greater the value of ϵ that is needed to obtain a within-group ϵ-calibrated classifier, as one may have perhaps expected.

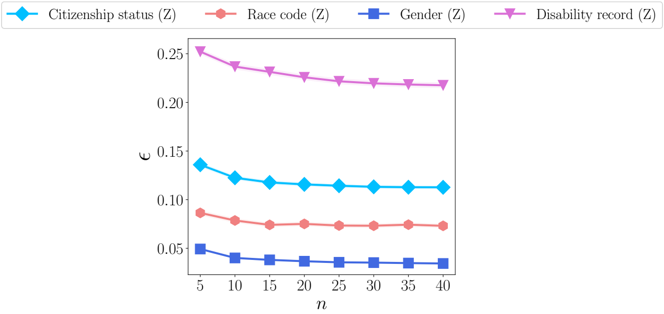


Figure 6: Minimum value of ϵ such that a within-group ϵ-calibrated $f_{\mathcal{B}^*_{\text{cal}}}$ exists against the number of bins n of the screening classifier f.

E.4 Experimental Results for Other Groups $\mathcal{Z}$

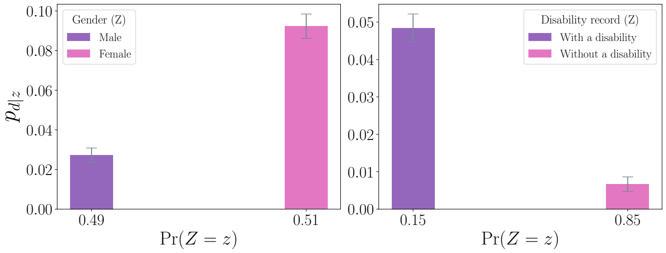
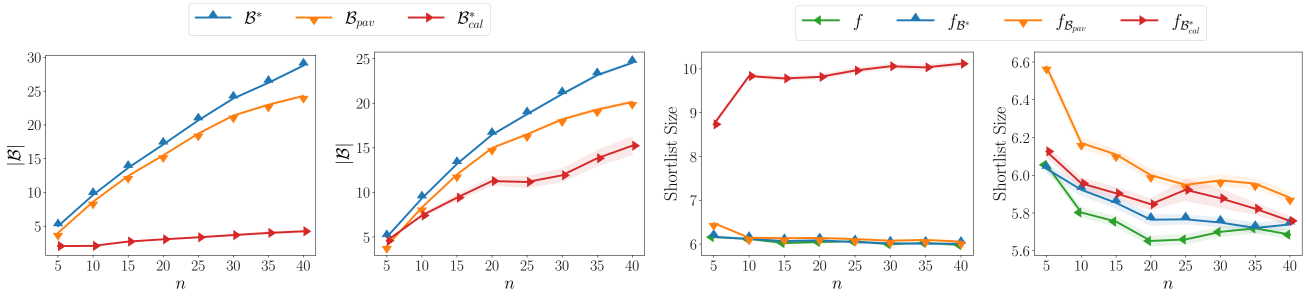


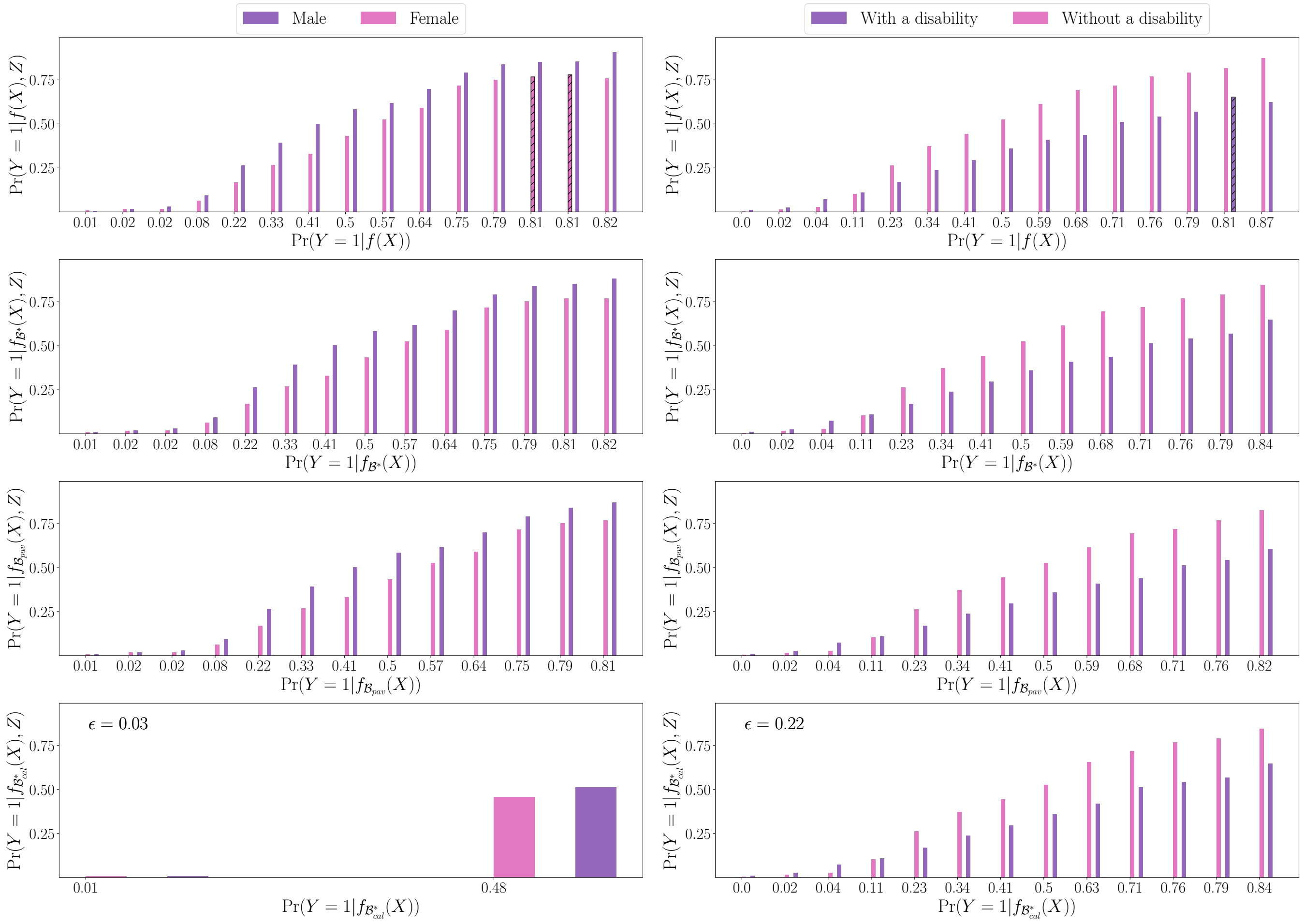
Figure 7: Probability $p_{d \mid z}$ that an individual from group z may suffer from within-group discrimination against $\Pr(Z = z)$ for $n = 15$.



(a) $|\mathcal{B}|$ vs. n

(b) Shortlist size vs. n

Figure 8: Quality of the partitions $\mathcal{B}_{\text{pav}}$, $\mathcal{B}^*$, and $\mathcal{B}^*_{\text{cal}}$ returned by Algorithms 2, 1 and 3, respectively, for screening classifiers f with an increasing number of bins n. Panel (a) shows the size $|\mathcal{B}|$ of the partitions provided by each algorithm (higher is better). Panel (b) shows the size of the shortlists created using the classifiers $f_{\mathcal{B}}$ induced by each partition $\mathcal{B}$ (lower is better).



(a) Gender (Z)

(b) Disability record (Z)

Figure 9: Quality score values $a = P(Y = 1 \mid f(X) = a)$ and group conditional quality score values $a_z = P(Y = 1 \mid f(X) = a, Z = z)$ of the screening classifier f and the modified classifiers $f_{\mathcal{B}_{\text{pav}}}$, $f_{\mathcal{B}^*}$, and $f_{\mathcal{B}^*_{\text{cal}}}$ induced by the partitions found by Algorithms 2, 1 and 3, respectively. In the first row, the hatched bars indicate within-group monotonicity violations and, in the last row, we report the smallest ϵ value such that a within-group ϵ-calibrated classifier $f_{\mathcal{B}^*_{\text{cal}}}$ exists.