Pushing the limits of fairness impossibility: Who's the fairest of them all?

Anonymous Author(s)
Affiliation
Address
email

Abstract

The impossibility theorem of fairness is a foundational result in the algorithmic fairness literature. It states that outside of special cases, one cannot exactly and simultaneously satisfy all three common and intuitive definitions of fairness - demographic parity, equalized odds, and predictive rate parity. This result has driven most works to focus on solutions for one or two of the metrics. Rather than follow suit, in this paper we present a framework that pushes the limits of the impossibility theorem in order to satisfy all three metrics to the best extent possible. We develop an integer-programming based approach that can yield a certifiably optimal post-processing method for simultaneously satisfying multiple fairness criteria under small violations. We show experiments demonstrating that our post-processor can improve fairness across the different definitions simultaneously with minimal model performance reduction. We also discuss applications of our framework for model selection and fairness explainability, thereby attempting to answer the question: *who's the fairest of them all?*

1 Introduction

While fairness in machine learning has received significant attention in recent years, most existing works focus on one of the many fairness criteria [3]. Consequently, practitioners are perhaps left with no choice but to use their best judgment to apply a single fairness criterion. We suspect that the conflicting nature of existing mathematical definitions of fairness might have led to this undesirable practice of narrowing down fairness-related measurement and mitigation to one chosen definition. This is somewhat analogous to the trade-off between precision and recall while evaluating model performance [7]. Instead of choosing one of precision or recall to evaluate the performance of a classification model, practitioners often evaluate the trade-off and choose models that can maintain a certain level of precision while optimizing recall or vice-versa [19]. In this paper, we provide a framework for explicitly addressing such trade-offs among multiple fairness criteria and model performance toward optimal model selection.

One of the most prolific examples of fairness in machine learning arose from the ProPublica recidivism study ([2]), in which a risk assessment tool called COMPAS was found to be biased against black defendants. But beyond the immediate implications in criminal justice, the study also prompted more general studies in algorithmic fairness and, in particular, led to a key result highlighted in [13], [21], and [25] which some colloquially refer to as the "impossibility theorem" in fairness [30, 24]. This theorem essentially states that three common definitions of algorithmic fairness - demographic parity [16], equalized odds [21], and predictive parity [3], cannot be simultaneously satisfied outside of pathological situations. Several works following these initial results have therefore focused on satisfying one metric ([21, 32]) or proposing adaptable methods for different metrics ([11, 37]). Yet, while we do not deny the conclusions of the impossibility theorem, we also believe there have not been sufficient efforts to reconcile the conflicting fairness definitions to the best extent possible. We

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

fill this gap in the literature by translating the trade-offs among multiple fairness criteria and model performance into a constrained optimization problem and propose a post-processing methodology for simultaneously achieving approximate fairness in the conflicting definitions simultaneously.

We believe our framework would alleviate the practitioner from making the hard choice in choosing a particular metric. Instead, if they have a partial ordering of importance amongst the metrics (which many possess), our framework would explicitly allow them evaluate such trade-offs. The application which we highlight in Section 4 discusses these in detail. Overall, we make three main contributions in this work:

1. We design a flexible optimization framework that returns a post-processing score transformation function that can make scores group-wise ϵ-fair along three definitions (demographic parity, equalized odds, and predictive rate parity) simultaneously. This framework can be applied to any binary classifier that produces a continuous score, can be configured for singular or multiple metrics of fairness, and also can account for fairness vs. performance trade-offs in terms of ROC-AUC.
2. We present a novel reformulation of this non-convex optimization problem as a Mixed Integer Linear Program (MILP) [41]. This reformulation allows us to find provably globally optimal solutions. We further show that in practice, we can consistently find better solutions through our global optimization method compared to local optimization methods in a reasonable time.
3. We discuss and extend our framework from a post-processing mechanism to a tool that can aid practitioners in better understanding their data and models' empirical fairness characteristics and trade-offs and compare these traits across models.

The rest of the paper is organized as follows. In Section 2 we mathematically define fairness metrics and the multiple fairness optimization problem. We discuss the optimal solution via MILP in Section 3. We try to answer the question of "who's the fairest of them all?" through our applications and experiments in Section 4. Finally, we conclude with a discussion in Section 5. We wrap up this section with a discussion of the related literature.

Related Work: Early works [13] exploring the conflicts between fairness definitions prove that for a binary predictor, predictive rate parity conflicts with equalized odds ([21]) unless base rates are equal or the model is perfectly predictive. Chouldechova [13] also considers trade-offs between the fairness definitions in the binary prediction case. Kleinberg et al. [25] generalizes this study by showing that statistical (i.e., demographic) parity is also inconsistent with predictive rate parity and equalized odds. The same paper studies these inconsistencies in a more general bin-wise prediction setting and shows that approximate fairness definitions (predictive rate parity, equalized odds) can simultaneously hold but only under ϵ-approximate equal base rates or ϵ-approximate perfect performance. This work further proves that there is an inherent trade-off between fairness and loss.

Beyond impossibility theorem results, several works have focused on trade-offs between fairness and model performance [21, 17, 37, 42]. They develop in-processing solutions aimed at reducing one metric while maintaining accuracy. A few authors have analyzed trade-offs or attempted to achieve multiple fairness. One example is Pleiss et al. [36], which shows that predictive rate parity and a relaxed form of equalized odds are reconcilable under a randomized prediction scheme. Another notable example is Celis et al. [11], which develops a flexible in-processing approach to achieve multiple types of fairness (potentially at the same time). Like Celis et al. [11], the framework we develop also aims for a flexible approach to focus on one or many fairness metrics simultaneously. However, our method is distinct in that it is a post-processing based solution and also more general as it works for continuous scores (rather than binary classification). Our work is most closely related to Nandy et al. [32] in terms of the underlying score transformation mechanism, and we leverage some of their methods. However, whereas [32] only targets equalized odds, we go further and include demographic parity and predictive rate parity in our framework—this posits computational challenges, which we address by proposing novel methodology based on integer programming.

As noted at the end of [25], some open questions are how to optimally assign scores to satisfy multiple criteria when base rates are equal and additionally, how to satisfy predictive rate parity and either equal TPR or TNR when one cost outweighs the other. To our knowledge, no prior work has attempted to reconcile all three fairness conditions (demographic parity, equalized odds, and predictive rate parity) simultaneously with model performance through a post-processing framework. Although the post-processing methods tend to be less flexible for fairness-performance trade-offs

than their in-processing counterparts, they can be much more easily added on top of any existing model training pipeline. This makes a post-processing approach modular, and in particular, more appealing in complex web-scale recommender systems that use (a combination of) certain prediction scores to rank a list of items.

2 Multiple Fairness Optimization

We consider a binary classification problem where the i-th observation is characterized by their label $y_i \in \{0, 1\}$, their group membership $g_i \in \mathcal{G}$, and model predicted probability (also known as a risk score [25]) $s_i \in [0, 1]$ for $i = 1, \ldots, N$. The corresponding random variables are denoted as Y, G and S respectively. To set up the problem, we discretize the scores into nonempty bins $b \in \mathcal{B} := \{1, \ldots, |\mathcal{B}|\}$ by using, for example, a quantile transformation (we will denote $|\mathcal{B}|$ as B). Additionally, let $N_{b+}^{[g]}$ denote the number of group g positive class ($y_i = 1$) instances in bin b, $N_b^{[g]}$ denote the total number of instances of group g instances in bin b, $N_+^{[g]}$ ($N_-^{[g]}$) be the total number of group g positive (negative) instances, and $N^{[g]}$ be the total number of group g instances. Lastly, our approach seeks to achieve fairness by moving instances from one bin to another bin: hence, we define variable $x_{bb'}^{[g]}$ as the probability of moving an instance of group attribute g and score in bin b into a new bin b' [1]. In other words, for every group g, the collection $\{x_{bb'}^{[g]}\}_{b,b'}$ can be represented as a $B \times B$ transition matrix (with additional constraints, as discussed next).

For the optimization framework and the remainder of this paper, we translate a single fairness definition (e.g. equal true positive rate) as a constraint that controls for the worst-case violations across all bins. Below, we discuss different fairness constraints that we consider in our framework, and describe how they can be represented in terms of the optimization variables $\{x_{bb'}^{[g]}\}$.

2.1 Fairness Constraints Under Binning Framework

Demographic Parity (DP): For simplicity, we assume that there are exactly two groups $g \in \{1, 2\}$ as we formulate the fairness metrics, although everything discussed below can be applied for non-binary groups by using pairwise constraints, which we elaborate on in Appendix A. Starting with demographic or statistical parity from [16], this condition states that the model's predicted score is independent of group membership. This is equivalent to

$$P(S = s \mid G = 1) = P(S = s \mid G = 2).$$

Our version of this constraint uses bins B to empirically approximate the probability $P(S = s \mid G = g)$ and we also relax the equality to an ϵ-approximate equality (for some pre-specified $\epsilon > 0$). Therefore, after transforming the scores using $\{x_{bb'}^{[g]}\}_{b,b'}$ the ϵ_{DP}-approximate DP can be expressed as the following as a linear constraint:

$$\left| \frac{1}{N^{[1]}} \sum_{b \in \mathcal{B}} x_{bb'}^{[1]} N_b^{[1]} - \frac{1}{N^{[2]}} \sum_{b \in \mathcal{B}} x_{bb'}^{[2]} N_b^{[2]} \right| \leq \epsilon_{DP} \qquad \forall \ b' \in \mathcal{B}, \tag{1}$$

where $N_b^{[g]}$ denote the number of observations from group g in bin b (before transformation), and $N^{[g]} = \sum_{b \in \mathcal{B}} N_b^{[g]}$. For reproducibility, ϵ_{DP} should be chosen to be larger than the approximation error $O(1/\sqrt{N^{[g]}})$ for replacing $P(S = s \mid G = g)$ with its empirical counterpart.

Equalized Odds (EOdds): The equalized odds condition for binary predictors, given in Hardt et al. 21, is a balance condition where the groups must have equal true positive and false positive rates. For continuous scores, it translates to having equal score distributions for each group conditional on their true labels [32]:

$$P(S = s \mid Y = y, G = 1) = P(S = s \mid Y = y, G = 2) \qquad \text{for } y \in \{0, 1\}.$$

[1] In applications, we can discretize the scores into B bins with a quantile discretizer and consider how we can move them across bins. More bins allow for more granular interpretation of the transformed scores at the cost of us solving a harder problem and vice versa.

Like demographic parity, our empirical score bin version requires that the distribution of positive or negative instances be ϵ_{EOdds}-approximately equal between groups in the new bins b'. Both equal true positive rate and false positive rate can be expressed as linear constraints, respectively:

$$\begin{aligned} &\left| \frac{1}{N_+^{[1]}} \sum_{b \in \mathcal{B}} x_{bb'}^{[1]} N_{b+}^{[1]} - \frac{1}{N_+^{[2]}} \sum_{b \in \mathcal{B}} x_{bb'}^{[2]} N_{b+}^{[2]} \right| \leq \epsilon_{EOdds} \ \ \forall \ \ b' \in \mathcal{B} \\ &\left| \frac{1}{N_-^{[1]}} \sum_{b \in \mathcal{B}} x_{bb'}^{[1]} N_{b-}^{[1]} - \frac{1}{N_-^{[2]}} \sum_{b \in \mathcal{B}} x_{bb'}^{[2]} N_{b-}^{[2]} \right| \leq \epsilon_{EOdds} \ \ \forall \ \ b' \in \mathcal{B}. \end{aligned} \tag{2}$$

Predictive Rate Parity (PRP): Lastly, we examine the predictive rate parity condition popularized in [13]. This condition states that the probability of being a positive instance is independent of group membership when we condition on the score. Formally:

$$P(Y = 1 \mid S = s, G = 1) = P(Y = 1 \mid S = s, G = 2).$$

Using the empirical score bin framework, an approximate version of the above implies that the proportion of positive instances in each bin must be ϵ_{PRP}-approximately equal among groups:

$$\left| \frac{\sum_{b \in \mathcal{B}} x_{bb'}^{[1]} N_{b+}^{[1]}}{\sum_{b \in \mathcal{B}} x_{bb'}^{[1]} N_{b}^{[1]}} - \frac{\sum_{b \in \mathcal{B}} x_{bb'}^{[2]} N_{b+}^{[2]}}{\sum_{b \in \mathcal{B}} x_{bb'}^{[2]} N_{b}^{[2]}} \right| \leq \epsilon_{PRP} \quad \forall \ \ b' \in \mathcal{B}. \tag{3}$$

Unlike constraints (1) and (2), which can be expressed as a linear function of the optimization variables $\{x_{bb'}^{[g]}\}$, condition (3) yields bilinear terms and is in general a non-convex constraint. A main technical difficulty of our framework arises from this non-convex fairness constraint—Section 3 presents an integer programming framework to handle this non-convexity, ensuring we can obtain a globally optimal solution to the resulting optimization problem.

Remark. Our definition of fairness as the worst-case violation across all bins aims to resemble approximations of the respective probabilistic definitions but we have not found identical definitions in other works. We comment on the differences and discuss why it does not contradict the traditional impossibility theorem of [25] in Appendix F.

2.2 MFOpt: Multiple Fairness Optimization Framework

We use constraints developed in Section 2.1 to state the multiple fairness optimization (MFOpt) problem:

$$\begin{aligned} \underset{\{x_{bb'}^{[g]}\}_{b,b',g}}{\text{minimize}} \quad & \sum_{g \in \mathcal{G}} \sum_{b \in \mathcal{B}} \sum_{b' \in \mathcal{B}} \left| \frac{N_b^{[g]}}{N} (\bar{s}_b - \bar{s}_{b'}) x_{bb'}^{[g]} \right| && \text{(4a)} \\ \text{s.t.} \quad & \sum_{b \in \mathcal{B}} x_{bb'}^{[g]} = 1 \qquad \forall \ \ b' \in \mathcal{B}, \ g \in \mathcal{G} && \text{(4b)} \\ & x_{bb}^{[g]} \geq 1 - \xi \qquad \forall \ \ b \in b' \in \mathcal{B}, \ g \in \mathcal{G} && \text{(4c)} \\ & x_{bb'}^{[g]} = 0 \qquad \forall \ \ b' \ s.t. \ |b' - b| \geq w, \ \forall g \in \mathcal{G} && \text{(4d)} \\ & \text{Fairness Constraints:} \quad (1), (2), (3) && \text{(4e)} \\ & \frac{\sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b+}^{[g]}}{\sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b}^{[g]}} \leq \frac{\sum_{b \in \mathcal{B}} x_{b(b'+1)}^{[g]} N_{b+}^{[g]}}{\sum_{b \in \mathcal{B}} x_{b(b'+1)}^{[g]} N_{b}^{[g]}} \qquad \forall \ \ b' \in \{1, ..., B-1\}, \ g \in \mathcal{G} && \text{(4f)} \\ & 0 \leq x_{bb'}^{[g]} \leq 1, \ \ \forall \ \ b, b', g. && \text{(4g)} \end{aligned}$$

Above, the optimization variables are the group-specific movement probability $x_{bb'}^{[g]}$ terms and all remaining terms are problem data and/or configurable hyperparameters. Starting with the objective (4a), we define $\bar{s}_b$ as the midpoint score in the bin and hence the objective is the product of the movement distance $\bar{s}_b - \bar{s}_{b'}$ weighed by the fraction of total samples moved $N_b^{[g]}/N$ and the amount of movement $x_{bb'}^{[g]}$. (4b) states that the total movement out of bin b, including the movement back to itself, must sum up to 1 and along with (4g) ensures that $x_{bb'}^{[g]}$ represent probabilities in a transition

matrix. (4c) states that the total movement from bin b back to itself must be lower bounded by hyperparameter ξ. This parameter controls how far we allow the new scores to stray from the original and is necessary to prevent zero denominators in 3 and 4f. Constraint (4d) represents window constraints to restrict extreme movements of scores beyond w bins. Constraint (4e) are the fairness constraints in Section 2.1. (4f) ensures that we preserve the rank-ordering of the scores and expected values, which is desirable for comparing bins against each other. While constraints (4d), (4f) are not fairness-related, we add them to retain the utility of the solution. The predictive rate parity constraint (3) and (4f) both introduce non-convexities into Problem (4). These two constraints also require us to assume overlap, or that each bin contains at least one member from each group. Without overlap, predictive rate parity is undefined since it is not possible to compare expectations across groups for a given bin and demographic parity is likely violated as it means one group has 0 probability of landing in a given bin.

We close this section with two major benefits of this framework compared to inprocessing solutions such as adding fairness regularization ([42, 17, 37]) or using an entirely different fairness-based model ([11], [43]). First, the number of optimization variables scales in the order of $\mathcal{O}(|\mathcal{G}||\mathcal{B}|^2)$ while the number of constraints scale in the order of $\mathcal{O}(|\mathcal{B}||\mathcal{G}|^2)$. This is significant as it entails that our framework can be applied in arbitrarily large data settings as long as score-based binning is possible. Applying the transformation is equally tractable, as it only requires binning observations and making independent draws from a multinomial distribution with B possible outcomes.

The second benefit of our framework is that it returns a highly interpretable solution as it returns one optimized $B \times B$ transition matrix per group. Hence given a newly scored instance, several facts can be read from the corresponding row of the matrix such as the likelihood of moving to a specific bin b, moving into any higher or lower bin, etc. These probabilities can also be controlled via constraints as shown with the window constraints (4d) and max movement constraints (4c). This interpretability is an advantage over model regularization frameworks, where it is difficult to know how an individual's score might change when switching from the base model to a fair model.

3 Finding Optimal Solutions via Mixed Integer Programming (MIP)

The primary difficulty of the above optimization problem are the predictive rate parity constraint (3) and rank-order constraint (4f) which turn the problem non-convex. Non-convex constrained optimization is generally NP-hard and traditional methods that seek locally optimal solutions include gradient-based interior point optimization ([40]), sequential quadratic programming ([18]), or algorithms specific to quadratically-constrainted-quadratic-programs (QCQPs) such as operator splitting methods, semidefinite relaxations, etc. (see [33] for an overview).

Rather than pursuing a locally optimal solution, we propose a novel reformulation of the problem into a tractable mixed-integer-linear-program (MILP) which can be solved to global optimality ([10], [9]). Our reformulation grants two benefits over traditional locally optimal solvers. First, global strategies theoretically enable us to find the best possible solution. Second, as our problem primarily scales with the number of bins, it is practically tractable when utilizing the power of modern MIP solvers.

3.1 Tractable reformulations for computational efficiency

We first observe that a direct reformulation of the fractional terms into bilinear terms in constraints (3) and (4f) will lead to bilinear terms in the order of $\mathcal{O}(B^3)$. We show that we can reduce this to $\mathcal{O}(B)$ bilinear terms through a substitution that exploits the problem structure. Next, we take advantage of the vastly reduced number of bilinear terms to apply the normalized multiparametric disaggregation technique (NMDT, [1]) which we explain in Section 3.1.2. This allows us to approximate products of continuous variables as products of binary variables, which can be easily linearized and handled by MIP solvers. Importantly, this transformation of xy (i.e, product) terms requires upper and lower bounds for x and y and we propose a method in Section 3.1.2 for generating and tightening these bounds by solving fractional linear program subproblems ([12]). Taken together, the reduction of bilinear terms combined with the bound-tightening procedure enable us to effectively apply the NMDT methodology and transform the problem from a non-convex QCQP to an MILP that can be solved to global optimality.

3.1.1 Step 1: Reducing number of bilinear variables

We reduce the number of bilinear variables in our problem by making a substitution for the fraction term by introducing new optimization variables $v_b^{[g]} \geq 0$ (to represent a sum) and $t_b^{[g]} \geq 0$ (to represent the fractional quantity) as additional variables (see Appendix D for more details). We can then use them to write equivalent constraints with only $\mathcal{O}(B)$ bilinear terms. Let,

$$v_{b'}^{[g]} = \sum\nolimits_{b \in \mathcal{B}} x_{bb'}^{[g]} N_b^{[g]} \qquad \text{and} \qquad t_{b'}^{[g]} v_{b'}^{[g]} = \sum\nolimits_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b+}^{[g]} \qquad \forall \ b' \in \mathcal{B}.$$

Then we have the following:

$$\text{Constraint (3)} \iff \left| t_{b'}^{[1]} - t_{b'}^{[2]} \right| \leq \epsilon_{PRP} \ \ \forall \ b' \in \mathcal{B},$$
$$\text{Constraint (4f)} \iff t_{b'}^{[g]} \leq t_{b'+1}^{[g]} \ \ \forall \ b' \in \{1, \ldots, B-1\}$$

3.1.2 Step 2: NMDT and bound tightening through fractional LP subproblems

Linearizing bilinear terms (NMDT): We show how we can model each bilinear term $t_b^{[g]} v_b^{[g]}$ by using a binary expansion for the continuous variables $t_b^{[g]}, v_b^{[g]}$, and by observing that the product of binary variables can be modeled via integer programming (see [27, 39] for reference). To this end, we make use of the NMDT transformation [1]. We recap this method below as formulated in [1][2] Given any bounded optimization variable $x \in [x_L, x_U]$, and precision factor p, a negative integer, we can represent this variable exactly as $x = (x_U - x_L)\lambda + x_L$ where

$$\lambda = \sum_{l \in \{-p, \ldots, -1\}} 2^l z_l + \Delta\lambda$$

where $0 \leq \Delta\lambda \leq 2^p$ is a remainder term and $z_l \in \{0, 1\}$ are binary optimization variables. Dropping the remainder term $\Delta\lambda$ gives us the approximate form and product forms of xy become dot products of several integer variables, which can be effectively handled via modern MIP solvers, such as [20]. Although we are solving an approximation (e.g. precision of $1e^{-4}$) this is not a practical problem since it is precise enough for reasonable choices of ϵ and we do not expect the constraints to hold exactly when we apply the post-processor on the testing data.

Bounds on $v_b^{[g]}, t_b^{[g]}$ via Fractional LPs: A key requirement to apply NMDT is that all optimization variables in the bilinear terms ($t_b^{[g]}$ and $v_b^{[g]}$ in our case) must be bounded and we need to be able to accurately estimate these bounds (i.e., tighter bounds leads to faster runtimes [1]). We propose obtaining these lower and upper by minimizing and maximizing respectively, the sub-problems while keeping all fairness constraints except the quadratic constraints. Firstly, note that bounds on $v_b^{[g]}$ can be solved as a simple LP (details in Appendix D). Meanwhile, obtaining a bound on $t_b^{[g]}$ requires solving a nonlinear problem due to fact that $t_b^{[g]}$ represents a fractional objective (ratio of two affine terms of optimization variables). However, we observe that we can apply the Charnes-Cooper transformation [12] to reformulate the nonlinear problem into a simple LP (details in Appendix D).

3.2 Choice of algorithm: QCQP (heuristic) vs MIP (optimal solution)

In this section we show the results and benefits of our reformulation from a non-convex QCQP to an MILP. In Table 1, we take each dataset, create a 60/40 train-test split, train a grid-searched random forest model, and score the training data. Next, we discretize the scores into bins, parameterize the problem (# bins, ϵ, max movement, window size, solve time) on the scored training data, and compare our MILP solution solved by Gurobi ([20]) against the QCQP problem solved by IPOPT ([28]), which is a generic interior-point log-barrier penalty method for nonlinear constrained optimization. We discuss the datasets and problem parameters for all experiments in the Appendix E. For each metric such as AUC, we use AUC_{INT} and AUC_{IP} to denote the average result of applying the interior-point (INT, for short) or integer programming (IP) method, respectively. The metrics used are the

[2]The open-source implementation can be found in `https://github.com/joaquimg/QuadraticToBinary.jl` (MIT License) which we utilize.

objective value, optimality gap ($\%\Delta$), [3] and AUC[4]. We also report the statistical significance of the improvement based the p-value from the Wilcoxon signed-rank test to determine if $\%\Delta_{IP} \leq \%\Delta_{INT}$ is a consistent result. Bold figures indicate statistical significance w.r.t. 1 standard deviation[5] [6].

Table 1: Interior Point Solution vs. MIP Solution

Dataset	Obj_{INT}	Obj_{IP}	$\%\Delta_{INT}$	$\%\Delta_{IP}$	p-value	AUC_{INT}	AUC_{IP}
ACS Income	2.0809	1.9682	15.076 ± 6.461	10.621 ± 3.402	**0.0029**	0.9041	0.9044
ACS Insurance	0.9769	0.9599	3.432 ± 0.225	**1.715 ± 0.169**	**0.0010**	0.7411	0.7413
ACS Mobility	2.4580	2.3781	5.37 ± 0.803	**2.193 ± 0.138**	**0.0010**	0.7971	0.7973
ACS Poverty	2.0693	2.0526	3.756 ± 0.435	**2.972 ± 0.324**	**0.0010**	0.8440	0.8440
ACS Coverage	8.9361	1.9665	79.711 ± 0.782	**7.878 ± 2.207**	**0.0010**	0.5420	0.8149
ACS Travel	2.3935	2.3859	2.554 ± 0.254	2.242 ± 0.28	**0.0010**	0.7725	0.7725
Heart Disease	1.8871	1.3035	26.385 ± 17.401	**3.81 ± 0.864**	**0.0010**	0.8302	0.8629
COMPAS	7.4551	3.1300	62.88 ± 13.407	**17.055 ± 7.482**	**0.0010**	0.5143	0.7378

As the results show, the MIP reformulation consistently beats the interior point solver applied on the raw optimization problem, even when solving for only 10 minutes. We also observe that in most cases, regardless of the method we choose, we can quickly find near optimal solutions that are high performing in the AUC sense. This is a significant result as it means that even a locally optimal solution to our optimization problem can yield a practically useful post-processing result. Additional experiments on these datasets showing the effectiveness of MFOpt with respect to performance and fairness in a training/testing data scenario can be found in Appendix H.

We conclude this section by reiterating two benefits of the reformulation. First, solving a MIP method yields lower bounds that can be used to prove optimality or otherwise gauge the quality of a feasible solution. Second, by framing the problem as a MIP, we can always theoretically continue improving the solution to optimality based on the acceptable time limits. There are other applicable global optimization methods, such as spatial branch and bound. We considered these solutions but ultimately opted for a MIP approach due to the maturity and availability of solvers (see Appendix G for details).

4 Applications

In this section, we illustrate a few methods of applying our framework and how it can be used to help model developers select and understand models from a fairness perspective. To apply these procedures, we first require developing an efficient frontier of fairness solutions to understand which ϵ configurations are feasible for a given model type and dataset. To generate this frontier, we solve the problem over a grid of parameters $\epsilon_{DP}, \epsilon_{EOdds}, \epsilon_{PRP}$. Each feasible solution will yield a point $s \in \{(AUC, \epsilon_{DP}, \epsilon_{EOdds}, \epsilon_{PRP})\}$ and the collection of non-dominated points from the solution set yields a efficient frontier. We show the 2-d profile shots of our 4-d fairness surface in Figure 1 as an example, where the axes represent one of three fairness metrics and the point color gradient represents AUC. In theory, we could obtain a true Pareto-optimal frontier since we have devised a method of obtaining globally optimal solutions. However, we generate this frontier using IPOPT due to practical limitations as we are solving a $7 \times 7 \times 7$ grid of ϵ parameters.

4.1 Understanding fairness tradeoffs

After the Pareto frontier is generated, the modeler can pick an operating point s based on the desired AUC and tolerable fairness violations ϵ. However, when communicating fairness properties to stakeholders and accounting for potential changes in strategy, it can be helpful to additionally understand the cost of further increasing fairness in terms of $\epsilon_{DP}, \epsilon_{EOdds}, \epsilon_{PRP}$.

[3] The optimality gap is defined $\%\Delta = \frac{UpperBound - LowerBound}{UpperBound}$ where the upper bound is the best feasible solution and the lower bound is produced by the branch-and-bound method. The INT method does not have the benefit of providing lower bounds, hence we use the bound produced by the IP method to compute this.

[4] We describe how we computed the AUC for the bin-wise probabilities in Appendix C.

[5] We are limited in the number of trials we can run and actively chose to prioritize the variety of datasets we apply on rather than a large number of trials for a single dataset. As such, we expect relatively large standard errors but we reflect the consistency of our method through the p-value.

[6] See Appendix E for an explanation of the stark underperformance of IPOPT on the ACS Coverage and COMPAS datasets.

Figure 1: Efficient frontier of solutions for ACS West Insurance data

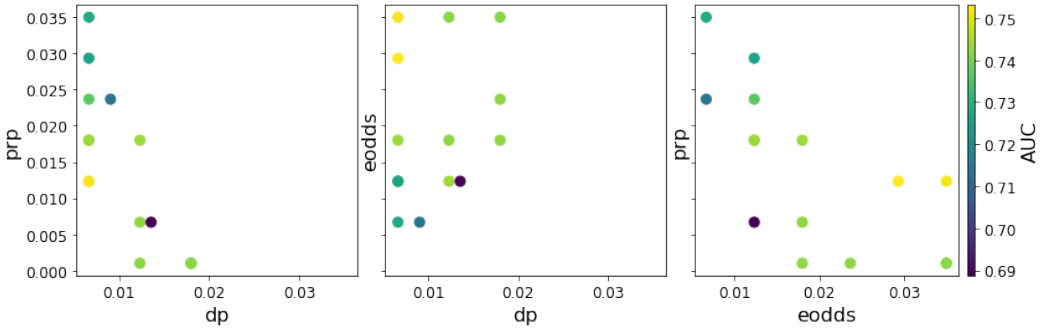

Table 2: Performance Fairness Trade-off Analysis

Trade...	For...	s^*_{AUC}	$s^*_{\epsilon_{DP}}$	$s^*_{\epsilon_{EOdds}}$	$s^*_{\epsilon_{PRP}}$
Base	Base	0.7434	0.0123	0.018	0.0123
AUC	ϵ_{DP}	-	-	-	-
AUC	ϵ_{Eodds}	-	-	-	-
AUC	ϵ_{PRP}	0.7422	0.0123	0.0180	0.0067
ϵ_{DP}	ϵ_{PRP}	0.7436	0.0152	0.0123	0.0067
ϵ_{Eodds}	ϵ_{PRP}	-	-	-	-
ϵ_{Eodds}	ϵ_{DP}	0.7436	0.0152	0.0123	0.006

This tradeoff can be understood by looking at the characteristics of points on the frontier near the operating point. Suppose we are at an operating point s. If we want to trade AUC for ϵ_{EOdds}, then we would find a point s' with at least as good $\epsilon_{DP}, \epsilon_{PRP}$ but worse AUC and better ϵ_{EOdds}. More generally, if we pick trade performance/fairness characteristic c (cost) for characteristic b (benefit), then we hold all other factors constant and find a point with better b and worse c. We illustrate this in the Table 2. The first row shows a hypothetical operating point while the following rows show other points on the efficient frontier that we could move to when we make a certain trade. Blank rows indicate that no such point was found that permits the desired trade-off. This trade-off perspective can enable developers to better understand and communicate the costs to performance or other fairness metrics when trying to close the disparity in one fairness metric.

4.2 Performance Comparison

Lastly, we compare our framework against two methods and show that we can satisfy fairness constraint(s) just as well or better, while obtaining significantly stronger performance. Our comparison is done as follows (details in E), in each iteration we randomly split and pre-process the data, tune and fit a random forest model, and score the training and testing data to get the base scores $\hat{y}_0$. Next, we run the methodology that we are comparing against (i.e. build a model or apply the postprocessor) to get method scores $\hat{y}_m$. We then bin the outputs of the base model and compared method and compute the AUC along with the fairness metrics (ϵ_0, ϵ_m). Next, we solve our constrained optimization problem on the training data, where we set the parameters ϵ to $\frac{1}{2}min(\epsilon_0, \epsilon_m)$. After optimizing, we can apply the optimal solutions $x^{[g]}_{bb'}$ to assign new bins in the testing data based on the original score bins. One method of assigning a group g instance with score $s \in b$ (denote as $s^{[g]}_b$) would be to randomly draw from a multinomial distribution parameterized by probabilities $(x^{[g]}_{b1}, x^{[g]}_{b2}, \ldots, x^{[g]}_{b|\mathcal{B}|})$. We propose alternative methods to this stochastic assignment method in the Appendix B. Lastly, we compute the resulting AUC and fairness metrics on remapped bins for the testing data (and do the same for $\hat{y}_0$ and $\hat{y}_m$ on the testing data). These figures are shown in Table 3. We only show the results on the test set due to space constraints and have placed the results for the training set in Appendix E. Bold figures indicate that a metric is statistically significant to 1 standard deviation.

First, we compare our framework against the in-processing framework in Rezaei et al. [37], which is a robust optimization-based logistic regression model for reducing equality of opportunity violation. We found that this method works well compared to standard logistic regression and managed to decrease

Table 3: Comparison with other fairness methods

Method	Metric	Testing Data Base	Testing Data Method	Testing Data MF-Opt
Rezaei	AUC	0.7471 ± 0.003	0.6619 ± 0.0022	0.747 ± 0.003
	ϵ_{DP}	0.0117 ± 0.0014	0.0124 ± 0.0013	**0.0088 ± 0.001**
	ϵ_{EOdds}	0.0266 ± 0.007	0.0291 ± 0.0059	**0.0167 ± 0.0029**
	ϵ_{PRP}	0.109 ± 0.0145	0.1091 ± 0.0143	0.0986 ± 0.0133
Pleiss	AUC	0.8319 ± 0.0033	0.8149 ± 0.0087	0.831 ± 0.0032
	ϵ_{DP}	0.0212 ± 0.0016	0.0137 ± 0.0016	**0.0106 ± 0.0011**
	ϵ_{EOdds}	0.0329 ± 0.0042	0.023 ± 0.0038	**0.0142 ± 0.0028**
	ϵ_{PRP}	0.1465 ± 0.0178	0.4147 ± 0.1537	0.1547 ± 0.0293

fairness violations while maintaining the similar performance. It improves in ϵ_{Eodds} compared to a random forest as well. However, its weakness is that the underlying model is still logistic regression and therefore has limited expressiveness. In comparison, MFOpt can be applied on top of any model class and thus the performance advantages of more flexible models are better maintained.

Next, we compare our framework against the post-processing framework in Pleiss et al. [36]. In this method, the authors use randomization to maintain the model's calibration while simultaneously satisfying a relaxed equalized odds condition (whereby a linear combination of TPR and TNR are satisfied). Again, we see that this method maintains close performance as the base model, successfully shrinks equalized odds violations, and even decreases demographic parity violations too. However, it results in large violations of predictive rate parity based on our definition in contrast to our method.

4.3 Who's the fairest of them all?

In industry, machine learning model selection is guided by many factors including performance, speed, interpretability, among others. Yet, the fairness dimension is commonly overlooked unless the developers specifically induce it in their model. Even then, picking a specific fairness metric to optimize for can be a nebulous task. Rather than focus on a single metric, we propose a simple and intuitive method of gauging a model's efficiency in trading between different fairness definitions.

To do so, we first construct the frontier for the two models in question and then filter all points on the efficient frontier with tolerable performance $AUC \geq AUC_{min}$. Next, we find the point on the respective frontiers with minimum Euclidean distance to the origin. The model with the shorter distance to their frontier can then be declared as the model that has better tradeoff properties. We do not show an example due to lack of space, but remark that this procedure and the trade-off analysis can be useful when a developer is iterating between models that were not designed for fairness, but wants a model that can be flexibly made more fair through the postprocessing framework that we propose. As fairness requirements may change over time, the model that can yield the best tradeoffs between different definitions can offer the most overall utility.

5 Discussion

In our study, we have devised a flexible, tractable, and interpretable post-processing method which we apply to push the limits of the impossibility theorem. We show that while theoretical limitations remain undisputed, there is a path forward to practically reconciling the conflicting fairness definitions. These results extend the findings of [38], which state that the trade-off of fairness and accuracy are negligible in practice. Our work reinforces this claim but also adds on that trade-offs between fairness definitions can be negligible as well. For further research, one area is to improve the consistency of the PRP violation reduction, as we observed the largest standard error in reducing this metric. This could be due to us using random forests for all experiments, which is known to be an uncalibrated model [6]. One method to address this is therefore first calibrating the model through other methodologies such as Platt scaling ([35]) or binning-based calibration ([26]) before applying MFOpt. We could also consider incorporating uncertainty in the training data through principled approaches such as stochastic or robust optimization.

References

[1] T. Andrade, F. Oliveira, S. Hamacher, and A. Eberhard. Enhancing the normalized multiparametric disaggregation technique for mixed-integer quadratic programming. *J. of Global Optimization*, 73(4):701–722, apr 2019. ISSN 0925-5001. doi: 10.1007/s10898-018-0728-9. URL https://doi.org/10.1007/s10898-018-0728-9.

[2] J. Angwin, J. Larson, S. Mattu, and L. Kirchner. Machine bias, May 2016. URL https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

[3] S. Barocas, M. Hardt, and A. Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2019. http://www.fairmlbook.org.

[4] R. K. E. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilovic, S. Nagar, K. N. Ramamurthy, J. Richards, D. Saha, P. Sattigeri, M. Singh, K. R. Varshney, and Y. Zhang. AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias, Oct. 2018. URL https://arxiv.org/abs/1810.01943.

[5] K. Bestuzheva, M. Besançon, W.-K. Chen, A. Chmiela, T. Donkiewicz, J. van Doornmalen, L. Eifler, O. Gaul, G. Gamrath, A. Gleixner, L. Gottwald, C. Graczyk, K. Halbig, A. Hoen, C. Hojny, R. van der Hulst, T. Koch, M. Lübbecke, S. J. Maher, F. Matter, E. Mühmer, B. Müller, M. E. Pfetsch, D. Rehfeldt, S. Schlein, F. Schlösser, F. Serrano, Y. Shinano, B. Sofranac, M. Turner, S. Vigerske, F. Wegscheider, P. Wellner, D. Weninger, and J. Witzig. The SCIP Optimization Suite 8.0. Technical report, Optimization Online, December 2021. URL http://www.optimization-online.org/DB_HTML/2021/12/8728.html.

[6] H. Boström. Calibrating random forests. In *2008 Seventh International Conference on Machine Learning and Applications*, pages 121–126, 2008. doi: 10.1109/ICMLA.2008.107.

[7] M. Buckland and F. Gey. The relationship between recall and precision. *Journal of the American society for information science*, 45(1):12–19, 1994.

[8] S. Burer and A. Saxena. The milp road to miqcp. In J. Lee and S. Leyffer, editors, *Mixed Integer Nonlinear Programming*, pages 373–405, New York, NY, 2012. Springer New York. ISBN 978-1-4614-1927-3.

[9] P. M. Castro. Tightening piecewise mccormick relaxations for bilinear problems. *Computers & Chemical Engineering*, 72:300–311, 2015. ISSN 0098-1354. doi: https://doi.org/10.1016/j.compchemeng.2014.03.025. URL https://www.sciencedirect.com/science/article/pii/S0098135414001069. A Tribute to Ignacio E. Grossmann.

[10] P. M. Castro. Normalized multiparametric disaggregation: An efficient relaxation for mixed-integer bilinear problems. *J. of Global Optimization*, 64(4):765–784, apr 2016. ISSN 0925-5001. doi: 10.1007/s10898-015-0342-z. URL https://doi.org/10.1007/s10898-015-0342-z.

[11] L. E. Celis, L. Huang, V. Keswani, and N. K. Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. FAT* '19, page 319–328, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361255. doi: 10.1145/3287560.3287586. URL https://doi.org/10.1145/3287560.3287586.

[12] A. Charnes and W. W. Cooper. Programming with linear fractional functionals. *Naval Research Logistics Quarterly*, 9(3-4):181–186, 1962. doi: https://doi.org/10.1002/nav.3800090303. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800090303.

[13] A. Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2):153–163, 2017. doi: 10.1089/big.2016.0047. URL https://doi.org/10.1089/big.2016.0047.

[14] F. Ding, M. Hardt, J. Miller, and L. Schmidt. Retiring adult: New datasets for fair machine learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 6478–6490. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/32e54441e6382a7fbacbbbaf3c450059-Paper.pdf.

[15] I. Dunning, J. Huchette, and M. Lubin. Jump: A modeling language for mathematical optimization. *SIAM Review*, 59(2):295–320, 2017. doi: 10.1137/15M1020575.

[16] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

[17] B. Fish, J. Kun, and Ádám D. Lelkes. *A Confidence-Based Approach for Balancing Fairness and Accuracy*, pages 144–152. doi: 10.1137/1.9781611974348.17. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611974348.17.

[18] P. E. Gill and E. Wong. Sequential quadratic programming methods. 2012.

[19] M. Gordon and M. Kochen. Recall-precision trade-off: A derivation. *Journal of the American Society for Information Science*, 40(3):145–151, 1989.

[20] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2022. URL https://www.gurobi.com.

[21] M. Hardt, E. Price, E. Price, and N. Srebro. Equality of opportunity in supervised learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 3315–3323. Curran Associates, Inc., 2016.

[22] Q. Huangfu and J. A. J. Hall. Parallelizing the dual revised simplex method, 2015. URL https://arxiv.org/abs/1503.01889.

[23] A. Janosi, W. Steinbrunn, M. Pfisterer, R. Detrano, and D. Aha. Adult dataset, 1996. URL https://archive.ics.uci.edu/ml/datasets/adult.

[24] K. Karthik. The impossibility theorem of machine fairness - A causal perspective. *CoRR*, abs/2007.06024, 2020. URL https://arxiv.org/abs/2007.06024.

[25] J. Kleinberg, S. Mullainathan, and M. Raghavan. Inherent Trade-Offs in the Fair Determination of Risk Scores. In C. H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 43:1–43:23, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-029-3. doi: 10.4230/LIPIcs.ITCS.2017.43. URL http://drops.dagstuhl.de/opus/volltexte/2017/8156.

[26] A. Kumar, P. S. Liang, and T. Ma. Verified uncertainty calibration. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, 2019.

[27] H.-L. Li and C.-T. Chang. An approximate approach of global optimization for polynomial programming problems. In *European Journal of Operations Research*, pages 625–632. European Journal of Operations Research Volume 107, 1996.

[28] R. Lougee-Heimer. The common optimization interface for operations research: Promoting open-source software in the operations research community. *IBM Journal of Research and Development*, 47(1):57–66, 2003. doi: 10.1147/rd.471.0057.

[29] G. P. Mccormick. Computability of global solutions to factorable nonconvex programs: Part i – convex underestimating problems. *Math. Program.*, 10(1):147–175, dec 1976. ISSN 0025-5610. doi: 10.1007/BF01580665. URL https://doi.org/10.1007/BF01580665.

[30] T. Miconi. The impossibility of "fairness": a generalized impossibility result for decisions, 2017. URL https://arxiv.org/abs/1707.01195.

[31] H. Nagarajan, M. Lu, E. Yamangil, and R. Bent. Tightening mccormick relaxations for nonlinear programs via dynamic multivariate partitioning. In M. Rueher, editor, *Principles and Practice of Constraint Programming*, pages 369–387, Cham, 2016. Springer International Publishing. ISBN 978-3-319-44953-1.

[32] P. Nandy, C. Diciccio, D. Venugopalan, H. Logan, K. Basu, and N. E. Karoui. Achieving fairness via post-processing in web-scale recommender systems, 2020. URL `https://arxiv.org/abs/2006.11350`.

[33] J. Park and S. Boyd. General heuristics for nonconvex quadratically constrained quadratic programming, 2017. URL `https://arxiv.org/abs/1703.07870`.

[34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[35] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press, 1999.

[36] G. Pleiss, M. Raghavan, F. Wu, J. Kleinberg, and K. Q. Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems*, pages 5680–5689, 2017.

[37] A. Rezaei, R. Fathony, O. Memarrast, and B. Ziebart. Fairness for robust log loss classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):5511–5518, apr 2020. doi: 10.1609/aaai.v34i04.6002. URL `https://doi.org/10.1609%2Faaai.v34i04.6002`.

[38] K. T. Rodolfa, H. Lamba, and R. Ghani. Empirical observation of negligible fairness–accuracy trade-offs in machine learning for public policy. *Nature Machine Intelligence*, 3(10):896–904, oct 2021. doi: 10.1038/s42256-021-00396-x. URL `https://doi.org/10.1038%2Fs42256-021-00396-x`.

[39] J. P. Teles, P. M. Castro, and H. A. Matos. Univariate parameterization for global optimization of mixed-integer polynomial problems. *European Journal of Operational Research*, 229(3): 613–625, 2013. ISSN 0377-2217. doi: https://doi.org/10.1016/j.ejor.2013.03.042. URL `https://www.sciencedirect.com/science/article/pii/S0377221713002750`.

[40] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106:25–57, 2006.

[41] L. A. Wolsey and G. L. Nemhauser. *Integer and combinatorial optimization*, volume 55. John Wiley & Sons, 1999.

[42] M. B. Zafar, I. Valera, M. G. Rogriguez, and K. P. Gummadi. Fairness Constraints: Mechanisms for Fair Classification. In A. Singh and J. Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 962–970. PMLR, 20–22 Apr 2017. URL `https://proceedings.mlr.press/v54/zafar17a.html`.

[43] B. H. Zhang, B. Lemoine, and M. Mitchell. Mitigating unwanted biases with adversarial learning. AIES '18, page 335–340, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450360128. doi: 10.1145/3278721.3278779. URL `https://doi.org/10.1145/3278721.3278779`.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] In the discussion 5
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] In Section E
   (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? [N/A] We do not derive any theoretical results and all claims (e.g. NP-hardness of non-convex programs) are either well-known or a source is referenced.
   (b) Did you include complete proofs of all theoretical results? [N/A]
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] Yes, all code is included in a zip file uploaded as the supplementary materials. Instructions and data used in all experiments are clearly outlined in E and 4
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Referenced in E
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (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] Referenced in E
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] License and references can be found in the Appendix Section E
   (b) Did you mention the license of the assets? [Yes] License are mentioned alongside open-source package references
   (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? [Yes] Under the Experimental Data Section E
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]

A Mapping from bins to scores

Extending the MFOpt framework to non-binary groups requires creating pairwise constraints for all groups in each bin to achieve the same fairness effect. This means that the number of constraints scaled in the order of $\mathcal{O}(|\mathcal{B}||\mathcal{G}|^2)$ where in practice, we expect $|G|$ to be in the range of 3-4. This pairwise procedure does not affect the scaling of the number of optimization variables, which remains at $\mathcal{O}(|\mathcal{B}|^2|\mathcal{G}|)$. However, we note that the scaling factor of the constraints is not a major bottleneck due to our problem reformulation as presented in Section 3. Without the reformulation, the pairwise constraints would generate $\mathcal{O}(|\mathcal{B}|^3|\mathcal{G}|^2)$ bilinear terms from the cross-multiplication of the numerator/denominator sums and return a much more computationally intensive problem.

B Mapping from bins to scores

As mentioned, the most straightforward method of applying the score transformation after solving the optimization problem is to sample from a multinomial distribution. However, this is a less granular approach as we are assuming that all observations in the bin are indistinguishable. To overcome this, we recommend the idea proposed in ([32]) which is a linear projection. This strategy proposes that if an observation with score s falling into a bin a with upper and lower bounds $[a_l, a_u]$ gets mapped from the random draw into a new bin b_1 with bounds $[b_{1l}, b_{1u}]$, then we assign it a linearly interpolated score given by:

$$s' = b_{1l} + \frac{s - a_l}{a_u - a_l}(b_{1u} - b_{1l})$$

This allows us to maintain rank-ordering of scores that receive the same assignment from a to b.

A more deterministic manner of mapping from bin to score would be to take the expected score mapping. After solving the optimization problem, we know the transitions probabilities a to $\{b_1, b_2, \ldots, b_B\}$ (denoted as $P(a \longrightarrow b_i)$) based on the optimization variables and from the previous method, we also know the score assignment if a were moved into b_i (denote as s_i). Hence, a deterministic map would transform score s to $s' = \sum_{i \in B} s_i P(a \longrightarrow b_i)$.

C Computing AUC from bins and using AUC as an objective

Another idea we leverage from ([32]) is the Riemann approximation of AUC from the bins. Essentially, ROC AUC be approximated by the FPR at bin k and TPR of the cumulative bins $b \in \{k, \ldots, B\}$. Another consideration is that we could have changed the objective to maximizing AUC rather than minimizing score movement. However, in our experience, maximizing AUC (quadratic objective) as the objective led to a harder time finding better feasible solutions compared to minimizing score movement (linear objective).

D Details on the fractional LP subproblem for bound tightening

We elaborate on the methodology in Section 3.1.2. Recall that our goal is to find bounds for:

$$v_{b'}^{[g]} = \sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_b^{[g]} \qquad \text{and} \qquad t_{b'}^{[g]} v_{b'}^{[g]} = \sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b+}^{[g]} \qquad \forall \ b' \in \mathcal{B}.$$

Where $t_{b'}^{[g]}$ is meant to represent the fractional quantity:

$$t_{b'}^{[g]} = \frac{\sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b+}^{[g]}}{\sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_b^{[g]}} = \frac{\sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b+}^{[g]}}{v_{b'}^{[g]}}$$

We will do this by fixing $\bar{g}$ and $\bar{b}$ such that we first tighten bounds for $v_{\bar{b}}^{\bar{g}}$ and then use the optimal solution to tighten bounds for $t_{\bar{b}}^{\bar{g}}$. First, it is easy to see that maximizing/minimizing $v_{\bar{b}}^{\bar{g}}$ is an LP as we have dropped the quadratic constraints, leaving us with a linear objective and linear constraint set. Now let $v_{\bar{b},min/max}^{\bar{g}*}$ represent the optimal values of the min/max objective for $v_{\bar{b}}^{\bar{g}}$. We now turn

to bounding $t_{\bar{b}}^{\bar{g}}$ which has the same linear constraints but a fractional (nonlinear) objective. To deal with this, we utilize the Charnes-Cooper transformation ([12]). Essentially, this reformulation trick handles the denominator by removing it from the objective and passing it to all constraints while maintaining linearity. To illustrate this in detail, we first define new optimization variables:

$$\xi_{bb'}^{[g]} = \frac{x_{bb'}^{[g]}}{\sum_{b\in\mathcal{B}} x_{b\bar{b}}^{[\bar{g}]} N_b^{[\bar{g}]}} \qquad \phi_{\bar{b}}^{[\bar{g}]} = \frac{1}{\sum_{b\in\mathcal{B}} x_{b\bar{b}}^{[\bar{g}]} N_b^{[\bar{g}]}} \tag{5}$$

Using (5), we can express the min/max problem for $t_{\bar{b}}^{[\bar{g}]}$ as problem (6).

$$\begin{aligned}
\underset{\xi_{bb'}^{[g]},\phi}{\text{Min or Max}} \quad & t_{\bar{b}}^{[\bar{g}]} = \sum_{b\in\mathcal{B}} N_{b+}^{[\bar{g}]} \xi_{b\bar{b}}^{[\bar{g}]} \\
\text{subject to} \quad & \sum_{b\in\mathcal{B}} \xi_{b\bar{b}}^{[g]} N_b^{[\bar{g}]} = 1 \\
& \xi_{bb'}^{[g]} \geq (1-m)\phi \qquad \forall \;\; b = b' \\
& \xi_{bb'} = 0 \qquad \forall \;\; b' \; s.t. \; |b'-b| \geq w \\
& \left| \frac{1}{N^{[1]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[1]} N_b^{[1]} - \frac{1}{N^2} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[2]} N_b^{[2]} \right| \leq \epsilon_{DP}\phi \qquad \forall \;\; b' \in B \\
& \left| \frac{1}{N_+^{[1]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[1]} N_{b+}^{[1]} - \frac{1}{N_+^{[2]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[2]} N_{b+}^{[2]} \right| \leq \epsilon_{EOdds}\phi \qquad \forall \;\; b' \in B \\
& \left| \frac{1}{N_-^{[1]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[1]} N_{b-}^{[1]} - \frac{1}{N_-^{[2]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[2]} N_{b-}^{[2]} \right| \leq \epsilon_{EOdds}\phi \quad \forall \;\; b' \in B \\
& \frac{1}{v_{b'max}^{g*}} \leq \phi \leq \frac{1}{v_{b'min}^{g*}} \qquad 0 \leq \xi_{bb'}^{[g]} \leq \frac{1}{v_{b'min}^{g*}}
\end{aligned} \tag{6}$$

By solving these subproblems and taking the objective value as bounds for $t_b^{[g]}$, we can reduce the feasible region of the problem and enhance our solutions. The sub-problems are bounded and if any of them are infeasible, then it also implies that the MFOpt problem is also infeasible as we drop the PRP constraints in these sub-problems:

E Experiment data descriptions and problem parameters

We use three primary data sources for our experiments, the more recently developed American Community Survey (ACS) data as well as two more classical datasets, Heart Disease and COMPAS. We elaborate on each dataset in this section. The ACS data is a dataset made publicly available by the US Census Bureau. Specifically, Ding et. al [14] have created an excellent Python package[7] that enables users to pull model-ready data (for a requested year and geographic region) for a set of pre-defined binary classification tasks, such as predicting high income, health insurance coverage, whether they move or not, among others. The tasks are detailed in the paper and we use all of the pre-defined tasks without any additional modification except for Employment. We do not use the Employment task because of the assumption detailed in Section 2.2 regarding overlap. Experiments with this task occasionally yielded models that did not have overlap which made this task unsuitable for demonstrating our methodology. We reiterate that this is not a practical issue if one just ignored the non-overlapping bins, but requires a lengthy and technical fairness interpretation that we felt were beyond the purpose of our study. In terms of time and geography, we use 2020 data for all experiments while the geography varies. In the experiments shown on Table 1, we use the West Coast US states (California, Oregon, Washington). In 3 we wanted a larger dataset as we required a sufficiently large testing split, hence we used the West Coast States ('CA', 'OR', 'WA', 'NV', 'AZ') with the "ACS Mobility" dataset and a 60/40 train-test split for the inprocessing comparison and East

[7] https://github.com/zykls/folktables (MIT License)

Table 4: Experiment Problem Parameters

# Trials	Bins	ϵ	Max Movement	Window Size	Solve Time	Precision
10	50	0.03	0.5	13	600s	1e-5

Table 5: Comparison with other fairness methods

Method	Metric	Base	Train Method	MF-Opt
Rezaei	AUC	0.7471 ± 0.003	0.6619 ± 0.0022	0.747 ± 0.003
	ϵ_{DP}	0.0117 ± 0.0014	0.0124 ± 0.0013	0.0088 ± 0.001
	ϵ_{EOdds}	0.0266 ± 0.007	0.0291 ± 0.0059	0.0167 ± 0.0029
	ϵ_{PRP}	0.109 ± 0.0145	0.1091 ± 0.0143	0.0986 ± 0.0133
Pleiss	AUC	0.8314 ± 0.0045	0.8145 ± 0.0104	0.8306 ± 0.0044
	ϵ_{DP}	0.0208 ± 0.0029	0.0145 ± 0.0023	0.0105 ± 0.0008
	ϵ_{EOdds}	0.0325 ± 0.0062	0.0257 ± 0.005	0.0144 ± 0.0017
	ϵ_{PRP}	0.1405 ± 0.0214	0.4149 ± 0.1824	0.1319 ± 0.0204

Coast States ('ME','NH','MA','RI', 'CT','NY', 'NJ', 'DE','MD','VA', 'NC','SC','GA','FL') and same with the "ACS Poverty" dataset for the postprocessing comparison. There was no particular reason for selecting these geographies aside from obtaining a large enough sample that we can feasibly run multiple trials on. Though we are using census data there is no PII information nor any endangerment to the subjects in the data. However, we note that in practice, it is important to exercise caution and equity in picking groups to mitigate for, as selective mitigation of favored groups by a malicious practitioner can result in underperformance for deserving groups.

The Heart Disease Dataset ([23]) is a publicly available dataset where the task is to predict whether or not an individual has heart disease. Most applications of this data use the standard processed "Cleveland" data and we use sex as the group variable. We could not find a standard and preprocessed version of this data and did it ourselves by one-hot-encoding categorical variables.

The COMPAS dataset is based on the recidivism study noted in ([2]). We use the preprocessed version made available in the publicly available AIF360 package ([4]) [8] without any additional modification. In this dataset, the task is to predict whether or not an individual will recidivate and we use ethnicity as the group variable.

We list the problem parameters used to create the results in Table 1 in Table 4. We use the same parameters for all tasks. A particular note is that in Table 1, the IPOPT method performs very poorly compared on the COMPAS and ACS Coverage data. This is because across the 10 trials we ran, the IPOPT algorithm had frequently failed to converge within the 10-minute time limit for these two datasets. From the outputs, we saw that convergence failure is accompanied by heavy violation of the predictive rate parity constraint (demographic parity and equalized odds are still satisfied) and a high objective value. The exact reason for frequent failure in these two datasets is unclear, however, we hypothesize that it is due to a relatively high predictive rate parity gap in the data which led to numeric issues. Such failures were not observed in the IP formulation while both solvers were provided the exact same problem parameters.

All experiments were run on a MacBook Pro with a 2.4GHz 8-Core Intel Core i9 processor with 32 GB RAM. We did not use the GPU for solving. Data, preprocessing steps, and the random forest models utilize Python's scikit-learn ([34], BDS License) package. The optimization model is coded through Julia's JuMP package ([15] MPL License). We use the Gurobi ([20] Academic License) and IPOPT ([28] Eclipse Public License) solvers for all problems.

Due to lack of space, we only showed the method comparison Table 3 for the testing data. We show the results on the training data in Table 5.

[8] https://github.com/Trusted-AI/AIF360

F Comparison to other fairness definitions

We compare our bin-wise worst-case fairness definition with other fairness definitions seen in literature and explain why it does not contradict previous impossibility theorem results. First, since we are considering score bins, our definition is a generalization of the definitions in ([13]), ([21]) and ([11]), which consider fairness metrics for binary $\{0, 1\}$ classifiers or assume that there is a threshold for mapping probabilities to 0,1 outcomes. In these cases, the overall FPR/FNR can be computed and EOdds refers to the equality of those rates. Our framework is a generalization since the same fairness metrics for binary classification can be achieved by specifying that all scores be moved into exactly one of two bins (representing {0,1} predictions) under our framework.

Since we are dealing with binned scores, our fairness definitions more resemble those seen in [25], which also has a notion of binned "risk assignments". The critical difference in fairness definitions is that Kleinberg's paper utilize the sum of scores in each bin compared against the number of positive or negative instances. Under this scheme, predictive rate parity refers to having the sum of scores be equal to the number of positive instances and true positive rate refers to the expected score of the positive instances in each bin (and analogously with FPR). The key difference is that rather than the sum of scores, our definitions are based on the expected number of {0,1} instances moved into each bin, irrespective of the instance's original scores. As such, we are not faced with the same strict fairness trade-offs.

G Commentary on other solution methods and solvers

While investigating a solution to our nonconvex problem, we considered another global integer programming based approach known as spatial branch and branch (SBB), which relies on a combination of spatial partitioning and solving local partitions using McCormick relaxations ([29], [9]) and other outer approximation variants ([8]). In our testing, Gurobi's nonconvex QCQP solver, which applies these SBB heuristics, worked remarkably well despite being relatively new and was sometimes able to beat both the interior point solution and the MIP solution. Open-source solver SCIP ([5]) also features a gender nonconvex SBB solver that works reasonably well. However, our main goal was to provide a widely accessible method of solving the problem to global optimality and as of writing, there are significantly more developed open-source MILP solvers, such as SCIP, HiGHS ([22]), and CBC ([28]), than SBB solvers. Another reason we opted for the MILP approach is that we saw more potential in the NMDT reformulation for taking advantage of our reformulation and bound tightening procedure. Nonetheless, our bound tightening procedure is theoretically beneficial for both methods and as other open-source algorithms/solvers for SBB become more developed, such as Couenne ([28]) and Alpine ([31]), we encourage a future re-evaluation of solution methods and comparisons.

Finally, we note that we chose Gurobi in our experiments for its speed and effectiveness since we are repeatedly solving many problems. We acknowledge that Gurobi is a very powerful commercial solver and the results solved over 10 minutes may be worse with open source solvers such as HiGHS ([22]). Nonetheless, the important fact is that all MIP solvers target the global optimum and hence even less powerful solvers can yield strong solutions given more time.

H Additional Experiments Using Expected Assignment on Testing Data

We list additional experiments focusing on the performance of our method on the testing data in Tables 6 to 11. All results are based on 20 trials that are run with a similar procedure in the comparison section 4 except that we compute the testing metrics based on expected assignment rather than stochastic assignment (explained below), which we feel better reflects the *average* performance of MFOpt. In each trial, we tune a random forest via grid-search, find the base fairness violations, set up the parameters of MFOpt to reduce the violations by a half, and optimize. Then, we compute the results (AUC and fairness violations) on the testing data (baseline with no modifications vs. MFOpt) and show the 1-Standard deviation error margins as well as the p-value corresponds to a one-sided Wilcoxon signed rank test which evaluates if the distribution of differences of the Base - MFOpt stats

(higher AUC, lower fairness violations) is symmetric around zero (null) or instead favors the base (alternative).

The major difference in this evaluation compared to the results shown in section 4 is that in section 4, we assign observations to bins in the testing data based on random draws from a multinomial distribution (explained in Appendix B). However, a single draw per train-test split may not properly reflect the expected performance of MFOpt, even if we average over 20 trials. Instead, we feel a more accurate representation of the expected performance is if we apply the expected bin assignment to obtain the post-movement number of $\{0, 1\}$ and total samples for each bin in the testing set. Under this procedure, we do not move individual observations across bins, but rather move all of them together. Concretely, suppose that we have optimized parameters $(\hat{x}^{[g]}_{1b'}, \hat{x}^{[g]}_{2b'}, \ldots, \hat{x}^{[g]}_{Bb'})$, which represented the probabilities of observations from each bin moving into bin b'. Then we propose that the number of $\{1\}$ outcomes at bin b' after the expected bin assignment procedure is:

$$\hat{N}^{[g]}_{b'+} = \sum_{b \in \mathcal{B}} \hat{x}^{[g]}_{bb'} N^{[g]}_{b+}$$

The same computation applies for the expected number of negative samples $\hat{N}^{[g]}_{b'-}$ and total samples $\hat{N}^{[g]}_{b'}$, which we use to compute the AUC and fairness violations and report in the tables. We find that across different datasets, the decrease in AUC is miniscule in terms of both absolute amount and variance (less than 1%). We obviously do not expect better AUC from the MFOpt solution compared to the unconstrained model and thus this result is remarkable as it indicates that some degree of fairness can be afforded practically for free under our framework. The second observation is that we can reduce all three fairness metrics simultaneously and consistently across all datasets, as we find p-values below 0.05 in all cases. We do observe variance of the PRP violation is relatively higher than that of DP or EOdds. We noted this in our conclusion Section 5 as an area for future work and provide some hypotheses for methods that can address this inconsistency. Nonetheless, improving conflicting definitions of fairness simultaneously is another significant result as it provides empirical evidence that there is a path forward towards multiple fairness.

Table 6: ACS West Travel

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.7439 ± 0.0039	0.7437 ± 0.0039	0.999999
DP	0.0313 ± 0.0057	0.0208 ± 0.0045	0.000001
EOdds	0.0404 ± 0.0055	0.0268 ± 0.0081	0.000001
PRP	0.1743 ± 0.0326	0.1481 ± 0.0306	0.000001

Table 7: ACS West Income

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.8907 ± 0.0012	0.8902 ± 0.0016	1.000000
DP	0.0172 ± 0.0021	0.0126 ± 0.0019	0.000001
EOdds	0.0362 ± 0.006	0.0231 ± 0.0035	0.000001
PRP	0.1994 ± 0.0745	0.1507 ± 0.0295	0.000024

Table 8: ACS West Mobility

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.7413 ± 0.0033	0.7412 ± 0.0033	0.999284
DP	0.0183 ± 0.0057	0.0149 ± 0.0033	0.008591
EOdds	0.0469 ± 0.0153	0.0327 ± 0.0068	0.000031
PRP	0.197 ± 0.0318	0.1738 ± 0.0311	0.000024

Table 9: ACS West Insurance

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.7183 ± 0.0028	0.7182 ± 0.0029	0.999916
DP	0.0603 ± 0.0089	0.0322 ± 0.0041	0.000001
EOdds	0.0676 ± 0.0072	0.0579 ± 0.0062	0.000052
PRP	0.3732 ± 0.11	0.205 ± 0.0446	0.000018

Table 10: ACS West Poverty

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.8319 ± 0.0039	0.8316 ± 0.0039	1.000000
DP	0.0246 ± 0.0038	0.0154 ± 0.0017	0.000001
EOdds	0.0396 ± 0.0086	0.0226 ± 0.0024	0.000001
PRP	0.1305 ± 0.0172	0.1173 ± 0.019	0.000018

Table 11: ACS West Public Coverage

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.7932 ± 0.0016	0.7924 ± 0.0018	1.000000
DP	0.03 ± 0.0041	0.0204 ± 0.0028	0.000001
EOdds	0.0403 ± 0.0061	0.0236 ± 0.0028	0.000001
PRP	0.159 ± 0.0245	0.1443 ± 0.0297	0.029129

I Role of Bins Parameter and Ablation Study

In this section, we first comment on the choice of the number of bins, $|\mathcal{B}|$, as a hyperparameter and then show results of additional testing with respect to the choice of the bin parameter. In practice, the choice of $|\mathcal{B}|$ should reflect the required bin granularity of the outputs for usage in, for example, rank-ordering or threshold-based classification. Our work therefore approaches choosing $|\mathcal{B}|$ purely from a computational complexity perspective as the number of optimization variables scales in the order of $\mathcal{O}(|\mathcal{G}||\mathcal{B}|^2)$. We used $|\mathcal{B}| = 50$ across our experiments as we found that this parameter gave a reasonable degree of granularity in the resulting bins while also being computationally tractable enough to solve (our MIP solver could frequently find solutions with <10% optimality gap within 10 minutes). From a more theoretical perspective, the choice of $|\mathcal{B}|$ determines how well we can estimate the score transformation function, with higher $|\mathcal{B}|$ giving us better estimates in the training data. However, having too few samples within each bin (in the training data) results in low bias but high variance estimates of the score transformation function and the corresponding performance/fairness metrics, leading to bad generalization on the testing set. Having too few bins can, on the other hand, lead to an under-parameterized score transformation function. While it is clear that $|\mathcal{B}|$ should scale with respect to the per-bin sample size, we consider a rigorous analysis of this choice to be out of scope as a current limitation of our work. We hence recommend choosing $|\mathcal{B}|$ in practice based on the desired granularity of results and/or treating it as a parameter to optimize via cross-validation.

To run an experiment to study the effects of changing $|\mathcal{B}|$, we must first define $|\mathcal{B}|$-agnostic metrics to evaluate the post-processing output in terms of performance and fairness. The primary difficulty is that since fairness is defined on a bin-wise basis, evaluating the worst-case violation for example using $|\mathcal{B}| = 10$ is not comparable to $|\mathcal{B}| = 50$. Hence, we do the following: First, after optimizing for $x_{bb}^{[g]*}$, we apply the linear interpolation based mapping as described in Appendix B so that we have new scores s'. To compare the accuracy across different $|\mathcal{B}|$, we compute the ROC and precision-recall (PR) AUC based on s'. To compare the fairness, we discretize s' in 100 bins (regardless of which $|\mathcal{B}|$ we used) and compute the worst case violation for each fainess metric across all 100 bins. The reason we use 100 bins to assess fairness is that we specifically use a large dataset (500k+ instances in both training and testing) to obtain more granular metrics. To generate the results, we use the ACS Income data as presented in previous sections. The difference is that we now use data across all states to obtain a large sample for a single random 60/40 train-test split, each with over 500k

samples. Following the procedure from before, we train and tune a random forest model, score the training and testing data, and proceed with our evaluation methodology. Since the linear interpolation mapping methodology utilizes stochastic draws, we sample 300 draws per bin and compute the 1 standard-deviation errors shown in the tables.

Table 12 shows the performance metrics across different $|\mathcal{B}|$, where we show the average metric and 1-standard deviation error margins. We observe that performance is very similar across the different choices, but is maximized at the higher bin counts of of 50-60. The performance also appears to plateau beyond a certain point, suggesting that it is not necessary to select a large $|\mathcal{B}|$ for performance purposes. Interestingly, we also see that the solutions are near-optimal based on the optimality gap. This shows that although the number of variables scales with $|\mathcal{B}|$, it does not necessarily preclude us from high quality solutions. On the contrary, the optimality gap for lower bin ranges such as 25 is larger with lower performance. This could be due to the fact that since we are optimizing over the same ϵ fairness criteria, having fewer degrees of freedom with smaller bins makes it more difficult to find feasible solutions.

Table 13 shows the results for fairness, where we show the average worst fairness violation and the 1-standard deviation error margins. The story is less clear from this angle but we emphasize that training the problem for different $|\mathcal{B}| \neq 100$ bins using the same ϵ parameters and evaluating it on $|\mathcal{B}| = 100$ for fairness is an unintended method of using our framework which we are only doing to have comparable results for the ablation study. Here, we see that the average metrics are similar for different parameters with $|\mathcal{B}| = 40$ having the best overall result. Demographic parity violation is surprisingly minimized at $|\mathcal{B}| = 30$ in this example, but given the other fairness and performance metrics $|\mathcal{B}| = 40$ appears to be the best choice. Under this method of evaluation, the fairness parameters do not appear to be very sensitive to the choice of $|\mathcal{B}|$.

Table 12: Bin Ablation Study (Performance) - ACS Income

Num. Bins	Optimality Gap	ROC AUC	PR AUC
25	0.1483	0.8712 ± 0.0001	0.7933 ± 0.0002
30	0.0672	0.8733 ± 0.0	0.7969 ± 0.0002
35	0.0326	0.8739 ± 0.0	0.7981 ± 0.0002
40	0.0095	0.874 ± 0.0	0.8001 ± 0.0001
45	0.0096	0.8742 ± 0.0	0.8 ± 0.0001
50	0.0144	0.8743 ± 0.0	0.8002 ± 0.0001
55	0.0150	0.8742 ± 0.0	0.8005 ± 0.0001
60	0.0212	0.8743 ± 0.0	0.8011 ± 0.0001

Table 13: Bin Ablation Study (Fairness) - ACS Income

Num Bins	Demographic Parity	Equalized Odds	Predictive Rate Parity
25	0.0174 ± 0.0004	0.0224 ± 0.0008	0.1447 ± 0.013
30	0.0144 ± 0.0004	0.0206 ± 0.0005	0.1702 ± 0.0101
35	0.0177 ± 0.0003	0.0235 ± 0.0019	0.1535 ± 0.0149
40	0.0179 ± 0.0004	0.0195 ± 0.0005	0.1295 ± 0.013
45	0.0174 ± 0.0003	0.0234 ± 0.0008	0.1406 ± 0.0182
50	0.0215 ± 0.0003	0.0239 ± 0.0004	0.1456 ± 0.0143
55	0.0165 ± 0.0004	0.021 ± 0.0006	0.1323 ± 0.0127
60	0.0179 ± 0.0004	0.0237 ± 0.0007	0.1503 ± 0.0178