FROC: Building Fair ROC from a Trained Classifier

Avyukta Manjunatha Vummintala, Shantanu Das, Sujit Gujar

International Institute of Information Technology, Hyderabad
avyukta.v@research.iiit.ac.in, shantanu.das31@gmail.com, sujit.gujar@iiit.ac.in

Abstract

This paper considers the problem of fair probabilistic binary classification with binary protected groups. The classifier assigns scores, and a practitioner predicts labels using a certain cut-off threshold based on the desired trade-off between false positives vs. false negatives. It derives these thresholds from the ROC of the classifier. The resultant classifier may be unfair to one of the two protected groups in the dataset. It is desirable that no matter what threshold the practitioner uses, the classifier should be fair to both the protected groups; that is, the $\mathcal{L}_p$ norm between FPRs and TPRs of both the protected groups should be at most ε. We call such fairness on ROCs of both the protected attributes ε_p-Equalized ROC. Given a classifier not satisfying ε_1-Equalized ROC, we aim to design a post-processing method to transform the given (potentially unfair) classifier's output (score) to a suitable randomized yet fair classifier. That is, the resultant classifier must satisfy ε_1-Equalized ROC. First, we introduce a threshold query model on the ROC curves for each protected group. The resulting classifier is bound to face a reduction in AUC. With the proposed query model, we provide a rigorous theoretical analysis of the minimal AUC loss to achieve ε_1-Equalized ROC. To achieve this, we design a linear time algorithm, namely FROC, to transform a given classifier's output to a probabilistic classifier that satisfies ε_1-Equalized ROC. We prove that under certain theoretical conditions, FROC achieves the theoretical optimal guarantees. We also study the performance of our FROC on multiple real-world datasets with many trained classifiers.

Extended version — https://arxiv.org/abs/2412.14724

Code and Miscellaneous —
https://github.com/magnetar-iiith/FROC/tree/main

1 Introduction

The use of *Machine Learning based Models* (MLM) in decision-making is prevalent today. Practitioners use MLMs' predictions in college admissions, credit scores, recidivism, employment, recommender systems, etc. (Portugal, Alencar, and Cowan 2018; Berger, Frame, and Miller 2005). However, there have been several reports of such MLMs discriminating against individuals belonging to certain groups based on *protected attribute* such as gender, age, race, color, and religion. E.g., in Angwin et al. (2022), predictive models are found to be biased against the black population, or the Amazon recruitment team has to stop using the AI tool for shortlisting candidates as it was biased against females Dastin (2022). Bickel, Hammel, and O'Connell (1975); Berger, Frame, and Miller (2005); Zhao et al. (2018) show that many of such predictive models are unfair to females. Such unfair instances have driven researchers toward building a fair MLM.

An MLM that achieves fairness with the least possible compromise on traditional performance guarantees such as accuracy is *desirable* MLM. Building a desirable MLM involves two main steps: a) formalizing and quantifying a fairness measure and b) designing algorithms to train MLM for quantified fairness. Researchers proposed many fairness measures, majorly belonging to two categories: (i) *individual fairness* Dwork et al. (2012) – individuals with similar input features receive similar decision treatment irrespective of their protected attribute. (ii) *Group fairness* – a particular statistical property must be similar across each protected group, e.g., *Disparate Impact (DI)*,*Equalized odds (EO)* (Madras et al. 2018).

Building Fair MLM Fair machine learning models (MLMs) can be developed by targeting different stages of the model training cycle. Approaches include: (i) *Pre-processing* methods, which act on input data to eliminate bias (Feldman et al. 2015; Zemel et al. 2013). (ii) *In-processing* algorithms, which intervene during training to incorporate fairness as a constraint or within the learning objective (Padala and Gujar 2020). (iii) *Post-processing* methods, which adjust the outputs of trained MLMs to produce fair results, requiring access to sensitive attributes.

In-processing and pre-processing methods are tailored to specific fairness criteria and models, necessitating retraining for each new fairness definition. Post-processing methods, in contrast, are model-agnostic and do not depend on the training process, making them suitable for domain experts with limited MLM knowledge (Sleeman et al. 1995). These methods are especially favored when retraining is infeasible, such as in large-scale systems like recommender systems (Nandy et al. 2022).

Given a potentially biased scoring function, this paper addresses the challenge of constructing a fair probabilistic binary classifier with a binary-protected attribute. The goal is to ensure fairness without retraining the MLM, minimizing performance loss.

Fairness and Performance Trade-offs For classification, one of the desired characteristics of an MLM is *calibration* (Kleinberg, Mullainathan, and Raghavan 2017). Suppose a classifier predicts that a given input is accepted ($Y = 1$) with probability p, then calibration demands that the fraction of the accepted population, with the same features, is p. Kleinberg, Mullainathan, and Raghavan (2017); Chouldechova (2017) have shown that calibration and equalized odds cannot be satisfied simultaneously except for highly constrained cases. Hence, researchers have been focusing on building classifiers (MLMs) with an appropriate approximate version of fairness (Madras et al. 2018). When it comes to practitioners, they focus on *Receiver Operator Characteristics* (ROC) for evaluating a classifier as it best describes the classifiers. ROC measures the relative scores of the positive versus negative instances. The area under ROC-curve (AUC) is an appropriate performance metric to measure the predictive quality of such classifiers and to segregate positive and negative samples through ranking (Huang and Ling (2005); Clémençon, Lugosi, and Vayatis (2008); Zehlike, Yang, and Stoyanovich (2021)). AUC is particularly beneficial when the classifier is expected to segregate positive and negative labels, and the predictions must be fair across all threshold scores.

To make the practitioner's job effortless, we introduce a novel fairness measure, namely ε_p-Equalized ROC – no matter what threshold it uses for classification, the classifier is approximately fair, i.e., for all possible thresholds, the distance between the corresponding points of the ROC curves for both the protected group should be withing ε distance in the $\mathcal{L}_p$ norm. We aim to build a new probabilistic classifier that satisfies ε_1-Equalized ROC with the minimal loss in AUC w.r.t. to the scoring function s.

Our Approach: We assume query access to the ROC of s. First, we make sufficiently large k queries to the ROC for the protected groups and make a piece-wise linear approximation of the ROC curves of both the protected groups. Next, we transport ROCs within ε distance of each other to minimize the loss in AUC of the resultant ROC. We can achieve such transportation by randomizing scores across certain feasible classifiers for the given ROC curve. We call the space of these classifiers as *ROC Space* of s. The resultant classifier from such randomization across the ROC Space is a convex combination of these classifiers. In a nutshell, we *transform* the given s to a fair scoring function by such ROC transport. We refer to this procedure of ROC transport as `FROC`. We then geometrically prove that under certain conditions, `FROC` is *optimal*.

Our Contributions:

- We introduce a novel group fairness notion ε_p-Equalized ROC, enforcing fairness over all thresholds in a score-based classification, which is extremely useful for practitioners.
- Next, we model a post-processing problem as a problem of finding an optimal transformation $\mathcal{H}$ on a given scoring function s to minimize the performance loss due to transformation while ensuring ε_1-Equalized ROC.
- To achieve ε_1-Equalized ROC, we propose a ROC transport, `FROC`, a *post-processing* algorithm (Algorithm 1). Thus, it avoids re-training the existing MLM, which might not be fair. It also helps in explaining the decisions.
- We perform rigorous theoretical analysis. We prove that (under some conditions) `FROC` is optimal in terms of AUC loss. (Theorem 4.2).
- Finally, we demonstrate the efficacy of `FROC` via experiments.

1.1 Related Work

Fairness in Binary Classification and Ranking *Demographic Parity* (DP), *Disparate Impact* (DI), and *Equalized Odds* (EO) are widely studied group fairness notions. DP (Dwork et al. 2012) and DI (Feldman et al. 2015) ensure that the fraction of positive outcomes is identical across all sensitive groups. Barocas and Selbst (2016) introduced the 80% rule, requiring that the positive outcome rate for a minority group must be at least $4/5$ of that for the majority group. EO (Hardt, Price, and Srebro 2016) ensures similar distributions of error rates, specifically false positives and false negatives (Verma and Rubin 2018). Techniques to achieve fair MLMs include those discussed by Padala and Gujar (2020). Group fairness has been shown to be inadequate for score-based classifiers, which classify across all thresholds (Gorantla, Deshpande, and Louis 2021). Consequently, researchers have proposed fairness notions based on the area under the curve (AUC). Examples include *intra-group pairwise* AUC fairness (Beutel et al. 2019), *BNSP* (Borkan et al. 2019), and *inter-group pairwise* AUC (xAUC) fairness (Kallus and Zhou 2019). Yang et al. (2023) present a minimax learning and bias mitigation framework that integrates intra-group and inter-group AUC metrics to address algorithmic bias. Vogel, Bellet, and Clémençon (2021) examine fairness in ranking problems, developing a general class of AUC-based fairness notions. They demonstrate that AUC-based fairness notions do not capture all forms of bias, as AUC summarizes classifier performance. They propose a stronger notion called point wise ROC-based fairness and design an in-processing algorithm for this purpose.

Our fairness definition (ε_p-Equalized ROC) is inspired by equalized odds for all thresholds in ranking-based classification and is suitable for post-processing algorithms. It generalizes the approach of Chen and Wu (2020), which uses the Manhattan distance as its norm. We later demonstrate the equivalency of both fairness notions (ours ε_1). Note that the notion in (Chen and Wu 2020) is not motivated by the same error rates at all thresholds, and also, ours is more of a geometric approach from ROC curves, and theirs is an algebraic approach; ours is more general.

Post-processing for fair classification Post-processing techniques range from simple adjustments, such as thresholding or re-scaling, to complex methods like re-weighting or re-sampling. Hardt, Price, and Srebro (2016) argue that many existing fairness criteria are too restrictive, leading to sub-optimal solutions. They propose a fairness notion allowing some variation in prediction outcomes, defined by "equality of opportunity" constraints, ensuring the classifier is unbiased regarding the sensitive attribute. Their approach involves adjusting prediction thresholds for different groups based on their base rates to equalize false positive and false negative

rates across groups. However, it does not involve *transporting* ROC curves. Wei, Ramamurthy, and Calmon (2020) examine post-processing from the perspective of transformers, defining fairness as the expectation of scores and bounding the differences between true positive rates (TPRs) and false positive rates (FPRs) across protected groups. Cui et al. (2021) propose a model-agnostic post-processing framework for balancing fairness in bipartite ranking scenarios. Zhao (2024) introduces a novel approach using Wasserstein barycenters to quantify and address the cost of fairness, demonstrating that the complexity of learning an optimal fair predictor is comparable to learning the Bayes predictor. Ţifrea et al. (2024) propose a framework that transforms any regularized in-processing method into a post-processing approach, extending its applicability across a broader range of problem settings. Cruz and Hardt (2023) identifies two key methodological errors in prior work through empirical analysis: comparing methods with different unconstrained base models and differing levels of constraint relaxation. Jang, Shi, and Wang (2022) introduce a method to optimize multiple fairness constraints through group-aware threshold adaptation, learning classification thresholds for each demographic group by optimizing the confusion matrix estimated from the model's probability distribution. Unlike Jang, Shi, and Wang (2022), our approach starts with the fairness notion that differences between TPRs and FPRs of different groups must be bounded. Mishler, Kennedy, and Chouldechova (2021) use the bounded difference of counterfactual TPRs and FPRs as their fairness criterion, which differs from our ε_p-Equalized ROC definition. Our ε_p-Equalized ROC focuses on the bounded difference between TPRs and FPRs of different groups as the fairness criterion.

2 Preliminaries

Consider a practitioner interested in binary classification, each data point having a binary-protected attribute. He/she is equipped with a scoring-based classifier trained on dataset $D = \{(x_i, a_i, y_i)_{i\in 1:n}\}$. Here, for ith data sample, $x_i \in \mathcal{X} \subset \mathbb{R}^d$ denotes features, $y_i \in \{0, 1\}$ denotes the binary label, and $a_i \in \mathcal{A} = \{0, 1\}$ denotes its binary protected attribute. We consider all these three as drawn from random variables X, A, Y, respectively. There could be two scenarios - when the protected attribute is included or excluded from training (Wei, Ramamurthy, and Calmon (2020))—our post-processing works for both cases as long as protected attributes are accessible during post-processing.

The random variables X, A, Y are jointly distributed according to an unknown probability distribution over (x_i, a_i, y_i). The cumulative conditional distributions on $X \mid (Y = 1)$ and $X \mid (Y = 0)$ are denoted by G, H, respectively. G^a, H^a are the corresponding distributions conditioned on $A = a$ (i.e. G^a denotes the distribution of $X \mid (Y = 1, A = a)$)

2.1 Probabilistic Binary Classification

Probabilistic Binary Classifier is equipped with a scoring function $s : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ mapping the feature space to a score. A deterministic classifier returns $s(X) \in \{0, 1\}$ and a randomized one returns $s(X) \in [0, 1]$. The higher the score $s(x)$, the higher the chance of the corresponding label $y = 1$. The model prediction $\widehat{Y}$, based on certain threshold $t \in [0, 1]$, is given by $\widehat{Y} = \mathbb{I}(s(X) \geq t)$. $\mathcal{S}$ denotes the space of such scoring functions.

The practitioner decides the threshold t depending on the corresponding true positive rate (*TPR*) and false positive rate (*FPR*) (Provost (2000); Zhou and Liu (2005)). For deciding t, he is supplied with ROC – *receiver operator characteristic curve* for s. The ROC depicts the relation between TPR ($G_s(t)$) and FPR ($H_s(t)$) for s at all possible thresholds t.

We define $G_s(t) \triangleq \mathbb{P}(s(X) \geq t \mid Y = 1)$ and $H_s(t) \triangleq \mathbb{P}(s(X) \geq t \mid Y = 0)$. Furthermore, we define $G^a_s(t) \triangleq \mathbb{P}(s(X) \geq t \mid Y = 1, A = a)$ and $H^a_s(t) \triangleq \mathbb{P}(s(X) \geq t \mid Y = 0, A = a)$.

2.2 ROC Curve and AUC

The plot of a ROC-curve (Definition (2.1)) is used to visualize homogeneity between two cumulative distributions (Vogel, Bellet, and Clémençon (2021)). The ROC curve is defined as:

Definition 2.1 (ROC-Curve)**.** *For any two cumulative distributions g_1, g_2 defined over the set $\mathbb{R}$, the ROC-curve is defined as the plot of $ROC_{g_1,g_2}(\alpha) \triangleq 1 - g_1 \circ g_2^{-1}(1 - \alpha)$ with domain $\alpha \in [0, 1]$.*

The area under ROC-curve, *AUC*, represents a summary of point-wise dissimilarity between the concerned distributions. Formally, let S, S' be two independent random variables distributed according to g_1, g_2 respectively, then $AUC_{g_1,g_2} = \mathbb{P}(S' > S) + \frac{1}{2}\mathbb{P}(S' = S)$.

For a given scoring function s, we get two RVs, G_s and H_s, by varying decision thresholds. We call the corresponding ROC curve $\texttt{ROC}_s$. The area under $\texttt{ROC}_s$, i.e., $\texttt{AUC}_s = AUC_{H_s,G_s}$, is used to measure the ranking performance of a score function $s(.)$ (Cortes and Mohri (2003); Clémençon, Lugosi, and Vayatis (2008)). For a perfect classifier, $\texttt{AUC}_s = 1$, but such a classifier does not exist. Therefore, the optimal scoring function s^* maximizes the $\texttt{AUC}_s$ amongst a certain subset of $\mathcal{S}' \subset \mathcal{S}$. Formally, $s^* \in \arg\max_{s\in\mathcal{S}'} \texttt{AUC}_s$. In section 3.4, we illustrate how a sub-optimal score function with lower TPRs can be achieved by randomizing outputs of $s(\cdot)$. This process is crucial in ensuring fairness. Let $\mathcal{S}|_s$ be the space of possible scoring functions through such randomization. We call it ROC-space of s. Before designing our fair classifier, we formally define our notion of fairness in the next section.

2.3 Fairness in Classification

The typical group fairness notions in binary classifiers such as *Demographic Parity* (DP) and *Equalized Odds* (EO) are defined on deterministic predictions, i.e., in score-based classification, they work with a single threshold on scoring function s. Let t^* be the threshold set by the practitioner. The resultant classifier is said to satisfy DP if $G^0_s(t^*) + H^0_s(t^*) = G^1_s(t^*) + H^1_s(t^*)$. It satisfies the equivalence of *acceptance rates* across groups. Similarly, EO enforces equality of

positive and negative error rates across protected groups, $1 - G_s^0(t^*) = 1 - G_s^1(t^*)$ and $H_s^0(t^*) = H_s^1(t^*)$.

ε_p-Equalized ROC As discussed earlier, all group fairness notions are characterized by equality of a particular statistic across both the protected groups. In scoring-based probabilistic classifiers, these fairness notions depend on the selected threshold. To achieve fairness across all thresholds, the practitioner can choose to retrain the model and achieve the right trade-offs between TPR and FNR. However, retraining is expensive. Therefore, a desirable solution is To offer fair treatment to both protected groups using the pre-trained classifier. However, this leads to invoking the post-processing technique every time the practitioner needs to update the threshold t^*. Instead, we propose a novel fairness measure to simplify the practitioner's job. We perform post-processing on the given classifier once, and it ensures that no matter what threshold t^* they choose to make decisions, the classifier offers similar treatment to both the protected groups. That is, the individual ROCs (Here on, we shall denote the ROCs of the protected groups, i.e., $ROC_{H_s^0, G_s^0}$ and $ROC_{H_s^1, G_s^1}$ by $\texttt{ROC}_s^0$ and $\texttt{ROC}_s^1$ respectively) should be within ε distance ($\mathcal{L}_p$ norm) of each other. We call it *ε_p-Equalized ROC*. More formally,

Definition 2.2 (ε_p-Equalized ROC)**.** *A scoring function for binary classification s with label prediction $\widehat{Y} = \mathbb{I}(s(x) \geq t)$ is said to satisfy -Equalized ROC if for all $\alpha \in (0, 1)$ the following holds:*

$$|| \; ROC_s^1(\alpha) - ROC_s^0(\alpha) \; ||_p \leq \varepsilon \tag{1}$$

In ε_p-Equalized ROC, we utilize standard metrics (i.e. $\mathcal{L}_p$ norms) as the fairness statistic to quantify fairness. Thus, ε_p-Equalized ROC is feasible for post-processing algorithms.

Furthermore, if FROC is effective for $\mathcal{L}_1$, it necessarily extends to all p-norms. This conclusion follows from the inequality:

$$|a|^p + |b|^p \leq |a| + |b|, \quad \forall p \geq 1, \, a, b \in [0, 1].$$

However, while FROC ensures fairness, it does not guarantee optimality for $p > 1$.

Next, we formulate the problem of fair post-processing. Note: ε_1-Equalized ROC is a generalization of Equalized Odds to all the given thresholds of the scoring function. The proofs and detailed discussion are in Appendix B.

2.4 Problem Formulation

Given $s \in \mathcal{S}$, we would like to find $h \in \mathcal{S}|_s = \mathcal{H}(s)$ – a transformation of a given scoring function such that h satisfies ε_1-Equalized ROC. Additionally, we want the loss in AUC due to transformation $\mathcal{H}$ minimal. That is, $\mathcal{L}_F = \texttt{AUC}_s - \texttt{AUC}_h$ must be minimal to retain the maximum performance guarantee of s. Thus, our goal is to get transformation $\mathcal{H}$ that solves the following optimization problem and returns the optimal transformed score h^*:

$$h^* \in \underset{h \in \mathcal{S}|_s}{\arg\max} \; \texttt{AUC}_h \tag{2}$$

$$\text{s.t.} \; || \; \texttt{ROC}_h{}^0(\alpha) - \texttt{ROC}_h{}^1(\alpha) \; ||_1 \leq \varepsilon, \; \forall \alpha \in [0, 1]$$

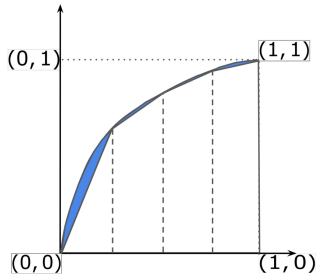

Figure 1: Shaded Area indicates $\mathcal{L}_{PLA}$

3 Our Approach

First, we explain query access to $\texttt{ROC}_s$ to sample from the desired statistic at various thresholds and its piece-wise linear approximation in Section 3.1 and Section 3.2, respectively. Since we cannot sample a continuum of thresholds, our $\texttt{ROC}_s$ will be discrete. In Section 3.3, we describe the transport of ROCs. Finally, we summarize our transformation as FROC in Section 3.4.

3.1 Query Model

Let $\mathcal{T} = \{t_1, \ldots t_k\}$ be the set of thresholds at which we sample $\texttt{ROC}_s$ for each sensitive group ($t_i = \frac{i}{k}$). Let $\mathcal{Q}^a(t_i)$ denote the query output at threshold t_i for sensitive group $A = a$ on the $\texttt{ROC}_s^a$. $\mathcal{Q}^a(t_i) \triangleq ROC_{H_s^a, G_s^a}(t_i)$.

Abusing notations, we use $\mathcal{Q}^a(t_i)$ and $\mathcal{Q}_i^a$ interchangeably. Let $\mathcal{Q}^a = (\mathcal{Q}_1^a, \ldots, \mathcal{Q}_k^a)$ be the sequence of all query outputs for group a. In the next section, we construct the piece-wise linear approximation of the group-wise ROC curves using the group-wise query outputs $\mathcal{Q}^a$.

3.2 Piece-wise Linear Approximation (PLA) of ROC-curves

To obtain the piece-wise linear approximation (PLA), we sample k points from ROC and construct a straight line from $\mathcal{Q}_i^a$ to $\mathcal{Q}_{i+1}^a$ for all $i = 1 \ldots k-1$. Lastly, we join $(0, 0)$ to $\mathcal{Q}_1^a$ (see Figure 1). Following these steps on the query sets $\mathcal{Q}^a$ will generate the PLAs for protected groups $a \in \{0, 1\}$. We denote by $\widehat{G_s^a}, \widehat{H_s^a}$, the cumulative distributions induced by the linear approximation of the ROC-curve on s.

Due to PLA, we incur a loss $\mathcal{L}_{LPA}$ in AUC_{H_s, G_s}(shaded region in Figure (1)). $\mathcal{L}_{LPA}$ is inversely proportional to the number of queries k, see Section 4.1 for bounds on this loss. Hence, we shall ignore this loss in our fairness analysis as it can be brought arbitrarily close to 0 by increasing k.

3.3 Transporting ROCs for ε_1-Equalized ROC

Since we are using post-processing technique to ensure fairness, it is impossible to shift any ROC above its current position, i.e., build a classifier corresponding to any point in the epigraph (the points above the ROC curve) of $\texttt{ROC}_s$ just with the help of s. Interestingly, a classifier representing a point in the hypograph (points below the curve) of $s \cap \mathcal{S}$ can be obtained through randomization on the predicted scores (see Chapter 3 in Barocas, Hardt, and Narayanan (2023)).

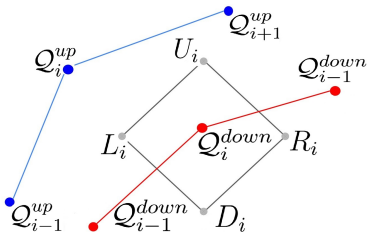

Figure 2: Norm Boundary

The key idea involves abstracting out the convex hull formed by the three points $(0,0)$, $(1,1)$ and $\mathcal{Q}_i^{up}$, and sampling outcomes from classifiers representing $(0,0)$, $(1,1)$[1] and $\mathcal{Q}_i^{up}$ with specific probabilities. By taking convex combinations of the three aforementioned points in the ROC space, we can represent any point lying in their convex hull. The exact convex combinations are described in **C2**. We leverage this property to achieve ε_1-Equalized ROC. We denote this space as *ROC-space of* $s - \mathcal{S}|_s$. Each point in $\mathcal{S}|_s$ represents a binary classifier in terms of its performance at a certain threshold t. Each point is of the form $(FPR(t), TPR(t))$. This method is discussed in detail in the Appendix.

In the realm of binary classification, it is a common occurrence for one group to be subject to discrimination. Specifically, if we plot $\texttt{ROC}_s^0, \texttt{ROC}_s^1$, we will find that one of the ROCs is notably situated below the other. For this study, the ROC predominantly above the other will be designated as ROC_{up}, while the other ROC will be referred to as ROC_{down}. We believe this is a reasonable assumption because we observed that in most classifiers (for which present the results and others we explored on the datasets mentioned in Section E3) the ROCs don't intersect or intersect at regions where $FPR \leq 0.2$ or $TPR \geq 0.5$. Typically, no practitioner will work in those areas of ROCs. We leave for future work to address intersecting ROCs.

Let $\mathcal{Q}^{up}, \mathcal{Q}^{down}$ be the corresponding set of query points for $\texttt{ROC}_{up}, \texttt{ROC}_{down}$ respectively. We also denote their fair counterparts by $\widetilde{\mathcal{Q}}^{up}, \widetilde{\mathcal{Q}}^{down}$.

Algorithm Definitions We need to transport $\texttt{ROC}_{up}$ towards $\texttt{ROC}_{down}$ such that the new ROCs are within ε distance of each other. Our approach is geometric. We need to identify certain points/curves in the epigraph of $\texttt{ROC}_{down}$ as follows.

Definition 3.1 (Norm Boundary)**.** *The set of all points within* ε *distance* (ℓ_1 *norm) from* $\mathcal{Q}_i^{down}$ *is known as the* norm set $\mathfrak{C}_i$*. Formally, we have*

$$\mathfrak{C}_i \triangleq \{x : x \in [0,1]^2, ||x - \mathcal{Q}_i^{down}||_1 \leq \varepsilon\}$$

The set of all points exactly ε *distance (in* $\mathcal{L}_1$ *norm) from* $\mathcal{Q}_i^a$ *is known as* Norm Boundary $\mathfrak{B}_i$*. Formally,*

$$\mathfrak{B}_i \triangleq \{x : x \in [0,1]^2, ||x - \mathcal{Q}_i^{down}||_1 = \varepsilon\}$$

Additionally, we denote the vertices of the Norm Boundary Rhombus (starting from the top most point and moving clockwise) as U_i*,* R_i*,* D_i*, and* L_i*.*

[1]Note that $(0,0)$ and $(1,1)$ represent 'always reject' and 'always accept' classifiers.

We say that an index $i \in [1, 2, \ldots, k]$ is a Boundary Cut index when ROC_{up} intersects the Norm Boundary $\mathfrak{B}_i$. Formally,

Definition 3.2 (Boundary Cut)**.** *Index* $i \in [1, 2, \ldots, k]$ *is a* Boundary Cut index *when* $\mathfrak{B}_i \cap ROC_{up} \neq \phi$*.*

We now define the three kinds of shifts that will be used in our Algorithm: For a given $i \in [1, 2, \ldots, k]$, Upshift is the transportation of $\mathcal{Q}_i^{up}$ to the point U_i.

Definition 3.3 (UpShift)**.** *For a given* $i \in [1, 2, \ldots, k]$*, Upshift is the transportation of* $\mathcal{Q}_i^{up}$ *to the point* U_i*. Formally,* UpShift *can be defined as the function that returns a fair threshold* $\widetilde{\mathcal{Q}}_i^{up}$ *(i.e.* U_i*) by taking the* $\mathcal{Q}_i^{down}$ *and* ε *as the arguments.*

For a given $i \in [1, 2, \ldots, k]$, Leftshift is the transportation of $\mathcal{Q}_i^{up}$ to the point L_i. Formally,

Definition 3.4 (LeftShift)**.** *LeftShift is a function that returns a fair threshold* $\widetilde{\mathcal{Q}}_i^{up}$ *(i.e.* L_i*) by taking the* $\mathcal{Q}_i^{down}$ *and* ε *as the arguments.*

Definition 3.5 (CutShift)**.** *For a given* $i \in [1, 2, \ldots, k]$ *(representing the index of the* ROC_{down}*), we run through all the points of the* ROC_{up} *and return the set of all points that intersect the Norm Boundary* $\mathfrak{B}_i$*. Formally, we define* Cutshift *as a function that takes* $\mathcal{Q}_i^{down}$ *and* ε *as the arguments and returns* $ROC_{up} \cap \mathfrak{B}_i$*. The set* $ROC_{up} \cap \mathfrak{B}_i$ *can be represented as* $\{p_{left}, p_{right}\}$ *denoting the points at the intersection of* ROC_{up} *at the* ***left-side*** *of the Norm Boundary and the* ***right-side*** *of the Norm Boundary respectively.*

Now, we elaborate on the above procedure to transport points from ROC_{up} towards ROC_{down}.

Algorithm for ROC Transport We provide a geometric algorithm that returns a classifier equivalent to the scoring function h^* in $\mathcal{S}|_s$.

Note that, Algorithm 1 treats ROC_{down} as *implicitly* fair. Also, by $Area(\square ABCD)$, we denote the area of the quadrilateral whose vertices are A, B, C, and D. This area is easily found in this context by splitting $\square ABCD$ into two disjoint triangles- ΔABC and ΔACD and using the Herons formula (Kendig 2000) on each triangle.

For example, consider $Area(\Delta \mathcal{Q}_i^{up} \mathcal{Q}_{i-1}^{up} L_i)$. Let $a = ||\mathcal{Q}_i^{up} \mathcal{Q}_{i-1}^{up}||_2$, $b = ||\mathcal{Q}_i^{up} L_i||_2$ and $c = ||\mathcal{Q}_{i-1}^{up} L_i||_2$. Additionally, we define $s = \frac{a+b+c}{2}$. Then, it is true that:

$$Area(\Delta \mathcal{Q}_i^{up} \mathcal{Q}_{i-1}^{up} L_i) = \sqrt{s(s-a)(s-b)(s-c)}$$

3.4 Obtaining Fair Classifier from the Updated ROCs

The algorithm described in the previous subsection returns the fair ROC curves according to ε_1-Equalized ROC. As a final step, we need to find the transformed classifier. We call it $\texttt{ConstructClassifier}(FairROC_{up}, FairROC_{down}, \texttt{ROC}_s^0, \texttt{ROC}_s^1)$ which returns a probabilistic binary classifier representing $h = \mathcal{H}(s)$ such that it represents the FairROCs. We construct one using the procedure explained in Section 3.3. Now, we establish the optimality of our solution within specific assumptions.

Algorithm 1: FAIRROC ALGORITHM

Require: $ROC_{up}, ROC_{down}, \varepsilon$
Ensure: $FairROC_{up}, FairROC_{down}$
Initialize $i \leftarrow 1$, $k \leftarrow \text{length}(ROC_{up})$
$FairROC_{up} \leftarrow \emptyset, FairROC_{down} \leftarrow ROC_{down}$
while $i < k - 1$ **do**
  $i \leftarrow i + 1$
  if BOUNDARYCUT(i, ε) == TRUE **then**
    $p_{left}, p_{right} \leftarrow$ CUTSHIFT$(i, ROC_{up}, ROC_{down})$
    if $FPR(\mathcal{Q}_i^{up}) \geq FPR(\mathcal{Q}_i^{down})$ **then**
      $\widetilde{\mathcal{Q}}_i^{up} \leftarrow p_{right}$
    else
      $\widetilde{\mathcal{Q}}_i^{up} \leftarrow p_{left}$
    end if
  else if $\mathcal{Q}_i^{up} \in$ HYPOGRAPH(ROC_{down}) **then**
    $\widetilde{\mathcal{Q}}_i^{up} \leftarrow \mathcal{Q}_i^{up}$
    continue
  else
    if Area$(\square \mathcal{Q}_{i+1}^{up} \mathcal{Q}_i^{up} \mathcal{Q}_{i-1}^{up} L_i) \geq$ Area$(\square \mathcal{Q}_{i+1}^{up} \mathcal{Q}_i^{up} \mathcal{Q}_{i-1}^{up} U_i)$ **then**
      $\widetilde{\mathcal{Q}}_i^{up} \leftarrow U_i$
    else
      $\widetilde{\mathcal{Q}}_i^{up} \leftarrow L_i$
    end if
  end if
  $FairROC_{up} \leftarrow$ APPEND$(\widetilde{\mathcal{Q}}_i^{up})$
end while

4 Theoretical Analysis

As described in Section (3.2), we work with PLA of the ROC curves $ROC_{H_s^a, G_s^a}$, $a \in \{0, 1\}$. This causes a *loss* in area under ROC. We denote this loss by $\mathcal{L}_{PLA}$ and is quantified as the difference in AUCs of $ROC_{H_s^a, G_s^a}$ and $ROC_{\widehat{H_s^a}, \widehat{G_s^a}}$.

In Section 3.3, transporting the ROC query points, $\mathcal{Q}^{up}$ introduces a decrease of the area under the ROC curve due to the transformation of scoring function s to h. We denote this loss by $\mathcal{L}_{AUC}$. This loss can be quantified as the difference in AUCs of $ROC_{\widehat{H_s^a}, \widehat{G_s^a}}$ and $ROC_{H_h^a, G_h^a}$ The total loss in AUC, $\mathcal{L}$, induced by FROC is given by: $\mathcal{L} = \mathcal{L}_{PLA} + \mathcal{L}_{AUC}$

4.1 PLA Loss analysis

We start our analysis by making a few standard assumptions regarding the continuity and differentiability of the cumulative distributions on the family of scoring functions $\mathcal{S}$. We adopt a less stringent assumption than that presented in (Vogel, Bellet, and Clémençon 2021), as we impose only an upper bound on the slopes. This contrasts with the approach in (Vogel, Bellet, and Clémençon 2021), which necessitates both an upper and lower bound on the slopes.

Assumption 4.1. *We assume that the rate of change (with respect to the thresholds t) of the TPRs and FPRs are upper bounded. I.e. we assume that $\exists\, u_T, u_F \in \mathbb{R}$ such that $\frac{d\,TPR}{dt} \leq u_T$ and $\frac{d\,FPR}{dt} \leq u_F$.*

Theorem 4.1. *Let $ROC_{\widehat{H_s^a}, \widehat{G_s^a}}$ be the* PLA *of $ROC_{H_s^a, G_s^a}$ over the query set of k equidistant thresholds, $\mathcal{T} = \{t_i \mid t_i = i/k \; \forall i \in [k]\}$. The corresponding $\mathcal{L}_{PLA}$ is bounded as: $\mathcal{L}_{PLA} \leq \frac{1}{2} \frac{u_T u_F}{k}$*

4.2 AUC Loss analysis

We start our analysis by making a few assumptions regarding the spacing of the ROC thresholds and the ROC curve.

Assumption 4.2. *We have two assumptions:*
- *$\forall i \in \{1, 2, \ldots, k\}$, we assume that $FPR(\mathcal{Q}_{i-1}^{down}) \leq FPR(\mathcal{Q}_i^{up}) \leq FPR(\mathcal{Q}_{i+1}^{down})$.*
- *We assume that the ROC_{up} can intersect any Norm boundary (i.e. $(\mathfrak{B}_i)_{i \in \{1,2,\ldots,k\}}$) at most 2 times.*

We note that even if **Assumption 4.2** does not hold, FROC remains operational and continues to produce outputs that are ε_1-Equalized ROC fair. However, under these conditions, the optimality with respect to AUC is not guaranteed, as **Theorem 4.4** no longer applies. The necessity of these assumptions is discussed in greater detail in the extended version of this paper.

Theorem 4.2. *If a given classifier s is piece-wise linear and satisfies assumption 4.2, the ROCs returned by FROC represent the classifier solving optimization problem 2.*

4.3 Optimally Fair points and Norm Boundary

This section proves that all optimally fair points must lie on some Norm Boundary. We do this by establishing that the performance of any point in the Norm Set can be improved by appropriate transportation to a point on the Norm Boundary.

Theorem 4.3. *(Norm Boundary) If $(\widetilde{\mathcal{Q}}_i^{up})_{i \in \{1,2,\ldots,k\}}$ is the set of optimal fair (points that maximize the AUC and also satisfy the ε_1 -Equalized ROC) thresholds must necessarily be a subset of $(\mathfrak{B}_i)_{i \in \{1,2,\ldots,k\}}$.*

Theorem 4.4. *(CutShift) If index i is a Boundary cut point, then the CutShift operation must be performed. Of the 2 points (p_{left} and p_{right}) returned by the Cutshift operation, the point that is closer to $\mathcal{Q}_i^{up}$ must be chosen i.e. $\widetilde{\mathcal{Q}}_i^{up} = argmin_{p \in \{p_{left}, p_{right}\}} |FPR(\mathcal{Q}_i^{up}) - FPR(p)|$*

Theorem 4.5. *(UpShift) If index i is not a Boundary cut point and if $Area(\square \mathcal{Q}_{i+1} \mathcal{Q}_i \mathcal{Q}_{i-1} L_i \geq Area(\square \mathcal{Q}_{i+1} \mathcal{Q}_i \mathcal{Q}_{i-1} U_i)$, then UpShift operation must be performed. The resulting point (U_i) is the new fair point $\widetilde{\mathcal{Q}}_i^{up}$. Otherwise, the LeftShift operation must be performed. The resulting point (L_i) is the new fair point $\widetilde{\mathcal{Q}}_i^{up}$.*

The proofs of all the above theorems are given in the appendix. However, the following is brief sketch of the proof:
Step 1: We prove that all optimally fair points $(\widetilde{\mathcal{Q}}_i^{up})_{i \in \{1,2,\ldots,k\}}$ must lie on the Norm Boundaries of the corresponding $\mathcal{Q}_i^{down}$. (i.e. $(\mathfrak{B}_i)_{i \in \{1,2,\ldots,k\}}$)
Step 2: We then prove that if $\mathfrak{B}_i \cap ROC_{up} \neq \phi$, then the CutShift transportation is the optimal transportation.
Step 3: We then prove that if $\mathfrak{B}_i \cap ROC_{up} = \phi$, then, based on the Cover and aforementioned area condition, the UpShift or the LeftShift transportation is the optimal transportation.

In the next section, we experimentally analyze FROC.

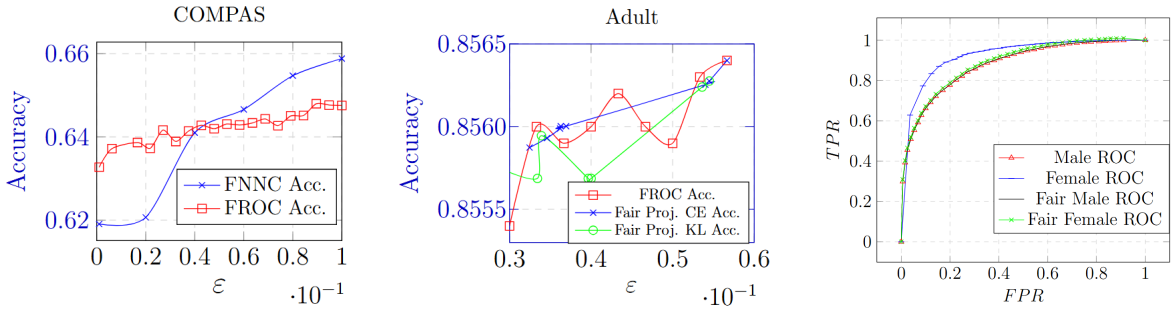

Figure 3: Comparison of different methods: (a) C1 vs. C1-FROC, (b) C3-Fair Fair vs. C3-FROC, and (c) C2 Before and After FROC.

5 Empirical Analysis

5.1 Experimental Setup

Datasets: We train different classifiers on the widely-used ADULT (Becker and Kohavi 1996) and COMPAS (Angwin et al. 2022) benchmark datasets, selecting MALE and FEMALE as protected groups in ADULT, and BLACK and OTHERS in COMPAS. ROCs are generated, with additional experiments on datasets like CelebA in Appendix E and F.

Classifiers: We test FROC on ROCs from the following classifiers: [2]. C1: *FNNC*(Padala and Gujar (2020)): This is a neural network-based classifier with a target parameter for fairness. C2: *Logistic Regression* and C3: *Random Forest* We used the code from the author's GitHub for C1 and sklearn implementations for C2 and C3.

Post-Processing methods: We compare FROC against the following baselines: B1: *FairProjection-CE* and *FairProjection-KL* (Alghamdi et al. 2022): Transforms the score to achieve mean equalized odds fairness through information projection.

5.2 Experiments

We train C1 on both datasets, C2 and C3 on the Adult dataset, and generate their ROCs for all the protected groups. FNNC, we train by ignoring its fairness components in the loss function and then generate ROC. We then invoke FROC for different ε values and check the best possible threshold for accuracy. We refer to the new classifier as C1-C3-FROC.

Baseline Post-Processing Method: We evaluate FROC, and the baselines B1 on ADULT dataset against the fairness metric *mean equalized odds*(B2) (Alghamdi et al. 2022) in Figs. 3(b). For consistent comparison, we adopt the training parameters for base classifiers from (Alghamdi et al. 2022) and keep it identical across all experiments.

5.3 Results

We show the results on the COMPAS and Adult dataset (using FNNC and FROC) here, along with a comparison with existing post-processing baselines. The remaining experimental observations are detailed in the supplementary. **Figure 3(c)** displays the ROC curves (Before and After FROC) for both males and females, on the ADULT dataset for C2. The female ROC consistently occupies the higher position, indicating a positive bias for males. This establishes ROC_0 as our counterpart to ROC_{down}. Thus, we apply FROC to the alternate curve, ROC_1, showcased in the figure. Before FROC, the maximum difference between Male ROC and Female ROC is 0.08. However, after post-processing with FROC, the loss in accuracy is $< 0.1\%$ for $\varepsilon = 0.05$. In general, across all experiments (more experiments in Appendix), we observe a 7-8% improvement in fairness, FROC incurs at most a 2% drop in accuracy. As seen in **Figure 3(a)** and **Figure 3(b)** for smaller values of ε, we also observe the performance may beat FNNC and the post-processing methods. We assign it to the fact that FNNC (and the other methods) may overachieve the target fairness for smaller values of ε (Evident from Table 2 (Padala and Gujar 2020)). FROC drops AUC minimally to achieve target fairness.

[2]We choose these classifiers as per the availability of experiment hyper-parameters from other in-processing and post-processing benchmarks.

6 Conclusion

In this work, we addressed the problem of practitioners aiming to achieve fair classification without retraining MLMs. Specifically, we provide a post-processing framework that takes a potentially unfair classification score function and returns a probabilistic fair classifier. The practitioner need not worry about fairness across different thresholds, so we proposed a new notion ε_1-Equalized ROC (Definition 2.2), which ensures fairness for all thresholds. To achieve ε_1-Equalized ROC, we proposed FROC (Algorithm 1), which transports the ROC for each sensitive group within ϵ distance while minimizing the loss in AUC of the resultant ROC. We geometrically proved its optimality conditions (Theorem 4.2) and bounds under certain technical assumptions. We observed empirically that its performance might differ at most by 2% compared to an in-processing technique while ensuring stronger fairness and avoiding retraining. We leave it for future work to explore the possibility of different distance metrics for fairness and optimizing for different performance measures.

References

Alghamdi, W.; Hsu, H.; Jeong, H.; Wang, H.; Michalak, P.; Asoodeh, S.; and Calmon, F. 2022. Beyond Adult and COM-

PAS: Fair Multi-Class Prediction via Information Projection. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 38747–38760. Curran Associates, Inc.

Angwin, J.; Larson, J.; Mattu, S.; and Kirchner, L. 2022. Machine bias. In *Ethics of data and analytics*, 254–264. Auerbach Publications.

Barocas, S.; Hardt, M.; and Narayanan, A. 2023. *Fairness and machine learning: Limitations and opportunities*. MIT press.

Barocas, S.; and Selbst, A. D. 2016. Big data's disparate impact. *Calif. L. Rev.*, 104: 671.

Becker, B.; and Kohavi, R. 1996. Title. https://archive.ics.uci.edu/dataset/2/adult. Accessed: 2024-08-01 , DOI: https://doi.org/10.24432/C5XW20.

Berger, A. N.; Frame, W. S.; and Miller, N. H. 2005. Credit scoring and the availability, price, and risk of small business credit. *Journal of money, credit and banking*, 191–222.

Beutel, A.; Chen, J.; Doshi, T.; Qian, H.; Wei, L.; Wu, Y.; Heldt, L.; Zhao, Z.; Hong, L.; Chi, E. H.; et al. 2019. Fairness in recommendation ranking through pairwise comparisons. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2212–2220.

Bickel, P. J.; Hammel, E. A.; and O'Connell, J. W. 1975. Sex Bias in Graduate Admissions: Data from Berkeley: Measuring bias is harder than is usually assumed, and the evidence is sometimes contrary to expectation. *Science*, 187(4175): 398–404.

Borkan, D.; Dixon, L.; Sorensen, J.; Thain, N.; and Vasserman, L. 2019. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion proceedings of the 2019 world wide web conference*, 491–500.

Chen, M.; and Wu, M. 2020. Towards threshold invariant fair classification. In *Conference on Uncertainty in Artificial Intelligence*, 560–569. PMLR.

Chouldechova, A. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2): 153–163.

Clémençon, S.; Lugosi, G.; and Vayatis, N. 2008. Ranking and empirical minimization of U-statistics.

Cortes, C.; and Mohri, M. 2003. AUC optimization vs. error rate minimization. *Advances in neural information processing systems*, 16.

Cruz, A. F.; and Hardt, M. 2023. Unprocessing seven years of algorithmic fairness. *arXiv preprint arXiv:2306.07261*.

Ţifrea, A.; Lahoti, P.; Packer, B.; Halpern, Y.; Beirami, A.; and Prost, F. 2024. FRAPPÉ: a group fairness framework for post-processing everything. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Cui, S.; Pan, W.; Zhang, C.; and Wang, F. 2021. Towards model-agnostic post-hoc adjustment for balancing ranking fairness and algorithm utility. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 207–217.

Dastin, J. 2022. Amazon scraps secret AI recruiting tool that showed bias against women. In *Ethics of data and analytics*, 296–299. Auerbach Publications.

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

Feldman, M.; Friedler, S. A.; Moeller, J.; Scheidegger, C.; and Venkatasubramanian, S. 2015. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 259–268.

Gorantla, S.; Deshpande, A.; and Louis, A. 2021. On the problem of underranking in group-fair ranking. In *International Conference on Machine Learning*, 3777–3787. PMLR.

Hardt, M.; Price, E.; and Srebro, N. 2016. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29.

Huang, J.; and Ling, C. X. 2005. Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on knowledge and Data Engineering*, 17(3): 299–310.

Jang, T.; Shi, P.; and Wang, X. 2022. Group-aware threshold adaptation for fair classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 6988–6995.

Kallus, N.; and Zhou, A. 2019. The fairness of risk scores beyond classification: Bipartite ranking and the xauc metric. *Advances in neural information processing systems*, 32.

Kendig, K. 2000. Is a 2000-year-old formula still keeping some secrets? *The American Mathematical Monthly*, 107(5): 402–415.

Kleinberg, J.; Mullainathan, S.; and Raghavan, M. 2017. Inherent Trade-Offs in the Fair Determination of Risk Scores. In Papadimitriou, C. H., ed., *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 43:1–43:23. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-029-3.

Madras, D.; Creager, E.; Pitassi, T.; and Zemel, R. 2018. Learning adversarially fair and transferable representations. In *International Conference on Machine Learning*, 3384–3393. PMLR.

Mishler, A.; Kennedy, E. H.; and Chouldechova, A. 2021. Fairness in risk assessment instruments: Post-processing to achieve counterfactual equalized odds. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 386–400.

Nandy, P.; DiCiccio, C.; Venugopalan, D.; Logan, H.; Basu, K.; and El Karoui, N. 2022. Achieving Fairness via Post-Processing in Web-Scale Recommender Systems. *2022 ACM Conference on Fairness, Accountability, and Transparency*.

Padala, M.; and Gujar, S. 2020. Fnnc: Achieving fairness through neural networks. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence,{IJCAI-20}, International Joint Conferences on Artificial Intelligence Organization*.

Portugal, I.; Alencar, P.; and Cowan, D. 2018. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*, 97: 205–227.

Provost, F. 2000. Machine learning from imbalanced data sets 101. In *Proceedings of the AAAI'2000 workshop on imbalanced data sets*, volume 68, 1–3. AAAI Press.

Sleeman, D.; Rissakis, M.; Craw, S.; Graner, N.; and Sharma, S. 1995. Consultant-2: Pre-and post-processing of machine learning applications. *International journal of human-computer studies*, 43(1): 43–63.

Verma, S.; and Rubin, J. 2018. Fairness definitions explained. In *Proceedings of the international workshop on software fairness*, 1–7.

Vogel, R.; Bellet, A.; and Clémençon, S. 2021. Learning fair scoring functions: Bipartite ranking under roc-based fairness constraints. In *International conference on artificial intelligence and statistics*, 784–792. PMLR.

Wei, D.; Ramamurthy, K. N.; and Calmon, F. P. 2020. Optimized score transformation for fair classification. *Proceedings of Machine Learning Research*, 108.

Yang, Z.; Ko, Y. L.; Varshney, K. R.; and Ying, Y. 2023. Minimax auc fairness: Efficient algorithm with provable convergence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 11909–11917.

Zehlike, M.; Yang, K.; and Stoyanovich, J. 2021. Fairness in ranking: A survey. *arXiv preprint arXiv:2103.14000*.

Zemel, R.; Wu, Y.; Swersky, K.; Pitassi, T.; and Dwork, C. 2013. Learning fair representations. In *International conference on machine learning*, 325–333. PMLR.

Zhao, H. 2024. Fair and Optimal Prediction via Post-Processing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 22686–22686.

Zhao, J.; Wang, T.; Yatskar, M.; Ordonez, V.; and Chang, K.-W. 2018. Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 15–20. New Orleans, Louisiana: Association for Computational Linguistics.

Zhou, Z.-H.; and Liu, X.-Y. 2005. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on knowledge and data engineering*, 18(1): 63–77.