Achieving Fairness via Post-Processing in Web-Scale Recommender Systems

Preetam Nandy ¹ Cyrus DiCiccio ¹ Divya Venugopalan ¹ Heloise Logan ¹ Kinjal Basu ¹ Noureddine El Karoui ¹

Abstract

Building fair recommender systems is a challenging and extremely important area of study due to its immense impact on society. We focus on two commonly accepted notions of fairness for machine learning models powering such recommender systems, namely equality of opportunity and equalized odds. These measures of fairness make sure that equally "qualified" (or "unqualified") candidates are treated equally regardless of their protected attribute status (such as gender or race). In this paper, we propose scalable methods for achieving equality of opportunity and equalized odds in rankings in the presence of position bias, which commonly plagues data generated from recommendation systems. Our algorithms are model agnostic in the sense that they depend only on the final scores provided by a model, making them easily applicable to virtually all web-scale recommender systems. We conduct extensive simulations as well as real-world experiments to show the efficacy of our approach.

1. Introduction

Fairness in classification and ranking problems has been an active area of research in recent years due to its tremendous impact on society as a whole (Barocas et al., 2017). As more and more businesses are relying on machine learning (ML) algorithms to recommend goods and services that affect people, fairness is becoming ever more important. ML based systems may contain implicit biases and can serve to reproduce or reinforce the biases present in society. Thus, it is crucial to be able to mitigate these biases. In this paper, we develop tools for this purpose that are both theoretically sound and widely applicable, including in the extremely large data setting that are common to internet applications.

Fairness mitigation strategies commonly fall in one of three categories: pre-, in-, or post-processing. Pre-processing

modifies training data to reduce potential sources of bias (Zemel et al. (2013b), Calmon et al. (2017), Gordaliza et al. (2019)). In-processing (also known as training time) mitigation methods modify the model training objective to incorporate fairness, often by adding constraints or regularization penalties (Kamishima et al., 2012; Bechavod & Ligett, 2017; Mary et al., 2019; Zafar et al., 2017b; Agarwal et al., 2018a), though many of these methods do not scale well to extremely large datasets. Finally, post-processing methods transform model scores to ensure fairness according to a provided definition. Post-processing methods learn (protected-attributespecific) transformations of model scores to achieve fairness objectives (Hardt et al., 2016; Pleiss et al., 2017; Kamiran et al., 2012). The post-processing approaches are very appealing in industrial practice as they do not require changes to an existing model training pipeline. Virtually any model can be easily adjusted by a post-processing algorithm to achieve the desired fairness goal.

We aim to derive post-processing approaches providing fairness in ranked lists of items generated by a recommender system. Hardt et al. (2016) provide such methods for equality of opportunity or equalized odds in the binary classification setting. However, these methods are not applicable to the ranking problems since fairness needs to hold with respect to the ranks of the items. Another challenge present in ranked data is position bias (Joachims et al., 2017; Wang et al., 2018), i.e., the bias in an end user's response depending on an item's position, which is not a concern in the binary classification setting. In this paper, we develop post-processing approaches for achieving equality of opportunity and equalized odds for recommender systems. We explicitly handle the position bias issue in our solution and also provide simple mechanisms for controlling the fairness versus performance trade-off.

The remainder of the paper is organized as follows. Sections 2 and 3 develop post-processing techniques to adjust for equality of opportunity and equalized odds respectively. Section 4 addresses the position bias issues present in the ranking context for both the mechanisms. Simulated experiments are shown in Section 5, followed by a real-world application in Section 6. We conclude with a discussion in Section 7. Proofs of all the results are given in the appendix. We end this section with a brief overview of related work.

¹LinkedIn Corporation, Sunnyvale, CA, USA. Correspondence to: Preetam Nandy <pnandy@linkedin.com>, Cyrus DiCiccio <cdiciccio@linkedin.com>.

Related Work: Most early work in fairness has been on classification tasks and strives to achieve fairness notions such as equalized odds on protected attributes (Hardt et al., 2016; Agarwal et al., 2018b; Feldman et al., 2015; Zafar et al., 2017a;c; Goh et al., 2016; Wu et al., 2019). Also, many techniques for training models that guarantee other definitions of fairness, such as equality opportunity (Hardt et al., 2016) and demographic parity, have been widely studied in recent years (Dwork et al., 2012; Zemel et al., 2013a; Goel et al., 2018; Johndrow et al., 2019)

Penalty methods for bringing in fairness constraints has received a great deal of attention (Kamishima et al., 2012; Bechavod & Ligett, 2017; Mary et al., 2019; Zafar et al., 2017b). Although they are sound theoretical methods, many of them do not scale well due to the iterative nature of these algorithms (Agarwal et al., 2018a) and hence can become a bottleneck in many large-scale applications. Thus, post-processing techniques, which do not interfere with the model training algorithm, are often preferable.

Most large-scale recommender systems are ranking systems, returning a list of ranked results to the users. Although, there are several scalable post-processing methodologies (Hardt et al., 2016; Pleiss et al., 2017; Kamiran et al., 2012), most of them focus on classification problems and their methodology do not directly translate to the ranking problem. Position bias plays a critical role in these ranking frameworks which we actively try to address through our post-processing reranker. For some of the early work on ranking systems please see Singh & Joachims (2019b); Celis et al. (2017).

2. Equality of Opportunity in Rankings

We first recall the definition of Equality of Opportunity (*EOpp*) in the context of binary classification (Hardt et al., 2016).

Definition 1 (EOpp in classification). A binary predictor satisfies equal opportunity with respect to a (protected) characteristic C and (decision) \hat{Y} if

$$\mathbb{P}(\hat{Y} = 1 \mid C = c_1, Y = 1) = \mathbb{P}(\hat{Y} = 1 \mid C = c_2, Y = 1),$$

for all c_1, c_2 . In other words, \hat{Y} is independent of C given that Y = 1.

It is natural to extend this requirement to non-binary \hat{Y} , for instance, if they are scores. In this case, if X denotes features, and s is a score function, we would want $\mathbb{P}(s(X) \leq t \mid C = c_1, Y = 1) = \mathbb{P}(s(X) \leq t \mid C = c_2, Y = 1)$, for all c_1 , c_2 and t. In other words, we want the distribution of the scores to be independent of C given that Y = 1 (see, e.g., Mary et al. (2019), Section 2.1). For threshold based classifiers (i.e., $\hat{Y}(X) = 1$ if and only if s(X) > t, for a certain threshold t), this would ensure EOpp for the associated classifiers at all thresholds.

In industrial applications, this is a natural requirement as scores could be passed downstream to other machine learning systems before yielding final recommendations. Note that for recommender systems, this requirement implies an identical distribution of ranks for each value of protected characteristic C. Thus, we define EOpp in rankings as follows.

Definition 2 (EOpp in rankings). A binary predictor satisfies equal opportunity with respect to a (protected) characteristic C and score s(X) if

$$\mathbb{P}(s(X) \le t \mid C = c_1, Y = 1) = \mathbb{P}(s(X) \le t \mid C = c_2, Y = 1),$$

for all c_1 , c_2 and t.

Note that the solution proposed in Hardt et al. (2016) is based on Definition 1, while in the following subsection, we propose a solution based on Definition 2.

2.1. Algorithm to achieve Equality of Opportunity

We present a simple post-processing algorithm that achieves EOpp at all thresholds t.

Lemma 1 (Algorithm for EOpp). Let $F_{c,1}$ be the cumulative distribution function (CDF) of scores in group C=c and Y=1. Then for each c, we transform the scores of group C=c as $s(X) \to F_{c,1}(s(X))$. These transformations guarantee EOpp for all thresholds t in the range of s(X).

The CDF transformations in Lemma 1 map the scores into [0,1]. We can apply an additional transformation $F^{-1}(\tilde{F}(s))$ to bring the scores back to the original scale, where F and \tilde{F} are the CDF of the scores before applying any transformation and after applying the transformations in Lemma 1. Note that this step will not affect the EOpp problem or the ROC curve, since the latter is invariant under increasing transformations. This step might be useful in industrial settings where scores are used in more than one machine learning systems. This line of reasoning is related to quantile normalization in (bio)statistics (Amaratunga & Cabrera, 2001; Bolstad et al., 2003).

2.2. Variants of EOpp algorithm

These post-processing approaches could also allow us to maintain EOpp between retraining of the models by updating $F_{c,1}$'s online, as user engagement can be dynamic, requiring adjustment to the algorithm (D'Amour et al., 2020). If keeping these CDFs in memory is too costly, they can be discretized (for instance, at every percentile or every 10^{-4}) and create a linear or higher-order interpolation increasing between the points.

We note that the algorithm immediately generalizes to an arbitrary number of characteristics. Furthermore, one could

relax the strict *EOpp* constraint, by considering the following modification to the transformation of the scores:

$$\widetilde{s}_c(\alpha) = \alpha \times F^{-1}(\widetilde{F}(F_{c,1}(s_c))) + (1 - \alpha) \times s_c, \quad (1)$$

where s_c denote the score restricted to C=c and $0 \le \alpha \le 1$. We can tune α to achieve a desirable performance-fairness trade-off, where larger values of α would bring more fairness (in terms of EOpp), possibly at the expense of a lower performance.

3. Equalized Odds in Rankings

Equalized odds (*EOdds*) is a fairness definition which extends equality of opportunity. In the context of binary classification, Hardt et al. (2016) defines equalized odds as follows.

Definition 3 (EOdds in classification). A binary predictor satisfies equalized odds with respect to a (protected) characteristic C and (decision) \hat{Y} if

$$\mathbb{P}(\hat{Y} = 1 \mid C = c_1, Y = 1) = \mathbb{P}(\hat{Y} = 1 \mid C = c_2, Y = 1),$$

$$\mathbb{P}(\hat{Y} = 1 \mid C = c_1, Y = 0) = \mathbb{P}(\hat{Y} = 1 \mid C = c_2, Y = 0),$$

for all c_1 , and c_2 . That is, \hat{Y} is independent of C given Y.

Similar to the *EOpp* definition, *EOdds* is naturally extended to handle the ranking case, where ranks are assigned according to a scoring function s(X). The analogous condition becomes $\mathbb{P}(s(X) \leq t \mid C = c_1, Y = y) = \mathbb{P}(s(X) \leq t \mid C = c_2, Y = y)$, for all c_1, c_2, t and $y \in \{0, 1\}$. This is equivalent to the requirement that the distribution of the scores are independent of C given the outcome Y.

Definition 4 (EOdds in rankings). A binary predictor satisfies equalized odds with respect to a (protected) characteristic C and score s(X) if

$$\mathbb{P}(s(X) \le t \mid C = c_1, Y = y) = \mathbb{P}(s(X) \le t \mid C = c_2, Y = y),$$

for all c_1, c_2, t and $y \in \{0, 1\}$.

Even in the case of classification, applying the "ranking" definition of equalized odds can be preferable. One reason is that in many internet applications, the threshold t for a score based classifier is chosen through A/B experimentation to yield a desirable tradeoff of business metrics (Agarwal et al., 2018c). As such, it is generally not known a priori which threshold is needed, and this definition affords the robustness and flexibility of guaranteeing fairness regardless of the threshold chosen.

3.1. Re-Ranking for Equalized Odds

In the context of classifiers, it is simple to randomize classifications in a way that achieves equalized odds. Consider randomly changing the decision of a classification $\hat{Y} = y$

for group C=c with probability $p_{y,c}$. The resulting randomized classification, satisfies equalized odds whenever the simple set of linear constraints

$$\begin{split} &(1-p_{y,c_1}) \cdot \mathbb{P}(\hat{Y} = y \mid C = c_1, Y = y) \\ &+ p_{1-y,c_1} \cdot \mathbb{P}(\hat{Y} = 1 - y \mid C = c_1, Y = y) \\ &= (1-p_{y,c_2}) \cdot \mathbb{P}(\hat{Y} = y \mid C = c_2, Y = y) \\ &+ p_{1-y,c_2} \cdot \mathbb{P}(\hat{Y} = 1 - y \mid C = c_2, Y = y) \end{split}$$

for all c_1 , c_2 and y are satisfied. We extend this reasoning to the ranking context. Without loss of generality, assume that the ranking scores $s(\cdot)$ fall in the interval [0,1). Note that if the scores do not fall in this interval, they can easily be transformed to this interval in a way that does not affect rankings, for instance, by applying an inverse logit transformation.

Let $I_1, ..., I_K$ be a partition of the score domain. That is, choose intervals $I_k = [i_k, i_{k+1})$ for k = 1, ..., K, where $0 = i_1 < i_2 < \cdots < i_{K+1} = 1$. We will derive a score achieving equalized odds by randomizing the original scores between these intervals. For each category $C=c_i$, define p_{k,k',c_i} (such that $\sum_{k'} p_{k,k',c_i} = 1$ for all k and c_i) to be the probability that the score from an item with characteristic c_i is randomly moved from interval indexed by k to interval indexed by k' (for notational compactness, we will write P as the vectorized version of these probabilities). More concretely, suppose that s(X) is observed for an item Xwith characteristic $C = c_i$. Suppose that s(X) belongs to interval I_k , i.e. that $i_k \leq s(X) < i_{k+1}$. Let k' be a draw from a multinomial distribution on 1, ..., K with probabilities $p_{k,1,c_i},...,p_{k,K,c_i}$. Then, define $\bar{s}(X;P)$ (or more compactly $\bar{s}(X)$) to be a score chosen in the interval $I_{k'}$ according to an arbitrary distribution on $I_{k'}$. Note that

$$\mathbb{P}(\bar{s}(X) \in I_{k'} \mid C = c_1, Y = y) = \sum_{k} p_{k,k',c_i} \cdot \mathbb{P}(s(X) \in I_k \mid C = c_1, Y = y).$$
(2)

Consequently, the randomized score $\bar{s}(X)$ satisfies equalized odds whenever the probabilities are chosen such that

$$\mathbb{P}(\bar{s}(X) \in I_{k'} | C = c_1, Y = y) = \\ \mathbb{P}(\bar{s}(X) \in I_{k'} | C = c_2, Y = y)$$

for all k' and $y \in \{0,1\}$. It is readily seen from Equation (2) that this specifies a system of linear equations in the interval transition probabilities. Not only does a solution to this system always exist, but there are infinitely many solutions. Ideally, we would like to use a solution which gives good model performance. To identify a "best" solution, let $\phi(F_{X,Y,C}, P)$ be a functional of the data generating distribution $F_{X,Y,C}(\cdot)$ and the interval transition probabilities P such that $\phi(F_{X,Y,C}, P)$ captures some aspect of predictive performance such as mean squared error or area under the receiver operating characteristic curve (AUC). Note that $F_{X,Y,C}(\cdot)$ is generally not known, but is easily replaced by

the empirical version in such situations. Finding the optimal interval transition probabilities can be formulated as a maximization problem

Maximize
$$\phi\left(P_{X,Y,C},P\right)$$

subject to
$$\mathbb{P}(\bar{S}(X) \in I_k \mid C = c_1, Y = y)$$

$$= \mathbb{P}(\bar{S}(X) \in I_k \mid C = c_2, Y = y)$$
(3)

for all c_1, c_2 , and $y \in 0, 1$.

Theorem 1. The randomized score $\bar{s}(X)$ derived from s(X) using interval transition probabilities found as the solution to the optimization problem (3) satisfies equalized odds in the ranking sense. That is,

$$\mathbb{P}(\bar{s}(X) \le t \mid C = c_1, Y = y)$$

= $\mathbb{P}(\bar{s} \le t \mid C = c_2, Y = y),$

for all c_1 , c_2 , y, and t.

While the choice of the objective function ϕ can be arbitrary, we give two possible choices. Typically, the original scores are carefully chosen to give optimal model performance. One choice is to define ϕ as $E|s(X)-\bar{s}(X)|$, which gives the least possible average movement of the scores. The empirical version of this condition can easily be written as a linear equation, which gives the computational convenience that the optimization problem (3) becomes a linear program (LP). Another choice is to maximize for AUC. A Riemann approximation to the AUC based on the partition $I_1, ..., I_K$ can be computed as

$$\sum_{c} \mathbb{P}(C=c) \cdot \sum_{k=1}^{K} \cdot \mathbb{P}(\bar{s}(X) \in I_k \mid Y=0)$$
$$\cdot \sum_{k' \geq k} \mathbb{P}(\bar{s}(X) \in I_{k'} \mid Y=1).$$

It is readily seen from Equation (2) that this is a quadratic function of the transition probabilities. Therefore, when using this choice of objective, the optimization problem (3) becomes a quadratic program (QP).

3.2. Extensions of Equalized Odds

We note that the variants of the *EOpp* algorithm discussed in Section 2.2 are also applicable to *EOdds*. Furthermore, *EOdds* has several natural extensions that are easily handled by adjustments to the constraints in the optimization based methodology for re-ranking. Here, we discuss the extension to the case of categorical outcomes with more than two possible outcomes, as well as relaxations of the condition of equalized odds.

Suppose that items are ranked according to a score s(X), and the rankings result in an outcome $y \in 1,...,M$. For example, in a news feed context, users of a site may engage with articles in multiple ways captured by actions such as "like," "comment," or "share." We extend the definition of equalized odds to ensure that the rankings are fair with

respect to any of these outcomes across all characteristics. Definition 5 (Multi-Outcome EOdds in Rankings). A score based ranker satisfies equalized odds with respect to (protected) characteristic C and score s(X) if

$$\mathbb{P}(s(X) \le t \mid C = c_1, Y = y) \\ = \mathbb{P}(s(X) \le t \mid C = c_2, Y = y),$$

for all $c_1, c_2, y \in \{1, ..., M\}$ and t.

Achieving multi-outcome *EOdds* simply requires extending the constraints in optimization problem (3) to

$$\mathbb{P}(\bar{s}(X) \in I_{k'} \mid C = c_1, Y = y) = \\ \mathbb{P}(\bar{s}(X) \in I_{k'} \mid C = c_2, Y = y)$$

for all c_1, c_2 and $y \in \{1, ..., M\}$ which are again a system of linear equations. When finding a randomized scoring function as a solution to this modified optimization problem, an analogous result to Theorem 1 holds, but for multi-outcome EOdds.

Another simple modification is to relax the strictness of the equalized odds condition.

Definition 6 (ϵ_0 , ϵ_1 -differentially *EOdds* in Rankings). A score based ranker satisfies ϵ_0 , ϵ_1 -differentially equalized odds with respect to (protected) characteristic C and score s(X) if

$$\exp\left(-\epsilon_y\right) \le \frac{\mathbb{P}(s(X) \le t \mid C = c_1, Y = y)}{\mathbb{P}(s(X) \le t \mid C = c_2, Y = y)} \le \exp\left(\epsilon_y\right)$$

for all c_1, c_2, t and $y \in \{0, 1\}$.

Ensuring that a randomized ranking score $\bar{s}(X)$ satisfies ϵ_0, ϵ_1 -differentially *EOdds* requires conditions that can be expressed as linear constraints, for example through

$$\mathbb{P}(\bar{s}(X) \le t \mid C = c_1, Y = 1) -$$

$$\exp(\epsilon) \mathbb{P}(\bar{s}(X) \le t \mid C = c_2, Y = 1) \le 0$$

and similarly for the remaining constraints required. Once again, a simple modification of the constraints in optimization problem (3) leads to a randomized ranking score such that an analogous result to Theorem 1 holds, but for ϵ_0, ϵ_1 -differentially EOdds. An interesting special case is ∞, ϵ_1 -differentially equalized odds, which reduces to ϵ_1 -differentially EOpp, providing a method to re-rank for an alternative relaxation of EOpp. Adjusting the ϵ_0, ϵ_1 allows the practitioner to strike a desireable balance between fairness and model performance.

4. Position Bias Adjustment

To understand the effect of position bias in achieving EOpp and EOdds, consider a recommendation system where for each query q, a candidate set of J items $\{d_1^{(q)},\ldots,d_J^{(q)}\}$ are ranked according to a scoring function s(X) defined on a set of features X. The viewer's response to $d_i^{(q)}$ in the recommended list not only depends on the quality of $d_i^{(q)}$

(relative to the viewer) but also depends on the position of $d_i^{(q)}$ in the list. The position bias refers to the fact that the chance of observing a positive response (e.g., a click) on an item appearing at a higher position (where the highest position is Position 1) is higher than the chance of observing the same in a lower position.

To define *EOpp* or *EOdds* in the presence of the position bias, we need to take into account the dependency of the response variable Y on the position where the item is shown. To this end, we denote the counterfactual response when an item appears at position j by Y(j). Furthermore, we use γ to denote the position of an item in the ranking generated by s(X). Therefore, the observed response is given by $Y(\gamma)$. Definition 7. A scoring function s(X) of a recommendation system satisfies *EOpp* or *EOdds* with respect to a characteristic C if

$$\mathbb{P}(s(X) \le t \mid C = c_1, Y(\gamma) = y)$$

$$= \mathbb{P}(s(X) \le t \mid C = c_2, Y(\gamma) = y), \quad (4)$$

for all t, c_1, c_2 , and for y = 1 for EOpp and $y \in \{0, 1\}$ for EOdds.

Note that the post-processing approaches discussed in the previous section work under the assumption that Y(i)'s are identical for all j. To see this, let $\tilde{s}(X)$ denote the score of an item after applying the *EOpp* or *EOdds* post-processing algorithm and let $\tilde{\gamma}$ denote the corresponding position of the item in the ranking generated by $\tilde{s}(X)$. Then $(\tilde{s}(X), \gamma)$ would satisfy Equation (4) for γ equals the position of the item in the ranking without post-processing. This does not guarantee that $(\tilde{s}(X), \tilde{\gamma})$ would satisfy Equation (4) unless Y(j)'s are identical for all j.

We make the following assumptions on the position bias.

Assumption 1. Let s(X), Y(j) and C be as in Definition 7. For all queries q and all candidate items $d_i^{(q)}$ and all j > 1, we assume

1. Homogeneity:

$$\begin{split} \mathbb{P}\left(Y(j) = 1 \mid Y(1) = 1, \ d_i^{(q)}\right) \\ &= \mathbb{P}(Y(j) = 1 \mid Y(1) = 1), \ \textit{ and } \end{split}$$

2. Preservation of Hierarchy:

$$\mathbb{P}\left(Y(j) = 1 \mid Y(1) = 0, \ d_i^{(q)}\right) = 0.$$

The first assumption states that the position bias is homogeneous over all queries and candidate items. This is a common assumption in the literature (Singh & Joachims. 2018; 2019a; Wang et al., 2018). The second assumption states that if a candidate item in a given position does not get a positive response, it cannot get a positive response in a lower position (which is reasonable in most practical settings).

Under Assumption 1, we formally define the position bias

as the following positive response decay factor

$$w_i := \mathbb{P}(Y(i) = 1 \mid Y(1) = 1) \text{ for } i = 1 \dots, J.$$
 (5)

4.1. Tackling Position Bias for Equality of Opportunity

We achieve Equation (4) for EOpp by learning the conditional CDFs of weighted scores (c.f. Lemma 1) where the weights are given by the inverse of the decay factor defined in Equation (5).

Theorem 2. Under Assumption 1, let w_i be as in Equation (5) and let $F_{c,1}^*$ denote the CDF of the conditional scores s(X) given $Y(\gamma) = 1$ and C = c with weights $1/w_{\gamma}$. Then the transformed scores $\tilde{s}(X) := \sum_{c} F_{c,1}^*(s(X)) 1_{\{C=c\}}$

1.
$$\mathbb{P}(\tilde{\gamma} = j \mid C = c_1, Y(1) = 1) = \mathbb{P}(\tilde{\gamma} = j \mid C = c_2, Y(1) = 1)$$
, and

$$c_{2}, Y(1) = 1$$
), and
2. $\mathbb{P}(\tilde{s}(X) \le t \mid C = c_{1}, Y(\tilde{\gamma}) = 1) = \mathbb{P}(\tilde{s}(X) \le t \mid C = c_{2}, Y(\tilde{\gamma}) = 1)$,

for all j, t and c_1, c_2 ,, where $\tilde{\gamma}$ denotes the position of an item based on $\tilde{s}(X)$.

The first part of Theorem 2 shows that the weighted CDF transformations guarantee "fairness of exposure" with respect to the items with a (counterfactual) positive response at position 1. This is also related to the notion of group fairness parity in Singh & Joachims (2019a).

Corollary 1. For a characteristic C = c, let $M_c :=$ $\mathbb{P}(Y(1) = 1 \mid C = c)$ and $v_{obs}(c) := \mathbb{P}(Y(\tilde{\gamma}) = 1 \mid C = c)$ c) be the average merit and the observed exposure (with respect to the scoring function $\tilde{s}(X)$ defined in Theorem 2), respectively. Then under Assumption 1, $\tilde{s}(X)$ achieves the group fairness parity, given by the following constraint: $v_{obs}(c_1)/M_{c_1} = v_{obs}(c_2)/M_{c_2}$, for all c_1, c_2 .

4.2. Tackling Position Bias for Equalized Odds

We achieve Equation (4) for *EOdds* by carefully adjusting the counts of positive and negative labels in certain data segments defined by $\{C=c,Y(\gamma)=y,s(X)\in I_k,\gamma=1\}$ j}. To motivate our approach, we first show that if we had access to the counterfactual label Y(1), then the method defined in Section 3.1 would have worked. Then we describe how we can estimate $\mathbb{P}(\tilde{s}(X) \in I_k \mid C = c, Y(1) = y)$ by adjusting for the position bias.

Theorem 3. Under Assumption 1, let $\tilde{s}(X)$ be such that

$$\mathbb{P}(\tilde{s}(X) \le t \mid C = c_1, Y(1) = y) = \\ \mathbb{P}(\tilde{s}(X) \le t \mid C = c_2, Y(1) = y),$$

for $k \in \{1, ..., K\}$, $y \in \{0, 1\}$ and for all c_1, c_2 . Then $\tilde{s}(X)$ satisfies the equalized odds conditions given in Definition 7.

To use Theorem 3 in conjunction with the optimization given in 3, we need to estimate $p_{c,y,k}^* := \mathbb{P}(s(X) \in I_k \mid C =$

Algorithm 1 Position Bias Adjusted Equality of Opportunity

1: Reranker Training

- 2: Input: Score, position, label and characteristic data $(s_i, \gamma_i, y_i, c_i), i = 1, ..., n$
- 3: Compute the weighted empirical CDF $\hat{F}_{c,1}^*$ of the conditional scores s(X) given $Y(\gamma)=1$ and C=c with weights $1/\hat{w}_{\gamma}$ computed as in Section 4.3
- 4: Output: The empirical distribution functions $\hat{F}_{c,1}^*$
- 5: Reranker Scoring
- 6: Input: Score S, characteristic C
- 7: Output: Fair score, $\tilde{S} = \sum_{c} \hat{F}_{c,1}^{*}(S) 1_{\{C=c\}}$

c,Y(1)=y). Note that if the counterfactual label Y(1) were known $p_{c,y,k}^*$ estimated as $n_{c,y,k}^*/(\sum_k n_{c,y,k}^*)$ where $n_{c,y,k}^*=\sum_i 1_{\{C_i=c,Y_i(1)=y,s_i(X)\in I_k\}}.$

Let $n_{c,y,k}^{(j)*} = \sum_i 1_{\{C_i=c,Y_i(1)=y,s_i(X)\in I_k,\gamma=j\}}$ such that $n_{c,y,k}^* = \sum_j n_{c,y,k}^{(j)*}$. Now we approximate $n_{c,y,k}^{(j)*}$ using the observed counts $n_{c,y,k}^{(j)} = \sum_i 1_{\{C=c,Y(\gamma)=y,s(X)\in I_k,\gamma=j\}}$ by adjusting for position bias.

Recall $w_j = \mathbb{P}(Y(j) = 1 \mid Y(1) = 1)$ defined in (5). To adjust for the positive response decay, we inflate the positive label count and adjust the negative label count as follows.

$$\begin{split} n_{c,1,k}^{(j)\prime} &= n_{c,1,k}^{(j)}/w_j \text{ and } n_{c,0,k}^{(j)\prime} = (n_{c,0,k}^{(j)} + n_{c,1,k}^{(j)}) - n_{c,1,k}^{(j)\prime}. \end{split}$$
 Finally, we estimate $\mathbb{P}(s(X) \in I_k \mid C = c, Y(1) = y)$ as

$$\hat{\mathbb{P}}(s(X) \in I_k \mid C = c, Y(1) = y) = \frac{\sum_j n_{c,y,k}^{(j)'}}{\sum_{j,k} n_{c,y,k}^{(j)'}}.$$
 (6)

The correctness of this adjustment follows from the following result.

Theorem 4. Under Assumption 1, $\hat{\mathbb{P}}(s(X) \in I_k \mid C = c, Y(1) = y)$ converges to $\mathbb{P}(s(X) \in I_k \mid C = c, Y(1) = y)$ almost surely for all k, c, y.

4.3. Position Bias Estimation

The weighted CDF transformations defined in Theorem 2 require the weights to be known. To estimate the position bias w_j , we may collect data by randomizing slots and estimate the ratio of click-through-rates (CTRs) at position j and position 1 (CTR at j = the probability of observing a positive response at position j), i.e.

$$\hat{w}_{j} = \frac{\left(\sum_{i} 1_{\{Y_{i}(\gamma)=1, \ \gamma=j\}}\right) / \left(\sum_{i} 1_{\{\gamma=j\}}\right)}{\left(\sum_{i} 1_{\{Y_{i}(\gamma)=1, \ \gamma=1\}}\right) / \left(\sum_{i} 1_{\{\gamma=1\}}\right)}.$$
 (7)

Without the randomization, we will end up underestimating the CTR ratio by using Equation (7) as the items served at position j are expected to be of lower quality than the items served at position 1. More precisely, in the observational data where the items are ranked according to s(X), the conditional distribution of s(X) given $\gamma = 1$ is expected to

Algorithm 2 Position Bias Adjusted Equalized Odds

1: Reranker Training

- 2: Input: Score, position, label and characteristic data $(s_i, \gamma_i, y_i, c_i)$, i = 1, ..., n and a partition of the score space $I_1, ..., I_K$
- 3: Estimate empirical conditional bin probabilities as in Equation (6) with weights estimated as in Section 4.3
- 4: Find *P*, the solution to Optimization Problem (3)
- 5: Output: Vector of interval transition probabilities, P
- 6: Reranker Scoring
- 7: Input: Score S, characteristic C, a partition of the score space $I_1, ..., I_K$, interval transition probabilities $P = \{p_{k,k',c}\}$ and distribution functions $F_1, ..., F_K$
- 8: Determine k such that $S \in I_k$
- 9: Choose k' from a multinomial distribution with probabilities $\{p_{k,1,C}, \dots, p_{k,K,C}\}$
- 10: Randomly select a point \tilde{S} within I_k according to $F_k(\cdot)$
- 11: Output: Fair score, \tilde{S}

be stochastically larger than the conditional distribution of s(X) given $\gamma = j$.

However, randomly shuffling all items can undesirably harm viewers' experience. A less harmful alternative is to shuffle pairs of items randomly (Joachims et al., 2017; Wang et al., 2018). To estimate the weights from observational data (Wang et al., 2018) applied the EM algorithm with a parametric click model. Below, we propose a non-parametric approach to estimate the weights from observational data based on importance sampling. To this end, we first estimate the response bias at position j relative to position j-1 by correcting for the discrepancy in the distribution of scores in those positions with importance weighting as follows.

Let $f_j(\cdot)$ denote the conditional density of the observed score at position j. For $j \geq 2$, define

$$\eta_{j} = \frac{\mathbf{E}\left(Y(\gamma) \frac{f_{j-1}(s(X))}{f_{j}(s(X))} \mid \gamma = j\right)}{\mathbf{E}\left(Y(\gamma) \mid \gamma = j - 1\right)}.$$

It is straightforward to estimate η_j by replacing the conditional density and the conditional expectations with the corresponding empirical estimates. We denote this estimator by $\hat{\eta}_j$. Finally, we estimate w_j by $\hat{w}_j = \prod_{r=2}^j \hat{\eta}_r$.

The reason for not estimating the w_j directly by using the importance weights $f_1(s(X))/f_j(s(X))$ is that the distribution of scores at position 1 might be much different from its counterpart at position j, even for not-so-large large j. Our adjacent-pairwise importance sampling approach tends to have a lower variance than the direct importance sampling approach since the score distributions of the adjacent positions are expected to be relatively close to each other. However, for practical purposes we recommend to use a truncated version $\hat{w}_j = \prod_{r=2}^{\min(j,T)} \hat{\eta}_r$ for some threshold

T. Note that this is equivalent to assuming $\eta_j=1$ for all j>T, which is a reasonable practical assumption for most recommendation systems. The overall algorithms for position adjusted EOpp and EOdds are given in Algorithms 1 and 2 respectively.

5. Experiments

Here we do a thorough simulation study to validate the need and the efficacy of our algorithms to achieve fairness in web-scale recommender systems. First we describe the datageneration mechanism and then focus on the experiments for the two large-scale fairness re-rankers.

Data Generation: We generate a population of p=50,000 items, where each item consists of id i, characteristic $C_i \sim \{0,1\}$, $Y_i(1) \in \{0,1\}$ and relevance R_i . We independently generate C_i 's from a Bernoulli(0.6) distribution. The conditional distribution $Y_i(1)$ given $C_i=0$ is Bernoulli(0.4), and the conditional distribution $Y_i(1)$ given $C_i=1$ is Bernoulli(0.5). Finally, $R_i \mid (C_i, Y_i(1))$ is generated from

$$\mathcal{N}(0.6Y_i(1) + 2C_i, 0.5) + (1 - C_i)Uniform[0, (1 + Y_i(1))].$$

We consider a recommendation system with K=50 slots. For each query, we randomly select 50 items from the population and assign a score $s_i=R_i+\mathcal{N}(0,0.1)$ to each selected item $i\in\{1,\ldots,50000\}$. The selected items are then ranked according to s_i (in a descending order) and assigned position according to $\operatorname{rank}(i)$. Finally, the item at position j gets observed response $Y(j)=Y(1)\times Bernoulli(w_j)$ with position bias $w_j=1/\log_2(1+j)$. We generate a training dataset based on 100,000 queries and a validation dataset based on 50,000 queries.

5.1. Results

We use the adjacent-pairwise importance sampling approach with the threshold T=30 (see Figure 1) for the position bias estimation based on the training dataset.

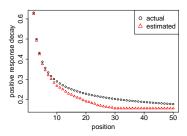


Figure 1. Estimating the position bias $w_j = 1/\log 2(1+j)$ for j = 2, ..., 50, using the adjacent-pairwise importance sampling approach with the threshold T = 30.

Next, we learn the weighted empirical CDF for *EOpp* based on the training data. To learn the position bias adjusted *EOdds* re-ranker based on the training data, we apply an inverse-logit transformation to the score, discretize the score

using 100 equally spaced intervals and solve the optimization problem given in (3) with $E|s(X) - \bar{s}(X)|$ as the objective function.

We apply these transformations on the validation data scores and each time regenerate the online labels (with position bias) based on the rankings of the items given by the transformed scores. Figure 2 validates the usefulness of our algorithms in achieving *EOpp* and *EOdds*. Prior to these transformations, it is seen that the conditional score distributions differ greatly between the characteristics. Post-transforming, the conditional score distributions are identical, as required for equalized odds. Recall that the *EOpp* transformation only guarantees to produce identical score distributions across all groups for positive labels, while *EOdds* guarantees the same for positive labels as well as for negative labels. These are reflected in Figure 2. Note that we transformed the scores back to the original scale using a CDF and inverse CDF transformation as described in Section 2.1.

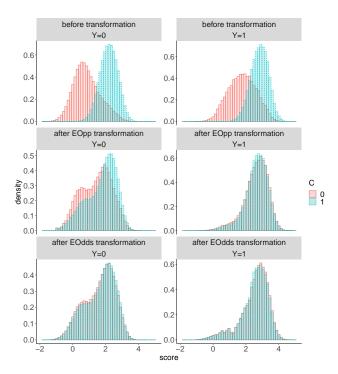


Figure 2. Score distributions corresponding to online negative responses (left column) positive responses (right column) for groups C=0 (red) and C=1 (green) (i) before transformation (top panel), (ii) after EOpp transformation (middle panel) and (iii) after EOdds transformation (bottom panel).

We implemented¹ the Algorithms in R. Based on the training data with 5 million samples (100k queries with 50 slots), the position bias estimation took 7 seconds, the *EOpp* learning took 9 seconds, and the *EOdds* learning with 100 bins took 100 seconds on a Macbook Pro with 3.5 GHz

¹Code is available in the supplementary material.

Dual-Core Intel Core i7 processor and 16 GB 2133 MHz LPDDR3 memory, demonstrating the scalability of the proposed algorithms.

Finally, Figure 3 demonstrates how a desirable performancefairness trade-off can be achieved via a linear combination of the transformed scores and the original scores given by,

 $\alpha \times (\text{transformed score}) + (1 - \alpha) \times (\text{original score})$

where $\alpha \in [0, 1]$ is a tuning parameter (see (1)). Here, we observe the following:

- 1. The unfairness decreases to zero monotonically as α increases, except for EOpp corresponding to negative labels (as expected).
- 2. The unfairness corresponding to the *EOpp* transformation is strictly lower for all values of α , while the performance of *EOpp* is strictly better for $\alpha \geq 0.4$.
- Surprisingly, the performance corresponding to the EOpp transformation is monotonically increasing with α. This serves as a counterexample to the popular belief that fairness and performance are always conflicting properties of recommender systems.

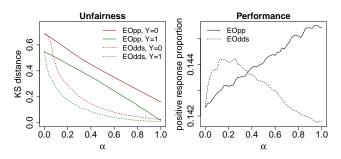


Figure 3. Unfairness and performance for a number of values of the tuning parameter α , where unfairness is measured by the Kolmogorov-Smirnov distance between the distribution of scores with positive responses (green lines) and the distribution of scores with negative responses (red lines), and the performance is measured by the proportion of positive responses (i.e., click-through rate when clicks are the positive responses).

6. Real-World Application

The friend or connection recommendation system is a social network product in several large internet companies such as Facebook, Instagram, Twitter, LinkedIn, etc. This recommender system suggests members to connect with others, in order to build their network. In this system, a member sending an invitation to connect with a recommended candidate can be viewed as a positive outcome.

The training data used for this product arises from historical recommendations and the labels are whether or not an invitation has been sent. The training data usually have a lower representation of infrequent members (IMs; members who are less active on the platform) in comparison to frequent

members (FMs) as candidate recommendation, due to their lower rates of engagement. This means there is potential for the model to not only be biased against IMs, but for that bias to be reinforced over time, leading to a system that is optimized for the benefit of members who are already highly engaged on the site. In this literature this is commonly known as the "popularity bias" or "rich getting richer" phenomenon (Abdollahpouri et al., 2019).

We see an opportunity to apply fairness notions, as a debiasing mechanism, to adjust for the exposure of IMs as candidates being recommended, thus giving them opportunities to be shown and invited. In our experiments, we applied both the *EOpp* and *EOdds* reranker to give qualified IMs and FMs equal representation in recommendations. We build the required transformations using two weeks of training data. While serving we apply the transformation on the top 100 candidates ensuring that we do not introduce noise by transforming candidates at all positions. We chose $\alpha = 0.99$ in order to push more towards a better representation of IMs (potentially at some cost to FMs). The serving was done via a parametrized CDF for EOpp and through the estimated transition probabilities for *EOdds*. Due to the simple transformation in both approaches, we did not see any drastic gain in latency, which is a core-requirement in large-scale recommender systems. The results of the A/B tests on real member traffic are shown in Table 1.

Invitation	EOpp		EOdds	
Metrics	IM	FM	IM	FM
Sent	+5.72%	Neutral	+2.77%	Neutral
Accepted	+ 4.85%	Neutral	+ 2.26%	Neutral

Table 1. A/B Experimentation results for the two fairness rerankers. In both setups, we observed improved metrics of invitations sent and accepted by IMs without any statistically significant impact to the same metrics corresponding to FMs.

Two of the cornerstone metrics for evaluating such experiments are invitations sent, and invitations accepted (connection made). While one might expect that this post-processing approach would shift invitations away from FMs to IMs and may be detrimental to the metrics for FMs, we were heartened to see more invites being sent to and accepted by IMs without any negative impact on the FMs. This highlights that re-ranking approaches pursuing fairness have improved recommendation quality overall.

7. Conclusion

We have proposed post-processing methods for handling equality of opportunity and equalized odds in rankings, requiring handling of problems induced by position bias. They are applicable in many internet-industry applications as they are tailored towards scalability, have a relatively low engineering footprint, and can be relatively easily added on top

of existing machine-learning/AI pipelines.

References

- Abdollahpouri, H., Mansoury, M., Burke, R., and Mobasher, B. The unfairness of popularity bias in recommendation. *arXiv* preprint arXiv:1907.13286, 2019.
- Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., and Wallach, H. A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 60–69. PMLR, 2018a.
- Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., and Wallach, H. A reductions approach to fair classification. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 60–69, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018b. PMLR.
- Agarwal, D., Basu, K., Ghosh, S., Xuan, Y., Yang, Y., and Zhang, L. Online Parameter Selection for Web-based Ranking Problems. In *KDD*, pp. 23–32, New York, NY, USA, 2018c. ACM. ISBN 978-1-4503-5552-0.
- Amaratunga, D. and Cabrera, J. Analysis of data from viral dna microchips. *Journal of the American Statistical Association*, 96(456):1161–1170, 2001. doi: 10.1198/016214501753381814.
- Barocas, S., Hardt, M., and Narayanan, A. Fairness in machine learning. *NIPS Tutorial*, 1, 2017.
- Bechavod, Y. and Ligett, K. Penalizing unfairness in binary classification. arXiv:1707.00044, 2017.
- Bolstad, B., Irizarry, R., Åstrand, M., and Speed, T. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, 01 2003. ISSN 1367-4803. doi: 10.1093/bioinformatics/19.2.185.
- Calmon, F., Wei, D., Vinzamuri, B., Natesan Ramamurthy, K., and Varshney, K. R. Optimized pre-processing for discrimination prevention. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 3992–4001. Curran Associates, Inc., 2017.
- Celis, L. E., Straszak, D., and Vishnoi, N. K. Ranking with fairness constraints, 2017.
- Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pp. 214–226, 2012.

- D'Amour, A., Srinivasan, H., Atwood, J., Baljekar, P., Sculley, D., and Halpern, Y. Fairness is not static: Deeper understanding of long term fairness via simulation studies. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, pp. 525–534. Association for Computing Machinery, 2020.
- Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 259–268, 2015.
- Goel, N., Yaghini, M., and Faltings, B. Non-discriminatory machine learning through convex fairness criteria. In *Proceedings of the 2018 AAAI/ACM Conference on AI*, *Ethics, and Society*, pp. 116–116, 2018.
- Goh, G., Cotter, A., Gupta, M., and Friedlander, M. P. Satisfying real-world goals with dataset constraints. In Advances in Neural Information Processing Systems, pp. 2415–2423, 2016.
- Gordaliza, P., Barrio, E. D., Fabrice, G., and Loubes, J.-M. Obtaining fairness using optimal transport theory. In Chaudhuri, K. and Salakhutdinov, R. (eds.), Proceedings of the 36th International Conference on Machine Learning, volume 97 of Proceedings of Machine Learning Research, pp. 2357–2365. PMLR, 09–15 Jun 2019.
- Hardt, M., Price, E., Price, E., and Srebro, N. Equality of opportunity in supervised learning. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R. (eds.), Advances in Neural Information Processing Systems 29, pp. 3315–3323. Curran Associates, Inc., 2016.
- Joachims, T., Swaminathan, A., and Schnabel, T. Unbiased learning-to-rank with biased feedback. In *Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, pp. 781–789. Association for Computing Machinery, 2017.
- Johndrow, J. E., Lum, K., et al. An algorithm for removing sensitive information: application to race-independent recidivism prediction. *The Annals of Applied Statistics*, 13(1):189–220, 2019.
- Kamiran, F., Karim, A., and Zhang, X. Decision theory for discrimination-aware classification. In 2012 IEEE 12th International Conference on Data Mining, pp. 924–929. IEEE, 2012.
- Kamishima, T., Akaho, S., Asoh, H., and Sakuma, J. Fairness-aware classifier with prejudice remover regularizer. In Flach, P. A., De Bie, T., and Cristianini, N. (eds.), *Machine Learning and Knowledge Discovery in Databases*, pp. 35–50, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-33486-3.

- Mary, J., Calauzenes, C., and El Karoui, N. Fairness-aware learning for continuous attributes and treatments. In *International Conference on Machine Learning*, pp. 4382–4391, 2019.
- Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. On fairness and calibration. In *Advances in Neural Information Processing Systems*, pp. 5680–5689, 2017.
- Singh, A. and Joachims, T. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, pp. 2219–2228, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450355520. doi: 10.1145/3219819.3220088.
- Singh, A. and Joachims, T. Policy learning for fairness in ranking. In *Advances in Neural Information Processing Systems 32*, pp. 5426–5436. Curran Associates, Inc., 2019a.
- Singh, A. and Joachims, T. Policy learning for fairness in ranking. *arXiv preprint arXiv:1902.04056*, 2019b.
- Wang, X., Golbandi, N., Bendersky, M., Metzler, D., and Najork, M. Position bias estimation for unbiased learning to rank in personal search. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM)*, pp. 610–618, 2018.
- Wu, Y., Zhang, L., and Wu, X. On convexity and bounds of fairness-aware classification. In *The World Wide Web Conference*, pp. 3356–3362, 2019.
- Zafar, M. B., Valera, I., Gomez Rodriguez, M., and Gummadi, K. P. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*, pp. 1171–1180, 2017a.
- Zafar, M. B., Valera, I., Rogriguez, M. G., and Gummadi, K. P. Fairness Constraints: Mechanisms for Fair Classification. In Singh, A. and Zhu, J. (eds.), Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, volume 54 of Proceedings of Machine Learning Research, pp. 962–970, Fort Lauderdale, FL, USA, 20–22 Apr 2017b. PMLR.
- Zafar, M. B., Valera, I., Rogriguez, M. G., and Gummadi, K. P. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pp. 962–970. PMLR, 2017c.
- Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. Learning fair representations. In Dasgupta, S. and McAllester, D. (eds.), *Proceedings of the 30th Interna*tional Conference on Machine Learning, volume 28 of

- *Proceedings of Machine Learning Research*, pp. 325–333, Atlanta, Georgia, USA, 17–19 Jun 2013a. PMLR.
- Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. Learning fair representations. In Dasgupta, S. and McAllester, D. (eds.), *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 325–333, Atlanta, Georgia, USA, 17–19 Jun 2013b. PMLR.

8. Appendix

8.1. Proofs

Proof of Lemma 1. Let $s_{c,1}$ denote the random variable corresponding to the score s(X) restricted to C=c and Y=1. Since $F_{c,1}$ is the CDF of $s_{c,1}$, the distribution of $F_{c,1}(s_{c,1})$ is Uniform[0,1] for all c. Hence these transformed scores satisfy equality of opportunity across all thresholds.

Proof of Theorem 2. We first prove the results using the following claims and then prove the claims.

Claim 1: For all i > 1 and for all t,

$$P(s(X) \le t \mid \gamma = j, Y(j) = 1, C = c)$$

= $P(s(X) \le t \mid \gamma = j, Y(1) = 1, C = c)$.

Claim 2: For all i > 1,

$$\begin{split} P(\gamma = j \mid Y(\gamma) = 1, \ C = c) \\ &= \frac{w_j P(\gamma = j \mid Y(1) = 1, \ C = c)}{\sum_r w_r P(\gamma = r \mid Y(1) = 1, \ C = c)}. \end{split}$$

Using Claims 1 and 2, we show that the CDF of s(X) given Y(1) = 1 and C = c equals $F_{c,1}^*$.

$$\begin{split} F_{c,1}^*(t) &:= \sum_j \left\{ P(s(X) \leq t \mid \gamma = j, \, Y(j) = 1, \, C = c) \right. \\ &\times \frac{P(\gamma = j \mid Y(\gamma) = 1, C = c) / w_j}{\sum_r P(\gamma = r \mid Y(\gamma) = 1, C = c) / w_r} \right\} \\ &= \sum_j \left\{ P(s(X) \leq t \mid \gamma = j, \, Y(1) = 1, \, C = c) \right. \\ &\times \frac{P(\gamma = j \mid Y(1) = 1, C = c)}{\sum_r P(\gamma = r \mid Y(1) = 1, C = c)} \right\} \\ &= \sum_j \left\{ P(s(X) \leq t \mid \gamma = j, \, Y(1) = 1, \, C = c) \right. \\ &\times P(\gamma = j \mid Y(1) = 1, C = c) \right\} \\ &= \sum_j P(s(X) \leq t, \, \gamma = j \mid Y(1) = 1, \, C = c) \\ &= P(s(X) \leq t \mid Y(1) = 1, \, C = c). \end{split}$$

The first equality follows from the definition of $F_{c,1}^*$, we use Claims 1 and 2 in the second equality, the third equality follows from the fact that $\sum_j P(\gamma=j\mid Y(1)=1,C=c)=1$, and the fourth equality follows from the fact that

$$\cup_j \big(\{ s(X) \le t \} \cap \{ \gamma = j \} \big) = \{ s(X) \le t \}.$$

We have shown that the conditional distribution of s(X) given Y(1) = 1 and C = c equals $F_{c,1}^*$. This implies that the conditional distribution of $\tilde{s}(X) := F_{c,1}^*(s(X))$ given Y(1) = 1 and C = c is Uniform[0, 1] for all c.

Therefore, the conditional distributions of the position $\tilde{\gamma}$ given Y(1)=1 and C=c are identical for all c. To see this, note that

$$P(\tilde{\gamma} \le j \mid Y(1) = 1, \ C = c)$$

= $P(\tilde{s}(X) > U_{(K-i)} \mid Y(1) = 1, \ C = c), \ (8)$

where K is the total number of positions, $U_{(0)}=0$ and for j < K, $U_{(K-j)}$ is the (K-j)-th order statistic corresponding to K i.i.d. uniform samples. Note that the right hand side of (8) does not depend on c since the conditional distribution of $\tilde{s}(X):=F_{c,1}^*(s(X))$ given Y(1)=1 and C=c is $Uniform[0,\ 1]$ for all c. This completes the proof of the first part of the theorem.

To prove the second part, we will use Claims 1 and 2 with $(\tilde{s}(X), \tilde{\gamma})$ instead of $(s(X), \gamma)$.

$$\begin{split} &P(\tilde{s}(X) \leq t \mid C = c, \ Y(\tilde{\gamma}) = 1) \\ &= \sum_{j} \left\{ P(\tilde{s}(X) \leq t \mid \tilde{\gamma} = j, \ Y(j) = 1, \ C = c) \right. \\ &\times P(\tilde{\gamma} = j \mid Y(\tilde{\gamma}) = 1, C = c) \right\} \\ &= \frac{\sum_{j} w_{j} \ P(\tilde{s}(X) \leq t, \ \tilde{\gamma} = j \mid Y(1) = 1, \ C = c)}{\sum_{r} w_{r} P(\tilde{\gamma} = r \mid Y(1) = 1, C = c)}. \end{split}$$

Now note that it follows from the first part of theorem that the denominator is identical for each c. Furthermore, the numerator does not depend on c, since it follows from (8) and the independence of $(\tilde{s}(X) \mid Y(1) = 1)$ and C that

$$\begin{split} &P(\tilde{s}(X) \leq t, \ \tilde{\gamma} = j \mid Y(1) = 1, \ C = c) \\ = &P(U_{(K-j)} < \tilde{s}(X) \leq \min\{U_{(K-j+1)}, \ t\} \mid Y(1) = 1). \end{split}$$

This completes the second part of the theorem.

Proof of Claim 1: From the second part of Assumption 1, it follows that

$$\begin{split} &P(s(X) \leq t \mid \gamma = j, \; Y(j) = 1, \; C = c) \\ &= P(s(X) \leq t \mid \gamma = j, \; Y(j) = 1, \; Y(1) = 1, \; C = c). \end{split}$$

Therefore, the results follows from the first part of Assumption 1 that ensures that Y(j) is independent of s(X), γ and C given Y(1).

Proof of Claim 2: For notational convenience, we denote the conditional probabilities given C = c by $P_c(\cdot)$. Using the second part of Assumption 1 and then applying

the Bayes' theorem, we get

$$\begin{split} &P_c(\gamma=j\mid Y(\gamma)=1)\\ =&P_c(\gamma=j\mid Y(\gamma)=1,\; Y(1)=1)\\ =&\frac{P_c(Y(\gamma)=1\mid Y(1)=1,\; \gamma=j)P_c(\gamma=j\mid Y(1)=1)}{\sum_r P_c(Y(\gamma)=1\mid Y(1)=1,\; \gamma=r)P_c(\gamma=r\mid Y(1)=1)}\\ =&\frac{P(Y(j)=1\mid Y(1)=1)P_c(\gamma=j\mid Y(1)=1)}{\sum_r P(Y(r)=1\mid Y(1)=1)P_c(\gamma=r\mid Y(1)=1)}\\ =&\frac{w_j P_c(\gamma=j\mid Y(1)=1)}{\sum_r w_r P_c(\gamma=r\mid Y(1)=1)}. \end{split}$$

The seconds last equality follows from the first part of Assumption 1, and the last equality follows from the definition of w_i .

Proof of Corollary 1. For notational convenience, we denote the conditional probabilities given C=c by $P_c(\cdot)$. Note that

$$\begin{split} &\frac{v_{obs}(c)}{M_c} \\ &= \frac{\sum_{j} P_c(Y(j) = 1 \mid Y(1) = 1, \ \tilde{\gamma} = j) P_c(Y(1) = 1, \ \tilde{\gamma} = j)}{P_c(Y(1) = 1)} \\ &= \frac{\sum_{j} P(Y(j) = 1 \mid Y(1) = 1) P_c(Y(1) = 1, \ \tilde{\gamma} = j)}{P_c(Y(1) = 1)} \\ &= \sum_{j} w_j P_c(\tilde{\gamma} = j \mid Y(1) = 1), \end{split}$$

where w_j is as in Theorem 2 and the second equality follows from the first part of Assumption 1. Therefore, the result follows from the first part of Theorem 2.

Proof of Theorem 1. Fix a $t \in [0, 1)$. Assume $t \in I_k$, i.e. $i_k \le t < i_{k+1}$. Then,

$$\begin{split} P(\bar{s}(X) &\leq t \mid C = c, Y = y) \\ &= P(\bar{s}(X) \leq t \mid \bar{s}(X) \in I_k, C = c, Y = y) \\ &\times P(\bar{s}(X) \in I_k \mid C = c, Y = y) \\ &= F_k(t) \cdot P(\bar{s}(X) \in I_k \mid C = c, Y = y) \end{split}$$

Now, $F_k(t)$ does not depend on y or c, and the constraints satisfied in the optimization problem imply that

$$P(\bar{s}(X) \in I_k \mid C = c, Y = y)$$

does not depend on c conditionally on y. It follows that equalized odds is satisfied in the ranking sense.

Proof of Theorem 3. Note that without loss of generality, we can assume that the $\tilde{s}(X)$) given Y(1)=1 and C=c is Uniform[0,1]. This is because we can transform the score using the same CDF function (since they have the same distribution) to make them Uniform[0,1] for all c. Then it follows from the arguments given the proof of Theorem 2 that $P(\tilde{s}(X) \leq t \mid C=c, Y(\tilde{\gamma})=1)$ is independent of c.

Proof of Theorem 4. For notational convenience, we de-

note the conditional probabilities given C=c by $P_c(\cdot)$. By applying Bayes' Theorem, we get

$$\mathbb{P}_c(s(X) \in I_k \mid Y(1) = y)$$

$$= \frac{\mathbb{P}_c(Y(1) = y, \ s(X) \in I_k)}{\sum_{\ell} \mathbb{P}_c(Y(1) = y, \ s(X) \in I_{\ell})}.$$

Next, note that

$$\mathbb{P}_{c}(Y(1) = 1, s(X) \in I_{\ell})$$

$$= \sum_{j} \mathbb{P}_{c}(Y(1) = 1, \gamma = j, s(X) \in I_{\ell}).$$

$$= \sum_{j} \left\{ \mathbb{P}_{c}(s(X) \in I_{\ell} \mid Y(1) = 1, \gamma = j) \times \mathbb{P}(Y(1) = 1, \gamma = j) \right\}$$

$$\times \mathbb{P}(Y(1) = 1, \gamma = j)$$

$$\times \mathbb{P}_{c}(s(X) \in I_{\ell} \mid Y(j) = 1, \gamma = j)$$

$$\times \mathbb{P}_{c}(Y(1) = 1, \gamma = j)$$

$$= \sum_{j} \frac{\mathbb{P}_{c}(s(X) \in I_{\ell}, Y(j) = 1, \gamma = j)}{\mathbb{P}_{c}(Y(j) = 1 \mid Y(1) = 1, \gamma = j)}$$

$$= \sum_{j} \frac{\mathbb{P}_{c}(s(X) \in I_{\ell}, Y(j) = 1, \gamma = j)}{\mathbb{P}(Y(j) = 1 \mid Y(1) = 1)}$$

$$= \sum_{j} \frac{\mathbb{P}_{c}(s(X) \in I_{\ell}, Y(j) = 1, \gamma = j)}{\mathbb{P}_{c}(s(X) \in I_{\ell}, Y(j) = 1, \gamma = j)}.$$

The first, second and the fourth equalities follow from the definition of conditional probability, the third equality follows from Claim 1 in the proof of Theorem 2, the fifth equality follows from Assumption 1 and the last equality follows from the definition of w_i .

Now it follows from the strong law of large numbers that $\frac{n_{c,1,\ell}^{(j)}}{n_c}$ converges almost surely to $\mathbb{P}_c(s(X) \in I_\ell, Y(j) = 1, \ \gamma = j)$, where n_c is the number of samples corresponding to C = c. Hence, we have

$$\hat{\mathbb{P}}_c(s(X) \in I_k \mid Y(1) = 1) \xrightarrow{a.s} \mathbb{P}_c(s(X) \in I_k \mid Y(1) = 1),$$
 where $\stackrel{a.s}{\longrightarrow}$ denotes almost sure convergence.

Finally, note that

$$\begin{split} & \mathbb{P}_c(Y(1) = 0, \ s(X) \in I_{\ell}) \\ & = \mathbb{P}_c(s(X) \in I_{\ell}) - \mathbb{P}_c(Y(1) = 1, \ s(X) \in I_{\ell}) \\ & = \mathbb{P}_c(s(X) \in I_{\ell}) - \sum_i \frac{\mathbb{P}_c(s(X) \in I_{\ell}, Y(j) = 1, \ \gamma = j)}{w_j}. \end{split}$$

Therefore,

$$\hat{\mathbb{P}}_c(s(X) \in I_k \mid Y(1) = 0) \xrightarrow{a.s} \mathbb{P}_c(s(X) \in I_k \mid Y(1) = 0)$$
 follows from the strong law of large numbers similarly. \square