General Post-Processing Framework for Fairness Adjustment of Machine Learning Models

Léandre Eberhard, Nirek Sharma, Filipp Shelobolin, Aalok Ganesh Shanbhag

April 24, 2025

1 Abstract

As machine learning increasingly influences critical domains such as credit underwriting, public policy, and talent acquisition, ensuring compliance with fairness constraints is both a legal and ethical imperative. This paper introduces a novel framework for fairness adjustments that applies to diverse machine learning tasks, including regression and classification, and accommodates a wide range of fairness metrics. Unlike traditional approaches categorized as pre-processing, in-processing, or post-processing, our method adapts in-processing techniques for use as a post-processing step. By decoupling fairness adjustments from the model training process, our framework preserves model performance on average while enabling greater flexibility in model development. Key advantages include eliminating the need for custom loss functions, enabling fairness tuning using different datasets, accommodating proprietary models as black-box systems, and providing interpretable insights into the fairness adjustments. We demonstrate the effectiveness of this approach by comparing it to Adversarial Debiasing, showing that our framework achieves a comparable fairness/accuracy tradeoff on real-world datasets.

2 Introduction

In machine learning, the primary goal when building a model is to optimize a given performance metric on a dataset. However, in many fields, such as credit underwriting, public policy, talent acquisition, and others, it is necessary to satisfy other fairness constraints in addition to the performance metric for which the model is optimized. Often, the performance metric is at odds with the fairness metric and optimizing for performance alone can yield a model that does not satisfy the fairness constraints. Numerous works, such as [Hardt et al.(2016), Agarwal et al.(2018), Donini et al.(2018), Zafar et al.(2017)], have formally studied these fairness requirements, noting that purely optimizing a baseline loss may lead to disparate outcomes for different groups. In practical applications like credit analysis, public policy, or job hiring, large models are frequently used, and sometimes neural networks or boosted trees can outperform each other depending on data characteristics [McElfresh et al.(2024)].

A variety of open-source toolkits have emerged to facilitate bias detection and mitigation, such as the IBM AI Fairness 360 Toolkit [Bellamy et al.(2018)], which has implemented many existing methods. In academic benchmarks, the UCI repository [Dua and Graff(2017)] and the COMPAS

dataset [Angwin et al.(2016)] have been commonly used to demonstrate bias or measure fairness improvements.

2.1 Related Work

Traditional fairness bias mitigation strategies can be classified into three classes based on the stage of the model-building process at which they are applied [Mehrabi et al.(2022)]:

Pre-Processing. The goal of pre-processing methods is to edit the training data used to fit the model in order to improve the fairness of the resulting model, such as by sampling [Calders et al.(2009), Iosifidis and Ntoutsi(2018)] or perturbation of the data [Wang et al.(2019), Hajian and Domingo-Ferrer(2012)], however both of these methods may change the true data distribution, which can decrease model performance.

In-Processing. In contrast, in-processing methods modify the training algorithm to remove bias during model training time, usually by adding regularization to the loss function [Jung et al.(2020), Kim et al.(2018), Dwork et al.(2011)]. However, these methods are heavily dependent on the learning algorithm and can be difficult or impossible to adapt to different algorithms. Some recent work has proposed reductions or probabilistic approaches to unify these constraints under a single framework [Donini et al.(2018)]. A particularly high-performing method is adversarial debiasing, which uses adversarial learning to debias the model [Grari et al.(2019b), Celis and Keswani(2019), Wadsworth et al.(2018), Zhang et al.(2018)].

Post-Processing. This class of methods is useful when changing the training data or model algorithm is infeasible. The advantage is that the underlying model can be treated as a black-box and does not need to be retrained. Traditional approaches in this class usually involve reassigning predicted labels [Lohia et al.(2018), Menon and Williamson(2018), Kamiran et al.(2012)]. Recent models have also attempted to adjust final task-specific parts of the model [Du et al.(2021)] or using an attention mechanism to explain and adjust trained models [Mehrabi et al.(2021)].

We generalize existing in-processing techniques into an "adjuster model" framework that is applied as a post-processing step, utilizing adversarial debiasing to achieve high-performance. The adversarial debiasing method applies to gradient-based optimization approaches (for instance, backpropagation or tree boosting [Grari et al.(2019a)]). At each iteration of training our model, a secondary adversary model tries to predict some undesirable part of our model. For example, if we want group parity in our model, then we would set up our adversary to predict group membership as a function of our model prediction. Making this adversary model worse by changing our prediction in effect improves the fairness criteria (group parity) that we have selected. Given a primary model with weights W and a primary loss function L_P, adversarial debiasing updates W according to

$$\nabla_W L_P \ - \ \nabla_W L_A,$$

where L_A is the adversary's loss and $\alpha \in \mathbb{R}$ a tunable parameter controlling the strength of fairness. The sign of the adversary term is negative because the main model aims to maximize the adversary's loss. In practice, once the main model's gradient is computed, the adversary is also updated via $\nabla_{W_A} L_A$. Intuitively, a higher adversary loss means it becomes more difficult to infer the protected attribute from the main predictions.

Although we focus on adversarial debiasing, our fairness adjuster framework is also compatible with any approach where the final loss can be written as $L_P + c\,L_A$. An example is the prejudice remover approach in [Kamishima et al.(2012)], which adds a regularizer to remove prejudice in the classification process. Our post-processing perspective can accommodate such in-processing constraints while letting the baseline model remain a black-box, potentially trained with automated techniques like FLAML [Wang et al.(2021)].

2.2 Main Contribution

This paper proposes a general bias mitigation framework that adapts techniques from the in-processing class of methods, but applied as a post-processing step, which we call the **fairness adjuster**. This combines the advantages of in- and post-processing methods. Our framework allows us to tune a model's predictions for fairness without changing its training data or learning algorithm, while maintaining performance near state-of-the-art in-processing methods.

We demonstrate theoretical bounds on the performance difference between applying a technique as a post-processing step rather than an in-processing step. In practice, we find that the our fairness adjuster is able to match the performance of in-processing methods almost exactly. Perhaps the most surprising result we find is that we are able to decompose regular in-processing methods into two steps: the first involving only the performance objective, and the second involving only the predictions from the non-debiased model as well as the fairness constraint.

3 Method

For data $\{(x_i, y_i)\}_{i=1}^n$ with $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$ or $y_i \in \{0, 1\}$, some accuracy loss function $L(X, y) \to \mathbb{R}$ and some fairness loss function $L_a(X, Y) \to \mathbb{R}$ define three objectives.

A model h trained with traditional in-processing methods optimizes the following objective jointly for accuracy and fairness. As noted above, our method applies to a general class of in-processing methods, but from now on we focus on adversarial debiasing:

Joint Optimization

$$L_{AD} = L\big(h(X),\, Y\big) + \lambda L_a\big(h(X)\big)$$

This optimization yields a single model that is jointly trained for accuracy and fairness. In contrast, our proposed fairness adjuster methodology separates the optimization for accuracy and fairness into two distinct steps.

1. **Baseline Optimization** First, we train the baseline model f focusing solely on performance:

$$L_B = L\big(f(X),\, Y\big)$$

2. **Fairness Adjustment** In the second step, we fix the baseline predictions $\hat{Y}_i = f^*(X)$ from the optimization of L_B in the first step. Defining the adjuster model as g, we minimize:

$$L_O = L\big(\hat{Y} + g(X),\, \hat{Y}\big) + \lambda\, L_a\big(\hat{Y} + g(X)\big)$$

Notice here that we do not use the true label Y in the loss function in the adjustment step! This is an important distinction, as it allows us to separately train the adjuster without access to the baseline model's training labels. In the following section, we bound the difference in performance between the joint training approach and adjuster approach, and show empirically that both approaches are able to achieve a similar fairness-accuracy tradeoff.

4 Theoretical Properties

In this section, we derive some properties of the adjuster in comparison with the joint optimization approach. We focus our attention on two common supervised learning tasks [1]:

1. **Regression**, where $L(u, y) = \sum_{i=1}^{n} (u_i - y_i)^2$ is the mean squared error (MSE) loss.

2. **Binary Classification**, where $L(u, y) = -\sum_{i=1}^{n} \left[y_i \ln(\sigma(u_i)) + (1 - y_i) \ln(1 - \sigma(u_i)) \right]$ is the binary cross entropy (BCE) loss.

4.1 Regression

The results in this section characterize the change in performance and fairness of the fairness adjuster model. First, we characterize the differnece in performance between the fairness adjuster and the baseline model. Next, we state results comparing the solutions of the adversarial debiasing and fairness adjuster approaches. Finally, we connect the change in fairness to the size of the adjustment.

Proposition 4.1 (Performance loss of adjustment from optimal model)**.** *Let $L(\hat{y}, y) = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2$ denote the Mean Squared Error (MSE) loss. For any perturbation vector $g \in \mathbb{R}^n$, the change in the MSE loss when g is added to the prediction $\hat{y}$ given by*

$$\Delta_{MSE}(f, g) = L(\hat{y} + g, y) - L(\hat{y}, y),$$

can be expressed as

$$\Delta_{MSE}(f, f + g) = \sum_{i=1}^{n} g_i^2 + 2 \sum_{i=1}^{n} (\hat{y}_i - y_i) g_i.$$

Alternatively, it can be decomposed into:

$$\Delta_{MSE}(f, f + g) = L(\hat{y} + g, \hat{y}) + 2 \sum_{i=1}^{n} (\hat{y}_i - y_i) g_i,$$

where $L(\hat{y} + g, \hat{y}) = \sum_{i=1}^{n} g_i^2$ represents the MSE between $\hat{y}$ and $\hat{y} + g$.

Proof.

$$\Delta_{\text{MSE}}(f, f + g) = L(\hat{y} + g, y) - L(\hat{y}, y)$$
$$= \sum_{i=1}^{n} \left[(\hat{y}_i + g_i - y_i)^2 - (\hat{y}_i - y_i)^2 \right]$$
$$= \sum_{i=1}^{n} g_i^2 + 2 \sum_{i=1}^{n} (\hat{y}_i - y_i) g_i$$

□

---

[1]The method also works with other loss functions as well, for example with survival analysis.

If both g_i and $(\hat{y}_i - y_i)$ are uncorrelated and symmetric around zero then the extra term will be centered around zero [2]. We can more conservatively bound with a Cauchy-Scharwz as follows:

$$\Delta_{\text{MSE}} \le \sum_{i=1}^{n} g_i^2 \; + \; 2\sum_{i=1}^{n} |\hat{y}_i - y_i|\,|g_i|$$

If $\|g\|$ is small or if $\hat{y}_i \approx y_i$, then Δ_{MSE} remains small.

Proposition 4.2. *For optimal solutions g^* optimizing L_O and h^* optimizing L_{AD}, assume the following conditions hold:*

1. *Equal fairness: $L_a\big(h^*(X)\big) = L_a\big(\hat{Y} + g^*(X)\big)$*

2. *Sufficient model complexity: $\hat{Y} + g^*(X)$ can equal $f(X)$ and $h(X)$ with appropriate parameter choices.*

3. *Convergence: All models converge to the global minima within their model class.*

Then, the change in the objective Δ_{MSE} can be bounded by:

$$\Delta_{MSE} = L(\hat{y} + g^*(X)) - L(h^*(X))$$
$$\le \sum_{i=1}^{n} 2(\hat{y}_i - y_i)(g_i^* - (h_i^* - \hat{y}_i))$$
$$= 2(\hat{y} - y)^T(g^* - (h^* - \hat{y}^*))$$

Proof. Using the assumptions that $f^* + g^*$ minimizes L_O, and $h^* - f^*$ is in the same model class,

$$L_O(f^* + g^*) = \sum_{i=1}^{n} g^{*2}_i + \lambda L_a\big(f^* + g^*\big)$$
$$\le \sum_{i=1}^{n} (h_i^* - f_i^*)^2 + \lambda L_a\big(h^*\big)$$
$$= \sum_{i=1}^{n} (h_i^* - f_i^*)^2 + \lambda L_a\big(f^* + g^*\big),$$

where the final equality follows from the equal fairness assumption. Hence, $\sum_{i=1}^{n} g^{*2}_i \le \sum_{i=1}^{n} (h_i^* -$

---

[2]Naturally, MSE is also minimized when $g_i = (\hat{y}_i - y_i)$.

$f_i^*)^2$ and

$$\Delta_{\text{MSE}} = L(\hat{y} + g^*(x)) - L(h^*(x))$$

$$= \sum_{i=1}^{n} \left[\left((\hat{y}_i - y_i) + g_i^* \right)^2 - \left((h_i^* - \hat{y}_i) - (y_i - \hat{y}_i) \right)^2 \right]$$

$$= \sum_{i=1}^{n} 2(\hat{y}_i - y_i)g_i^* + g^{*2}_i - (h_i^* - \hat{y}_i)^2 - 2(h_i^* - \hat{y}_i)(y_i - \hat{y}_i)$$

$$\leq \sum_{i=1}^{n} 2(\hat{y}_i - y_i)g_i^* + (h_i^* - f_i^*)^2 - (h_i^* - \hat{y}_i)^2 - 2(h_i^* - \hat{y}_i)(y_i - \hat{y}_i)$$

$$= \sum_{i=1}^{n} 2(\hat{y}_i - y_i)g_i^* - 2(h_i^* - \hat{y}_i)(y_i - \hat{y}_i)$$

$$= \sum_{i=1}^{n} 2(\hat{y}_i - y_i)(g_i^* - (h_i^* - \hat{y}_i))$$

$\square$

We've bounded the loss in accuracy of the adjuster relative to the adversarial debiasing estimator in terms of the cross product between baseline model's residual and the difference in adjustments to the baseline model's predictions, where we reinterpret $h^* - \hat{y}^*$ as an adjustment to the baseline predictions $\hat{y}^*$, similar to g^*. Intuitively we would expect the adversarial debiasing model to cause slightly less inaccuracies holding fairness gains the same, because it has fairness in its objective function.

Interpreting this bound, there are three mechanisms that can cause the two methods to have a similar training loss.

1. The baseline model is very accurate with $\hat{y} \approx y$.

2. The fairness adjuster and adversarial debiasing predictions are close: $g^* \approx h^* - f^*$.

3. The model residuals and difference in adjustments to the base model are orthogonal.

In our results, we observe the third case where the residuals and difference and adjustments are near-orthogonal, suggesting that both the adjuster and adversarial debiasing have a similar fairness-accuracy tradeoff in practice.

Note that this result does not say anything about the individual predictions being close, nor that the two models have the same performance on the test set. However, in the case of linear regression, we can show a stronger result, where the solutions to both problems do coincide exactly.

Proposition 4.3 (Equivalence in Linear Regression Case). *If all the models (baseline, adversarial debiasing, adjuster) are fit using Ordinary Least Squares Linear Regression, the fairness loss is convex and X is full rank, then the adjuster model coincides with the adversarial debiasing model.*

Proof. For the baseline model, we have:

$$\beta_f = \min_{\beta} L_B$$

$$= \min_{\beta} \sum_{i=1}^{n} (x_i^\top \beta - y_i)^2$$

$$= \min_{\beta} (X^T \beta - Y)^T (X^T \beta - Y) \tag{1}$$

Which has a matrix solution:

$$\beta_f = (X^\top X)^{-1} X^\top y \tag{2}$$

Next, consider the adversarial debiasing estimator that satisfies:

$$\beta_a = \min_{\beta} L_{AD} = \min_{\beta} \Big\{ \sum_{i=1}^{n} (x_i^\top \beta - y_i)^2 \; + \; \lambda L_a (X^\top \beta_a) \Big\},$$

By convexity:

$$\frac{\partial L_{AD}}{\partial \beta} \Big|_{\beta = \beta_a} = 2 X^\top (X \beta_a - Y) \; + \; \lambda X^\top \Delta L_a (X \beta_a) = 0$$

$$\Rightarrow 2 X^\top (X \beta_a - Y) = -\lambda X^\top \nabla L_a (X \beta_a) \tag{3}$$

For the *adjuster*, fix β_f and define

$$L_O = \sum_{i=1}^{n} (\hat{y}_i + x_i^\top \beta - \hat{y}_i)^2 \; + \; \lambda L_a (\hat{Y} + X^\top \beta)$$

Since $\hat{y}_i = x_i^\top \beta_f$, the first term is $\sum_i (x_i^\top \beta_g)^2$. Thus:

$$\frac{\partial L_O}{\partial \beta} \Big|_{\beta = \beta_g} = 2 X^\top X \beta_g \; + \; \lambda X^\top \nabla L_a (X (\beta_f + \beta_g))$$

Consider $\beta_g = \beta_a - \beta_f$. Then

$$\hat{y} + \beta_g X = \beta_f X + (\beta_a - \beta_f) X = \beta_a X$$

$$\Rightarrow \frac{\partial L_O}{\partial \beta} \Big|_{\beta = \beta_g} = 2 X^\top X (\beta_a - \beta_f) \; + \; \lambda X^\top \nabla L_a ((X \beta_a))$$

$$= 2 X^\top X \beta_a - 2 X^\top X \beta_f \; + \; \lambda X^\top \nabla L_a ((X \beta_a))$$

$$= 2 X^\top X \beta_a - 2 X^\top X (X^T X)^{-1} X^T Y \; + \; \lambda X^\top \nabla L_a ((X \beta_a)) \qquad \text{(from (2))}$$

$$= 2 X^\top X \beta_a - 2 X^T Y \; + \; \lambda X^\top \nabla L_a ((X \beta_a))$$

$$= 2 X^\top X \beta_a - 2 X^T Y \; - \; (2 X^\top X \beta_a - 2 X^T Y) \qquad \text{(from (3))}$$

$$= 0$$

By convexity, the adjuster solution that adjusts the baseline to the adversary model minimizes the adjuster loss. Note that essential to this proof is that the derivatives f(x) and g(x) are not functions of their parameter β and so evaluate to the same value. $\qquad \square$

Finally, we provide a bound on the size of the adjustment in terms of the increase in fairenss as measured by L_a.

Proposition 4.4 (Accuracy-Fairness Trade-Off)**.**

$$Adjuster\ Objective : \|g\|^2 + L_a(\hat{y} + g)$$
$$At\ g = 0 : \|0\|^2 + L_a(\hat{y}) = L_a(\hat{y})$$
$$At\ g \neq 0 : \|g\|^2 + L_a(\hat{y} + g)$$
$$If\ g^* \neq 0 : \|g^*\|^2 + L_a(\hat{y} + g^*) \ \leq \ L_a(\hat{y})$$
$$\Rightarrow \|g^*\|^2 \ \leq \ L_a(\hat{y}) - L_a(\hat{y} + g^*)$$

Thus, the nonzero adjuster causes a decrease in L_a at least $\|g\|^2$.

This is helpful when our fairness loss is somewhat interpretable. Consider a fairness penalty that attempts to minimize the difference in overprediction error between two partitions of our data G_1 and G_2

Define the *overprediction* in group k by the average $\frac{1}{|G_k|} \sum_{i \in G_k} (f_i - y_i)$.

$$L_a(f) = \frac{1}{|G_1|} \sum_{i \in G_1} (f_i - y_i) - \frac{1}{|G_2|} \sum_{j \in G_2} (f_j - y_j).$$

$$L_a(f + g) = \frac{1}{|G_1|} \sum_{i \in G_1} (f_i + g_i - y_i) - \frac{1}{|G_2|} \sum_{j \in G_2} (f_j + g_j - y_j).$$

Due to shared terms, f and y cancel out in the difference:

$$L_a(f + g) - L_a(f) = \left[\frac{1}{|G_1|} \sum_{i \in G_1} g_i \right] - \left[\frac{1}{|G_2|} \sum_{j \in G_2} g_j \right] \geq \|g\|^2$$

Therefore, the *adjustment* g shifts the overprediction difference at least by the average squared offset, which can be a useful property in model selection.

4.2   Binary Classification

In the classification setting, we are able to characterize the difference in model performance between adversarial debiasing and the fairness adjuster analogously to the regression case. In the classification setting, the three objectives are:

- **Baseline:**
$$L_B(f) \ = \ -\sum_{i=1}^{n} \Big[y_i \ \ln(\sigma(f(x_i))) \ + \ (1 - y_i) \ \ln(1 - \sigma(f(x_i))) \Big].$$

  Here, $f(x)$ is the baseline logit; $\sigma(z) = 1/(1 + e^{-z})$.

- **Adversarial Debiasing:**
$$L_{AD}(h) \ = \ -\sum_{i=1}^{n} \Big[y_i \ \ln(\sigma(h(x_i))) + (1 - y_i) \ \ln(1 - \sigma(h(x_i))) \Big] \ + \ \lambda L_a(h(X)).$$

  Logit $h(x)$ now aims to reduce both cross-entropy and a fairness penalty $L_a(\cdot)$.

8

- **Adjuster:** Let $\hat{y}_i = \sigma\big(f(x_i)\big)$. Then

$$
\begin{aligned}
L_O\big(f+g\big) \;=\; &-\sum_{i=1}^{n}\Big[\hat{y}_i\,\ln\!\big(\sigma(f(x_i)+g(x_i))\big) \;+\; \big(1-\hat{y}_i\big)\,\ln\!\big(1-\sigma(f(x_i)+g(x_i))\big)\Big] \\
&+\; \lambda L_a\big(f(X)+g(X)\big).
\end{aligned}
$$

Here, $\hat{y}_i$ is treated as a pseudo-label, and $g(x)$ is an adjuster to the baseline logit $f(x)$.

Proposition 4.5 (Bounding Cross-Entropy for Adjuster vs. Adversarial Debiasing). *In the setting above, make the same assumptions as 4.2. Then, the following bound on cross-entropy with respect to the true labels Y holds:*

$$
\mathrm{BCE}(f+g,Y) \leq \mathrm{BCE}(h,Y) + (\sigma(f)-Y)^\top\big[g-(h-f)\big].
$$

Proof. We first recall a convenient identity for the cross-entropy expansion. For any vector $u = (u_1,\ldots,u_n)$ and a label vector $\hat{y} = (\hat{y}_1,\ldots,\hat{y}_n)$, we write:

$$
\mathrm{BCE}(u,\hat{y}) = \mathrm{BCE}(u,y) - \sum_{i=1}^{n}\big[\sigma(f_i)-y_i\big]\,\log\frac{\sigma(u_i)}{1-\sigma(u_i)}.
$$

Since $\log\!\big(\frac{\sigma(u_i)}{1-\sigma(u_i)}\big) = u_i$, this becomes

$$
\mathrm{BCE}(u,\hat{y}) = \mathrm{BCE}(u,y) - \sum_{i=1}^{n}\big[\sigma(f_i)-y_i\big]\,u_i.
$$

By assumption of optimality of the adjuster solution $f+g$, we have

$$
\mathrm{BCE}(f+g,\hat{y}) + \lambda L_a(f+g) \leq \mathrm{BCE}(h,\hat{y}) + \lambda L_a(h).
$$

Given $L_a(f+g) = L_a(h)$, this implies

$$
\mathrm{BCE}(f+g,\hat{y}) \leq \mathrm{BCE}(h,\hat{y}).
$$

We next apply the cross-entropy identity to both $\mathrm{BCE}(f+g,\hat{y})$ and $\mathrm{BCE}(h,\hat{y})$:

$$
\mathrm{BCE}(f+g,\hat{y}) = \mathrm{BCE}(f+g,y) - \sum_{i=1}^{n}\big[\sigma(f_i)-y_i\big]\big(f_i+g_i\big),
$$

$$
\mathrm{BCE}(h,\hat{y}) = \mathrm{BCE}(h,y) - \sum_{i=1}^{n}\big[\sigma(f_i)-y_i\big]\,h_i.
$$

Since $\mathrm{BCE}(f+g,\hat{y}) \leq \mathrm{BCE}(h,\hat{y})$, we rearrange to get

$$
\mathrm{BCE}(f+g,y) - \sum_{i=1}^{n}\big[\sigma(f_i)-y_i\big]\big(f_i+g_i\big) \leq \mathrm{BCE}(h,y) - \sum_{i=1}^{n}\big[\sigma(f_i)-y_i\big]\,h_i.
$$

9

Equivalently,

$$\mathrm{BCE}(f + g, y) \leq \mathrm{BCE}(h, y) + \sum_{i=1}^{n} \big[\sigma(f_i) - y_i\big]\big(g_i - (h_i - f_i)\big).$$

This can be written in vector form as

$$\mathrm{BCE}(f + g, y) \leq \mathrm{BCE}(h, y) + (\sigma(f) - y)^{\top}\big[g - (h - f)\big].$$

$\square$

Notice that this mirrors the MSE bound [3]:

$$L_B(f + g) \ \leq \ L_B(h) \ + \ 2(f - y)^{\top}\big[g - (h - f)\big].$$

5   Experiments

In this section, we present experimental evaluation of the proposed methodology on three datasets: the Adult [Dua and Graff(2017)], COMPAS [Angwin et al.(2016)], German credit [Dua and Graff(2017)]. We implement the method using a combination of XGBoost as the classifier and a simple logistic regression as the adversary [4]. For each dataset, we ran a Bayesian hyperparameter search [5] using the FLAML package [Wang et al.(2021)] and then computed fairness metrics using the IBM's AIF360 package [Bellamy et al.(2018)]. We first describe the experimental setup and subsequently provide the experimental results.

5.1   Experimental Setup

We evaluated the **fairness adjuster** using the following datasets:

Adult This dataset contains features, demographic information, and a binary target encoding if income is greater than \$50k for 48k individuals. We treated sex as the protected attribute.

German This dataset contains features, demographic information, and a binary target encoding good or bad credit risk for 1000 individuals. We treat age (≥ 26 years) as the protected attribute.

COMPAS This dataset contains features, demographic information, and a binary target encoding if an individual re-offended within 2 years. We treated race as the protected attribute.

---

[3]Interestingly, we can write both cases as $L(f + g) \leq L(h) + (\nabla_f L)^T(g - (h - f))$. It's unclear whether this is a coincidence or indicative of deeper structure.

[4]Gradient boosted decision trees, and XGBoost in particular, have been shown to achieve more favorable performance than Neural Networks on tabular data, especially on skewed data [McElfresh et al.(2024)]. On the other hand, NNs are more flexible than gradient boosted tree methods, in which a tree is fixed once it is generated, since they are able to update all parameters throughout the training process. Since the input feature of the NN is a changing output, a GBT method would not be well-suited as the adversary.

[5]Our implementation of adversarial debiasing relies on training the XGBoost model for more than 10 iterations. In cases where the optimal hyperparameters were fewer than 10 iterations, we multiplied the optimal number of iterations by a constant n and scaled the learning rate by $1/n$ to compensate. All three models (baseline, adversarial debiasing and adjuster) were then trained with the same hyperparameters.

For each dataset we fit a baseline learner to establish the fairness of a model trained solely to optimize classification accuracy. Each dataset is a classification task, so the baseline model minimizes the BCE (binary cross-entropy) loss. We executed the same preprocessing procedure as implemented in the AIF360 package on the raw feature values. In order to provide robust evaluation metrics we used 50 random seeds to execute a 5-fold cross validation routine. Within each fold and seed combination, we fit following algorithms:

Baseline XGBoost binary classifier, optimized without any fairness constraint.

Adversarial Debiasing XGBoost binary classifier, optimized with the fairness loss constraint in the objective function. This provided the main comparison between the proposed methodology and an existing, in-processing alternative algorithm to achieve demographic parity as shown in [Grari et al.(2019a)].

Fairness Adjuster We implemented the methodology proposed in this paper using the baseline XGBoost classifier in combination with an XGBoost regressor issuing the adjustments to the classification predictions.

Within each seed and fold combination, we evaluated each of the above algorithms both Classifications Accuracy and Disparate Impact defined as follows:

Classification Accuracy The proportion of correct classifications issued by the model $\frac{\sum_{i=1}^{n} \hat{y} = y}{n}$.

Disparate Impact defined as the proportion of favorable outcomes predicted for the protected group divided by the proportion of favorable outcomes predicted for the non-protected group. For each of the datasets, we can clearly define the favorable outcome as

Adult High income

German Good credit risk

COMPAS Low recidivism risk

Finally, through a linear search, we found the corresponding weights for the adversary and adjuster algorithms which approximately achieved equalized odds [Hardt et al.(2016)] for the protected and non-protected groups.

5.2 Results

For all data sets tested, we found that the **fairness adjuster** achieved nearly equivalent accuracy to adversarial debiasing when set to achieve equalized odds (i.e. Disparate Impact = 1).

To compare between methodologies, we calculated the mean Classification Accuracy and Disparate Impact (called "Fairness" in the table) scores across all folds and constructed a 95% confidence interval about the mean.

Finally, we calculated the differences between the adversarial debiasing and the fairness adjuster loss functions $\Delta\text{Loss} = \frac{1}{n}(\sigma(f) - y)^T(g - (h - f))$[6] derived above. We find these differences to be close to 0 for all datasets, and consequently the adversarial debiasing and adjuster models have similar performance.

---

[6]We add a factor of $\frac{1}{n}$ to control for the number of observations in each dataset.

Dataset	German	Adult	Compas
Baseline Accuracy	0.7055±0.0011	0.8036±0.0001	0.6629±0.0005
AD Accuracy	0.6980±0.0004	0.7802±0.0007	0.6588±0.0005
Adjuster Accuracy	0.6977±0.0004	0.7795±0.0006	0.6584±0.0005
Baseline Fairness	0.8215±0.0098	0.0000±0.0000	0.6999±0.0032
AD Fairness	1.0027±0.0009	1.0337±0.0232	1.0004±0.0110
Adjuster Fairness	1.0040±0.0006	1.0466±0.0246	1.0098±0.0100
Δ Loss	0.00071	4.5045×10^{-5}	-0.00090

(a) German dataset

(b) Adult dataset
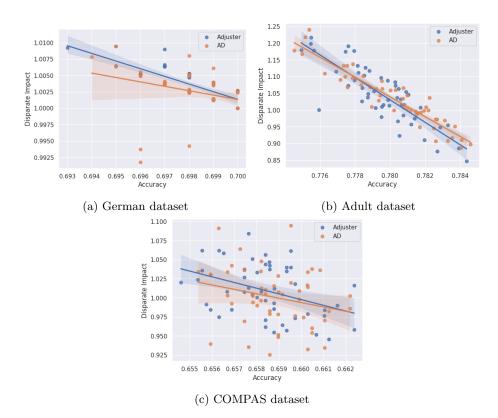
(c) COMPAS dataset

Figure 1: 50-seed, 5-fold CV results for three datasets: German, Adult, and COMPAS. The figures show the fairness-accuracy tradeoff for Adversarial Debiasing and the fairness adjuster resulting from the random CV splitting. Each point represents the average of the metrics over the 5 folds for either method. We fit a regression line with confidence bounds to better visualize the results.

6 Discussion

The fairness adjuster method has the following advantages:

Decoupling From Model Implementation Our method eliminates the need to modify the loss function of the underlying model. In contrast, competing approaches often require manual implementation of custom loss functions, which is both technically challenging and highly context-specific.

Flexibility in Training Data Unlike traditional methods that require both demographic data and training labels on the training data to optimize fairness, our approach demonstrates that training labels for the dataset used to tune fairness are not necessary. This flexibility allows for the base model to be trained on one dataset and subsequently fine-tuned for fairness using a different dataset, even one for which demographic data is available, but targets lacking.

Compatibility with Complex or Proprietary Models The framework supports fairness tuning of complex or proprietary base models by treating them as black-box systems. This enables the adjustment of more complex base models for which implementing in-processing methods would be difficult or impossible, e.g. an ensemble of multiple models. Furthermore, our method allows separate entities to independently optimize models for accuracy and fairness, potentially reducing conflicts of interest and facilitating post-hoc fairness adjustments without altering the original model.

Enhanced Interpretability The fairness adjuster provides greater transparency into the effects of fairness adjustments at the individual level. Unlike combined models that output a single prediction, the adjustment-based approach produces both the original prediction and the adjustment applied to that prediction. This separation enables more granular interpretability and control over the final model, such as identifying fairness-relevant features or imposing constraints on the adjustment.

7 Future Work

While our current framework focuses on a single fairness constraint based on adversarial debiasing, other directions warrant further exploration:

1. **Adapting to Additional Fairness Metrics.** We have primarily demonstrated our method with adversarial loss functions or group-based overprediction metrics. Future work could incorporate alternative definitions such as Equalized Odds, Calibration, or other constraints that may require specialized loss surrogates. A challenge is ensuring that the pseudo-label choice still admits an offset solution with minimal accuracy degradation.

2. **Extending to Complex Model Architectures.** Our offset approach can be applied to models with structured outputs (e.g. sequences) or to more intricate neural network structures. Investigating how to design the "adjuster" component for architectures like large language models or deep generative models could open new avenues of post-hoc fairness tuning.

3. **Robustness and Generalization Guarantees.** While we outlined theoretical results comparing in-processing vs. offset solutions, more rigorous bounds on generalization error under fairness constraints remain an open question. For instance, combining Rademacher complexity arguments with fairness constraints might yield improved theoretical guarantees for large-scale offset solutions.

4. **Handling Multiple Protected Attributes.** Much of the literature, including our main demonstration, focuses on one sensitive attribute at a time (e.g. race, gender). Future work could investigate how to incorporate additional attributes or intersectional fairness constraints, ensuring that our offset framework remains tractable.

8   Conclusion

We introduced a *post-processing* framework, the **fairness adjuster**, that adapts in-processing techniques for use on top of an existing model's predictions. Our theoretical analysis showed that, under certain assumptions, this offset approach can reproduce a similar solution as a directly debiased model, at a minimal accuracy loss for a comparable fairness improvement. Empirical results on real datasets (Adult, German, and COMPAS) demonstrated that our fairness adjuster achieves a near-identical fairness-accuracy tradeoff to the in-processing adversarial debiasing method, while offering important practical benefits: the elimination of custom model-training constraints, separation of the baseline model's optimization from fairness constraints, and interpretability of the offset itself. By treating models as black boxes, it also provides flexibility in datasets used for fairness tuning and accommodates proprietary systems. We hope that this methodology fosters a broader adoption of fairness improvements with minimal impact on existing machine learning pipelines.

9   References

References

[Agarwal et al.(2018)] A. Agarwal, A. Beygelzimer, M. Dudik, J. Langford, and H. Wallach. 2018. A Reductions Approach to Fair Classification. (2018). arXiv:1803.02453 [cs.LG] https://arxiv.org/abs/1803.02453

[Angwin et al.(2016)] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. https://github.com/propublica/compas-analysis

[Bellamy et al.(2018)] Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. 2018. AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. (Oct. 2018). https://arxiv.org/abs/1810.01943

[Calders et al.(2009)] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. 2009. Building Classifiers with Independency Constraints. In *2009 IEEE International Conference on Data Mining Workshops*. 13–18. https://doi.org/10.1109/ICDMW.2009.83

[Celis and Keswani(2019)] L. Elisa Celis and Vijay Keswani. 2019. Improved Adversarial Learning for Fair Classification. (2019). arXiv:1901.10443 [cs.LG] https://arxiv.org/abs/1901.10443

[Donini et al.(2018)] M. Donini, L. Oneto, S. Ben-David, J. Shawe-Taylor, and M. Pontil. 2018. Empirical Risk Minimization under Fairness Constraints. (2018). arXiv:1802.08626 [cs.LG] https://arxiv.org/abs/1802.08626

[Du et al.(2021)] Mengnan Du, Subhabrata Mukherjee, Guanchu Wang, Ruixiang Tang, Ahmed Awadallah, and Xia Hu. 2021. Fairness via representation neutralization. *Advances in Neural Information Processing Systems* 34 (2021), 12091–12103.

[Dua and Graff(2017)] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[Dwork et al.(2011)] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Rich Zemel. 2011. Fairness Through Awareness. (2011). arXiv:1104.3913 [cs.CC] https://arxiv.org/abs/1104.3913

[Grari et al.(2019a)] Vincent Grari, Sylvain Lamprier Boris Ruf, and Marcin Detyniecki. 2019a. Fair Adversarial Gradient Tree Boosting. (2019). https://arxiv.org/pdf/1911.05369

[Grari et al.(2019b)] Vincent Grari, Boris Ruf, Sylvain Lamprier, and Marcin Detyniecki. 2019b. Fair Adversarial Gradient Tree Boosting. (2019). arXiv:1911.05369 [cs.LG] https://arxiv.org/abs/1911.05369

[Hajian and Domingo-Ferrer(2012)] Sara Hajian and Josep Domingo-Ferrer. 2012. A methodology for direct and indirect discrimination prevention in data mining. *IEEE transactions on knowledge and data engineering* 25, 7 (2012), 1445–1459.

[Hardt et al.(2016)] Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems 29*. Curran Associates Inc, Red Hook, NY, United States, 3323–3331. https://arxiv.org/abs/1610.02413

[Iosifidis and Ntoutsi(2018)] Vasileios Iosifidis and Eirini Ntoutsi. 2018. Dealing with bias via data augmentation in supervised learning scenarios. *Jo Bates Paul D. Clough Robert Jäschke* 24, 11 (2018).

[Jung et al.(2020)] Christopher Jung, Michael Kearns, Seth Neel, Aaron Roth, Logan Stapleton, and Zhiwei Steven Wu. 2020. An Algorithmic Framework for Fairness Elicitation. (2020). arXiv:1905.10660 [cs.LG] https://arxiv.org/abs/1905.10660

[Kamiran et al.(2012)] Faisal Kamiran, Asim Karim, and Xiangliang Zhang. 2012. Decision Theory for Discrimination-Aware Classification. In *2012 IEEE 12th International Conference on Data Mining*. 924–929. https://doi.org/10.1109/ICDM.2012.45

[Kamishima et al.(2012)] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-Aware Classifier with Prejudice Remover Regularizer. In *Machine Learning and Knowledge Discovery in Databases*. Springer Berlin Heidelberg, Berlin, Heidelberg, 35–50.

[Kim et al.(2018)] Michael Kim, Omer Reingold, and Guy Rothblum. 2018. Fairness through computationally-bounded awareness. *Advances in neural information processing systems* 31 (2018).

[Lohia et al.(2018)] Pranay K. Lohia, Karthikeyan Natesan Ramamurthy, Manish Bhide, Diptikalyan Saha, Kush R. Varshney, and Ruchir Puri. 2018. Bias Mitigation Post-processing for Individual and Group Fairness. (2018). arXiv:1812.06135 [cs.LG] https://arxiv.org/abs/1812.06135

[McElfresh et al.(2024)] Duncan McElfresh, Sujay Khandagale, Jonathan Valverde, Vishak Prasad C, Benjamin Feuer, Chinmay Hegde, Ganesh Ramakrishnan, Micah Goldblum, and Colin White. 2024. When Do Neural Nets Outperform Boosted Trees on Tabular Data? (2024). arXiv:2305.02997 [cs.LG] https://arxiv.org/abs/2305.02997

[Mehrabi et al.(2021)] Ninareh Mehrabi, Umang Gupta, Fred Morstatter, Greg Ver Steeg, and Aram Galstyan. 2021. Attributing Fair Decisions with Attention Interventions. (2021). arXiv:2109.03952 [cs.AI] https://arxiv.org/abs/2109.03952

[Mehrabi et al.(2022)] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2022. A Survey on Bias and Fairness in Machine Learning. (2022). arXiv:1908.09635 [cs.LG] https://arxiv.org/abs/1908.09635

[Menon and Williamson(2018)] Aditya Krishna Menon and Robert C. Williamson. 2018. The cost of fairness in binary classification. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency (Proceedings of Machine Learning Research, Vol. 81)*, Sorelle A. Friedler and Christo Wilson (Eds.). PMLR, 107–118. https://proceedings.mlr.press/v81/menon18a.html

[Wadsworth et al.(2018)] Christina Wadsworth, Francesca Vera, and Chris Piech. 2018. Achieving Fairness through Adversarial Learning: an Application to Recidivism Prediction. (2018). arXiv:1807.00199 [cs.LG] https://arxiv.org/abs/1807.00199

[Wang et al.(2021)] Chi Wang, Qingyun Wu, Markus Weimer, and Eric Zhu. 2021. FLAML: A Fast and Lightweight AutoML Library. In *Proceedings of the 4th MLSys Conference*. Microsoft Research, Redmond WA USA.

[Wang et al.(2019)] Hao Wang, Berk Ustun, and Flavio P. Calmon. 2019. Repairing without Retraining: Avoiding Disparate Impact with Counterfactual Distributions. (2019). arXiv:1901.10501 [cs.LG] https://arxiv.org/abs/1901.10501

[Zafar et al.(2017)] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P. Gummadi. 2017. Fairness Constraints: Mechanisms for Fair Classification. (2017). arXiv:1507.05259 [stat.ML] https://arxiv.org/abs/1507.05259

[Zhang et al.(2018)] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. 2018. Mitigating Unwanted Biases with Adversarial Learning. (2018). https://arxiv.org/abs/1801.07593