PRAGUE UNIVERSITY
OF ECONOMICS
AND BUSINESS

Acta Informatica Pragensia
2024, Volume 13, Issue 3, pp. 359–373
https://doi.org/10.18267/j.aip.241

Article Open Access

The Fairness Stitch: A Novel Approach for Neural Network Debiasing

Modar Sulaiman, Kallol Roy

Institute of Computer Science, University of Tartu, Tartu, Estonia

Corresponding author: Kallol Roy (kallol.roy@ut.ee)

Abstract

The pursuit of fairness in machine learning models has become increasingly crucial across various applications, including bank loan approval and face detection. Despite the widespread use of artificial intelligence algorithms, concerns persist regarding biases and discrimination within these models. This study introduces a novel approach, termed "The Fairness Stitch" (TFS), aimed at enhancing fairness in deep learning models by combining model stitching and training jointly, while incorporating fairness constraints. We evaluate the effectiveness of TFS through a comprehensive assessment using two established datasets, CelebA and UTKFace. The evaluation involves a systematic comparison with the existing baseline method, fair deep feature reweighting (FDR). Our analysis demonstrates that TFS achieves a better balance between fairness and performance compared to the baseline method (FDR). Specifically, our method shows significant improvements in mitigating biases while maintaining performance levels. These results underscore the promising potential of TFS in addressing bias-related challenges and promoting equitable outcomes in machine learning models. This research challenges conventional wisdom regarding the efficacy of the last layer in deep learning models for debiasing purposes. The findings suggest that integrating fairness constraints into our proposed framework (TFS) can lead to more effective mitigation of biases and contribute to fairer AI systems.

Keywords

Artificial intelligence; AI bias; Deep learning; Fairness in machine learning; Finetune; Model stitching; Overfitting.

Citation: Sulaiman, M., & Roy, K. (2024). The Fairness Stitch: A Novel Approach for Neural Network Debiasing. *Acta Informatica Pragensia*, 13(3), 359–373. https://doi.org/10.18267/j.aip.241

Academic Editor: Zdenek Smutny, Prague University of Economics and Business, Czech Republic

1 Introduction

The widespread adoption of machine learning algorithms has transformed numerous sectors, ranging from healthcare and finance to education and criminal justice. Machine learning algorithms have demonstrated exceptional capabilities, enabling data-driven decision making at an unprecedented scale. However, the increased reliance on automated systems has raised concerns about fairness, as biases and discrimination can be inadvertently embedded in algorithmic processes, exacerbating societal inequalities and perpetuating systemic biases. The field of fairness in machine learning is a burgeoning area of research focused on preventing biases in data and model inaccuracies from resulting in unfavourable treatment of individuals based on characteristics such as race, gender, disabilities and sexual or political orientation. It is important to address fairness and ethics in machine learning because the outcome can be detrimental to users and the community when machine learning is not fair. For example, algorithms on social media sites may have sparked political tensions due to skewed or siloed news feeds (and fake news), when the intention was to deliver personalized recommendations for users.

To address the pressing need for fairness in machine learning, various techniques have been developed to mitigate unfairness for machine learning models at different stages of model development. These techniques are broadly categorized into pre-processing, in-processing and post-processing methods (Wan et al., 2023). Pre-processing and post-processing methods offer straightforward strategies to address unfairness. Pre-processing methods focus on adjusting the training data distribution to balance sensitive groups, while post-processing methods calibrate prediction results after model training. In contrast to the aforementioned techniques, in-processing debiasing methods have gained significant traction within the research community. These methods are noteworthy for their direct integration of fairness considerations into the model design process, resulting in the creation of intrinsically fair models (Dwork et al., 2012; Edwards & Storkey, 2015; Hashimoto et al., 2018; Kearns et al., 2018). By doing so, they address fairness issues at a fundamental level within machine learning models, promoting fairness as an integral aspect of model architecture. This approach not only reflects the growing importance of fairness but also represents a substantial step towards the development of equitable and ethically sound machine learning systems. In-processing methods offer advantages such as directly considering fairness in model optimization, enabling the converged model to achieve fairness even with biased input data (Caton & Haas, 2024). Additionally, in-processing methods can effectively fine-tune representations from pre-trained models to mitigate bias without requiring extensive re-training efforts. Based on the stage at which fairness is achieved in the model, in-processing debiasing techniques can be categorized into explicit and implicit methods (Wan et al., 2023). Explicit methods directly incorporate fairness constraints in training objectives, while implicit methods focus on refining latent representation learning. However, in our paper, we introduce a novel framework called "The Fairness Stitch" which combines the principles of model stitching and training with fairness constraint as an explicit in-processing debiasing method. This gives a comprehensive strategy to enhance fairness in deep learning models, providing a practical solution to mitigate biases and promote accurate outcomes at the same time.

1.1 Contribution

The major contributions of the paper are as follows:

- Proposing an innovative framework of "The Fairness Stitch" to better mitigate bias while preserving model performance.
- Empirically proving the limitation of last-layer fine-tuning to attain an optimal balance between fairness and performance.

The rest of the paper is structured as follows: Section 2 presents a literature review, Section 3 explains the mathematical notations and definitions used in the paper. Section 4 formally defines and elaborates on

"The Fairness Stitch" framework. Sections 5 and 6 delineate the datasets and design of experiments. Finally, in Section 7, we present the outcomes and findings achieved by the implementation of our framework.

2 Literature Review

Debiasing techniques are mainly categorized as (i) in-preprocessing, (ii) pre-processing and (iii) post-processing. In-processing debiasing techniques in machine learning aimed at mitigating disparity were studied by Wan et al. (2023). They proposed the use of adding a regularizer to reduce the correlation between sensitive attributes and prediction outcomes. Contrary studies (Cherepanova et al., 2021) have highlighted that in-processing techniques are less effective for over-parameterized large neural networks, as these models can easily overfit fairness objectives during training, especially when the training data are imbalanced. This fairness overfitting issue raised by Cherepanova et al. (2021) poses an open challenge. Over-parameterization of neural networks leads to highly flexible decision boundaries and attempting to meet fairness criteria for one attribute can negatively affect fairness with respect to another sensitive attribute. However, over-parameterization has been essential for achieving high prediction accuracy, particularly in neural networks designed for challenging tasks.

To address the problem of fairness criteria overfitting identified by Cherepanova et al. (2021), a novel framework was proposed by Mao et al. (2023) called last-layer fairness fine-tuning. A similar line of research is conducted by Kamishima et al. (2011) by introducing a prejudice index (PI) as a regularizer, which quantifies the level of dependence between a sensitive variable and a target variable. Jiang et al. (2020), on the other hand, used the information geometry metric of Wasserstein-1 distances between classifier outputs and sensitive information as a regularization in the optimization process. In a more practical setting, Beutel et al. (2019) proposed a new metric of conditional equality while implementing equality of opportunity. In the application of recommender systems, Beutel et al. (2019) proposed pairwise comparisons from randomized experiments as a tractable way to measure fairness.

A novel measure of decision boundary (un)fairness was proposed by Zafar et al. (2019) and Zafar et al. (2017). They used covariance between the sensitive attributes and the (signed) distance between the subjects' feature vectors and the decision boundary as a metric for the classifier. Mao et al. (2023) took a different route and avoided adding fairness constraint as a regularization for deep neural networks model. This is because large deep neural models have the tendency to overfit fairness criteria. Instead, they used pre-training and fine-tuned the last layer for fair training of their model. Their proposed method involves training a model using empirical risk minimization and subsequently fine-tuning only the last layer with various fairness constraints as an in-processing debiasing technique. In extensive experiments on benchmark image datasets with different fairness notions, the authors demonstrated the efficacy of last-layer fine-tuning in enhancing fairness. Other important works along this direction are Kirichenko et al. (2023) and Lee et al. (2022). The authors showed that only fine-tuning the last layer(s) while keeping the other layers frozen during the gradient descent achieves better performance. Last-layer fine-tuning of a pre-trained neural network on a smaller, more specific dataset achieved debiasing.

However, a recent counterclaim study by Kumar et al. (2022) observed that fine-tuning can achieve worse accuracy than linear probing out of distribution (OOD). This is particularly in cases where the pretrained features are of high quality and the distribution shift is substantial. The authors conducted experiments across ten distinct distribution shift datasets, revealing that while fine-tuning tends to achieve, on average, a 2% higher accuracy in distribution (ID), it results in a 7% lower accuracy when dealing with out-of-distribution (OOD) data, compared to the performance of linear probing. To address this disparity, they proposed a two-step strategy known as LP-FT, which begins with linear probing and is followed by full fine-tuning. This approach capitalizes on the strengths of both fine-tuning and linear probing and consistently leads to superior performance. In their empirical evaluations, Kumar et al. (2022)

demonstrated that LP-FT outperforms both fine-tuning and linear probing across various datasets, achieving a 1% higher accuracy in distribution and an impressive 10% higher accuracy out of distribution.

In this paper, we propose an innovative method (model stitching) for achieving fairness explained in Section 4. Our proposed method draws inspiration from the works of model stitching by Lenc & Vedaldi (2018) and Bansal et al. (2021). Model stitching is a tool to study the internal representations of deep neural models. Model stitching combines the bottom layers of one pre-trained and frozen model (referred to as Model A) with the top layers of another model (referred to as Model B) using a trainable layer positioned in between, resulting in the creation of what is termed a "stitched model". Our work makes use of this model stitching to explore the commonalities and disparities in the learned representations of various models and training strategies as a strategy to mitigate bias.

3 Preliminaries

In this section, we lay the groundwork by introducing a set of notations and fairness definitions, which serve as fundamental prerequisites for comprehending and interpreting the content presented in this research. Additionally, we offer a concise overview of pertinent information – theoretical concepts and model stitching – that is relevant for understanding the intuition behind our framework.

3.1 Notations

We formalize our classification setting as a triplet dataset $T = \{(x_i, a_i, y_i)\}_{i=1}^{N}$, where x_i is the feature drawn from a distribution over X; $a_i \in A = \{0,1\}$ is the sensitive attribute (race, gender, etc.); and $y_i \in Y = \{0,1\}$ is the output label. We define H as the hypothesis class of predictors (U) mapping from the input X to the output space Y.

3.1.1 Group fairness

Group fairness aims to ensure that different groups within a population are treated fairly, without discrimination, in the decision-making process. It involves assessing whether the outcomes or decisions produced by the algorithm are consistent and equitable across various predefined groups, typically distinguished by sensitive attributes (e.g., gender, race, age). Mathematically, group fairness can be defined using various fairness metrics or statistical tests, depending on the specific context and objectives of a given study (Narayanan, 2018). In our paper, our primary emphasis is on group fairness. We provide a succinct overview of three fairness definitions that are relevant to this paper. They also serve as fairness constraints during the fine-tuning/training of a neural network model. Moreover, we have adopted the same implementation as elucidated by Mao et al. (2023) for fairness definitions.

3.1.2 Equalized odds

Equal opportunity (EO) is a fairness criterion applied to classification tasks, ensuring equitable true positive and false positive rates between different groups. It is defined under the conditional probability distributions associated with distinct groups identified by sensitive attributes $a_i \in [0,1]$.

A classifier U satisfies equalized odds if the following condition holds (Hardt et al., 2016):

$$P(\hat{y}_i = 1 \mid a_i = 0, y_i = y) = P(\hat{y}_i = 1 \mid a_i = 1, y_i = y) \qquad (1)$$

where $y \in [0,1]$, $\hat{y}_i$ is the predicted label for the i-th sample, a_i is the sensitive attribute value for the i-th sample and y_i is the target label for the i-th sample.

This ensures elimination of disparities in different groups that are affected by both false negative (FNR) and false positive (FPR) rates in predictive models.

To operationalize the pursuit of equalized odds, practitioners often employ a training strategy that involves minimizing specific objectives as (Cherepanova et al., 2021; Padala & Gujar, 2020):

$$\min_{U}\{\hat{L}(U_w) + \rho(FPR + FNR)\} \tag{2}$$

where ρ is a predetermined weight, $\hat{L}_w$ signifies the cross-entropy loss and FPR and FNR are derived as:

$$FPR = \left|\frac{\sum_i p_i\,(1-y_i)a_i}{\sum_i a_i} - \frac{\sum_i p_i\,(1-y_i)(1-a_i)}{\sum_i(1-a_i)}\right| \tag{3}$$

$$FNR = \left|\frac{\sum_i(1-p_i)\,y_i a_i}{\sum_i a_i} - \frac{\sum_i(1-p_i)\,y_i(1-a_i)}{\sum_i(1-a_i)}\right| \tag{4}$$

In this scenario, p_i signifies the softmax output of model U in a binary prediction task, considering the sensitive attribute $a_i \in A$ and the true label $y_i \in Y$ corresponding to each feature vector $x_i \in X$ in the dataset.

3.1.3 Accuracy equality

Accuracy equality (AE) evaluates whether the subjects in protected and unprotected groups experience similar false positive rate (FPR) and false negative rate (FNR). The objective of AE is to ensure that misclassification rates are approximately equal across these sensitive groups, such as different demographic categories, to prevent discriminatory or biased outcomes in predictive models (Zafar et al., 2017). AE is defined as:

$$P(\hat{y}_i \neq y \mid a_i = 0) = P(\hat{y}_i \neq y \mid a_i = 1) \tag{5}$$

To enforce accuracy equality in practice, one way implemented in training is to minimize the following objective:

$$\min_{U}\left[\hat{L}(U_w) + \rho \mid \hat{L}^{-a}(U_w) - \hat{L}^{+a}(U_w) \mid\right] \tag{6}$$

for a given predefined weight ρ, where $\hat{L}^{+a}(U_w)$ is the cross-entropy loss of samples with $a = 1$ and $\hat{L}^{-a}(U_w)$ is the cross-entropy loss of samples with $a = 0$.

3.1.4 Max-min fairness

Max-min fairness (MMF) is a fairness principle that prioritizes and maximizes the performance of the least advantaged or worse-off group within a given context (Rawls, 2001). MMF optimizes the performance for the group with the lowest utility while still meeting overall performance goals, used in fairness-aware machine learning (Lahoti et al., 2020; Cherepanova et al., 2021). Max-min fairness is defined as:

$$arg\max_{\hat{y}\in Y} min\ _{a\in A}P(\hat{y} = y \mid a) \tag{7}$$

Max-min fairness is satisfied by minimizing the following objective (Cherepanova et al., 2021):

$$min\ _{w}max\left\{\,\hat{L}^{(y+,a+)}(U_w), \hat{L}^{(y+,a-)}(U_w), \hat{L}^{(y-,a+)}(U_w), \hat{L}^{(y-,a-)}(U_w)\right\} \tag{8}$$

where $\hat{L}^{(y\prime,a\prime)}(U_w)$denotes the cross-entropy loss on the training samples where $y = y'$ and $a = a'$.

3.1.5 Model stitching

The main idea of stitching was introduced by Lenc & Vedaldi (2018) for studying representations by learning their equivariant and equivalence properties. In general, model stitching (MS) is a method of combining the bottom layers of a pre-trained model (A) with the top layers of another pre-trained model (B). This combination of top layers with bottom layers is done using a trainable layer in between and the resulting model is termed a "stitched model" (Bansal et al., 2021). Particularly, for a neural network A, with an architecture A and $r: X \to R^d$ be a representation, the loss function L is defined by (Bansal et al., 2021):

$$L_l(r; A) = \ L(A_{>l} \circ s \circ r) \tag{9}$$

where S is the family of stitching layers, $A_{>l}$ mapping function from activations of the l-th layer of A to the final output, l is the index of layers of A and $\circ$ is function composition. $L_l(r;A)$ is the minimum loss of stiching r into all layers of A except the initial l layers, employing a stitching layer chosen from the set S.

In our proposed framework, we delve into a more specific instance of the previous broader concept of model stitching. We focus on a scenario where we possess only one pre-trained model and need to incorporate a trainable layer within it. Specifically, as depicted in Figure 1, if we have a pre-trained model M, we regard the bottom layers of another pre-trained model (A) as the bottom layers of M and similarly, the top layers of yet another pre-trained model (B) serve as the top layers of M. We may call this approach self-stitching, as it involves stitching together layers within the same pre-trained model. Generally, model stitching tells us the path (homotopy) through the stitching layer s of transforming a candidate representation r into first layers of A, in an appropiate subspace.

4 Proposed Model

In this paper, we introduce "The Fairness Stitch" (TFS), a specialized layer that aims to ensure equal opportunities for different groups during inference (based on sensitive attributes). "The Fairness Stitch" (TFS) transforms a candidate representation (a.k.a. biased representation) into an unbiased representation. The unbiased representation comes from multiple sources, including unbiased pre-trained layers. Our TFS acts as a path (homotopy) for transforming an unbiased representation to a biased representation. Thus, TFS is used as a path to debias. By integrating TFS, the machine learning model aims to achieve a more equitable representation of features, classes and data points, ultimately enhancing overall fairness and minimizing disparities in the learning process. We test our de-biasing method by our proposed TFS, through comprehensive experiments on popular image datasets. We use different pre-trained architectures with different fairness notions, to demonstrate the efficacy of our framework in enhancing fairness. In our paper, we challenge the conventional notion of achieving fairness through last-layer fine-tuning (Mao et al., 2023). We instead show that freezing the last layer is necessary and sufficient to strike a better balance between fairness and performance. Our proposed method, TFS, combines model stitching with fairness constraints as an in-processing debiasing method (transformation from an unfair representation to a fair representation). Experiments validate that TFS achieves jointly fairness and accuracy.

4.1 TFS: Trainable stitching layer for fairness

The deep learning architecture is formalized as a composition of two distinct blocks: (i) the last layer, and (ii) preceding layers, as shown in Figure 1. TFS is a two-step process. In the first phase, we train the model without the stitching layer denoted as M. In the second phase, we add a trainable stitching layer between the two frozen layer blocks. In our paper, we denote M_i as the i-th layer of the pre-trained model. The cost of adding a stitching layer is given by:

$$L^*(E;z;r) = \inf_{z\in Z}(L(E \circ z \circ r) + fairness\ constraint) \qquad (10)$$

where $E = \{M_i\}_{i=0}^{n-1}$ is the preceding layers, $r = \{M_n\}$ is the last layer of pre-trained model M and z is the stitching layer.

The stitching layer z is initialized with random weights and is trained by minimizing Equation (7). The added fairness constraints are explained in Section 3.1. In our paper, the family of stitching layers Z is sampled from the set of linear layers. The weights of the stitched model are frozen, except the stitching layer z during the training. In summary, we train the stitching layer, which incorporates fairness constraints on the (class and sensitive attribute) balanced dataset.

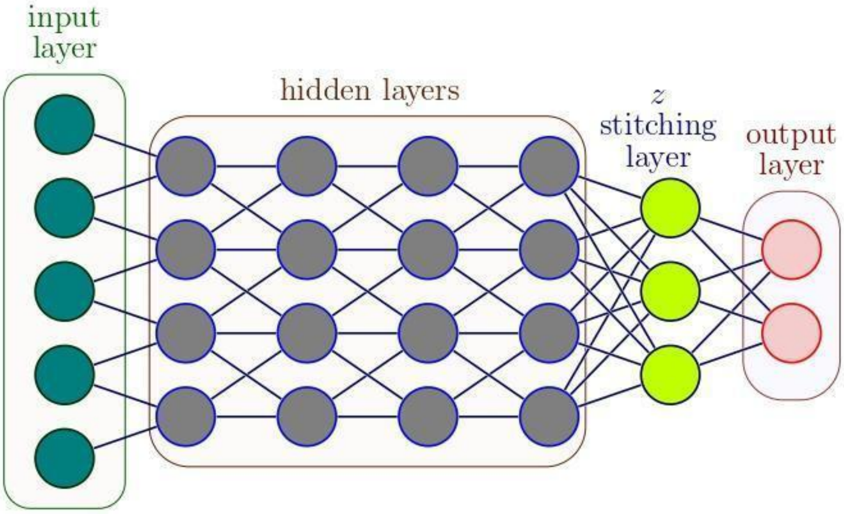


***Figure 1.** Illustration of TFS framework for our stitched model. Our pre-trained model M includes only the input, hidden and output layers, excluding the stitching layer z.*

5 Data Sources and Characteristics

The open-source CelebA and UTKFace datasets are used in the paper to design our experiments. Below, we provide a summary of the dataset characteristics, size and relevant attributes.

5.1 CelebA dataset

The CelebA dataset is a collection of celebrity faces, comprising more than 200,000 images annotated with 40 distinct attributes, including facial landmarks, gender, age, hair colour, glasses, etc. (Liu et al., 2015). For our experiments, we use hair colour (blond or non-blond) as the target label (y), while gender (male or non-male) serves as the sensitive attribute (a). We follow the dataset division methodology outlined in Liu et al. (2015) and Mao et al. (2023) and create Table 1. To create a balanced sub-dataset, we perform sampling from the initial training and validation subsets based on the image count in the minority subgroup. Precisely, we select 1,569 images for each (y, a) grouping, culminating in a total of 6,276 images within the balanced dataset. Furthermore the (male, blond hair) subgroup accounts for just 1% of the total image count, signifying the minority category within the dataset.

***Table 1.** (train / val / test) Overview of CelebA dataset.*

	Blond hair	**Non-blond hair**	**Total**
Male	1,387 / 182 / 180	66,874 / 8,276 / 7,535	68,261 / 8,458 / 7,715
Female	22,880 / 2,874 / 2,480	71,629 / 8,535 / 9,767	94,509 / 11,409 / 12,247
Total	24,267 / 3,056 / 2,660	138,503 / 16,811 / 17,302	162,770 / 19,867 / 19,962

5.2 UTKFace dataset

The UTKFace dataset is a publicly accessible face dataset (Zhang et al., 2017; Mao et al., 2023) that covers an extensive age range from newborns to individuals aged up to 116 years old. In our experiments, we use a random selection process to extract two subsets from the training dataset. Each subset accounts for 20% of the original data, to maintain proportionality. One of these subsets served as the validation dataset, while the other as the test dataset, as explained in Table 2. In our experiment, age is used as the target variable, while gender is the sensitive attribute. Age is categorized into two groups: "young" (≤ 35) and "old" (> 35) (Park et al., 2020). We follow a similar procedure (as before) of constructing a balanced sub-dataset by sampling an equivalent number of images from both the original training and validation datasets for each distinct (y, a) group (Mao et al., 2023). Each balanced group consists of 2,477 images, thereby yielding of 9,908 images in total and the minority category comprises females aged over 35 years.

Table 2. *(train / val / test) Overview of UTKFace dataset.*

	Young (≤ 35)	**Old (> 35)**	**Total**
Male	4,133 / 1,378 / 1,378	3,301 / 1,101 / 1,100	7,434 / 2,479 / 2,478
Female	4,931 / 1,643 / 1,644	1,858 / 619 / 619	6,789 / 2,262 / 2,263
Total	9,064 / 3,021 / 3,022	5,159 / 1,720 / 1,719	14,223 / 4,741 / 4,741

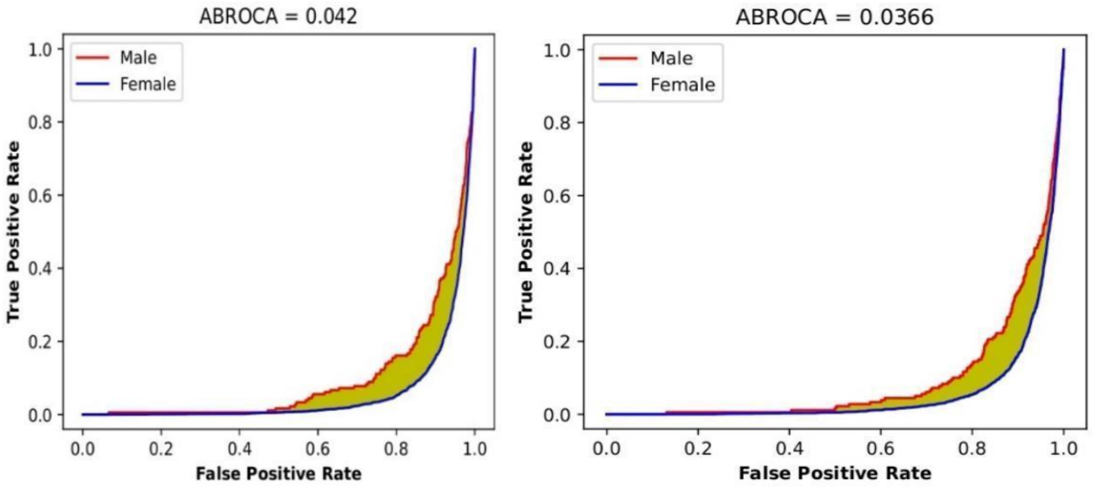
6 Experimental Setup

In this section, we elucidate the experimental setup to assess the efficacy of the TFS framework in promoting fairness. First, as part of our systematic assessment of the efficacy of our approach (TFS), we introduce the specifics of the baseline method (FDR). Both approaches fall under the category of two-step training-based debiasing methods as per the taxonomy of Parraga et al. (2023). Subsequently, we provide detailed insights into the architectural specifications of the deep learning model used for both FDR and TFS. Lastly, we quantitatively evaluate the trade-off between performance and fairness with an array of performance metrics.

6.1 Baseline method

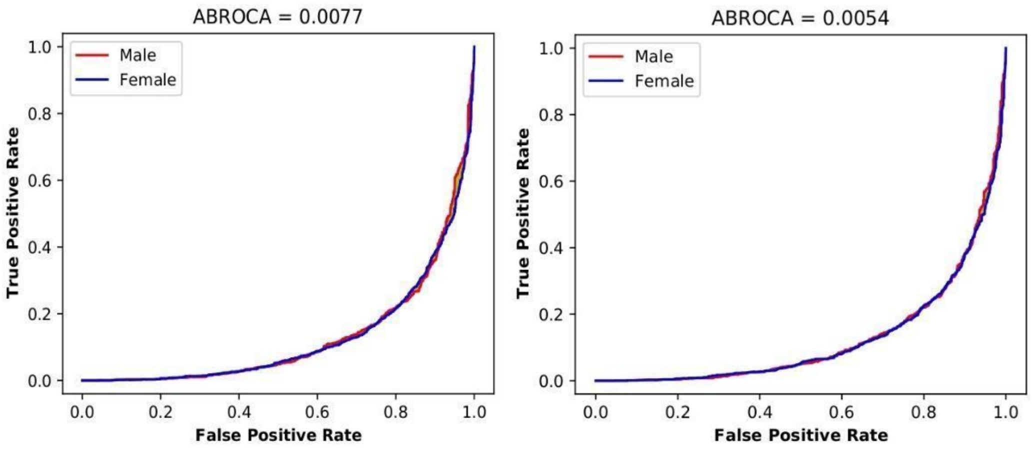
This section explains the FDR baseline method for our experimental evaluations, which acts as the point of reference for comparisons with our proposed TFS framework, see Figures 2 and 3. The FDR method, introduced by Mao et al. (2023), uses empirical risk minimization (ERM), fine-tuning the last layer and a balanced dataset concerning both class and sensitive attributes.

The balanced dataset is created by strategic sampling from the training and validation datasets. Furthermore, FDR augments its training process by incorporating fairness constraints into the objective function during fine-tuning. For a comprehensive understanding of the FDR method, we recommend referring to the original paper (Mao et al., 2023).



(a) FDR (b) TFS

Figure 2. *ABROCA results on CelebA. Figures (a) and (b) showcase ABROCA outcomes for both FDR and our TFS framework. Specifically, they showcase the use of equalized odds as the fairness constraint with ρ = 20 during fine-tuning (for FDR) and training (for TFS), respectively.*



(a) FDR (b) TFS

Figure 3. *ABROCA results on UTKFace. Figures (a) and (b) showcase ABROCA outcomes for both FDR and our TFS framework. Specifically, they showcase the use of equalized odds as the fairness constraint with $\rho = 2$ during fine-tuning (for FDR) and training (for TFS), respectively.*

6.2 Model architecture

In this paper, we train two distinct ResNet-18 models (Cherepanova et al., 2021): one on the CelebA dataset and the other on the UTKFace dataset to utilize them within the FDR and TFS frameworks. The ResNet-18 architecture consists of six blocks, including an input layer, and two convolutional layers within each of the four basic blocks. The final block corresponds to the last layer.

To ensure a harmonious alignment between the ResNet-18 model architecture and the conceptual framework of TFS illustrated in Figure 1, we map the ResNet-18 input layer to the input layer within the conceptual TFS framework. Similarly, we map the four basic blocks in the ResNet-18 model to the hidden layers in the conceptual TFS framework. Furthermore, the final layer of the ResNet-18 model is matched with the last layer depicted in the conceptual TFS architecture presented in Figure 1. In addition, we introduce a stitching layer denoted as z between the hidden layers and the last layer of the ResNet-18 model, as visually demonstrated in Figure 1. This stitching layer, z, is a fully connected layer designed with an input dimension matching that of the output from the hidden layers. Thus, it produces an output dimension that matches the input dimension of the final layer. Both final models in TFS and FDR frameworks utilize stochastic gradient descent (SGD) with the same specific hyperparameters, including a momentum of 0.9 and a weight decay of 5e-4.

6.3 Performance and fairness metrics

In our experiments, we use a set of performance and fairness metrics to evaluate the effectiveness of our proposed framework (TFS). These metrics are instrumental in quantifying both the performance and fairness aspects of the models in both FDR and TFS.

- Balanced accuracy (BACC) (Brodersen et al., 2010): BACC measures the ability of a model to correctly classify instances across different classes, considering the imbalance between classes. It provides a balanced view of the model performance.
- Area under the ROC curve (AUC) (Fawcett, 2004): AUC is used to evaluate the ability of a model to discriminate between positive and negative classes. A higher AUC indicates a better-performing model with stronger discrimination ability.

- Equalized odds difference (EO_Diff) (Hardt et al., 2016): EO_Diff quantifies disparities in the true positive rates (sensitivity or recall) and false positive rates (fallout or false alarm rate) between different demographic or protected groups. A smaller value of EO_Diff indicates better fairness.
- Accuracy equality difference (Mao et al., 2023): AE_Diff measures the difference in misclassification rates between the two groups. A smaller value of AE_Diff indicates better fairness, as it signifies that the predictive accuracy of the model is more balanced between the groups, reducing the likelihood of one group being disadvantaged in terms of predictive accuracy compared to the other.
- Worst accuracy (WA) (Mao et al., 2023): WA assesses the minimum accuracy among various group combinations, with a larger value indicating better fairness.
- Balanced accuracy and fairness (AF): The AF metric was introduced by Mao et al. (2023) and combines both balanced accuracy (BACC) and a fairness metric to holistically assess a machine learning model. A larger AF value indicates better fairness. It is calculated as AF = BACC – EO_Diff (for equalized odds), AF = BACC – AE_Diff (for accuracy equality) and AF = BACC + WA (for max-min fairness).
- Absolute between-ROC area (ABROCA) (Gardner et al., 2019): ABROCA is based on the receiver operating characteristics (ROC) and quantifies the discrepancy between the baseline and comparison group curves across all potential thresholds. It accomplishes this by summing up this discrepancy while disregarding which subgroup's model performs better at specific thresholds, as it considers the absolute values of these differences. ABROCA typically falls within the range from 0 to 1, but in practical scenarios, it often resides within the interval [0.5, 1]. This is because an AUC (area under the curve) value of 0.5 can be attained through random guessing.

7 Results

This section explains and provides the comparative evaluation results of TFS and FDR. We fine-tune our models for 1000 epochs using a (class and sensitive attribute) balanced dataset for the evaluation. Our objective is to assess the potential of both FDR and TFS in achieving an optimal trade-off between fairness and performance.

We initially choose the top-performing fine-tuned and trained models from FDR and TFS, respectively, determined by their performance across 1000 epochs on the validation dataset. Subsequently, we present our findings by testing the best fine-tuned TFS and FDR models on the test dataset. Our results affirm that TFS outperforms FDR in achieving a superior balance between fairness and performance.

7.1 Results without applying debiasing technique

Our research findings, as shown in Tables 3 and 4, reveal a distinct disparity in bias levels between the CelebA and UTKFace datasets before using any debiasing techniques. Specifically, the CelebA dataset exhibits significantly higher bias ratios, with EO_Diff = 0.586, AE = 0.213 and MMF = 0.183. In contrast, the UTKFace dataset demonstrates lower bias ratios, with EO_Diff = 0.143, AE = 0.023 and MMF = 0.675. This difference can be attributed to the varying distribution of samples among different sensitive attributes, particularly in terms of gender. The CelebA dataset contains a higher disparity in the numbers of male and female samples. Additionally, it is essential to emphasize that the CelebA dataset is considerably larger in size compared to the UTKFace dataset, emphasizing the need for dataset-specific adjustments to hyperparameters. For instance, in our subsequent experiments with FDR and TFS on the CelebA dataset, we set ρ to 20 for the EO and AE constraints. Contrary, in fine-tuning with the UTKFace dataset, we opted for a value of $\rho = 2$ for the EO and AE constraints. The differences highlight the importance of tailored hyperparameter selection based on dataset characteristics.

7.2 Results with applying TFS

In this section, we present the results obtained by applying the TFS framework to a ResNet-18 model for a binary classification task within the CelebA and UTKFace datasets. Our research findings shed light on the performance and effectiveness of TFS in promoting fairness while maintaining accurate predictions. We summarize the outcomes of applying the three distinct fairness criteria within the TFS framework to the CelebA and UTKFace datasets in Tables 3 and 4 as follows.

Table 3. *Results of our TFS approach with different fairness notions on CelebA dataset. For AUC, BACC, WA and AF, a larger value is considered better, while for EO_Diff and AE a smaller value is considered better.*

Without applying any fairness constraint										
	BACC			**AUC**				**EO_Diff**	**AE_Diff**	**WA**
	Train	Balanced	Test	Train	Balanced	Test		Test	Test	Test
ResNet-18 Model	0.903	0.792	0.853	0.987	0.966	0.971		0.586	0.213	0.183
Fairness notion 1: equalized odds										
	BACC			**AUC**			**EO_Diff**			**AF**
	Train	Balanced	Test	Train	Balanced	Test	Train	Balanced	Test	Test
FDR	0.898	0.896	0.876	0.959	0.956	0.942	0.014	0.015	0.110	0.766
TFS	0.887	0.884	0.874	0.953	0.947	0.940	0.032	0.026	0.081	0.793
Fairness notion 2: AE										
	BACC			**AUC**			**AE_Diff**			**AF**
	Train	Balanced	Test	Train	Balanced	Test	Train	Balanced	Test	Test
FDR	0.906	0.905	0.883	0.967	0.964	0.949	0.009	0.001	0.008	0.875
TFS	0.900	0.886	0.881	0.960	0.951	0.945	0.0002	0.011	0.0005	0.880
Fairness notion 3: MMF										
	BACC			**AUC**			**WA**			**AF**
	Train	Balanced	Test	Train	Balanced	Test	Train	Balanced	Test	Test
FDR	0.915	0.913	0.875	0.978	0.972	0.96	0.864	0.880	0.800	1.675
TFS	0.916	0.906	0.877	0.979	0.968	0.96	0.865	0.867	0.811	1.688

Our experimental findings highlight the superiority of the TFS framework over the FDR method when evaluating their performance and fairness under the equalized odds fairness constraint.

TFS consistently achieves significantly lower EO_Diff fairness values for both the CelebA and UTKFace datasets. Specifically, on the CelebA test dataset, FDR records an EO_Diff value of 0.110, whereas the TFS framework notably reduces this to 0.081. Similarly, on the UTKFace test dataset, FDR exhibits an EO_Diff value of 0.062, while TFS further diminishes it to 0.058. Furthermore, Figures 2 and 3 illustrate the ABROCA values, providing a comprehensive comparison of the performance and fairness achieved by the FDR and TFS frameworks when equalized odds are employed as a fairness constraint. These figures unequivocally demonstrate the superior fairness of our TFS framework (ABROCA = 0.0366) over FDR (ABROCA = 0.042).

Table 4. *Results of our TFS approach with different fairness notions on UTKFace dataset. For AUC, BACC, WA and AF, a larger value is considered better, while for EO_Diff and AE a smaller value is considered better.*

Without applying any fairness constraint										
	BACC			**AUC**				**EO_Diff**	**AE_Diff**	**WA**
	Train	Balanced	Test	Train	Balanced	Test		Test	Test	Test
ResNet-18 Model	0.998	0.947	0.803	1.000	0.983	0.885		0.143	0.023	0.675
Fairness notion 1: equalized odds										
	BACC			**AUC**			**EO_Diff**			**AF**
	Train	Balanced	Test	Train	Balanced	Test	Train	Balanced	Test	Test
FDR	0.996	0.951	0.796	1.000	0.986	0.876	0.005	0.010	0.062	0.734
TFS	0.996	0.951	0.793	1.000	0.985	0.876	0.003	0.012	0.058	0.735
Fairness notion 2: AE										
	BACC			**AUC**			**AE_Diff**			**AF**
	Train	Balanced	Test	Train	Balanced	Test	Train	Balanced	Test	Test
FDR	0.998	0.948	0.798	1.000	0.985	0.884	0.0008	0.003	0.016	0.782
TFS	0.992	0.946	0.796	1.000	0.984	0.879	0.010	0.001	0.0096	0.7864
Fairness notion 3: MMF										
	BACC			**AUC**			**WA**			**AF**
	Train	Balanced	Test	Train	Balanced	Test	Train	Balanced	Test	Test
FDR	0.998	0.949	0.797	1.000	0.984	0.881	0.997	0.933	0.739	1.536
TFS	0.997	0.949	0.797	1.000	0.984	0.880	0.995	0.934	0.744	1.541

Furthermore, TFS exhibits minimal and negligible deviations on BACC and AUC metrics when equalized odds are used as a fairness constraint during training on both the CelebA and UTKFace datasets. On the CelebA test dataset, our TFS framework achieves a BACC of 0.874 and an AUC of 0.940, while the FDR method records values of 0.876 for BACC and 0.942 for AUC. These slight variations underscore the consistency and reliability of the TFS framework in delivering both fairness and accuracy.

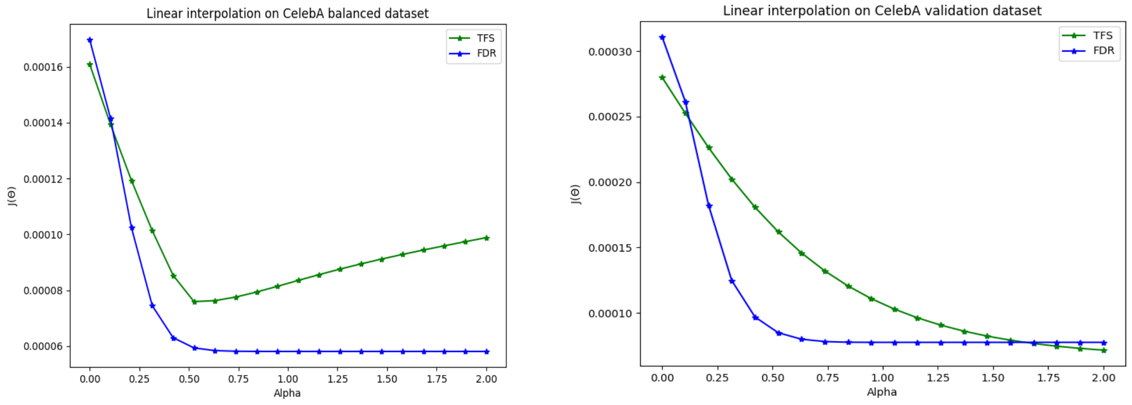
In addition, when we consider accuracy equality as a fairness constraint for both the CelebA and UTKFace datasets, TFS consistently achieves lower AE values compared to FDR. On the CelebA test dataset, TFS reduces the AE value from 0.008 (FDR) to an impressive 0.0005, emphasizing its robust capability to promote accuracy equality. Similarly, on the UTKFace dataset, FDR records an AE value of 0.016, whereas TFS further diminishes it to 0.0096. Moreover, TFS shows stability with minimal and inconsequential deviations in terms of both BACC and AUC metrics when accuracy equality is employed as a fairness constraint on both the CelebA and UTKFace datasets. Enforcing accuracy equality as a fairness constraint for the CelebA dataset, the FDR framework consistently achieves a BACC of 0.883 and an AUC of 0.949, while the TFS method yields values of 0.881 for BACC and 0.945 for AUC. These negligible variations emphasize the reliability and consistency of the TFS framework in simultaneously optimizing fairness and accuracy.

Finally, under the MMF constraint, TFS achieves higher WA values, indicating improved performance for both datasets. On the CelebA dataset, TFS achieves a WA value of 0.811, surpassing FDR’s 0.800. Similarly, on the UTKFace dataset, TFS attains a WA value of 0.744, outperforming FDR’s 0.739. Moreover, when employing MMF as a fairness constraint, TFS exhibits minimal and negligible deviations in BACC and

AUC metrics on both datasets, reinforcing its ability to strike a favourable balance between fairness and performance. Overall, these results underscore the effectiveness of our TFS framework in achieving better balance between performance and fairness, as demonstrated in Tables 3 and 4, compared to the baseline method.

8 Loss Function Visualization

To discern the distinctions between our proposed TFS framework and the baseline method FDR, we employ the one-dimensional linear interpolation approach outlined by Goodfellow et al. (2014). This technique allows us to visualize and compare the objective functions of both methods, facilitating a comprehensive analysis of their performance. To visualize the loss function using the one-dimensional linear interpolation approach, we select two parameter vectors, denoted as θ_0 and θ^*, and chart the loss function values along the line that connects these two points. This line can be parametrized by selecting a scalar parameter, denoted as alpha (α) and establishing the weighted average as $\theta(\alpha) = (1-\alpha)\theta_0 + \alpha\theta^*$. Here, θ^* represents the parameters (weights) of the trained/fine-tuned model, whether employing TFS or FDR, while θ_0 signifies the parameters (weights) of the corresponding model in its randomly initialized state, before undergoing the training or fine-tuning process. In Figure 4, we present a visual representation of the function $J(\theta)$, defined as $J(\theta) = L\big(\theta(\alpha)\big)$, evaluated on both the balanced and validation CelebA datasets, where L represents the loss function employed within the TFS and FDR frameworks. Figure 4 serves to underscore the differentiation between the TFS and FDR methods. Notably, our results indicate that FDR consistently outperforms TFS in terms of $J(\theta)$ on the balanced CelebA datasets. However, on the validation CelebA dataset, TFS consistently exhibits a better value at the final stage, signifying a more favorable trade-off between fairness and performance when compared to the FDR method. Furthermore, the outcomes depicted in Figure 4 align with the findings presented in Table 3. Our observations show that while FDR surpasses TFS on the balanced dataset, the situation is different when considering the validation and test datasets. This suggests that our TFS framework has the capacity to generalize more effectively in fairness settings, striking a better balance between fairness and performance compared to the FDR baseline method.



(a) Loss on CelebA balanced dataset (b) Loss on CelebA validation dataset

Figure 4. *Linear interpolation curves for ResNet-18 model using TFS and FDR frameworks on CelebA balanced and validation datasets.*

9 Conclusion and Future Work

In this paper, we presented an innovative method called "The Fairness Stitch" (TFS) as a debiasing technique. TFS combines model stitching and fairness constraints to mitigate the risk of bias. We tested the efficacy of our method by testing on two popular open-source datasets CelebA and UTKFace. We compared our results with the baseline method. Our research findings show that TFS beats the baseline method in fairness and accuracy. Our work presents a practical debiasing method with respect to computational complexity and sample complexity, especially in deep learning models. We hope our method will help contribute to advancing the active research area of fairness-aware machine learning. Our proposed method poses a challenge to the conventional wisdom of the effectiveness of the last layer in mitigating bias. TFS complements surgical fine-tuning (Lee et al., 2022) in the fairness context and provokes us to rethink the efficacy of the last layer.

In our future research work, we aim to investigate the computational and sample complexity of our TFS method with different non-linear stitchable layers on different pre-trained substrates. We will study the trade-off between fairness metrics and computational budgets. The other research direction in which we plan to extend our TFS is generative adversarial networks.

Additional Information and Declarations

Acknowledgments: The authors thank The High Performance Computing Center at the University of Tartu, Estonia, for providing the necessary computing resources.

Funding: This research has been funded by European Social Fund via IT Academy programme.

Conflict of Interests: The authors declare no conflict of interest.

Author Contributions: M.S.: Conceptualization, Methodology, Original draft preparation, Writing, Reviewing and Editing. K.R.: Supervision, Writing, Reviewing and Editing.

Data Availability: The data that support the findings of this study are openly available. The CelebA dataset is available at https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html, and the UTKFace dataset is available at https://susanqq.github.io/UTKFace/.

References

Bansal, Y., Nakkiran, P., & Barak, B. (2021). Revisiting Model Stitching to Compare Neural Representations. In *35th Conference on Neural Information Processing Systems (NeurIPS 2021).* NeurIPS.

Beutel, A., Chen, J., Doshi, T., Qian, H., Wei, L., Wu, Y., Heldt, L., Zhao, Z., Hong, L., Chi, E. H., & Goodrow, C. (2019). Fairness in Recommendation Ranking through Pairwise Comparisons. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (pp. 2212–2220). ACM. https://doi.org/10.1145/3292500.3330745

Beutel, A., Chen, J., Doshi, T., Qian, H., Woodruff, A., Luu, C., Kreitmann, P., Bischof, J., & Chi, E. H. (2019). Putting Fairness Principles into Practice. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, (pp. 453–459). ACM. https://doi.org/10.1145/3306618.3314234

Brodersen, K. H., Ong, C. S., Stephan, K. E., & Buhmann, J. M. (2010). The Balanced Accuracy and Its Posterior Distribution. In *2010 20th International Conference on Pattern Recognition*, (pp. 3121–3124). IEEE. https://doi.org/10.1109/ICPR.2010.764

Caton, S., & Haas, C. (2024). Fairness in Machine Learning: A Survey. *ACM Computing Surveys*, 56(7), Article 166. https://doi.org/10.1145/3616865

Cherepanova, V., Nanda, V., Goldblum, M., Dickerson, J. P., & Goldstein, T. (2021). Technical challenges for training fair neural networks. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2102.06764

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2012). Fairness Through Awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, (pp. 214–226). Springer. https://doi.org/10.1145/2090236.2090255

Edwards, H., & Storkey, A. (2015). Censoring Representations with an Adversary. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.1511.05897

Fawcett, T. (2004). ROC Graphs: Notes and Practical Considerations for Researchers. *Machine Learning*, 31(1), 1–38.

Gardner, J., Brooks, C., & Baker, R. (2019). Evaluating the Fairness of Predictive Student Models Through Slicing Analysis. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*, (pp. 225–234). ACM. https://doi.org/10.1145/3303772.3303791

Goodfellow, I. J., Vinyals, O., & Saxe, A. M. (2014). Qualitatively characterizing neural network optimization problems. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.1412.6544

Hardt, M., Price, E., & Srebro, N. (2016). Equality of Opportunity in Supervised Learning. In *30th Conference on Neural Information Processing Systems (NIPS 2016),* (pp. 1–9). NeurIPS.

Hashimoto, T., Srivastava, M., Namkoong, H., & Liang, P. (2018). Fairness Without Demographics in Repeated Loss Minimization. In *Proceedings of the 35th International Conference on Machine Learning*, (pp. 1929–1938). PMLR.

Jiang, R., Pacchiano, A., Stepleton, T., Jiang, H., & Chiappa, S. (2020). Wasserstein Fair Classification. In *Proceedings of the 35th Uncertainty in Artificial Intelligence Conference,* (pp. 862–872). PMLR.

Kamishima, T., Akaho, S., & Sakuma, J. (2011). Fairness-aware Learning through Regularization Approach. In *2011 IEEE 11th International Conference on Data Mining Workshops,* (pp. 643–650). IEEE. https://doi.org/10.1109/ICDMW.2011.83

Kearns, M., Neel, S., Roth, A., & Wu, Z.S. (2018). Preventing Fairness Gerrymandering: Auditing and Learning for Subgroup Fairness. In *Proceedings of the 35th International Conference on Machine Learning, (*pp. 2564–2572). PMLR.

Kirichenko, P., Izmailov, P., & Wilson, A.G. (2023). Last Layer Re-Training Is Sufficient for Robustness to Spurious Correlations. In *The Eleventh International Conference on Learning Representations* (1–37). ICLR. https://openreview.net/forum?id=Zb6c8A-Fghk

Kumar, A., Raghunathan, A., Jones, R., Ma, T., & Liang, P. (2022). Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2202.10054

Lahoti, P., Beutel, A., Chen, J., Lee, K., Prost, F., Thain, N., Wang, X., & Chi, E. H. (2020). Fairness without Demographics through Adversarially Reweighted Learning. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2006.13114

Lee, Y., Chen, A. S., Tajwar, F., Kumar, A., Yao, H., Liang, P., & Finn, C. (2022). Surgical Fine-Tuning improves adaptation to distribution shifts. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2210.11466

Lenc, K., & Vedaldi, A. (2018). Understanding image representations by measuring their equivariance and equivalence. *International Journal of Computer Vision*, 127(5), 456–476. https://doi.org/10.1007/s11263-018-1098-y

Liu, Z., Luo, P., Wang, X., & Tang, X. (2015). Deep Learning Face Attributes in the Wild. In *Proceedings of the IEEE International Conference on Computer Vision*, (pp. 3730–3738). IEEE. https://doi.org/10.1109/iccv.2015.425

Mao, Y., Deng, Z., Yao, H., Ye, T., Kawaguchi, K., & Zou, J. (2023). Last-Layer Fairness Fine-tuning is Simple and Effective for Neural Networks. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2304.03935

Narayanan, A. (2018). Translation Tutorial: 21 Fairness Definitions and Their Politics. https://www.youtube.com/watch?v=jIXIuYdnyyk

Padala, M., & Gujar, S. (2020). FNNC: Achieving Fairness through Neural Networks. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*. IJCAI. https://www.ijcai.org/proceedings/2020/0315.pdf

Park, S., Kim, D., Hwang, S., & Byun, H. (2020). README: REpresentation learning by fairness-Aware Disentangling MEthod. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2007.03775

Parraga, O., More, M. D., Oliveira, C. M., Gavenski, N. S., Kupssinskü, L. S., Medronha, A., Moura, L. V., Simões, G. S., & Barros, R. C. (2023). Fairness in Deep Learning: A survey on vision and language research. *ACM Computing Surveys,* (in press). https://doi.org/10.1145/3637549

Rawls, J. (2001). *Justice as Fairness: A Restatement*. Harvard University Press.

Shwartz-Ziv, R., & Tishby, N. (2017). Opening the Black Box of Deep Neural Networks via Information. *arXiv (Cornell University)*. https://doi.org/10.48550/arXiv.1703.00810

Wan, M., Zha, D., Liu, N., & Zou, N. (2023). In-Processing Modeling Techniques for Machine Learning Fairness: A survey. *ACM Transactions on Knowledge Discovery from Data*, 17(3), 1–27. https://doi.org/10.1145/3551390

Zafar, M. B., Valera, I., Rogriguez, M. G., & Gummadi, K. P. (2019). Fairness Constraints: A Flexible Approach for Fair Classification. *Journal of Machine Learning Research*, 20(1), 1–42. http://jmlr.org/papers/v20/18-262.html

Zafar, M. B., Valera, I., Rogriguez, M. G., & Gummadi, K. P. (2017). Fairness Constraints: Mechanisms for Fair Classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, (pp. 962–970). PMLR.

Zhang, Z., Song, Y., & Qi, H. (2017). Age Progression/Regression by Conditional Adversarial Autoencoder. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 5810–5818). IEEE. https://doi.org/10.1109/CVPR.2017.463

Editorial record: The article has been peer-reviewed. First submission received on 13 March 2024. Revision received on 9 June 2024. Accepted for publication on 17 June 2024. The editor in charge of coordinating the peer-review of this manuscript and approving it for publication was Zdenek Smutny.

Acta Informatica Pragensia is published by Prague University of Economics and Business, Czech Republic.

ISSN: 1805-4951