

Enforcing Fairness via Constraint Injection with FaUCI

Matteo Magnini¹, Giovanni Ciatto¹, Roberta Calegari¹ and Andrea Omicini¹

¹Dipartimento di Informatica – Scienza e Ingegneria, ALMA MATER STUDIORUM—Università di Bologna, Italy

Abstract

The problem of fairness in AI can be tackled by minimising bias in the data (pre-processing), in the algorithms (in-processing), or in the results (post-processing). In the particular case of in-processing applied to supervised machine learning, state-of-the-art solutions rely on a few well-known fairness metrics – e.g., demographic parity, disparate impact, or equalised odds – optimised during training—which, however, mostly focus on *binary* attributes and their effects on binary classification problems. Accordingly, in this work we propose FaUCI as a general purpose framework for injecting fairness constraints into neural networks (or, any model trained via stochastic gradient descent), supporting attributes of many sorts—there including binary, discrete, or continuous features. To evaluate its effectiveness and efficiency, we test FaUCI against several sorts of features and fairness metrics. Furthermore, we compare FaUCI with state-of-the-art solutions for in-processing, demonstrating its superiority.

Keywords

AI Fairness, FaUCI, in-processing, regularization, mitigation

1. Introduction

Nowadays, fairness has become a crucial issue in the development of AI systems, especially in the context of ML [1]. ML models are often trained on datasets that are biased toward certain groups of individuals, and the bias impacts the predictions of the model. This can lead to unfair outcomes, which may be damaging to the individuals from the disadvantaged groups—in terms of gender, ethnicity, age, religion, political views, etc.

One prerequisite for fairness enforcement in AI systems is *measuring* bias. In the literature, several metrics have been proposed to measure the degree of fairness of a predictive model.

Broadly speaking, these metrics score the extent to which a model treats different groups of individuals fairly and without prejudice. Technically speaking, they do so by relating sensitive attributes (e.g. gender, ethnicity, etc.) – and, in particular, their some protected values of their (e.g. female, black, etc.) – to some target variable of choice for the application at hand (e.g. loan approval, job hiring, etc.).

Once fairness metrics are defined, focus can be shifted to enforcing fairness in predictive models. Among the many techniques proposed in the literature, the so-called *in-processing* ones attempt to enforce fairness at the model level during training—as opposed to *pre-* or *post-processing* techniques, which operate before or after training, respectively.

Focussing on in-processing techniques, we acknowledge that state-of-the-art methods essentially rely on *regularisation* techniques to enforce fairness optimisation during training. In other words, these methods address ML models based on some loss-function minimization: the loss function is augmented with a fairness-related term – most commonly, derived from a fairness metric from the literature – to be minimised during training. While the approach is indeed effective, it is also limited by the fact that the fairness metrics used are mostly tailored to *binary* attributes and *binary* classification problems. This limitation may hinder the applicability of these methods to real-world datasets, where sensitive attributes may be of different sorts—e.g., categorical or continuous. Another practical issue characterising state-of-the-art methods is their reliance on several hyper-parameters, making them difficult to tune and use in practice.

AEQUITAS 2024: Workshop on Fairness and Bias in AI | co-located with ECAI 2024, Santiago de Compostela, Spain

✉ matteo.magnini@unibo.it (M. Magnini); giovanni.ciatto@unibo.it (G. Ciatto); roberta.calegari@unibo.it (R. Calegari); andrea.omicini@unibo.it (A. Omicini)

🆔 0000-0001-9990-420X (M. Magnini); 0000-0002-1841-8996 (G. Ciatto); 0000-0003-3794-2942 (R. Calegari);

0000-0002-6655-3869 (A. Omicini)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

To mitigate such issues, in this work, we propose FaUCI, a *general-purpose* method for enforcing Fairness Under Constraints Injection into any model trained via stochastic gradient descent (SGD). FaUCI advances the state-of-the-art of in-process fairness by letting users enforce the optimisation of different fairness metrics, over sensitive attributes of different sorts—namely, binary, categorical, and continuous. To let our method deal with non-binary sensitive attributes, we propose novel *variants* of the aforementioned fairness metrics, and we design FaUCI to support *pluggability* of these metrics. In other words, we let users choose the most adequate fairness metrics for their use case, while FaUCI takes care of the rest. Finally, to keep our method exploitable in practice, we design it to rely on a minimal set of hyper-parameters. In fact, apart from the fairness metric of choice, FaUCI users are only required to specify the *batch size* of the underlying training algorithm.

Despite its simplicity, it turns out that our method overcomes other similar state-of-the-art solutions in optimising fairness while retaining predictive accuracy. To verify this claim, we evaluate FaUCI on a real-world dataset and compare it with state-of-the-art methods. Experimental setup and results are presented in Section 4.

Accordingly, the paper is structured as follows: Section 2 provides background on fairness metrics and related works. Section 3 introduces variants of fairness metrics and our method, FaUCI, for enforcing fairness in ML models. The novelty of the approach is discussed. Section 4 presents results applying FaUCI to a real-world dataset, comparing with state-of-the-art methods. Section 5 concludes and discusses future work.

2. Background

In the literature, several metrics have been proposed to measure the degree of fairness of a predictive model. These metrics can be classified into two main categories: *group* vs. *individual fairness* [1]. Group fairness metrics aim at gauging the extent to which different groups are treated equally by some model (e.g., demographic parity [2], disparate impact [3], equalised odds [4]). Individual fairness metrics assess the extent to which similar individuals receive similar treatment from a given model (e.g., fairness through awareness [2] or through *unawareness* [5], and counterfactual fairness [6]). In this work, we focus on *group* fairness metrics, mainly because they are the most commonly used in the literature—*individual* fairness techniques being less common because of their computational cost.

Once fairness metrics are defined, the focus can be shifted to enforcing fairness in predictive models. There are three main approaches to this problem: *pre-*, *in-*, and *post-processing*. Pre-processing techniques aim at removing bias from the training data [7]. In-processing techniques modify the training algorithm to enforce fairness at the model level. Post-processing techniques treat the biased model as a black-box, and then reassign the predictions to enforce fairness. In this work, we focus on in-processing techniques, as we propose a new method falling into this category.

In the remainder of this section, we provide a brief overview of the most common fairness metrics from the literature, as well as most recent *in-processing* algorithms relying upon them.

2.1. Fairness Metrics

In AI, a fairness metric is a formula used to quantitatively measure the level of fairness or bias in the outcomes generated by AI algorithms. These metrics play a crucial role in evaluating whether the AI system treats various groups of individuals fairly and without prejudice. Fairness metrics encompass diverse formulæ, each specifically designed to target particular aspects of fairness.

In this study, we focus on three well-known and extensively used metrics from the literature, which guide our experiments. Additionally, in Section 3.1 we provide considerations and reformulations for each metric aimed at ensuring a consistent comparison with other state-of-the-art techniques.

Demographic Parity (DP) DP – also known as “statistical parity” – is a metric that measures whether the prediction of an ML model is independent w.r.t. a given protected attribute, i.e., values of

the sensitive feature do not affect the output of the model [2]. DP compares two different distributions: the distribution of the predictions of the model and the distribution of the conditioned – by the values of the protected attribute – predictions of the same model. For a *binary* classifier h and a *discrete* sensitive attribute A , DP can be defined as follows:

$$DP_{h,A}(X) = \sum_{a \in A} \|\mathbb{E}[h(X) | A=a] - \mathbb{E}[h(X)]\| \quad (1)$$

where X is a random variable ranging over the test data, A is the random variable ranging over the admissible values of the sensitive attributes – in turn, denoted by a –, \mathbb{E} is the expected value function, and $\|\cdot\|$ denotes the absolute value operator. A model h satisfies DP if its value is below an arbitrarily small bias threshold ϵ (a common value is 0.01).

DP can be useful in real applications like loan approval/denial. If a specific demographic group has been disproportionately denied loans in the past, DP can be used to ensure that approval rates are similar across different groups, helping to mitigate discrimination.

Disparate Impact (DI) DI measures how much a classifier *disproportionally affects* people given a sensitive attribute [3]. The minimal definition of DI for binary classification tasks and binary sensitive attributes is commonly provided as the ratio among the expected value of the model when $A=1$ and when $A=0$:

$$di_{h,A}(X) = \frac{\mathbb{E}[h(X) | A=1]}{\mathbb{E}[h(X) | A=0]}$$

However, this would make admissible values of DI unbound in value—which is undesirable for a scoring function. Hence, to guarantee that DI values are always in $[0, 1]$, the score is most commonly defined using the ratio standardisation function $\eta(x) = \min\{x, x^{-1}\}$:

$$DI_{h,A}(X) = \eta(di_{h,A}(X)) \quad (2)$$

Values of DI above 0.8 are commonly considered acceptable. In other words, values close to 0 imply high fairness violation, viceversa for values close to 1. For this reason, whenever requiring positive scores (i.e., scores where greater values are preferable), we may report the value $1 - DI$ (e.g., computing the loss function during the training of a neural network).

DI as well may be useful in real-life applications such as loans. It enables the identification of any disproportionate denial rates that may have adversely affected specific demographic groups, historically.

Equalised Odds (EO) EO is a fairness metric scoring the extent to which a classifier equally predicts a given class for all values of a sensitive attribute [4]. According to recent works [8, 9], in case of binary classification ($Y \in \{0, 1\}$), and if A is a categorical sensitive attribute, EO can be written as follows:

$$EO_{h,A}(X) = \sum_{(a,y)}^{A \times Y} eo_{h,A}(X, a, y) \quad (3)$$

where Y refers to the ground truth, the sum is aggregating all possible pairs (a, y) s.t. $a \in A$ and $y \in Y$, and eo is a shortcut defined as follows:

$$eo_{h,A}(X, a, y) = \|\mathbb{E}[h(X) | A=a, Y=y] - \mathbb{E}[h(X) | Y=y]\| \quad (4)$$

Similarly to DP, a classifier is considered fair w.r.t. a given sensitive attribute value and output value if its EO value is below an arbitrarily small bias threshold ϵ .

In real life contexts such as loan approvals, EO enables the identification of disparities in approval rates that may have disproportionately favored specific demographic groups.

2.2. In-Processing Methods for AI Fairness

Most recent state-of-the-art works for optimising fairness over neural classifiers¹ vary from one another in terms of (i) the type of sensitive attributes they support (among binary, discrete, or continuous) and for (ii) the fairness metrics they optimise [8, 9, 10, 11].

The first method – FNNC [8] – is based on Lagrangian multipliers to induce a fairness-based cost into a two-step stochastic gradient descent (SGD). Authors validate their method using all three fairness metrics (i.e., DP, DI, and EO) on real-world datasets. However, this approach comes with a strong limitation: the protected attribute must be a boolean variable so that categorical or continuous features cannot be handled.

The second work – Cho [9] – exploits kernel density estimation (KDE) methods. It approximates the computation of the cumulative distribution function of a Gaussian kernel with a Q-function [12]. This strategy enables the estimation of probabilities derived from the fairness formulæ. The approach has been validated on two metrics – DP and EO – on real-world datasets. Limitations of this work are the fine-tuning of additional hyper-parameters introduced by the KDE (e.g, bandwidth, factors in the Q-function) and the lack of tests for continuous sensitive features.

The third method – Wagner [10] – exploits the logic tensor network (LTN) framework [13]. Authors write the fairness constraints into formal logic rules that affect the training of the NN. DP and DI on non-continue protected features are used to validate the method.

Finally, the fourth method – Jiang [11] – is similar to KDE. It generalises the DP metric to take into account continuous features—see Equation (6). Authors validate their method on real-world datasets using a variant of DP metric – similar to the one we define in Section 3 –, but only on a continuous sensitive attribute. The work is limited by one single fairness metric and it is not validated on non-continuous protected features.

Other notable works for the fairness optimisation of ML models are the ones by Kamishima et al. [2012] and Berk et al. [2017]. The former proposes a regularisation method for probabilistic discriminative models, but it is tailored to the metrics the authors defined in that same work (i.e., prejudice index and underestimation index) that are not as common as other fairness metrics. The latter proposes a regularisation method for regression tasks.

3. Proposed Method

Here we introduce variants of the previously mentioned fairness metrics, extending those definitions to support categorical and continuous data. Then, we present our method for fairness enforcement, and we discuss how it differs from the related works. Finally, we provide an in-depth analysis of the functioning and the computational cost of our method.

3.1. Novel Metrics

Here we propose weighted and generalised variants of the fairness metrics presented in Section 2.1, to support categorical and continuous attributes, respectively. Broadly speaking, such variants generalise fairness metrics beyond binary data. This may help in measuring fairness in real-world datasets. For instance, these may be useful to measure / mitigate bias by taking into account the frequencies of sensitive attributes’ values. In the loan approval example, one may e.g. prioritise fairness for most underrepresented ethnicity groups—provided that more than 2 groups exist.

Weighted and Generalised Demographic Parity. For binary classification problems, $\mathbb{E}[h(X) | A=a]$ and $\mathbb{E}[h(X)]$ from Equation (1) are both in the range of $[0, 1]$, so the absolute value of their difference is still in range of $[0, 1]$. Therefore, the maximum theoretical value of DP depends on the number of possible values of the sensitive attribute A , i.e. $0 \leq DP \leq |A|$.

¹As our method is an in-processing method for neural networks (NN), we discuss existing in-processing methods tailored to NNs.

A metric whose upper bound is not fixed is undesirable since it prevents comparisons among DP values on different datasets, or even on different samples of the same dataset—which may lack some values for sensitive features. To avoid this issue, we propose two variants of DP, namely *weighted* (**WDP**) and *generalised* demographic parity (**GDP**), supporting the discrete and continuous cases, respectively. We define WDP as the weighted mean of the absolute value of the difference between the two distributions:

$$WDP_{h,A}(X) = \sum_{a \in A} \|\mathbb{E}[h(X) | A=a] - \mathbb{E}[h(X)]\| \cdot w_a \quad (5)$$

where w_a is the weight of the a -th value of the sensitive attribute A . The sum of the weights must be equal to 1. In this way, we can compare WDP values across different datasets since the upper bound is always 1. The values of the weights can be chosen according to the distribution of the sensitive attribute in the dataset, or they can be set to the same value (i.e., every value of the sensitive attribute is equally important). Doing so prevents the metric from being biased towards the most frequent values of the sensitive attribute. For instance, if we use the frequency of the values of the sensitive attribute as weights, the metric will be biased towards the most frequent values of the sensitive attribute. This could be undesirable when the dataset is unbalanced.

The definition of the GDP has been already introduced in a recent work [11]. This metric is a generalisation of DP in the continuous domain:

$$GDP'_{h,A}(X) = \int_l^u (\|\mathbb{E}[h(X) | A=a] - \mathbb{E}[h(X)]\| \cdot \mathbb{P}[A=a]) \cdot da \quad (6)$$

where $l = \min\{A\}$ (resp. $u = \max\{A\}$) is the minimum (resp. maximum) admissible value of A and $\mathbb{P}[A=a]$ is the probability density function of A . However, this definition of GDP could be blind to unbalanced distributions of the sensitive attribute due to the frequency weighting. So, we propose a variation of GDP where the weights can be chosen by the user:

$$GDP_{h,A}(X) = \int_l^u (\|\mathbb{E}[h(X) | A=a] - \mathbb{E}[h(X)]\| \cdot w_a) \cdot da \quad (7)$$

when $w_a = \mathbb{P}[A=a]$, Equation (7) is equivalent to Equation (6). Similarly, in WDP, the integral of the weights must be equal to 1.

Weighted and Generalised Disparate Impact The definition of DI can be straightforwardly extended to categorical sensitive attributes. It is worth noticing that DI for categorical sensitive features can be written with different aggregation functions. Here we use the weighted sum of all admissible values of A (this is not the only admissible choice: e.g., the minimum function is a valid alternative). In this way, we average over all admissible values of the sensitive attributes, similarly to Equation (5). Notably, when computing the conditional probability w.r.t. some admissible value a of the sensitive feature A , there are now two relevant cases: $A = a$ and $A \neq a$. Accordingly, we define the weighted DI (**WDI**) function as follows:

$$WDI_{h,A}(X) = \sum_{a \in A} \eta \left(\frac{\mathbb{E}[h(X) | A=a]}{\mathbb{E}[h(X) | A \neq a]} \right) \cdot w_a \quad (8)$$

Finally, we provide a continuous definition for disparate impact (**GDI**), similarly to 6:

$$GDI_{h,A}(X) = \int_l^u \eta \left(\frac{\mathbb{E}[h(X) | A=a]}{\mathbb{E}[h(X) | A \neq a]} \right) \cdot w_a \cdot da \quad (9)$$

where l, u have the same meaning of eq. (6).

Weighted and Generalised Equalised Odds Despite Equation (3) already handling both binary and categorical data for the protected feature, the metric suffers the lack of limits in the resulting value. To overcome this issue, and to be uniform with the other metrics as well, we define the weighted version of EO (**WEO**) as follows:

$$WEO_{h,A}(X) = \sum_{(a,y)}^{A \times Y} eo_{h,A}(X, a, y) \cdot w_a \quad (10)$$

There, eo is defined as in eq. (3).

Following the same generalisation approach used for DP, we extend EO defining the generalised equalised odds (**GEO**) for a binary classification task:

$$GEO_{h,A}(X) = \int_l^u (eo_{h,A}(X, a, 0) + eo_{h,A}(X, a, 1)) \cdot w_a \cdot da \quad (11)$$

3.2. FaUCI

Our method targets all ML models that are trained with stochastic gradient descent (SGD)—and, in particular, neural networks (NN). Similarly to other fairness algorithms, the main idea consists of introducing a specific cost factor in the loss function of the model. The cost factor depends on the specific fairness metric to be minimised. More precisely, the fairness metrics are the ones presented in Section 2.1—WDP and GDP, WDI and GDI, WEO and GEO for categorical and continuous versions of DP, DI, and EO, respectively. Unlike other methods, FaUCI (*i*) requires that the fairness metric to optimise is bounded in the range $[0, 1]$ (i.e., the training phase is not affected by the variation in the number of admissible values of the sensitive attribute), (*ii*) supports binary, categorical, and continuous sensitive attributes, and (*iii*) can be used to achieve intersectional fairness.

In particular, FaUCI prescribes the adaption of a loss function matching the following pattern:

$$\mathcal{L}_{h,A}(X, Y) = E(h(X), Y) + \lambda F_{h,A}(X, Y) \quad (12)$$

where E is the function scoring the error of the model, $F_{h,A}$ is an additive regularisation component aimed at scoring fairness w.r.t. A – and, potentially, also Y –, and $\lambda \in \mathbb{R}_{>0}$ is a hyperparameter that weights the regularisation term. The particular choice of E depends on the learning task at hand. So, for instance, classification tasks may rely on accuracy or cross-entropy. Similarly, F may be chosen among the fairness metrics presented in Section 2.1.

In practice, one key aspect of applying eq. (12) is w.r.t. to which input data (X) the fairness metric ($F_{h,A}$) is computed. Ideally, the bigger the dataset, the better. However, as a loss function aimed at steering the SGD process, $\mathcal{L}_{h,A}$ must be computed several times along the many epochs of a model’s training. The amount of data upon which the loss function is computed is called *batch size*, and it is commonly a hyperparameter of any implementation of SGD. So, the batch size is controlling the goodness of the fairness estimation provided by $F_{h,A}$, for the current epoch. This implies FaUCI is essentially subject to an implicit trade-off: the greater the batch size, the more accurate the fairness metric will be, at the price of a slower learning process.

About novelty. FaUCI differs from other related works in many aspects. First, it supports sensitive attributes of any kind – i.e., binary, categorical, and continuous –, and it is agnostic w.r.t. the fairness metric of choice—e.g., DP, DI and EO. The resulting loss function ($\mathcal{L}_{h,A}$) can be exploited in binary classification problems.

Second, FaUCI relies on a straightforward estimation of the probabilities that appear in the fairness metrics during training. Given a batch of size B , it is reasonable to consider the probabilities computed on the batch similar to the ones computed on the entire dataset if B is sufficiently large. This approximation does not hold if B is not high enough, i.e., if the distributions in the batch differ from the distributions in the whole dataset with statistical significance. Other approaches, e.g. the ones based on kernel density estimation (KDE) such as Cho, require the fine-tuning of additional hyper-parameters to estimate the

probabilities—and therefore to compute fairness metrics. By avoiding KDE, FaUCI does not require fine-tuning additional hyperparameters, other than λ in Equation (12).

Third, because in Section 3.1 we introduced the weighted versions of the fairness metrics, FaUCI does not suffer from unbalanced distributions of the sensitive attribute. Indeed, despite being a relatively small change, the weighted versions of the fairness metrics are paramount to avoid biases towards the most frequent values of the sensitive attribute. Unbalanced datasets are common in real-world applications, and the weighted fairness metrics are a key feature to ensure fairness in such scenarios. Using the non-weighted versions of the fairness metrics could lead to misleading values that suggest absence of bias when it is not the case.

Finally, FaUCI is general enough to be easily extendible with fairness metrics of other sorts. Moreover, one intriguing possibility is to use FaUCI to achieve intersectional fairness [16]. We will discuss this in Section 3.4.

3.3. Theoretical Analysis

During the training of a NN the training set is organised into batches. For each epoch of the training, all batches are used exactly one time to perform one step of inference (propagation) and one step of gradient descent (backpropagation). Between the two steps, the loss function is computed. Here we report the algorithms for computing the fairness cost factor for WDP, WDI and WEO metrics (continuous variants are omitted as algorithms are identical). For the sake of simplicity, we choose as weights the frequencies of the values of the sensitive attribute (note that this choice does not change the overall computational cost).

Algorithm 1 Weighted demographic parity (returns wdp)

Require: h	the predictive model
Require: A	sensitive attribute
Require: $batch$	a batch of the dataset

- 1: $wdp \leftarrow 0$
- 2: $estimated \leftarrow \mathbb{E}[h(batch)]$
- 3: **for all** a in A **do**
- 4: $estimated_a \leftarrow \mathbb{E}[h(batch) \mid A=a]$
- 5: $p_a \leftarrow \mathbb{P}[A=a]$
- 6: $wdp \leftarrow wdp + (\|estimated - estimated_a\| * p_a)$

Algorithm 1 describes the steps needed for computing the WDP metric within a single batch. The computational cost for computing the estimated probability in line 2 is $O(B)$, where B is the batch size. Similarly, $O(B)$ is the cost for computing the values at lines 4 and 5, while line 6 has a constant cost (we treat the operation of computing the output value of the model as having a constant cost). These steps are repeated for every admissible value of A . Accordingly, the resulting computational cost of the algorithm is $O(B) + 2O(BN) + O(N) = O(BN)$, where N is the number of different values of A .

Algorithm 2 Weighted disparity impact (returns wdi)

Require: $h, A, batch$ like in Algorithm 1

- 1: $wdi \leftarrow 0$
- 2: **for all** a in A **do**
- 3: $est_a \leftarrow \mathbb{E}[h(batch) \mid A=a]$
- 4: $est_not_a \leftarrow \mathbb{E}[h(batch) \mid A \neq a]$
- 5: $p_a \leftarrow \mathbb{P}[A=a]$
- 6: $wdi \leftarrow wdi + (\min\{\frac{est_a}{est_not_a}, \frac{est_not_a}{est_a}\} * p_a)$

Algorithm 2 elicits the steps for computing WDI. Values at lines 3, 4 and 5 have computational cost

$O(B)$, whereas line 6 has a constant cost and each of them is repeated N times. Overall, the resulting cost of the algorithm is $3O(BN) + O(N) = O(BN)$.

Algorithm 3 Weighted equalised odds (returns weo)

Require: $h, A, batch$ like in Algorithm 1,	Y : ground truth
1: $weo \leftarrow 0$	
2: $est_y_i \leftarrow \mathbb{E}[h(batch) \mid Y=i]$	$\forall i \in \{0, 1\}$
3: for all a in A do	
4: $est_a_y_i \leftarrow \mathbb{E}[h(batch) \mid A=a, Y=i]$	$\forall i \in \{0, 1\}$
5: $p_a \leftarrow \mathbb{P}[A=a]$	
6: $weo_i \leftarrow \ est_a_y_i - est_y_i\ $	$\forall i \in \{0, 1\}$
7: $weo \leftarrow weo + ((weo_0 + weo_1) * p_a)$	

Finally, Algorithm 3 shows the steps for WEO metric. As Equation (10) is designed for binary classification tasks, there is no need to iterate over an undefined number of admissible classes; rather, only two classes need to be considered. This simplifies the formulation of the algorithm and lowers the resulting computational cost. Values computed at lines 2 and 3 cost $O(B)$. Inside the loop, the computational cost of the values from lines 5-7 is also $O(B)$, for values from lines 8-10 is $O(1)$. So, the resulting cost of the algorithm is $2O(B) + 3O(BN) + 3O(N) = O(BN)$.

All three algorithms have a computational cost that is linear both in the batch size (B) and in the number of possible values of A (N). Their continuous variants have similar steps. The main difference is that instead of working with equalities and inequalities w.r.t. the values of A , we work with integrals. The implementation used for the experiments is histogram-based, i.e., several non-overlapping intervals of the same width. Then, the resulting computational cost depends on the number of intervals instead of the number of possible values of A .

3.4. Intersectional Fairness Extension

We extend FaUCI to support the optimisation of fairness in an intersectional setting, i.e., when the intersections of groups of protected attributes are considered (a.k.a., subgroups). FaUCI can support the optimisation of fairness metrics involving *multiple* protected attributes in two ways.

The first one consists in creating, before computing the loss cost, a dummy protected attribute by combining the original ones that the user wants to consider (i.e., computation of the subgroups). In this scenario, FaUCI can be used as-is (see Equation (12)) to optimise the fairness of the intersection of the original protected attributes. It is worth noticing that using the combination of multiple attributes could easily become intractable due to the computational cost of the underlying fairness metric (see Section 3.3). For this reason, we propose an alternative approach.

The second method consists in combining multiple loss factors, each one penalising the violation of the fairness of a single protected attribute.

$$\mathcal{L}_{h, \bar{A}}(X, Y) = E(h(X), Y) + \lambda_1 F_{h, A_1}(X) + \dots + \lambda_n F_{h, A_n}(X)$$

where \bar{A} is a set of protected attributes A_1, \dots, A_n . In this way, *multiple* fairness metrics can be *simultaneously* optimised for the same sensitive feature by keeping the computational cost of the fairness metric tractable (i.e., the cost of computing the fairness metric is linear in the number of protected attributes and not exponential). This approach is a trade-off between the computational cost and the complexity of the fairness metric. Further details and empirical results about FaUCI in the context of intersectional fairness will be presented in another work.

4. Experiments

We validate FaUCI on a well-known and widely-used real-world dataset for fairness algorithms: the UCI Adult dataset [17]. The dataset is already divided into two sets – training and test – with 32, 561

and 16,281 records respectively. There are 14 independent features of different types (i.e., boolean, categorical and continuous) and one dependent variable. Each record represents one individual and the task consists of classifying whether the individual earns more than 50,000 USD per year or less/equal to that amount. Adult is a standard de facto in the fairness community, mostly because it inherits the biases of the real world. Despite its limitations [18], it is still a valid benchmark for our purposes.

4.1. Setup

We endeavor to maintain a consistent setup with other works [10, 8] whenever possible. We use k-fold cross-validation with $k=5$ for each experiment. We first combine the two datasets into one single set of 48,842 records. Then we standardise it and apply the 80-20 rule to partition it into two new datasets. The second (9,768 records) is used as test set, while the first one (39,074 records) is used for generating the 5 folds. Conversely, for each configuration we run 5 experiments – the validation fold changes each time – and we report the mean values.

The model that we train is a feed-forward fully connected NN with 2 hidden layers of 100 and 50 neurons respectively. The number of epochs is up to 5,000 and the batch size is set to 500. We include two early stop conditions: the first one happens when both accuracy and the fairness metric reach a predefined target, while the second is triggered if there is no improvement for a specified amount of epochs (patience). In detail, we set the target accuracy to 0.9, and the thresholds for the fairness metrics to 0.01, 0.99, 0.01 for DP, DI, and EO respectively (we remind that the final value of DI metrics is expressed as $1 - DI$ to be compliant to the definition from the literature). The patience for the second condition is 100. These two conditions are evaluated at the end of each epoch on the validation fold. We additionally include in the experiments the evaluation of the NN without any fairness constraint, denoted as the “vanilla model”, serving as our baseline.

We conduct experiments with our method using WDP, GDP, WDI, GDI, WEO and GEO fairness metrics on three different sensitive attributes: sex (binary), ethnicity (categorical) and age (continuous). We decide to assign for the W^* metrics the same weight to each value of the protected attribute, i.e., $w_a = \frac{1}{N}$ for all $a \in A$ where N is the number of unique values of A . This choice is motivated by the fact that we want to give the same importance to each group of A since the attributes we want to protect are unbalanced (i.e., sex and ethnicity). Instead, for the G^* metrics we use the density of the continuous attribute (i.e., age) as weight since it is far less unbalanced than the other two attributes. We want to underline that the user can choose arbitrary weights as long as they fulfill the properties in Section 3.1.

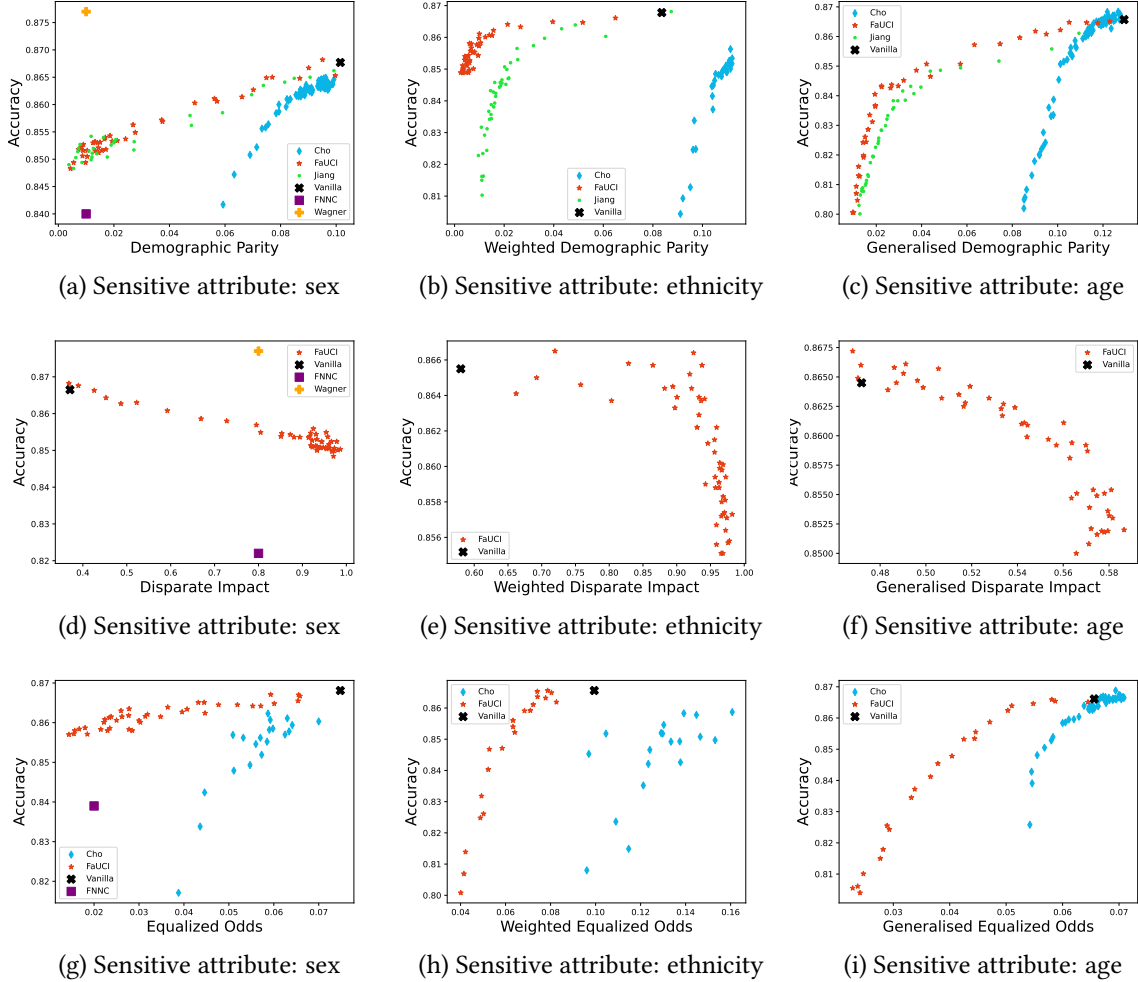
We are also able to fully reproduce and reuse the code of two state-of-the-art methods—Cho, Jiang. Unfortunately, we are unable to replicate the same procedure for the other two approaches—FNNC, Wagner. However, due to the similarity of their original setups with ours, we still report their results for discussion purposes.

FaUCI shares with Cho and Jiang one hyper-parameter – lambda (λ) – that weights the contribution of the fairness metric. All experiments are conducted by varying the value of λ . The other hyper-parameters of Cho and Jiang are the same used in their papers. The source code of the experiments is available at <https://anonymous.4open.science/r/FaUCI-aequitas-2024>.

4.2. Demographic Parity Results

Figure 1 shows the accuracy-demographic parity trade-off between FaUCI, Cho and Jiang methods (also FNNC and Wagner when using “sex” as sensitive attribute). Figure 1a reports the experiments using “sex” as a binary protected feature. FaUCI and Jiang have similar behaviour, while Cho is worse. FNNC and Wagner results are included, noting that only one accuracy-fairness metric result is reported by the authors, hence represented by a single dot in the plot. The accuracies are 84% and 87.7% respectively. FaUCI reaches an accuracy of 85.27% with $DP \leq 0.01$. While the result for FNNC could be coherent with the experiment setup, the one for Wagner is more uncertain. The authors reported that the accuracy of the model using their method is 87.7%, which is far more than the vanilla model

Figure 1: Experiments comparing FaUCI to related works, using different (variants of) the DP, DI, and EO fairness metrics. Each row corresponds to a different fairness metric (DP, DI, and EO). Each column corresponds to a different variant of the fairness metric (normal, for binary attributes, weighted for categorical attributes, and generalised for continuous attributes). Each dot of each chart is the average of 5 runs (5-folds cross-validation). In each chart, red stars represent FaUCI, blue diamonds Cho’s method, green dots Jiang’s method and the black cross the vanilla neural network (i.e., the NN without fairness constraints). For FNNC and Wagner we were not able to reproduce experiments (due to issues in their code), so we report the results as reported by the authors—which implies only one dot per chart.



(86.7%) that is trained without any fairness constraint. Figure 1b shows the experiments with “ethnicity” as categorical sensitive feature. We observe that FaUCI outperforms the other methods. Jiang has similar performances until DP is above 0.04, while Cho is always worse. We motivate this behaviour because the categorical sensitive feature “ethnicity” has 5 different groups, and they are unbalanced. Also “sex” is slightly unbalanced, but it has just 2 groups. Finally, Figure 1c reports the experiments for the continuous protected features “age”. In this scenario, our method and Jiang’s method have similar performance but ours is generally better. Cho’s method performs always worse than the others. Overall, our approach performs better than the others when optimising both accuracy and demographic parity for all kinds of sensitive attributes.

4.3. Disparate Impact Results

Figure 1 shows the trade-off between accuracy and disparate impact with FaUCI method. Here a comparison with other approaches can be done only with FNNC and Wagner when using a binary sensitive attribute and with the vanilla NN. Figure 1d describes the results with “sex” as protected

feature. The vanilla NN has an accuracy of 86.7% with $DI = 0.34$. FaUCI can reach $DI \geq 0.8$ – which is a commonly agreed threshold in the literature for DI – preserving an accuracy of 85.6%. Authors of FNNC say that their method with $DI \geq 0.8$ reaches 82.2%, under the same DI condition Wagner method can reach 87.7% of accuracy. We emphasize that the results of FNNC and Wagner presented here are reported by the original authors, and we are unable to independently reproduce them. For this reason, and because Wagner’s method reaches a higher accuracy than the vanilla NN (87.7% and 86.7% respectively), there is a possibility that the comparison is not straightforward. Figure 1e shows the performance of FaUCI when using the categorical sensitive attribute “ethnicity”. The vanilla NN has a DI of 0.47 and its accuracy is 86.8%. Our method can preserve the accuracy of the vanilla model with a satisfactory $DI \geq 0.8$. Finally, in Figure 1f there are the results of the experiments with “age” as continuous protected feature. The vanilla NN reaches an accuracy of 86.4% with $DI = 0.47$. FaUCI cannot achieve the desired value of 0.8 for DI. However, it improves the metric until $DI = 0.58$ with an accuracy of 85.5%. Overall, experiments show that FaUCI can significantly improve DI metric while preserving high accuracy.

4.4. Equalised Odds Results

Figure 1 reports the trade-off between accuracy and equalised odds using FaUCI and Cho (also FNNC when using “sex” as sensitive attribute). Figure 1g shows the results of the methods with “sex” as binary sensitive feature. We observe that FaUCI performs generally better than Cho, and it is able to maintain high accuracy ($\geq 85.9\%$) with an $EO \leq 0.02$. FNNC scores a reasonably good accuracy value (83.9%) while having an $EO \leq 0.02$. Figure 1h shows a peculiar behaviour of the performance of the two methods when using “ethnicity” as protected attribute. FaUCI maintains high accuracy while reducing the EO value, instead Cho drops its accuracy. One possible reason for this behaviour is that the underlying Q-function of Cho’s method might not adequately approximate the distributions of the categorical values. Indeed, also in Figure 1b there is a similar – yet less accentuated – behaviour. Another possible reason is that the method is vulnerable to unbalanced datasets. Experiments with the continuous sensitive attribute “age” are reported in Figure 1i. We observe that FaUCI outperforms Cho’s method in terms of accuracy and EO. Once again, we speculate that the behavior could be attributed to the Q-function. In this case, for continuous values, the approximation of distributions is better than our approach. Overall, the results with EO constraints show that FaUCI can improve the fairness metric while preserving a high accuracy-metric trade-off. Cho’s method can optimise the EO value for attributes “sex” and “age” but it fails to do so with “ethnicity”.

5. Conclusion

In this work we present FaUCI, a general purpose method to enforce fairness constraints in NN. The main features of this approach are the ability to work with different fairness metrics, the possibility to use it on any data type of sensitive attribute, the resilience to unbalanced datasets, and the ability to work with multiple sensitive attributes. We introduce a new formulation of some fairness metrics used by FaUCI enabling the utilization of these metrics with continuous sensitive attributes and enhancing resilience to unbalanced datasets. We validate our method on a real-world dataset and compare it with other state-of-the-art methods. Results show that FaUCI is able to improve the fairness metrics while preserving a high accuracy-metric trade-off. Future works will be devoted to investigating the potential application of FaUCI to enhance fairness in cases involving multiple sensitive attributes within the context of intersectionality. Another interesting direction is to compare FaUCI, and in general in-processing techniques, with post-processing methods.

References

- [1] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, A. Galstyan, A survey on bias and fairness in machine learning, *ACM Comput. Surv.* 54 (2022) 115:1–115:35. doi:10.1145/3457607.
- [2] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, R. S. Zemel, Fairness through awareness, in: S. Goldwasser (Ed.), *Innovations in Theoretical Computer Science 2012*, Cambridge, MA, USA, January 8-10, 2012, ACM, 2012, pp. 214–226. doi:10.1145/2090236.2090255.
- [3] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, S. Venkatasubramanian, Certifying and removing disparate impact, in: L. Cao, C. Zhang, T. Joachims, G. I. Webb, D. D. Margineantu, G. Williams (Eds.), *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney, NSW, Australia, August 10-13, 2015, ACM, 2015, pp. 259–268. doi:10.1145/2783258.2783311.
- [4] M. Hardt, E. Price, N. Srebro, Equality of opportunity in supervised learning, in: D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, R. Garnett (Eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016*, December 5-10, 2016, Barcelona, Spain, 2016, pp. 3315–3323. URL: <https://proceedings.neurips.cc/paper/2016/hash/9d2682367c3935defcb1f9e247a97c0d-Abstract.html>.
- [5] N. Grgic-Hlaca, M. B. Zafar, K. P. Gummadi, A. Weller, The case for process fairness in learning: Feature selection for fair decision making, in: *Machine Learning and the Law – Symposium @ NIPS 2016*, Barcelona, Spain, 2016. URL: <http://www.mlandthelaw.org/papers/grgic.pdf>.
- [6] M. J. Kusner, J. R. Loftus, C. Russell, R. Silva, Counterfactual fairness, in: I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, R. Garnett (Eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 4066–4076. URL: <https://proceedings.neurips.cc/paper/2017/hash/a486cd07e4ac3d270571622f4f316ec5-Abstract.html>.
- [7] B. d’Alessandro, C. O’Neil, T. LaGatta, Conscientious classification: A data scientist’s guide to discrimination-aware classification, *Big Data* 5 (2017) 120–134. doi:10.1089/big.2016.0048.
- [8] P. Manisha, S. Gujar, FNNC: achieving fairness through neural networks, in: C. Bessiere (Ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, ijcai.org, 2020, pp. 2277–2283. doi:10.24963/ijcai.2020/315.
- [9] J. Cho, G. Hwang, C. Suh, A fair classifier using kernel density estimation, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, December 6-12, 2020, virtual, 2020. URL: <https://proceedings.neurips.cc/paper/2020/hash/ac3870fcad1cfc367825cda0101eee62-Abstract.html>.
- [10] B. Wagner, A. S. d’Avila Garcez, Neural-symbolic integration for fairness in AI, in: A. Martin, K. Hinkelmann, H. Fill, A. Gerber, D. Lenat, R. Stolle, F. van Harmelen (Eds.), *Proceedings of the AAAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering (AAAI-MAKE 2021)*, Stanford University, Palo Alto, California, USA, March 22-24, 2021, volume 2846 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021. URL: <https://ceur-ws.org/Vol-2846/paper5.pdf>.
- [11] Z. Jiang, X. Han, C. Fan, F. Yang, A. Mostafavi, X. Hu, Generalized demographic parity for group fairness, in: *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event*, April 25-29, 2022, OpenReview.net, 2022. URL: <https://openreview.net/forum?id=YigKIMJwjye>.
- [12] M. López-Benítez, F. Casadevall, Versatile, accurate, and analytically tractable approximation for the gaussian q-function, *IEEE Trans. Commun.* 59 (2011) 917–922. doi:10.1109/TCOMM.2011.012711.100105.
- [13] S. Badreddine, A. S. d’Avila Garcez, L. Serafini, M. Spranger, Logic tensor networks, *Artif. Intell.* 303 (2022) 103649. doi:10.1016/j.artint.2021.103649.
- [14] T. Kamishima, S. Akaho, H. Asoh, J. Sakuma, Fairness-aware classifier with prejudice remover regularizer, in: P. A. Flach, T. D. Bie, N. Cristianini (Eds.), *Machine Learning and Knowledge*

Discovery in Databases - European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II, volume 7524 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 35–50. doi:10.1007/978-3-642-33486-3_3.

- [15] R. Berk, H. Heidari, S. Jabbari, M. Joseph, M. J. Kearns, J. Morgenstern, S. Neel, A. Roth, A convex framework for fair regression, CoRR abs/1706.02409 (2017). URL: <http://arxiv.org/abs/1706.02409>. arXiv:1706.02409.
- [16] K. Crenshaw, Demarginalizing the intersection of race and sex: A black feminist critique of antidiscrimination doctrine, feminist theory and antiracist politics, in: *Feminist legal theories*, Routledge, 2013, pp. 23–51. URL: <https://chicagounbound.uchicago.edu/uclf/vol1989/iss1/8>.
- [17] B. Becker, R. Kohavi, Adult, UCI Machine Learning Repository, 1996. DOI: <https://doi.org/10.24432/C5XW20>.
- [18] F. Ding, M. Hardt, J. Miller, L. Schmidt, Retiring adult: New datasets for fair machine learning, in: M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, J. W. Vaughan (Eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, 2021*, pp. 6478–6490. URL: <https://proceedings.neurips.cc/paper/2021/hash/32e54441e6382a7fbacbbaf3c450059-Abstract.html>.