Structured Prediction of Sparse Dependent Variables for Traffic State Etimation in Large-Scale Networks ☆

Andrija Petrović[a], Mladen Nikolić[b], Sandro Radovanović[a], Boris Delibašić[a], Miloš Jovanović[a]

[a] *University of Belgrade - Faculty of Organizational Sciences, Jove Ilica 154, Belgrade, Serbia*
[b] *University of Belgrade - Faculty of Mathematics, Studentski Trg 16, Belgrade, Serbia*

Abstract

With growing awareness of societal impact of artificial intelligence, fairness has become an important aspect of machine learning algorithms. The issue is that human biases towards certain groups of population, defined by sensitive features like race and gender, are introduced to the training data through data collection and labeling. Two important directions of fairness ensuring research have focused on (i) instance weighting in order to decrease the impact of more biased instances and (ii) adversarial training in order to construct data representations suitable for prediction of target variable, but uninformative about sensitive attributes. In this paper we propose a Fair Adversarial Instance Re-weighting (FAIR) model, which uses adversarial training to learn instance weighting to ensure fair predictions. We propose four different variants of the method and, among other things, demonstrating how such models can be cast in fully probabilistic framework. Additionally, theoretical analysis of FAIR models properties have been studied extensively. We compare FAIR models to 7 other related and state-of-the-art models and demonstrate that FAIR is able to achieve better trade-off between accuracy and unfairness. Moreover, to the best of our knowledge this is the first model that merges reweighting and adversarial approaches with a weighting function that can provide interpretable information about fairness of individual instances.

Keywords: Fairness, Adversarial training, Instance reweighting, Deep learning, Classification

---

☆

Email address: aapetrovic@mas.bg.ac.rs (Andrija Petrović)

1. Introduction

Machine learning algorithms have lead to many recent breakthroughs on different complex tasks that cannot be solved satisfactorily by domain specific algorithms, such as face detection [1], object detection [2], machine translation [3], facial expression recognition [4], sport prediction [5], etc. With this enormous success in practical applications, social issues related to machine learning algorithms are becoming increasingly important. One of the most prominent such issues is fairness of machine learning algorithms, as related to discrimination and bias [6].

It is well known that in many applications data reflects the disparities, distortions, and biases from the real world and the measurement process. Salary prediction [7], credit risk prediction [8], medical prediction [9], personnel planning and recruiting forecasting methods [10], are just some of the examples where data, collected from societal interactions, is biased with respect to age, gender, or race. Therefore, machine learning algorithms will extract and learn biases that are present in the data, including harmful ones. Improving fairness of biased data and decision procedures based on that data is not only a problem of society, but also a problem of machine learning. It is critical to guarantee that the prediction obtained by machine learning algorithms is based on appropriate information and that the outcomes are not biased towards certain groups of population defined by sensitive features like race and gender [11].

Current techniques for improving fairness fall into three different groups: preprocessing techniques [12, 13], techniques based on optimization at training time [14, 15, 16, 17], and post-processing based ones [18, 19]. The state-of-the-art techniques for mitigating bias by preprocessing are based on instance reweighing [20], a technique that assigns weights to instances as means of controlling their influence on the model during training.

Adversarial training has widely been used for finding Nash equilibrium in minimax (zero-sum) games [21]. Recently adversarial framework became extremely popular in debiasing deep learning models by introducing two networks, one for predicting output label and one for predicting sensitive attributes [22]. Developed architecture learns new feature space representation which allows fairly accurate prediction of the output label by the first network, while being maximally uninformative about the sensitive attributes, so that the second network has to fail in its task.

In this paper, we propose Fair Adversarial Instance Re-weighting (FAIR) – a novel model for mitigating bias in discriminative algorithms by using an adversarial framework to learn instance reweighing and not new data representation as is done in previous work. Also, in our approach, the weighting is not performed as preprocessing, but is integrated in the learning procedure so that the learning is performed end-to-end. FAIR consists of three neural networks: the first one is used for determining weights for each instance, the second one for predicting the sensitive attribute, and the third one for predicting the output label. FAIR proposes four different weighting methods. In the first method (FAIR-scalar), obtained scalar weights are used for weighting the log likelihood of corresponding instances, whereas in all other methods instance weights are random variables parametrized by the output of the first network. In the second method (FAIR-Bernoulli) the weights are distributed according to Bernoulli distribution and during learning, score function is used to evaluate the expectation of the log likelihood. Other two methods rely on beta distribution, but they differ in evaluation of the expectation of the log likelihood – the third one (FAIR-betaSF) uses score function and the fourth one (FAIR-betaREP) relies on reparametrization. Additionally, we discuss how to reduce the variance of FAIR-Bernoulli and FAIR-betaSF using baseline functions. We evaluated our models on four different real-world datasets and compared them to the state-of-the-art techniques. The results demonstrate that FAIR achieved the best results, with respect to fairness and classification performance. Furthermore, to the best of our knowledge this is the first model that merges reweighting and adversarial approaches with a weighting function that can provide interpretable information about fairness of individual instances.

In section 2 the related work is reviewed. Adversarial models for debiasing datasets in probabilistic and non probabilistic framework are presented in section 3. The proposed FAIR algorithm with different variants is described in 4. Experimental setup and results on real-world applications are shown in sections 5 and 6, respectively. Final conclusions are given in section 7.

2. Related Work

Notion of fairness. In context of decision-making, (un)fairness has several distinct notions, one of the most prominent being *disparate impact* [23]. It represents a situation in which decisions ($\hat{\mathbf{y}}$) made by classifier are disproportional between instances with different values of sensitive attributes ($\mathbf{s}$). We use three measures of disparate impact. First metric used is *absolute statistical parity difference*:

$$\mathbf{ASD} = |P(\hat{\mathbf{y}} = 1|\mathbf{s} = 0) - P(\hat{\mathbf{y}} = 1|\mathbf{s} = 1)| \tag{1}$$

Low values of **ASD** mean that both groups have approximately the same probability of being labeled 1 (e.g., bank loan granted) by the model. In such case, the classifier is said to have statistical parity. Second metric we used is *absolute equal opportunity difference*:

$$\mathbf{AEOD} = |TPR_{\mathbf{s}=0} - TPR_{\mathbf{s}=1}| \tag{2}$$

where TPR represents true positive rate (recall) of the prediction model. Recall reflects opportunity, so this measure can be interpreted as a difference of opportunities between unprivileged and privileged group. Value of **AEOD** close to 0 is desirable. Besides *absolute statistical parity difference* and *absolute equal opportunity difference* the third metric that we used is *average odds difference*. Average odds difference can be formulated as:

$$\mathbf{AOD} = \frac{1}{2}(|FPR_{s=unpriv} - FPR_{s=priv}| + |TPR_{s=unpriv} - TPR_{s=priv}|) \quad (3)$$

where FPR represent false positive rate (probability of false alarm), and TPR true positive rate (recall). Values of **AOD** close to zero are preferred.

Interested readers are referred to an extensive review on topic of fairness-aware machine learning algorithms which are presented in [24] and [25].

This paper focuses on two approaches for bias reduction – instance reweighting and adversarial training. Instance reweighting has shown impressive results compared even to some of the state-of-the-art models [26, 27]. It assigns lower importance to unfair examples or removes them from the learning process. More specifically, those examples will have lower impact on the likelihood function one tries to optimize. However, this is often not a complete solution to the problem. Adversarial training focuses on learning new data representation from which it is possible to predict the target variable, but not possible to predict the sensitive attribute. Ideally, one learns the model which is solving the problem at hand and not generating unwanted bias. However, this approach creates a trade-off between two goal functions and therefore reaching Nash equilibrium [21]. Our approach combines adversarial framework with instance reweighting, trying to obtain the best from both worlds.

Instance Reweighting. Instance reweighting, as a preprocessing technique, was traditionally used for the class imbalance problem. The idea is to assign larger weights to instances of lower cardinality class, so that the learning algorithm gives more importance to that class. This idea can be applied to the fairness problem as well. The simplest approach is to assign weights to instances so that sums of weights per value of sensitive feature is the same and all instances from a group have the same weight [20]. That approach was improved in [27] by utilizing adaptive sensitive reweighting procedure. More specifically, one can use variational fair auto encoder with Maximum Mean Discrepancy [28] which calculates distances between distributions using kernels. It is worth noticing that instance reweighting has shown to have lower disparate impact [26] compared to not applying any instance weighting strategy. However, if one wants to apply some instance reweighting strategy and create a discrimination-free predictive model one needs to perform a two step procedure – first to obtain instance weights, and then to use the weights by the learning algorithm.

This is a drawback of this approach since the weighting procedure is oblivious of the model representation and learning algorithm and therefore might choose suboptimal weights for them. Also, not all algorithms naturally support instance weighs.

Adversarial training. Adversarial training can be defined as a framework for mitigating biases by including a variable for the group of interest and simultaneously learning a predictor and an adversary. The idea is to learn a target prediction model, while at the same time lose the ability to predict the sensitive attribute. Adversarial training for fairness was first presented in [29]. Similar model was applied to recidivism prediction in order to remove racial bias [22]. An important approach of such kind is Fair Adversarial Discriminative model (FAD) [15]. Moreover, theoretical foundations of the relationship between the label classifier performance and the adversary's ability to predict the sensitive attribute value were explained. Also, in the same paper, a variation of the adversarial learning procedure is developed to increase diversity among elements of each mini-batch of the gradient descent training, in order to achieve a representation that does not suffer from "mode collapse". Another theoretical analysis of solving fairness problem via adversarial approach is presented in [30]. Another adversarial approach focuses on learning to select non-sensitive features on per instance basis [11]. The adversarial approach is employed to minimize the correlation between selected features and sensitive information.

3. Fairness via Adversarial Network

The dataset given by $D = \{(\mathbf{x}_i, \mathbf{y}_i, \mathbf{s}_i)\}_{i=1}^N$, consists of input features $\mathbf{x}$, the true label (or the target variable) $\mathbf{y}$ and sensitive features $\mathbf{s}$. It is generated by joint true underlying distribution $D \sim P(\mathbf{x}, \mathbf{y}, \mathbf{s})$. Generally, an unfair discriminative model predicts the label $\hat{\mathbf{y}}$ based on both input $\mathbf{x}$ and sensitive features $\mathbf{s}$. Due to the bias in the dataset, such model is inadequate. A naive approach to ensuring fairness would be to eliminate sensitive features $\mathbf{s}$ from the dataset. However, the information contained in sensitive features can also be approximated from other input features $\mathbf{x}$. For example, location of residence correlates with race, although it is not obviously sensitive itself. In this section, we discuss already mentioned fair adversarial discriminative model (FAD) in more detail due to some similarities with our own approach. Also, we propose our probabilistic formulation of this model based on normalizing flows. This is not the main contribution of our paper, though. Instead, we use it as a reasonable baseline for our other probabilistic approaches.

3.1. FAD Model

The architecture of FAD model consists of a neural network with one shared layer and two task specific layers. The goal of the shared layer is to map input features $\mathbf{x}$ to representation space $\mathbf{z} = f_\theta(\mathbf{x})$, so that the obtained representation $\mathbf{z}$ is uninformative of sensitive features $\mathbf{s}$, but includes information needed to predict label $\mathbf{y}$. The first task specific layer is a predictor $g_\phi(\mathbf{z})$ of the output label $\mathbf{y}$, whereas the second task specific layer $h_\psi(\mathbf{z})$ estimates sensitive features $\mathbf{s}$. Since sensitive information may also be important for estimating the label, there is a trade-off between model fairness and accuracy of prediction, related to the mapping from input feature space $\mathbf{x}$ to representation space $\mathbf{z}$. Fairness in the FAD model is achieved through adversarial learning of the mapping $g_\theta(\mathbf{x})$ and a classifier $h_\psi(\mathbf{z})$, while learning the predictor $g_\phi(\mathbf{z})$. This ensures that accuracy is not fully sacrificed for fairness. In other words, neural networks $f_\theta(\mathbf{x})$ and $g_\phi(\mathbf{z})$ play a minimax game with classifier $h_\psi(\mathbf{z})$. We denote probability functions corresponding to these networks as $P_\phi(\mathbf{y}|\mathbf{x})$ and $P_\psi(\mathbf{s}|\mathbf{x})$. Formally, the adversarial problem of FAD model is:

$$\min_{\theta,\phi} \max_{\psi} \mathbb{E}_{\substack{\mathbf{x},\mathbf{y},\mathbf{s} \sim P(\mathbf{x},\mathbf{y},\mathbf{s}) \\ \mathbf{z}=f_\theta(\mathbf{x})}} \left[\alpha \log P_\psi(\mathbf{s}|\mathbf{z}) - \log P_\phi(\mathbf{y}|\mathbf{z})\right]$$

where α is a hyper-parameter for tuning the trade-off between model fairness and accuracy. Increased value of α influences model to be more focused on fairness and, consequently, representation $\mathbf{z}$ will be less informative about sensitive features $\mathbf{s}$ and also to some extent of true label $\mathbf{y}$.

3.2. Probabilistic framework with normalizing flows

In case of the FAD model, the representation $\mathbf{z} = f_\theta(\mathbf{x})$ is an output of a neural network. We propose a fully probabilistic model (FAD-prob) based on the FAD model by considering the representation $\mathbf{z}$ as a random latent variable and modeling its distribution. Representing the hidden space as a random variable has several advantages. The main one is related to the possibility to marginalize over latent variable space and obtain better predictive performance [31]. Moreover, it can provide a possibility to predict structured outputs [32] representing sensitive features and labels.

The conditional probability distributions of sensitive features $\mathbf{s}$ and of output labels $\mathbf{y}$, given inputs $\mathbf{x}$ can be obtained by marginalization of joint distributions $P(\mathbf{s}, \mathbf{z})$ and $P(\mathbf{y}, \mathbf{z})$, respectively as

$$P(\mathbf{y}|\mathbf{x}) = \int_{\mathbf{z}} P(\mathbf{y}|\mathbf{z})P(\mathbf{z}|\mathbf{x})d\mathbf{z} = \mathbb{E}_{\mathbf{z}\sim P(\mathbf{z}|\mathbf{x})}[P(\mathbf{y}|\mathbf{z})]$$

$$P(\mathbf{s}|\mathbf{x}) = \int_{\mathbf{z}} P(\mathbf{s}|\mathbf{z})P(\mathbf{z}|\mathbf{x})d\mathbf{z} = \mathbb{E}_{\mathbf{z}\sim P(\mathbf{z}|\mathbf{x})}[P(\mathbf{s}|\mathbf{z})]$$

Marginalization can be performed using reparametrization trick and normalizing flows [33]. The reparametrization of latent variable can be performed as $\mathbf{z} = f(\mu(\mathbf{x}) + L(\mathbf{x}) \cdot \epsilon)$ where f is a nonlinear mapping of variable $\mathbf{z}$ obtained by assuming normal distribution with mean $\mu(\mathbf{x})$ and covariance matrix $\Sigma(\mathbf{x})$, which can be factorized as $L^T(\mathbf{x})L(\mathbf{x})$ by Cholesky decomposition.

Based on this, the overall adversarial objective function of FAD-prob model is:

$$\min_{\theta,\phi} \max_{\psi} \mathbb{E}_{\substack{\mathbf{x},\mathbf{y},\mathbf{s} \sim P(\mathbf{x},\mathbf{y},\mathbf{s}) \\ \mathbf{z} \sim P_\theta(\mathbf{z}|\mathbf{x})}} \left[\alpha \log P_\psi(\mathbf{s}|\mathbf{z}) - \log P_\phi(\mathbf{y}|\mathbf{z}) \right]$$

4. Fair Adversarial Instance Re-weighting - FAIR

Unfairness which AI models learn is introduced through data instances containing unfair decisions. Therefore, we strive to recognize if a particular instance in a dataset is unfair. The main principle of FAIR is to reweight log likelihood of each instance, according to the trade-off between fairness and prediction performance, in order to obtain a fair and useful predictor of the target variable.

FAIR consists of three neural networks: weighting network $f_\theta(\mathbf{x})$, predictor network $g_\phi(\mathbf{x})$ and sensitive network $h_\psi(\mathbf{x})$. For an instance $\mathbf{x}$ the weighting network outputs the weight of that instance $w_\mathbf{x} \in [0, 1]$, while predictor network and sensitive network output prediction of output labels $\mathbf{y}$ and sensitive features $\mathbf{s}$, respectively. In order to incorporate fairness objective, FAIR weights log likelihood of instances, so that the ones that are strongly informative of sensitive attribute, but not of target variable are assigned low weights and the ones that are informative of target variable, but not of sensitive attributes are assigned high weights. The weighting network is not used during inference, but can be helpful for assessing new instances.

Based on different weighting techniques, we present four different FAIR weighting methods. The first one FAIR-scalar is based on non-probabilistic weighting framework, whereas FAIR-Bernoulli, FAIR-betaSF and FAIR-betaREP are based on probabilistic framework. The graphical representation of FAIR with different weighting methods are given in Fig. 1.

4.1. FAIR – non-probablistic framework

Assume that each instance $\mathbf{x}$ is assigned a scalar weight $f_\theta(\mathbf{x}) \in [0, 1]$ by a weighting network. Then, FAIR-scalar adversarial problem is given by:

$$(\theta^*, \phi^*, \psi^*) = \arg \min_{\theta,\phi} \max_{\psi} \mathbb{E}_{\substack{\mathbf{x},\mathbf{y},\mathbf{s} \sim P(\mathbf{x},\mathbf{y},\mathbf{s}) \\ w = f_\theta(\mathbf{x})}} [w \cdot (\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x}))] \tag{4}$$

Similarly to FAD model, the hyperparameter α controls the trade-off between fairness and prediction performance of the predictor networks, but this trade-off will be given further theoretical analysis.

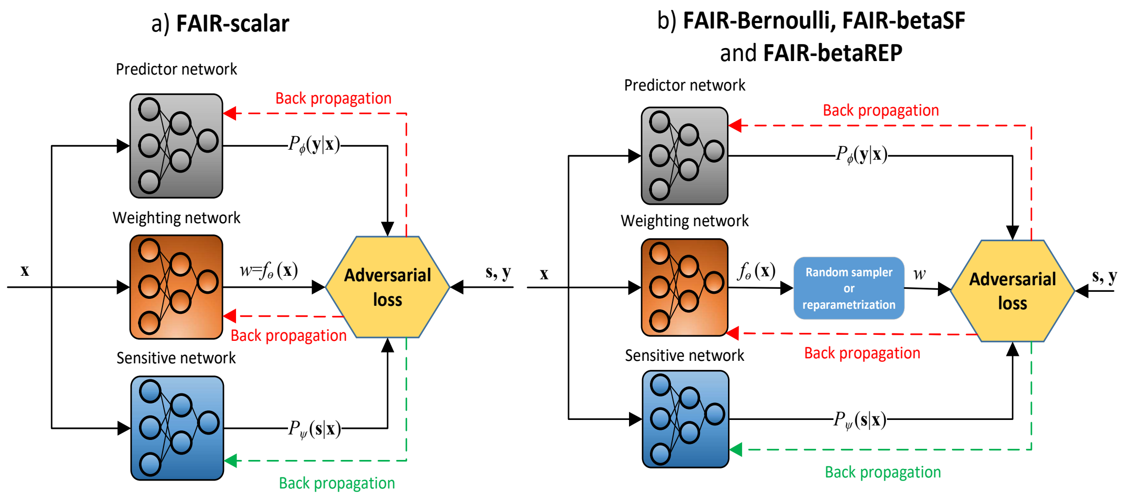

Figure 1: Graphical representations of FAIR with probabalistic and non-probabalistic frameworks

4.2. FAIR – probabilistic framework

In the the case of FAIR with probabilistic approach to weighting, it is assumed that weights of instances are random variables. In contrast to FAIR-scalar, in probabilistic framework, output of weighting network f_θ models a probability distribution of instance weights: $P(w_{\mathbf{x}}|\mathbf{x})$. Consequently, we can use different probability distribution models. We consider Bernoulli (FAIR-Bernoulli) and beta distribution (FAIR-betaSF and FAIR-betaREP).

FAIR-Bernoulli assumes that log likelihoods of instances, with respect to sensitive features $\log P_\psi(\mathbf{s}|\mathbf{x})$ and labels $\log P_\phi(\mathbf{y}|\mathbf{x})$ are weighted by integers $w_{\mathbf{x}} \in \{0,1\}$ such that it holds $P_\theta(w_{\mathbf{x}} = 1|\mathbf{x}) = f_\theta(x)$, meaning that the conditional probability of weights is Bernoulli distribution $\mathcal{B}(f_\theta(\mathbf{x}))$. The FAIR-Bernoulli adversarial loss $\mathcal{L}^{\mathcal{B}}_\alpha(\theta,\phi,\psi)$ is given by:

$$\mathbb{E}_{\substack{\mathbf{x},\mathbf{y},\mathbf{s}\sim P(\mathbf{x},\mathbf{y},\mathbf{s}) \\ w\sim P_\theta(w|\mathbf{x})}} \Big[w \cdot \left(\alpha \log P_\psi\left(\mathbf{s}|\mathbf{x}\right) - \log P_\phi\left(\mathbf{y}|\mathbf{x}\right)\right)\Big] \tag{5}$$

and the corresponding adversarial problem is $(\theta^*,\phi^*,\psi^*) = \arg\min_{\theta,\phi}\max_\psi \mathcal{L}^{\mathcal{B}}_\alpha(\theta,\phi,\psi)$ where the superscript $\mathcal{B}$ emphasizes Bernoulli assumption.

In order to optimize the loss, gradients with respect to θ, ϕ, and ψ need to be computed. Gradients with respect to ϕ and ψ are computed by standard backpropagation. However, gradient with respect to θ is trickier since θ defines the distribution of w over which the expectation is taken. Therefore, we derive gradient of adversarial loss $\nabla_\theta \mathcal{L}_\alpha(\theta, \phi, \psi)$ for FAIR-Bernoulli and FAIR-betaSF (Eq. 6) as:

$$\nabla_\theta \mathcal{L}_\alpha(\theta, \phi, \psi) = \nabla_\theta \mathbb{E}_{\substack{\mathbf{x},\mathbf{y},\mathbf{s} \sim P(\mathbf{x},\mathbf{y},\mathbf{s}) \\ w \sim P_\theta(w|\mathbf{x})}} \Big[w \cdot (\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x})) \Big]$$

The gradient operator ∇_θ can be propagated through the expectation as:

$$\nabla_\theta \mathcal{L}_\alpha(\theta, \phi, \psi) = \mathbb{E}_{\mathbf{y},\mathbf{x},\mathbf{s}} \Big[\int_w \nabla_\theta P_\theta(w|\mathbf{x}) \cdot w \cdot (\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi \left(\mathbf{y}|\mathbf{x}\right)) dw \Big]$$

Gradient of distribution $P_\theta(w|\mathbf{x})$ can be transformed as:

$$\begin{aligned} \nabla_\theta P_\theta(w|\mathbf{x}) &= P_\theta(w|\mathbf{x}) \cdot \frac{\nabla_\theta P_\theta(w|\mathbf{x})}{P_\theta(w|\mathbf{x})} \\ &= P_\theta(w|\mathbf{x}) \cdot \nabla_\theta \log P_\theta(w|\mathbf{x}) \end{aligned}$$

Following this transformation, the final form of gradient of the loss with respect to θ can be represented as:

$$\mathbb{E}_{\substack{\mathbf{x},\mathbf{s},\mathbf{y} \sim P(\mathbf{x},\mathbf{s},\mathbf{y}) \\ w \sim P_\theta(w|\mathbf{x})}} \Big[w \cdot \nabla_\theta \log P_\theta(w|\mathbf{x}) \cdot (\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x})) \Big] \tag{6}$$

Next, we assume that weights $w_\mathbf{x}$ are random variables distributed according to beta distribution which, in contrast to the case of FAIR-Bernoulli, take any value from the interval $[0, 1]$. The outputs of weighting network are parameters $\alpha_\mathbf{x}$ and $\beta_\mathbf{x}$ of beta distribution. The adversarial loss as defined by Eq. 5, but with beta distribution assumed instead of Bernoulli. We denote corresponding loss by $\mathcal{L}^\beta_\alpha(\theta, \phi, \psi)$ where β in the superscript emphasizes the assumed distribution. In optimization, the gradient $\nabla_\theta \mathcal{L}^\beta_\alpha(\theta, \phi, \psi)$ can be evaluated either by using score function as in Eq. 6 or by the reparametrization trick of beta distribution as shown in [34]. These two approaches we name FAIR-betaSF and FAIR-betaREP respectively.

Pseudocode of probabalistic FAIR with score function (FAIR-Bernoulli and FAIR-betaSF) is presented in Algortihm 1. FAIR losses are defined in terms of expectations. However, with finite samples, expectation is always approximated by sample mean, which is used in the algorithm.

4.2.1. FAIR-Bernoulli and FAIR-betaSF with baselines

Variance reduction methodology relying on baseline functions is commonly used in reinforcement learning algorithms. It is shown that introducing baseline in loss function does not introduce additional bias into the model thanks to the fact that it

Algorithm 1 Probabilistic FAIR with score function

Input: learning rates $\gamma_\theta, \gamma_\phi, \gamma_\psi$, dataset D, hyperparameter α, probabilistic model $\mathcal{P}$ of instance weights
Output: parameters θ, ϕ, ψ

Initialize θ, ϕ, ψ
while not converged **do**
  Sample a mini-batch $B \subseteq D$
  Sample $w_\mathbf{x} \sim \mathcal{P}(f_\theta(\mathbf{x}))$ for each $\mathbf{x}$ in B
  $d_\theta \leftarrow \gamma_\theta \frac{1}{|B|} \sum_{(\mathbf{x},\mathbf{y},\mathbf{s})\in B} \big[w_\mathbf{x} \nabla_\theta \log P_\theta(w_\mathbf{x}|\mathbf{x}) \cdot$
    $(\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x}))\big]$
  $d_\phi \leftarrow \gamma_\phi \nabla_\phi \mathcal{L}_\alpha^\mathcal{P}(\theta, \phi, \psi, B)$
  $d_\psi \leftarrow -\gamma_\psi \nabla_\psi \mathcal{L}_\alpha^\mathcal{P}(\theta, \phi, \psi, B)$
  $(\theta, \phi, \psi) \leftarrow (\theta, \phi, \psi) - (d_\theta, d_\phi, d_\psi)$

holds $\mathbb{E}_{P_\theta(w|\mathbf{x})}[\nabla_\theta \log P_\theta(w|\mathbf{x}) b(\mathbf{x})] = 0$ [35]. The goal of baseline network is to reduce the variance of estimates of gradient with respect to θ_g obtained by score function in FAIR-Bernoulli and FAIR-betaSF.

Keeping in mind that the variance can be represented as $\mathrm{Var}[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2$ (where squaring of a vector v means $v^T v$), the baseline loss can simplified due to the already stated fact that $\mathbb{E}_{P_\theta(w|\mathbf{x})}[\nabla_\theta \log P_\theta(w|\mathbf{x}) b_\mu(\mathbf{x})] = 0$:

$$\mathcal{L}_\alpha(\mu) = \mathbb{E}\Big[\Big(\nabla_\theta \log P_\theta(w|\mathbf{x}) \cdot w \cdot (\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x})) - b_\mu(\mathbf{x})\Big)^2\Big]$$

where the expectation is with respect to $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \sim P(\mathbf{x}, \mathbf{y}, \mathbf{s})$ and $w \sim P(w|\mathbf{x})$. Furthermore, we assumed the independence among the values involved in the expectation, and thus the expectation can be represented as:

$$\mathcal{L}_\alpha(\mu) = \mathbb{E}\Big[(\nabla_\theta \log P_\theta(w|\mathbf{x}))^2\Big] \cdot \mathbb{E}\Big[\Big(w \cdot (\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x})) - b_\mu(\mathbf{x})\Big)^2\Big]$$

Considering that the first factor is constant with respect to b_μ it can be omitted, so that the final form of baseline loss $\mathcal{L}_\alpha(\mu)$ is:

$$\mathbb{E}[\big(w \cdot (\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x})) - b_\mu(\mathbf{x})\big)^2]$$

Then, the gradient $\nabla_\mu \mathcal{L}_\alpha(\mu)$ is:

$$-\mathbb{E}[w \nabla_\mu b_\mu(\mathbf{x}) \cdot \big(\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x}) - b_\mu(\mathbf{x})\big)]$$

In Fig. 2 graphical representation of FAIR-betaSF with baseline is shown. The pseudo-code of FAIR-betaSF is presented in Algorithm 2.

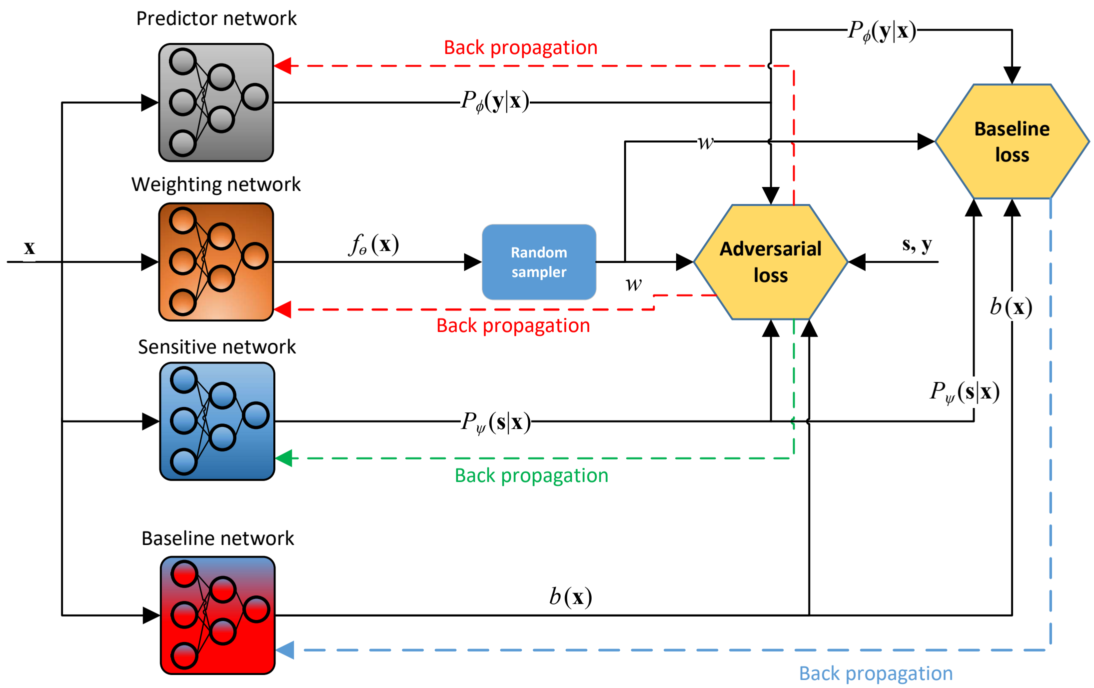

Figure 2: Graphical representations of FAIR-betaSF model with baseline

4.3. Analysis of model properties

In order to analyze properties of all our models in a uniform manner, we discuss instance weights as real values in the interval $[0, 1]$ and we emphasize dependence of the weight on the instance as $w_{\mathbf{x}}$ without explicating specifics of the dependence. Vector of all such weights is denoted $\mathbf{w}$ and it is denoted $\mathbf{w}^*$ if it is a part of the optimal solution of the corresponding adversarial problem. In practice, expectations are approximated by sample means (or sums since outmost constant factors are irrelevant in optimization), and losses are regularized. Therefore we consider regularized loss $\mathcal{L}_\alpha(\mathbf{w}, \phi, \psi)$:

$$\begin{aligned} &\sum_{(\mathbf{x},\mathbf{y},\mathbf{s})\in D} w_{\mathbf{x}}[\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x})] \\ &\text{s.t. } \|\theta\|_2^2 + \|\phi\|_2^2 + \|\psi\|_2^2 \leq \lambda \end{aligned} \tag{7}$$

where dependence of $w_{\mathbf{x}}$ on θ is not made explicit, but we are aware that it exists. To shorten the proofs, we formulate regularization in a constraint based manner

Algorithm 2 FAIR-betaSF with baseline

Input: learning rates $\gamma_\theta, \gamma_\phi, \gamma_\psi$, γ_b dataset D, hyperparameter α
Output: parameters θ, ϕ, ψ, μ

Initialize θ, ϕ, ψ, μ
while not converged **do**
  Sample a mini-batch $B \subseteq D$
  $\alpha_{\mathbf{x}}, \beta_{\mathbf{x}} \leftarrow f_\theta(\mathbf{x})$ for each $\mathbf{x} \in B$
  Sample $w_{\mathbf{x}} \sim \beta(\alpha, \beta)$ for each $\mathbf{x} \in B$
  $d_\theta \leftarrow \gamma_\theta \frac{1}{|B|} \sum_{(\mathbf{x},\mathbf{y},\mathbf{s}) \in B} \big[w_{\mathbf{x}} \nabla_\theta \log P_\theta(w_{\mathbf{x}}|\mathbf{x}) \cdot$
    $(\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x}) - b_\mu(\mathbf{x})) \big]$
  $d_\phi \leftarrow \gamma_\phi \nabla_\phi \mathcal{L}^{\mathcal{P}}_\alpha(\theta, \phi, \psi, B)$
  $d_\psi \leftarrow -\gamma_\psi \nabla_\psi \mathcal{L}^{\mathcal{P}}_\alpha(\theta, \phi, \psi, B)$
  $d_\mu \leftarrow -\gamma_\mu \frac{1}{|B|} \sum_{(\mathbf{x},\mathbf{y},\mathbf{s}) \in B} \big[w_{\mathbf{x}} \nabla_\mu b_\mu(\mathbf{x}) \cdot$
    $(\alpha \log P_\psi(\mathbf{s}|\mathbf{x}) - \log P_\phi(\mathbf{y}|\mathbf{x}) - b_\mu(\mathbf{x})) \big]$
  $(\theta, \phi, \psi, \mu) \leftarrow (\theta, \phi, \psi, \mu) - (d_\theta, d_\phi, d_\psi, d_\mu)$

[36], although it is more often formulated and implemented in a mathematically equivalent penalty based manner (note that the meaning of regularization parameter is reversed – in penalty based formulation case $\lambda = 0$ corresponds to infinite value of λ in constraint based formulation).

Now we prove important properties of our model. In a nutshell, hyperparameter α controls the trade-off between fairness and the quality of prediction of the target variable. Extreme case $\alpha = 0$ represents extreme emphasis on fairness and $\alpha \to \infty$ represents extreme emphasis on quality of prediction and disregard for fairness. Please note that a superficial glance at the adversarial problem would suggest vice versa, but we stress that it is not the case. The role of hyperparameter α in FAIR model is the opposite to its role in FAD model.

Lemma 1. *If λ is finite, there exist strictly negative constants c_ϕ, c'_ϕ, c_ψ, and c'_ψ such that it holds $c_\phi \leq \log P_\phi(\mathbf{y}|\mathbf{x}) \leq c'_\phi$ and $c_\psi \leq \log P_\phi(\mathbf{s}|\mathbf{x}) \leq c'_\psi$ for any $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{s}$, and any ϕ and ψ which satisfy regularization condition 7.*

Proof. Denote $\mathcal{B}$ the ball defined by $\|\theta\|_2^2 + \|\phi\|_2^2 + \|\psi\|_2^2 \leq \lambda$, representing the set of feasible solutions of the optimization problem. Denote $\bar{g}_\phi(\mathbf{x})$ the network $g_\phi(\mathbf{x})$ modelling $\mathbf{y}$ with sigmoid function at the output removed and $\bar{h}_\psi(\mathbf{x})$ the network $h_\psi(\mathbf{x})$ modelling $\mathbf{s}$ with sigmoid at the output removed. Since $\mathcal{B}$ is a compact set and $\bar{g}_\phi(\mathbf{x})$ and $\bar{h}_\psi(\mathbf{x})$ are continuous functions, they both attain their finite minimal and maximal values within $\mathcal{B}$. Since $\log P_\psi(\mathbf{s}|\mathbf{x})$ and $\log P_\phi(\mathbf{y}|\mathbf{x})$ are continuous functions of $\bar{h}_\psi(\mathbf{x})$ and $\bar{g}_\phi(\mathbf{x})$, respectively, which map the range of $\bar{h}_\psi$ and $\bar{g}_\phi$ from $(-\infty, \infty)$ to $(-\infty, 0)$, functions $\log P_\psi(\mathbf{s}|\mathbf{x})$ and $\log P_\phi(\mathbf{y}|\mathbf{x})$ attain their strictly negative and finite minimal and maximal values within $\mathcal{B}$. Therefore, the required constants exist, by which the lemma is proven. $\square$

Theorem 1. *If λ is finite, for $\alpha = 0$ it holds $\mathbf{w}^* = \mathbf{0}$.*

Proof. By Lemma 1, $P_\psi(\mathbf{s}|\mathbf{x})$ is bounded, so for $\alpha = 0$ it holds:

$$\begin{aligned}(\mathbf{w}^*, \phi^*, \psi^*) &= \arg\min_{\mathbf{w},\phi} \max_{\psi} \mathcal{L}_\alpha(\mathbf{w}, \phi, \psi) \\ &= \arg\min_{\mathbf{w},\phi} - \sum_{(\mathbf{x},\mathbf{y},\mathbf{s})\in D} w_\mathbf{x} \cdot \log P_\phi(\mathbf{y}|\mathbf{x})\end{aligned}$$

By Lema 1, $\log P_\phi(\mathbf{y}|\mathbf{x})$ is strictly negative, so $-\sum_{(\mathbf{x},\mathbf{y},\mathbf{s})\in D} w_\mathbf{x} \cdot \log P_\phi(\mathbf{y}|\mathbf{x})$ is zero or positive. Therefore its minimal value is 0 for $\mathbf{w} = \mathbf{0}$ regardless of ϕ. Therefore, it holds $\mathbf{w}^* = \mathbf{0}$. $\square$

Theorem 2. *For each instance $(\mathbf{x}, \mathbf{y}, \mathbf{s})$, it holds $w^*_\mathbf{x} = 1$ or $w^*_\mathbf{x} = 0$ or $\alpha \log P_{\psi^*}(\mathbf{s}|\mathbf{x}) = \log P_{\phi^*}(\mathbf{y}|\mathbf{x})$.*

Proof. Consider a partial derivative in the optimal solution:

$$\frac{\partial \mathcal{L}_\alpha}{\partial w_\mathbf{x}}(\mathbf{w}^*, \phi^*, \psi^*) = \alpha \log P_{\psi^*}(\mathbf{s}|\mathbf{x}) - \log P_{\phi^*}(\mathbf{y}|\mathbf{x})$$

If the derivative is negative, then there exists $d > 0$ such that it holds

$$\mathcal{L}_\alpha(\mathbf{w}^* + d\mathbf{e}_\mathbf{x}, \phi^*, \psi^*) < \mathcal{L}_\alpha(\mathbf{w}^*, \phi^*, \psi^*)$$

where $\mathbf{e}_\mathbf{x} = (0, \ldots, 1, \ldots, 0) \in \mathbb{R}^{|D|}$ where 1 is at the coordinate corresponding to $w_\mathbf{x}$. Therefore, if it holds $w^*_\mathbf{x} < 1$, $w^*_\mathbf{x}$ can be increased in order to decrease the loss and $(\mathbf{w}^*, \phi^*, \psi^*)$ is not an optimal solution, which is a contradiction. Therefore, it has to hold $w^*_\mathbf{x} = 1$. If the derivative is positive, $w^*_\mathbf{x} = 0$ is proven in an analogous manner. If the derivative is 0, the theorem holds due to its third case. $\square$

In the following propositions, we explicitly denote dependence of the optimal solution on α.

Lemma 2. *If λ is finite, for each instance $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in D$ it holds*

$$\frac{\partial \mathcal{L}_\alpha}{\partial w_\mathbf{x}}(\mathbf{w}^*_\alpha, \phi^*_\alpha, \psi^*_\alpha) \to -\infty \quad \textit{as} \quad \alpha \to \infty$$

Proof. Consider a partial derivative with respect to $w_{\mathbf{x}}$ in an optimum:

$$\frac{\partial \mathcal{L}_\alpha}{\partial w_{\mathbf{x}}}(\mathbf{w}^*_\alpha, \phi^*_\alpha, \psi^*_\alpha) = \alpha \log P_{\psi^*_\alpha}(\mathbf{s}|\mathbf{x}) - \log P_{\phi^*_\alpha}(\mathbf{y}|\mathbf{x})$$

According to Lemma 1, for any feasible ψ and ϕ there exists constants $c'_\psi < 0$ and c_ϕ such that it holds $\log P_\psi(\mathbf{s}|\mathbf{x}) \leq c'_\psi$ and $\log P_\phi(\mathbf{y}|\mathbf{x}) \geq c_\phi$. Therefore, the first term goes to $-\infty$ as $\alpha \to \infty$ and the second term is bounded, so the limit of the partial derivative is $-\infty$. □

Theorem 3. *If λ is finite, for each instance $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in D$, it holds $w^*_{\mathbf{x},\alpha} \to 1$ as $\alpha \to \infty$.*

Proof. According to Lemma 2, the limit of the values of the partial derivative $\frac{\partial \mathcal{L}_\alpha}{\partial w_{\mathbf{x}}}(\mathbf{w}^*_\alpha, \phi^*_\alpha, \psi^*_\alpha)$ in optima as $\alpha \to \infty$ is negative. Then, by the definition of the limit, there exists $\alpha_0 \in \mathbb{R}$ such that for all $\alpha > \alpha_0$ it holds

$$\frac{\partial \mathcal{L}_\alpha}{\partial w_{\mathbf{x}}}(\mathbf{w}^*_\alpha, \phi^*_\alpha, \psi^*_\alpha) < 0$$

For each such α, since derivative with respect to $w_{\mathbf{x}}$ is negative, by the same argument as in the proof of Theorem 2, it holds $w^*_{\mathbf{x},\alpha} = 1$. Hence, we can conclude that for each $\varepsilon > 0$, there exists α_0 such that for all $\alpha > \alpha_0$ it holds $w^*_{\mathbf{x},\alpha} > 1 - \varepsilon$ (since $w^*_{\mathbf{x},\alpha} = 1$). Therefore, by the definition of the limit, we conclude that it holds $w^*_{\mathbf{x},\alpha} \to 1$ as $\alpha \to \infty$. □

In case of infinite λ, overfitting might falsify our proof of Lemma 1 and in that case for some instance $\mathbf{x}$ it might hold $w^*_{\mathbf{x},\alpha} \to 0$ as $\alpha \to \infty$. However, this suggests an interesting diagnostic property – if for ever larger values of α one obtains $w_{\mathbf{x}} = 0$ for some $\mathbf{x}$, one has reasons to suspect overfitting.

Also note that the model of instance weights need not allow values 0 and 1. Nevertheless, the provided theorems inform us that the gradients will push the weights towards these values. Still, our probabilistic approaches might provide additional regularization by giving nonzero probability to other weight values except the optimal ones.

Provided theorems explain the role of hyperparameter α in our model – it is a threshold on the ratio of instance usefulness and instance fairness based on which the model decides if the instance should be discarded or used for learning. If it holds

$$\frac{\log P_\phi(\mathbf{y}|\mathbf{x})}{\log P_\psi(\mathbf{s}|\mathbf{x})} < \alpha$$

intuitively, the instance is fair enough considering its usefulness. Namely, for the ratio to be low, its predictive usefulness should be high (reflected by small negative value of log likelihood in the numerator) and its unfairness should be low (reflected by the large negative value of log likelihood in the denominator). In the extreme case of $\alpha = 0$ no instance is considered fair enough, since neither the log likelihood in the numerator can be exactly zero, nor the log likelihood in the denominator can be infinite. According to Theorem 1, in that case, all instances are discarded. In the other extreme, according to Theorem 3, as α tends to infinity, fairness is disregarded and all instances are used for learning. For values of α in between some instances are disregarded and some are used.

5. Experimental Evaluation

Datasets. The proposed framework was tested on four datasets, three of which are commonly used benchmarks. Two datasets (German credit and Adult income) comes from the UCI ML repository [37]. To the our knowledge the Hospital readmission dataset was used in this paper for the first time in the context of fairness.

The first, *Adult income* dataset [38] represents a binary classification task of predicting whether an income is greater than 50K dollars. Dataset contains 45,222 instances described by 14 attributes and including sensitive attribute Gender. Due to the constraint of usage of numerical features dummy coding was applied and total 93 attributes were created. The atributes used in dataset describes the individual's education level, age, gender, occupation, workclass, martial-status, relationship, capital loss and etc [39]. Total numbers of instances used in training, validation and testing are 31,655, 6,783 and 6,784, respectively.

The second, we used *Hospital readmission* dataset [40]. It represents a binary classification task where label 1 means that patient is readmitted within 30 days. The dataset consists of 66,994 instances and 931 attributes, including sensitive attribute Gender. Total number of instances used in training, validation and testing are 46,895, 10,049 and 10,050, respectively.

The third dataset, named *Hospital Expenditures*, comes from [41]. It represents a binary classification task of predicting whether a person would have 'high' or 'low' utilization of medical expenditures. The sensitive attribute is Race. Dataset contains 15,830 instances and 133 attributes, after usage of numerical features dummy codding total number of attributes used in this dataset is 138. In addition whereas 11,081, 2,374 and 2,375 instances were used during training, validation and testing, respectively.

As a fourth dataset, we used *German credit* dataset. *German credit* dataset has 1,000 instances where the task is to classify bank account holders into classes good or bad. The total number of attributes used in dataset, after applying one-hot encoding on categorical features is 58, including sensitive attributes. Following [12] definition of fairness for German credit dataset there are two sensitive attributes, one being Gender and other being Age (≥ 25 is considered as privileged class, and ≤ 25 as unprivileged class). Total numbers of instances used in training, validation and testing are 700, 150 and 150, respectively.

Da li je ≥ 25 i < 25 ili je > 25 i ≤ 25? German credit sensitive attribute

Models. The results obtained by FAIR models are compared with seven related and state-of-the-art algorithms FAD, its probabilistic variant (FAD-prob), reweighing preprocessing technique [20] combined with the random forest classifier (Reweighing - RF) and neural networks (Reweighing - NN), disparity impact remover [26] combined with random forest (DI - RF) and neural networks (DI - NN) and prejudice remover [17] (PR). Bearing in mind that all presented models have fairness hypeparameter that affects on trade-offs between fairness and pure performances of classifiers, the datasets are split on train, validation and test set. Note that such hyperparameters do not control models capacity and are not tuned to obtain maximal perfomance (like regularization hyperparameter would be), but is varied in order to illustrate model behaviour for different trade-offs. Model architectures and hyperparameters used in experiments are discussed in AppendixA.

Optimization. The hyperparameters α of FAIR and FAD models were varied in range $[0, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^{2}, 10^{3}]$, whereas in the case of other models hyperparameters were varied in range $[0, 10^{-3}, 10^{-2}, 10^{-1}, 1]$. Moreover, during optimization of FAIR and FAD models early stopping was used. In the early stopping procedure, the min and max objectives of adversarial training procedure on validation set were monitored. In case when there were no improvements in either of these two metrics for given number of epoch, the training procedure is stopped.

Metrics. Classification performance of all presented classifiers is quantified by the area under the ROC curve (AUC). As fairness metrics we use ASD and AEOD defined by Eqs. 1, 2 and 3. It is worth noticing that AUC is calculated for the output (y) and the sensitive attribute (s). Therefore, we present AUC_y and AUC_s for the output and sensitive attribute, respectively. If subscript is omitted, then AUC_y is presented. Also, every presented measure is calculated on the test set.

6. Results and Discussion

Results and discussion section present results obtained using above-presented experimental evaluation.

6.1. Results

The model hyperparameters were firstly tuned on validation set for each model separately. For each model family (e.g., FAD models obtained for different values of hyperparameter α), only models that belong to the Pareto efficient solutions on validation (models that are not dominated by other models in terms of both predictive performance and fairness) [42] are evaluated on test set. Consequently, in this section all presented results were evaluated on test sets and only Pareto efficient solutions were presented. Additional results can be observer in AppendixB.

Firstly, models performances obtained on *Adult income* dataset are illustrated in Fig. 3 by three fairness metrics (**AOD**, **ASD** or **AEOD**) and classification performance ($\mathbf{AUC_y}$). The models with greater AUC score and lower (un)fairness metric (upper left corner of plots) are preferred. It can be observed that FAIR models dominates Pareto optimal solutions with respect to the all fairness metrics. In addition, Reweighing-NN and FAIR-betaREP models dominate the upper left corner of Pareto fronts for AOD and AEOD, whereas the FAIR-scalar dominates the upper left corner of Pareto front for ASD metric. Moreover, in table 1 the Pareto optimal solutions obtained for all three fairness metrics and ($\mathbf{AUC_y}$) is presented. Similarly, it can be concluded that number of FAIR models is larger compared to the other models and FAIR can therefore be considered better then other models.

Secondly, the results obtained on *Hospital readmission* dataset are presented in Fig 4. It can be noticed that only FAIR models exist on Pareto front and consequently all other model are dominated by them. It can be observed that in the case of AOD and ASD metrics FAIR-scalar is the closest to the upper left corner and can therefore be considered better than others. The letter can be also confirmed in the table 2 where only FAIR models exist on overall Pareto front.

Thirdly, models performances obtained on *Hospital expenditures* dataset are illustrated in Fig. 3. It can be observed that the FAIR models dominates Pareto front in all presented metrics. In the case of ASD metric PR model is the closest to the upper left corner of Pareto front, whereas in the case of AOD and AEOD metrics similarly can be concluded for DI-RF model. However in table 3 where overall Pareto front is presented the FAIR models still dominates Pareto front.

Eventually, model performances obtained of *German credit* dataset for age and sex as sensitive attributes are presented in Figs. 7 and 6, respectively. Similarly as in previous datasets, overall Pareto fronts, for age and sex as sensitive attributes, are presented in Tables 4 and 5. It can be observed that in Fig. 6 all models are equally represented, whereas in Fig. 7 FAIR models dominates the Pareto front in all cases. Similar conclusion can be made in the case of overall Pareto fronts that are presented in tables 4 and 5. In table 4 it can be observed that all models are

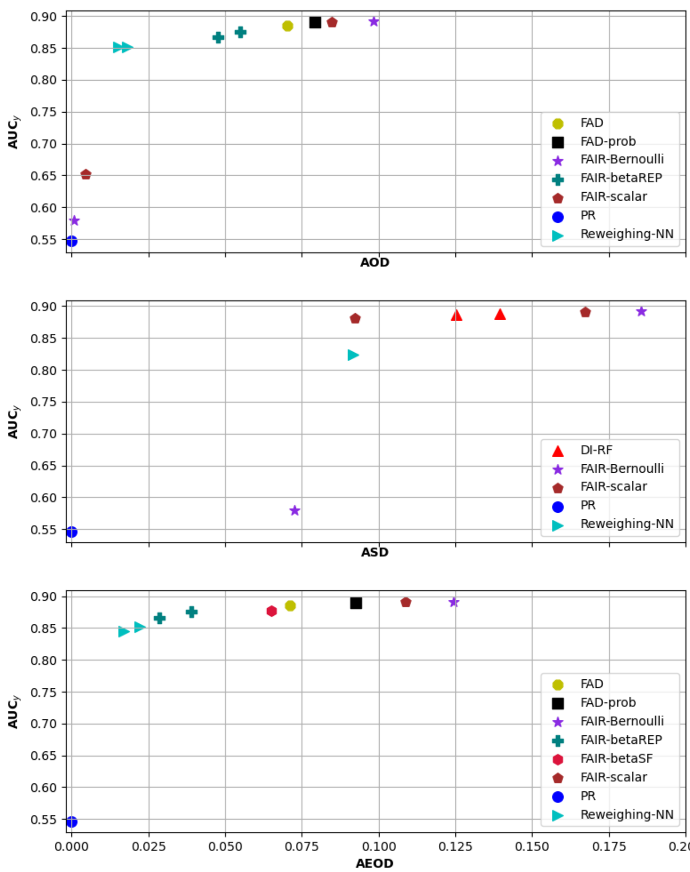

Figure 3: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** on the *Adult income* datasets

Table 1: Pareto optimal solutions - *Adult income* dataset

model	AUC	AOD	ASD	AEOD
PR	0.547	0.000	0.000	0.000
DI-RF	0.887	0.095	0.125	0.155
DI-RF	0.888	0.103	0.139	0.163
Reweighing-NN	0.852	0.018	0.152	0.022
Reweighing-NN	0.849	0.020	0.123	0.005
Reweighing-NN	0.824	0.040	0.092	0.117
Reweighing-NN	0.851	0.015	0.111	0.064
Reweighing-RF	0.888	0.104	0.167	0.143
FAD	0.886	0.070	0.172	0.071
FAIR-scalar	0.652	0.005	0.094	0.052
FAIR-scalar	0.881	0.115	0.092	0.154
FAIR-scalar	0.891	0.085	0.167	0.109
FAIR-betaSF	0.867	0.061	0.146	0.063
FAIR-betaSF	0.878	0.074	0.197	0.065
FAIR-betaSF	0.867	0.055	0.142	0.068
FAIR-Bernoulli	0.580	0.001	0.073	0.075
FAIR-Bernoulli	0.892	0.098	0.185	0.124
FAIR-betaREP	0.875	0.055	0.174	0.039
FAIR-betaREP	0.866	0.048	0.159	0.029
FAD-prob	0.890	0.079	0.172	0.093

Table 2: Pareto optimal solutions - *Hospital readmission* dataset

model	AUC	AOD	ASD	AEOD
FAIR-scalar	0.791	0.002	0.006	0.015
FAIR-betaSF	0.795	0.014	0.008	0.023
FAIR-betaSF	0.789	0.001	0.003	0.001
FAIR-Bernoulli	0.780	0.000	0.000	0.000
FAIR-Bernoulli	0.787	0.000	0.000	0.000
FAIR-Bernoulli	0.788	0.001	0.000	0.002

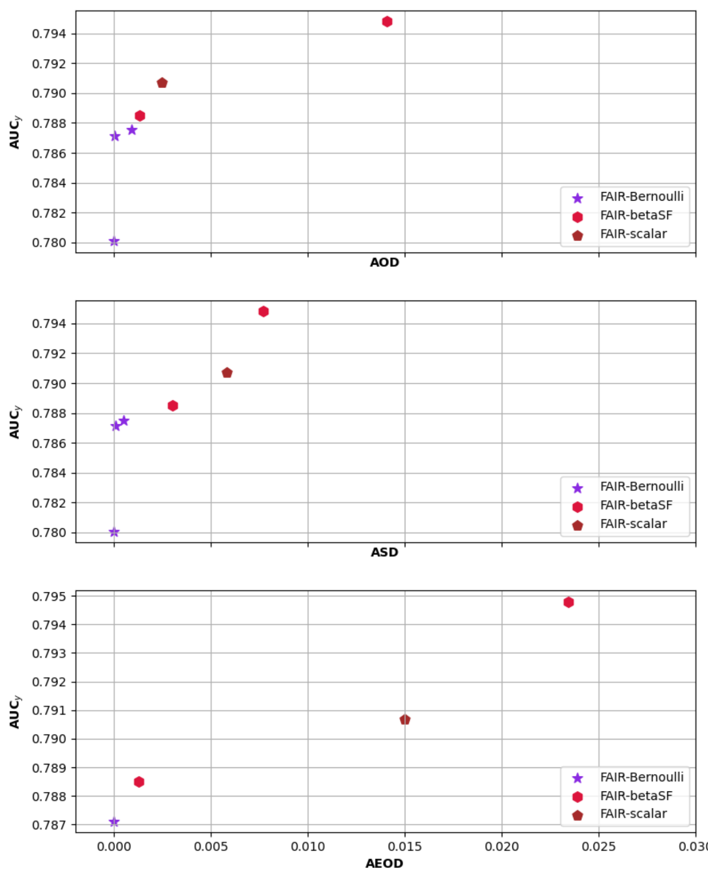

Figure 4: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** on the *Hospital readmission* dataset

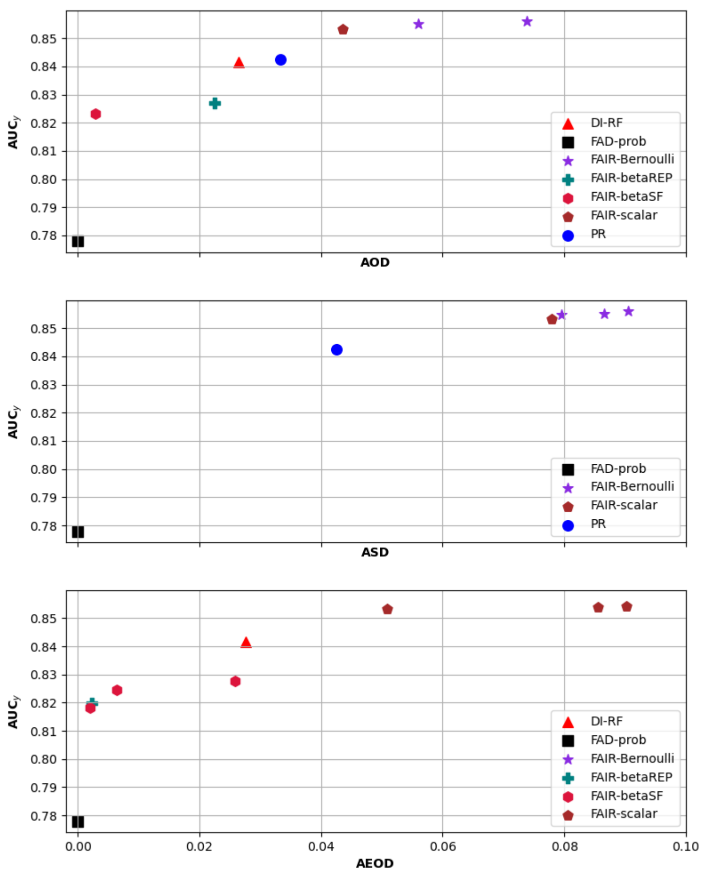

Figure 5: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** *Hospital expenditures* dataset

Table 3: Pareto optimal solutions - *Hospital expenditures* dataset

model	AUC	AOD	ASD	AEOD
PR	0.842	0.033	0.043	0.059
DI-RF	0.840	0.032	0.058	0.043
DI-RF	0.842	0.027	0.059	0.028
Reweighing-RF	0.842	0.040	0.065	0.057
FAIR-scalar	0.853	0.044	0.078	0.051
FAIR-betaSF	0.823	0.003	0.050	0.013
FAIR-betaSF	0.821	0.013	0.052	0.007
FAIR-betaSF	0.828	0.028	0.069	0.026
FAIR-betaSF	0.825	0.015	0.059	0.006
FAIR-betaSF	0.827	0.024	0.065	0.021
FAIR-betaSF	0.818	0.012	0.054	0.002
FAIR-Bernoulli	0.855	0.056	0.087	0.070
FAIR-Bernoulli	0.856	0.074	0.091	0.110
FAIR-Bernoulli	0.855	0.047	0.080	0.056
FAIR-betaREP	0.820	0.012	0.053	0.002
FAIR-betaREP	0.820	0.007	0.052	0.008
FAIR-betaREP	0.827	0.022	0.055	0.027
FAIR-betaREP	0.817	0.011	0.056	0.003
FAIR-betaREP	0.816	0.010	0.057	0.003
FAD-prob	0.778	0.000	0.000	0.000

Table 4: Pareto optimal solutions - *German credit* - age dataset

model	AUC	AOD	ASD	AEOD
PR	0.701	0.114	0.002	0.367
DI-NN	0.769	0.056	0.013	0.067
DI-NN	0.780	0.025	0.054	0.100
DI-NN	0.751	0.042	0.043	0.133
DI-RF	0.727	0.017	0.011	0.033
DI-RF	0.735	0.055	0.100	0.033
Reweighing-NN	0.710	0.033	0.005	0.167
Reweighing-NN	0.768	0.041	0.109	0.000
Reweighing-NN	0.761	0.001	0.087	0.100
FAIR-scalar	0.776	0.159	0.037	0.600
FAIR-scalar	0.787	0.085	0.103	0.033
FAIR-betaSF	0.596	0.000	0.000	0.000
FAIR-Bernoulli	0.765	0.226	0.002	0.567
FAIR-betaREP	0.665	0.003	0.034	0.067
FAIR-betaREP	0.780	0.198	0.037	0.400
FAD-prob	0.752	0.069	0.070	0.000

equally represented in Pareto fronts, whereas in table 5 the most dominant models are Reweighing-RF and FAD.

6.2. Discussion

Model behaviour of FAIR model with respect to change of hyperparameter α is shown in Fig. 8 on *Geman credit* dataset. It can be observed that as α decreases, instances which are unfair (but potentially useful for prediction of target variable) are being discarded, so AUC metrics for both the target variable and sensitive attribute decrease. This is experimental verification of theoretical model properties presented in section 4.

Furthermore, we increased the hyperparameter α in FAIR-scalar model from 0 to the first value where one of the instance in training dataset has weight that tends to 1. Based on theoretical formulation of model properties this is the most "fair" instance in dataset. Moreover, we kept to increase parameter α until the first two instances with weights that tends to 1 in opposite sex and label categories occurred. In table 6 the attributes of previously mentioned instances are presented.

Firstly, it could be observed that the most "fair" instance has good credit score mainly based on facts that he is employed as manager, does not have other debtors

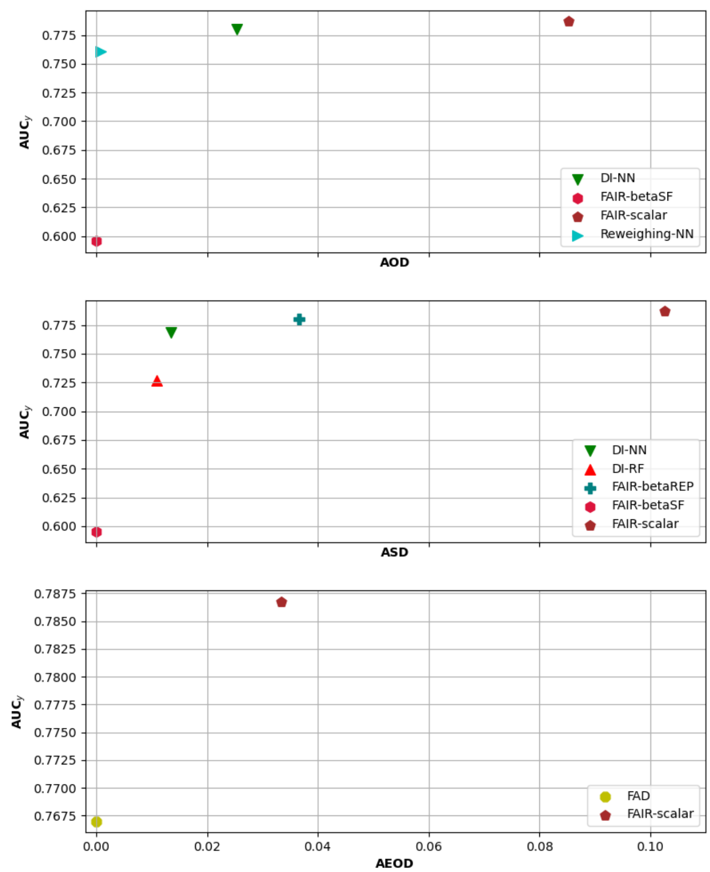

Figure 6: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** *German credit* (age) dataset

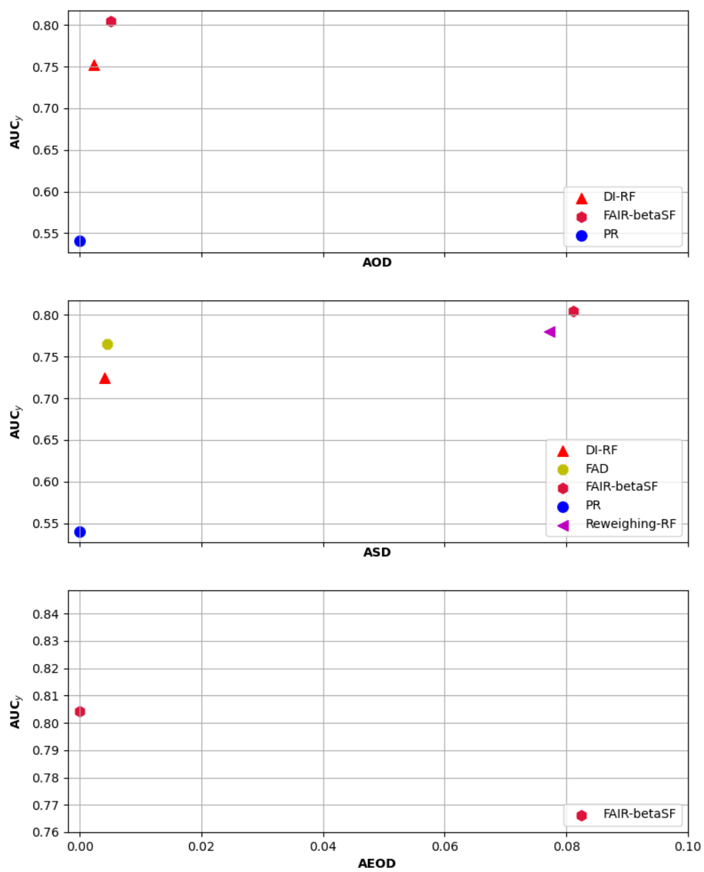

Figure 7: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** *German credit* (sex) dataset

Table 5: Pareto optimal solutions - *German credit* - sex dataset

model	AUC	AOD	ASD	AEOD
PR	0.540	0.000	0.000	0.000
DI-RF	0.727	0.017	0.023	0.033
DI-RF	0.725	0.018	0.004	0.017
Reweighing-NN	0.755	0.002	0.021	0.117
Reweighing-RF	0.739	0.025	0.014	0.050
Reweighing-RF	0.739	0.041	0.047	0.033
Reweighing-RF	0.780	0.018	0.077	0.083
Reweighing-RF	0.763	0.029	0.072	0.067
FAD	0.672	0.013	0.042	0.067
FAD	0.766	0.015	0.005	0.167
FAD	0.547	0.015	0.019	0.283
FAD	0.576	0.014	0.022	0.067
FAIR-betaSF	0.804	0.005	0.081	0.000

and credits taken, posses life insurance and house, is not a foreign worker, has small amount of money on checking account. Similar, attributes can be seen in the case of the first "fair" instance with good credit score that is female. She is not a foreign worker, employed as manager for 7 or more years, paid back duly existing credits and took credit for buying new car. She has small amount of money on checking account and does not have other debtors. Unlike this two instances, the first instance with bad credit score is unemployed man, that has other debtors, is foreign worker, does have house and car. It is obviously that unemployment and other debts has the most influential impact on labelling this instance as bad.

It can be concluded that all presented instances have reasonable explanations why they are labelled with bad or good credit score. Furthermore, it can be seen that sex does not have any kind of cause on final decision so FAIR-scalar successfully labelled them as "fair".

7. Conclusions

We introduced a Fair Adversarial Instance Re-weighting (FAIR) discriminative model, which uses adversarial training to learn instance weights to ensure fairness. We proposed four different variants of the model: a non probabilistic one and three models cast in fully probabilistic framework. In addition, we presented a possibility to introduce baseline to reduce variance of gradient estimation for models based

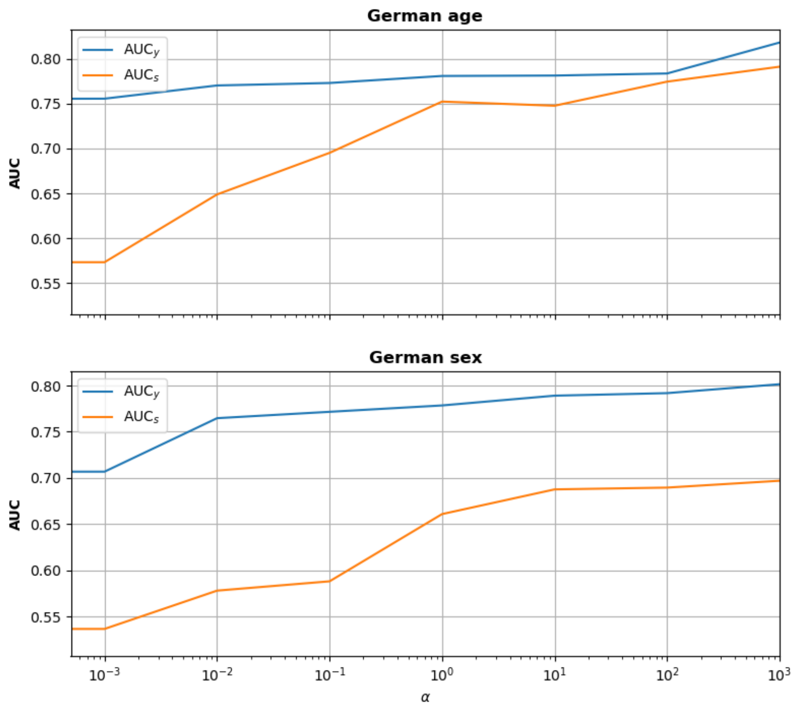

Figure 8: $\mathbf{AUC_y}$ and $\mathbf{AUC_s}$ as functions of the fariness hyperparameter α measured on the German credit - sex and Readmission datasets ($\mathbf{AUC_y}$ is preferred larger, and $\mathbf{AUC_s}$ smaller)

on score function. Theoretical analysis of FAIR model behaviour with respect to the change of the hyperparameter α is given. We proved that changing the value of the hyperparameter controls the trade-off between model fairness and predictive performance. In experimental evaluation on five real-world tasks we demonstrated that our models outperform previous state-of-the-art approaches with respect to fairness metrics and classification performance. Moreover, we showed experimental verification of presented results, and demonstrate that FAIR model is able to find "fair" instances for small values of the hyperparameter α.

Further studies should address extending FAIR models to numerical and categorical values of sensitive attributes and adding additional loss for individual fairness.

Table 6: German credit dataset instances with non zero weights

Credit duration	48	36	36
Credit amount	18424	14318	15857
Investment as income percentage	1	4	2
Residence since	2	2	3
No.of credits taken	1	1	1
No. of people liable to provide maintenance for	1	1	1
Status of checking account	<200 DM	<200 DM	<0DM
Credit history	no credits taken	existing credits paid back duly till now	existing credits paid back duly till now
Purpose	other	car (new)	other
Savings	<100 DM	<100 DM	<100 DM
Employment	1<4 years	>=7 years	unemployed
Other debtors	none	none	co-applicant
Properties	Life insurance	unknown	car or other
Installment plans	bank	none	none
Housing	own	for free	own
Skill level	management	management	self-employed
Telphone	yes, under customer name	yes, under customer name	none
Foreign worker	no	no	yes
Sensitive attribute - Sex	male	female	male
Credit score -Label	Good	Good	Bad

Acknowledgement

This work was supported in part by the ONR/ONR Global under Grant N62909-19-1-2008. In addition, this research is partially supported by the Ministry of Science, Education and Technological Development of the Republic of Serbia grants OI174021, TR35011 and TR41008. The authors would like to express gratitude to company Saga d.o.o Belgrade, for supporting this research.

[1] A. Kumar, A. Kaur, M. Kumar, Face detection techniques: a review, Artificial Intelligence Review 52 (2) (2019) 927–948.

[2] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis, Deep learning for computer vision: A brief review, Computational intelligence and neuroscience 2018 (2018).

[3] S. P. Singh, A. Kumar, H. Darbari, L. Singh, A. Rastogi, S. Jain, Machine translation using deep learning: An overview, in: 2017 International Conference on Computer, Communications and Electronics (Comptelix), IEEE, 2017, pp. 162–167.

[4] M. K. Domadiya, M. V. Gamit, M. K. Patel, A review on face detection and expression recognition (2019).

[5] R. P. Bunker, F. Thabtah, A machine learning framework for sport result prediction, Applied computing and informatics 15 (1) (2019) 27–33.

[6] S. Hajian, F. Bonchi, C. Castillo, Algorithmic bias: From discrimination discovery to fairness-aware data mining, in: Proceedings of the 22nd ACM SIGKDD

international conference on knowledge discovery and data mining, ACM, 2016, pp. 2125–2126.

[7] N. Innocenti, Mining the pay gap: Compensation inequality still exists, Law Prac. 42 (2016) 56.

[8] Y. Li, Credit risk prediction based on machine learning methods, in: 2019 14th International Conference on Computer Science & Education (ICCSE), IEEE, 2019, pp. 1011–1013.

[9] K. Boyd, D. Teres, J. Rapoport, S. Lemeshow, The relationship between age and the use of dnr orders in critical care patients: Evidence for age discrimination, Archives of Internal Medicine 156 (16) (1996) 1821–1826.

[10] P. T. Kim, Data-driven discrimination at work, Wm. & Mary L. Rev. 58 (2016) 857.

[11] X. Wang, H. Huang, Approaching machine learning fairness through adversarial network, arXiv preprint arXiv:1909.03013 (2019).

[12] F. Kamiran, A. Karim, X. Zhang, Decision theory for discrimination-aware classification, in: 2012 IEEE 12th International Conference on Data Mining, IEEE, 2012, pp. 924–929.

[13] F. Calmon, D. Wei, B. Vinzamuri, K. N. Ramamurthy, K. R. Varshney, Optimized pre-processing for discrimination prevention, in: Advances in Neural Information Processing Systems, 2017, pp. 3992–4001.

[14] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, K. P. Gummadi, Fairness constraints: A flexible approach for fair classification., Journal of Machine Learning Research 20 (75) (2019) 1–42.

[15] T. Adel, I. Valera, Z. Ghahramani, A. Weller, One-network adversarial fairness, in: Thirty-Third AAAI Conference on Artificial Intelligence, 2019.

[16] L. E. Celis, L. Huang, V. Keswani, N. K. Vishnoi, Classification with fairness constraints: A meta-algorithm with provable guarantees, in: Proceedings of the Conference on Fairness, Accountability, and Transparency, 2019, pp. 319–328.

[17] T. Kamishima, S. Akaho, H. Asoh, J. Sakuma, Fairness-aware classifier with prejudice remover regularizer, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2012, pp. 35–50.

[18] M. Hardt, E. Price, N. Srebro, Equality of opportunity in supervised learning, in: Advances in neural information processing systems, 2016, pp. 3315–3323.

[19] G. Pleiss, M. Raghavan, F. Wu, J. Kleinberg, K. Q. Weinberger, On fairness and calibration, in: Advances in Neural Information Processing Systems, 2017, pp. 5680–5689.

[20] F. Kamiran, T. Calders, Data preprocessing techniques for classification without discrimination, Knowledge and Information Systems 33 (1) (2012) 1–33.

[21] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in neural information processing systems, 2014, pp. 2672–2680.

[22] C. Wadsworth, F. Vera, C. Piech, Achieving fairness through adversarial learning: an application to recidivism prediction, arXiv preprint arXiv:1807.00199 (2018).

[23] S. Barocas, A. D. Selbst, Big data's disparate impact, Calif. L. Rev. 104 (2016) 671.

[24] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, S. Choudhary, E. P. Hamilton, D. Roth, A comparative study of fairness-enhancing interventions in machine learning, in: Proceedings of the Conference on Fairness, Accountability, and Transparency, ACM, 2019, pp. 329–338.

[25] S. Corbett-Davies, S. Goel, The measure and mismeasure of fairness: A critical review of fair machine learning, arXiv preprint arXiv:1808.00023 (2018).

[26] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, S. Venkatasubramanian, Certifying and removing disparate impact, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 259–268.

[27] E. Krasanakis, E. Spyromitros-Xioufis, S. Papadopoulos, Y. Kompatsiaris, Adaptive sensitive reweighting to mitigate bias in fairness-aware classification, in: Proceedings of the 2018 World Wide Web Conference, International World Wide Web Conferences Steering Committee, 2018, pp. 853–862.

[28] C. Louizos, K. Swersky, Y. Li, M. Welling, R. Zemel, The variational fair autoencoder, arXiv preprint arXiv:1511.00830 (2015).

[29] B. H. Zhang, B. Lemoine, M. Mitchell, Mitigating unwanted biases with adversarial learning, in: Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, ACM, 2018, pp. 335–340.

[30] D. Madras, E. Creager, T. Pitassi, R. Zemel, Learning adversarially fair and transferable representations, arXiv preprint arXiv:1802.06309 (2018).

[31] C. Tan, J. Tang, J. Sun, Q. Lin, F. Wang, Social action tracking via noise tolerant time-varying factor graphs, in: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, 2010, pp. 1049–1058.

[32] D. Koller, N. Friedman, Probabilistic graphical models: principles and techniques, MIT press, 2009.

[33] D. P. Kingma, M. Welling, et al., An introduction to variational autoencoders, Foundations and Trends® in Machine Learning 12 (4) (2019) 307–392.

[34] A. Shah, D. Knowles, Z. Ghahramani, An empirical study of stochastic variational inference algorithms for the beta bernoulli process, in: International Conference on Machine Learning, 2015, pp. 1594–1603.

[35] R. S. Sutton, A. G. Barto, Reinforcement learning: An introduction, MIT press, 2018.

[36] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society: Series B (Methodological) 58 (1) (1996) 267–288.

[37] A. Frank, A. Asuncion, et al., Uci machine learning repository, 2010, URL http://archive.ics.uci.edu/ml 15 (2011) 22.

[38] R. Kohavi, Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid., in: Kdd, Vol. 96, 1996, pp. 202–207.

[39] D. Dua, C. Graff, UCI machine learning repository (2017).
URL http://archive.ics.uci.edu/ml

[40] G. Stiglic, P. P. Brzan, N. Fijacko, F. Wang, B. Delibasic, A. Kalousis, Z. Obradovic, Comprehensible predictive modeling using regularized logistic regression and comorbidity based features, PloS one 10 (12) (2015).

[41] R. K. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilović, et al., Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias, IBM Journal of Research and Development 63 (4/5) (2019) 4–1.

[42] R. T. Marler, J. S. Arora, Survey of multi-objective optimization methods for engineering, Structural and multidisciplinary optimization 26 (6) (2004) 369–395.

AppendixA. Model architecture

In all experiments Reweighing - RF and DI - RF were used with 500 trees in random forest algorithm. Architectures, learning rates and maximum number of epochs used in models with neural networks for all datasets are presented in Table A.7, A.8 and A.9. Architecture, no. of epochs in early stopping procedure and learning rate were empirically determined as to optimize the performance of each model. In all models Adam was used as optimization algorithm.

AppendixB. Additional results

More detailed results of experimental evaluation are given in Figs. B.9, B.10, B.11, B.12 and B.13 by presenting Pareto fronts with all dominated and non-dominated models.

Table A.7: Architectures of models used

Model	No. of units per layer $P_\theta(w\|\mathbf{x})$ or $P_\theta(\mathbf{z}\|\mathbf{x})$	No. of units per layer $P_\phi(\mathbf{y}\|\mathbf{x})$	No. of cells per layer $P_\psi(\mathbf{y}\|\mathbf{x})$	Activation	Early stopping epoch / learning rate
Adult					
DI - NN	-	62/41/27/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
Reweighing - NN	-	62/41/27/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
FAD	62/41/27	18/12/1	18/12/1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-4}$
FAD-prob	46/23/23/23	11/1	11/1	ReLU + sigmoid (last layer)	$50/10^{-4}$
FAIR-scalar	62/41/27/1	62/41/1	62/1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-4}$
FAIR-betaSF	62/41/27/2	62/41/1	62/1	ReLU + Batch normalization + sigmoid or exp (last layer)	$50/10^{-4}$
FAIR-betaREP	62/41/27/2	62/41	62/1	ReLU + Batch normalization + sigmoid or exp (last layer)	$50/10^{-5}$
FAIR-Bernoulli	62/41/1	62/41/27/1	62/1	ReLU + Batch normalization + sigmoid (last layer)	$500/10^{-4}$
Readmission					
DI - NN	-	464/232/116/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
Reweighing - NN	-	464/232/116/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
FAD	464/232/116	58/1	58/1	ReLU + Batch normalization + sigmoid (last layer)	$40/10^{-4}$
FAD-prob	464/232/232/232	116/58/1	116/58/1	ReLU + sigmoid (last layer)	$40/10^{-4}$
FAIR-scalar	464/232/1	464/1	464/1	ReLU + Batch normalization + sigmoid (last layer)	$40/10^{-4}$
FAIR-betaSF	464/232/2	464/1	464/1	ReLU + Batch normalization + sigmoid or exp (last layer)	$40/10^{-5}$
FAIR-betaREP	464/232/2	464/1	464/1	ReLU + Batch normalization + sigmoid or exp (last layer)	$40/10^{-5}$
FAIR-Bernoulli	464/1	464/232/1	464/1	ReLU + Batch normalization + sigmoid (last layer)	$40/10^{-4}$

Table A.8: Architectures of models used

Model	No. of units per layer $P_\theta(w\|\mathbf{x})$ or $P_\theta(\mathbf{z}\|\mathbf{x})$	No. of units per layer $P_\phi(\mathbf{y}\|\mathbf{x})$	No. of cells per layer $P_\psi(\mathbf{y}\|\mathbf{x})$	Activation	Early stopping epoch / learning rate
Medical expenditures					
DI - NN	-	91/60/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
Reweighing - NN	-	91/60/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
FAD	68/34/17	8/1	8/1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-4}$
FAD-prob	68/34/34/34	17/1	17/1	ReLU + sigmoid (last layer)	$50/10^{-4}$
FAIR-scalar	68/34/1	68/1	68/1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-4}$
FAIR-betaSF	68/34/2	68/1	68/1	ReLU + Batch normalization + sigmoid or exp (last layer)	$50/10^{-5}$
FAIR-betaREP	68/34/2	68/1	68/1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-5}$
FAIR-Bernoulli	68/1	68/34/1	68/1	ReLU + Batch normalization + sigmoid or exp (last layer)	$50/10^{-4}$
German credit - sex					
DI - NN	-	37/24/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
Reweighing - NN	-	37/24/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
FAD	37/24/1	16/1	16/1	ReLU + Batch normalization + sigmoid (last layer)	$60/10^{-4}$
FAD-prob	28/14/14/14	7/1	7/1	ReLU + sigmoid (last layer)	$50/10^{-5}$
FAIR-scalar	37/1	1	1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-4}$
FAIR-betaSF	37/2	1	1	ReLU + Batch normalization + sigmoid or exp (last layer)	$60/10^{-5}$
FAIR-betaREP	37/2	1	1	ReLU + Batch normalization + sigmoid or exp (last layer)	$60/10^{-5}$
FAIR-Bernoulli	37/1	1	1	ReLU + Batch normalization + sigmoid (last layer)	$60/10^{-4}$

Table A.9: Architectures of models used

Model	No. of units per layer $P_\theta(w\|\mathbf{x})$ or $P_\theta(\mathbf{z}\|\mathbf{x})$	No. of units per layer $P_\phi(\mathbf{y}\|\mathbf{x})$	No. of cells per layer $P_\psi(\mathbf{y}\|\mathbf{x})$	Activation	Early stopping epoch / learning rate
German credit - age					
DI - NN	-	37/24/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
Reweighing - NN	-	37/24/1	-	ReLU + Batch normalization + sigmoid (last layer)	$10/10^{-3}$
FAD	37/24/1	16/1	16/1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-4}$
FAD-prob	28/14/14/14	7/1	7/1	ReLU + sigmoid (last layer)	$50/10^{-5}$
FAIR-scalar	37/1	37/1	1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-4}$
FAIR-betaSF	37/2	37/1	1	ReLU + Batch normalization + sigmoid or exp (last layer)	$50/10^{-5}$
FAIR-betaREP	37/2	37/1	1	ReLU + Batch normalization + sigmoid or exp (last layer)	$50/10^{-5}$
FAIR-Bernoulli	37/1	37/1	1	ReLU + Batch normalization + sigmoid (last layer)	$50/10^{-4}$

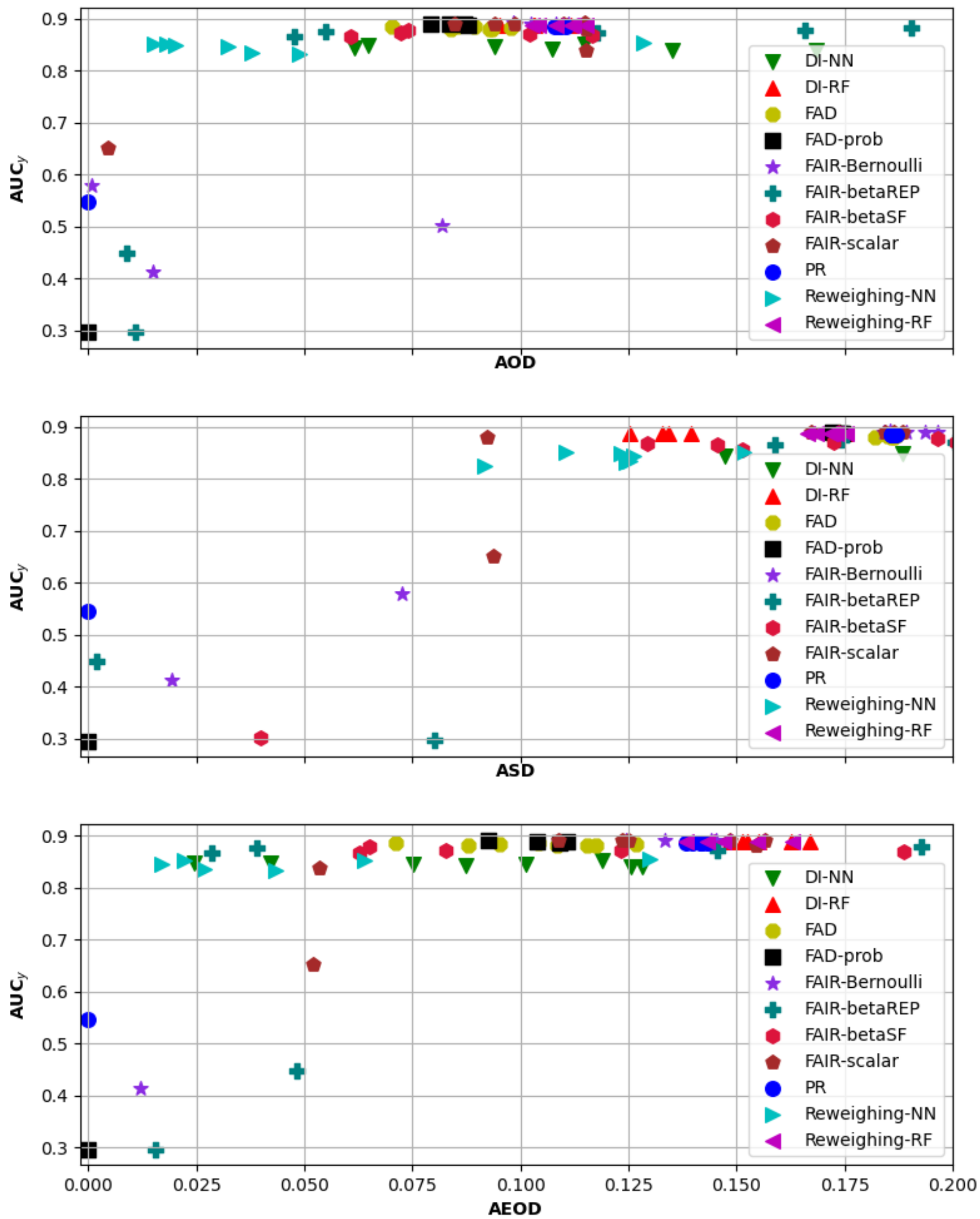

Figure B.9: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** on the *Adult income* dataset

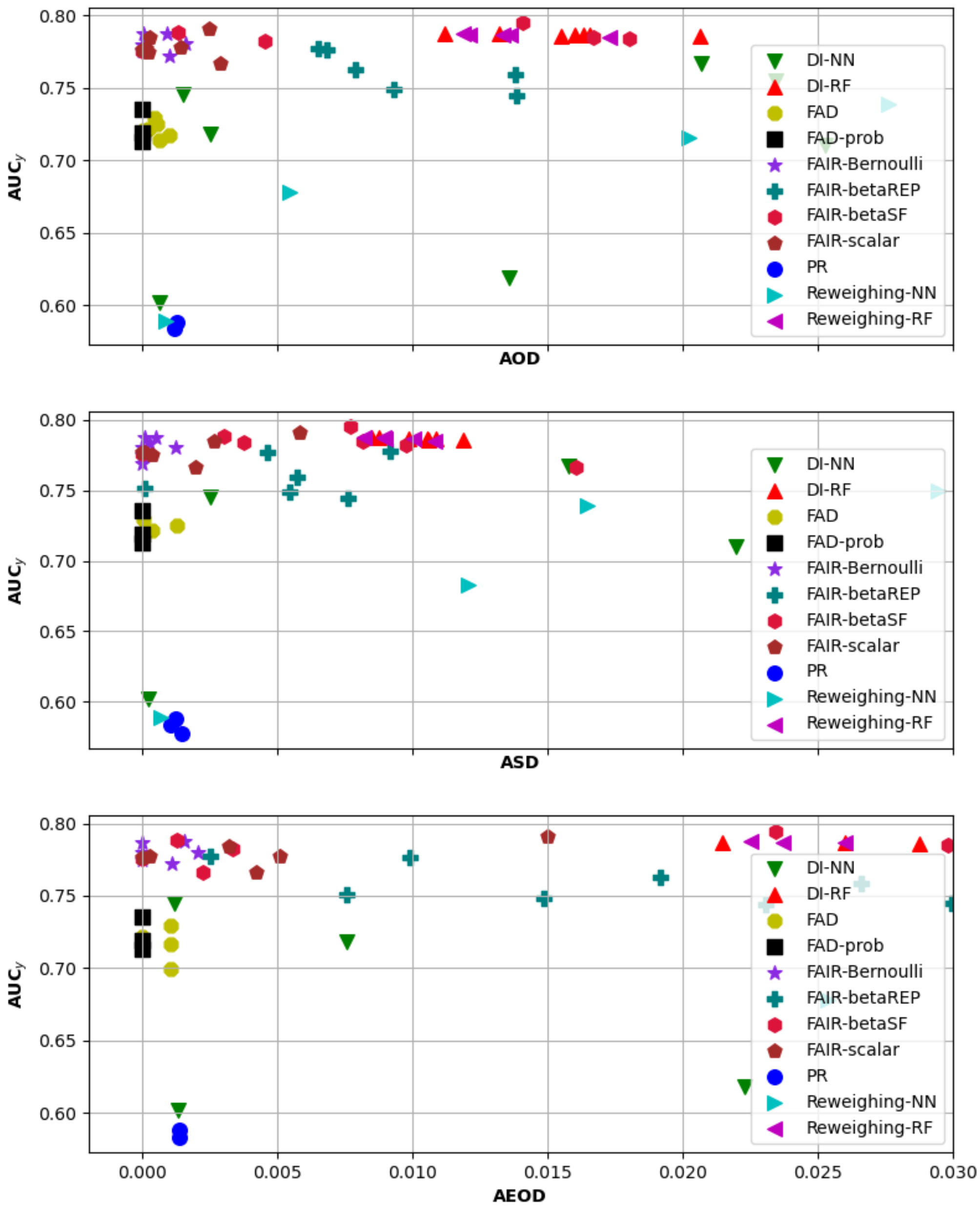

Figure B.10: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** on the *Hospital readmission* dataset

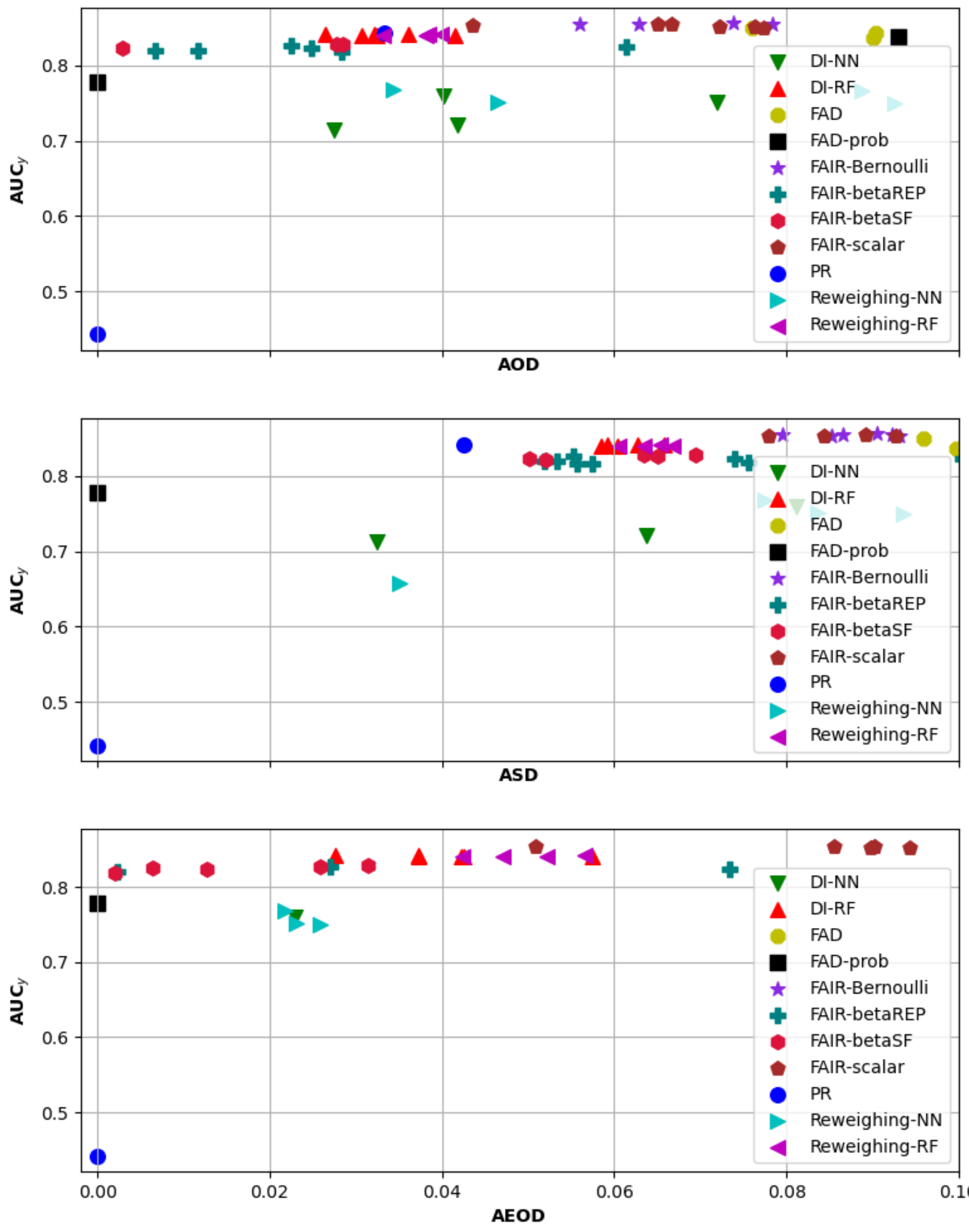

Figure B.11: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** on the *Hospital expenditures* dataset

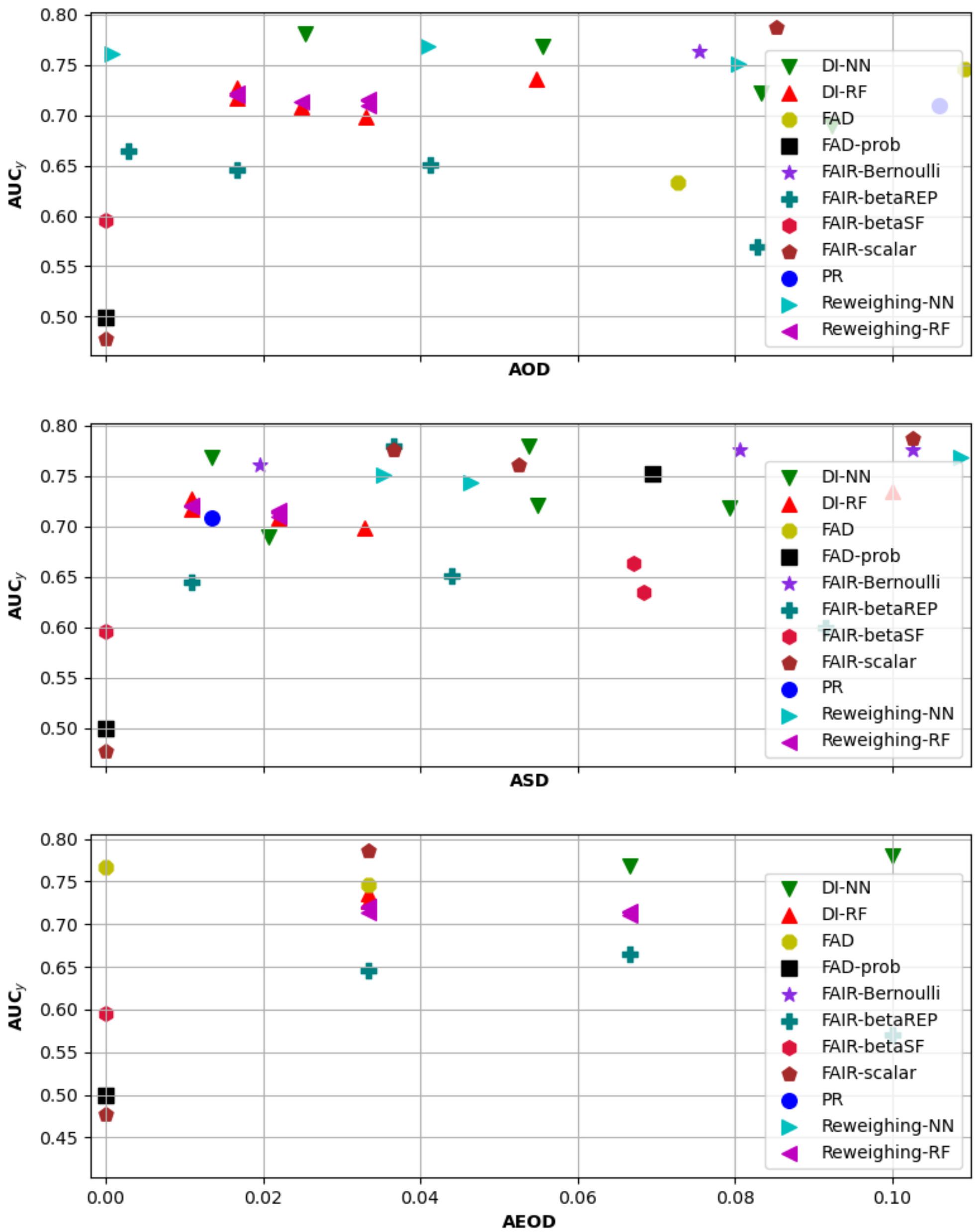

Figure B.12: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** on the *German credit* (age) dataset

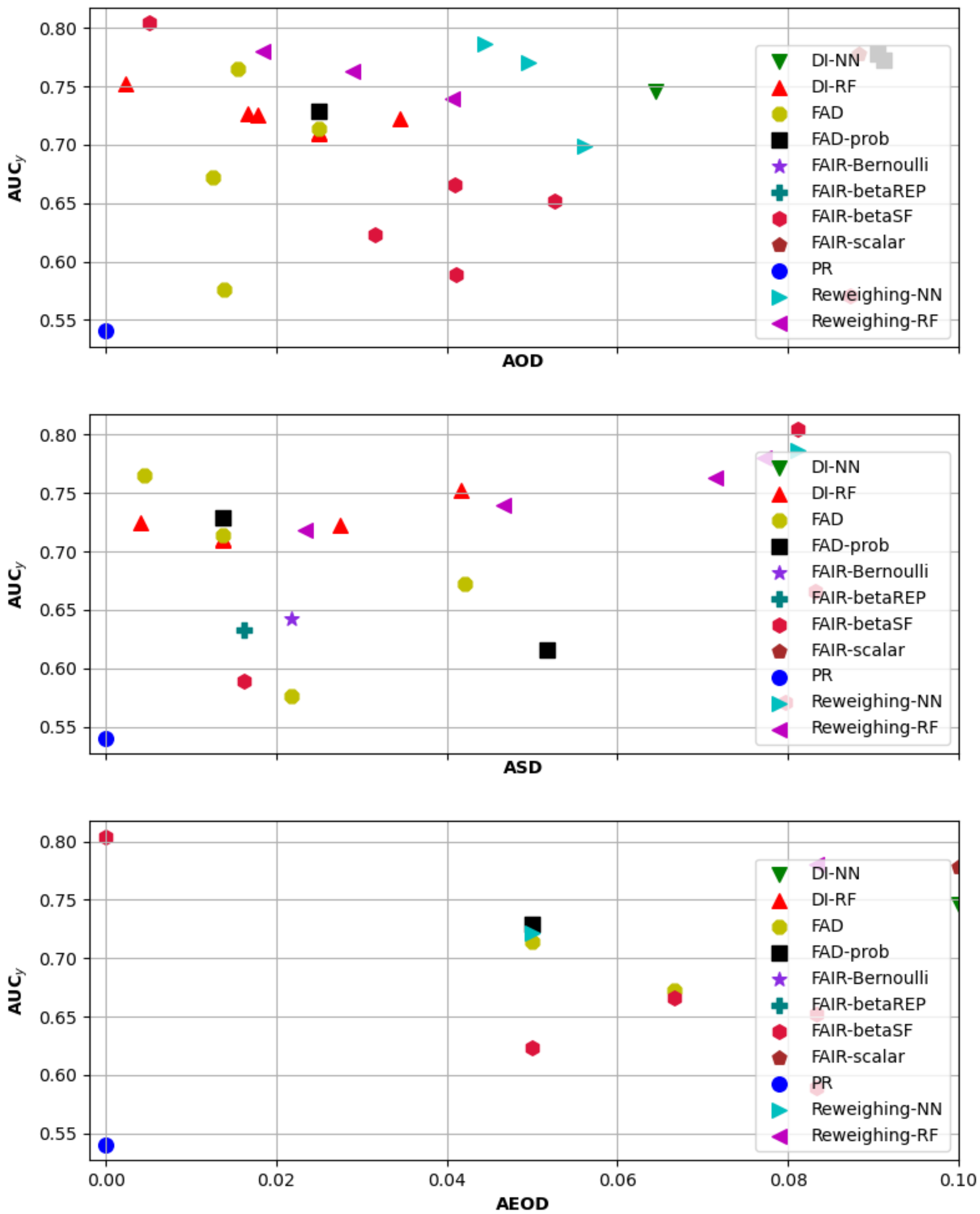

Figure B.13: Classification performance and fairness of models as measured by $\mathbf{AUC_y}$ and **AOD**, **ASD** or **AEOD** on the *German credit* (sex) dataset