Fairness guarantee in multi-class classification

Christophe Denis, Romuald Elie, Mohamed Hebiri, François Hu

Fairness guarantee in multi-class classification

Christophe Denis[1], Romuald Elie[1],

Mohamed Hebiri[1], and François Hu[2]

[1]Université Gustave Eiffel, [2]ENSAE-CREST

May 3, 2022

Abstract

Algorithmic Fairness is an established area of machine learning, willing to reduce the influence of biases in the data. Yet, despite its wide range of applications, very few works consider the multi-class classification setting from the fairness perspective. We extend both definitions of exact and approximate fairness in the case of *Demographic Parity* to multi-class classification. We specify the corresponding expressions of the optimal fair classifiers. This suggests a plug-in data-driven procedure, for which we establish theoretical guarantees. The enhanced estimator is proved to mimic the behavior of the optimal rule both in terms of fairness and risk. Notably, fairness guarantees are distribution-free. The approach is evaluated on both synthetic and real datasets and turns out to be very effective in decision making with a preset level of unfairness. In addition, our method is competitive with the state-of-the-art in-processing fairlearn in the specific binary classification setting.

1 Introduction

Algorithmic fairness has become very popular during the last decade [1, 2, 4, 5, 7, 8, 13, 23, 30, 32] because it helps addressing an important social problem: mitigating historical bias contained in the data. This is a crucial issue in many applications such as loan assessment, health care, or even criminal sentencing. The common objective in algorithmic fairness is to reduce the influence of a sensitive attribute on a prediction. Several notions of fairness have already been considered in the literature for the binary classification problem [4, 31]. All of them impose some independence condition between the sensitive feature and the prediction. In some applications (e.g. loan agreement), this independence is desired on some or all values of the label space, see *Equality of odds* or *Equal opportunity* [19]. In this paper, we focus on the well established *Demographic Parity* (DP) [5] that requires the independence between the sensitive feature and the prediction function, while not relying on labels. DP has a recognized interest in various applications, such as for loan agreement without gender attributes or for crime prediction without ethnicity discrimination [3, 14, 18, 22].

All the previously mentioned references focus either on the regression or the binary classification frameworks. However, many (modern) applications fall within the scope of multi-class classification, *e.g.,* image recognition, text categorization. Moreover, most real world applications can be tackled from the multi-class perspective. For instance, criminal recidivism is often treated as a binary problem, while it may be more suitable to distinguish between the different stratum of the problem

and provide thinner description of the criminal behavior. However, extension of previous works to the multi-class setting is tricky, in particular since the adequate notion of fairness in this framework is not clearly defined and should be handled with caution.

Up to our knowledge, imposing fairness constraint in the multi-class problem has only been briefly discussed in [29], while focusing on Support Vector Machine (SVM) fair prediction. However, their fairness approach relies on properly selecting a subset of the data that is unbiased from the fairness perspective, and hereby differs significantly from the one presented here. We aim at enforcing fairness using the dataset as a whole! Besides, from a high-level perspective, the procedure described in [29] chooses to impose fairness on each component of the score function. It is clear that such methodology can be generalized to any convex empirical risk minimization (ERM) problem such as SVM or quadratic risk. However, since the decision rule in the multi-class setting relies on the maximizer over scores, we do not adopt this quite unnatural approach and rather directly impose fairness on the maximizer itself. Our main contributions are the following:

- We extend DP notion of exact and approximate fairness to the multi-class classification problem;
- We give optimal solutions for the multi-class classifier problem under exact or approximate DP constraints;
- We build a data-driven procedure that mimics the performance of the optimal rule both in terms of risk and fairness. Notably, our fairness guarantees are *distribution-free*;
- The robustness of our approach is illustrated on synthetic datasets with various bias levels, as well as on several real datasets. It proves to be very effective for decision making with a preset level of unfairness.

Related works. There are mainly three ways to build fair prediction: i) *pre-processing* methods mitigate bias in the data before applying classical Machine Learning algorithms; ii) *in-processing* methods reduce bias during training; iii) *pro-processing* methods enforce fairness after fitting. The present work falls within the last category. In a related study, [8] exhibit fair binary classifiers under *Equal Opportunity* constraints. In contrast, we focus on the multi-class setting, while imposing *DP* constraints.

Another line of works considers algorithmic fairness from an optimal transport perspective [7, 10, 16, 17]. A fair prediction is built upon Wasserstein barycenters of conditional distributions with respect to the sensitive feature. Such argumentation extends with little effort to a multi-class setting, even though the proper fairness definition in this context remains a question to investigate. This approach provides a fair classifier based on empirical risk minimization (ERM) with fairness constraints on each underlying score. However, fairness constraints on scores do not properly translate to fairness at the level of the classifier in the multi-class setting. Hence, we opt in the present work to enforce fairness directly at the level of the classifier.

Up to our knowledge, only few works study fairness in the multi-class setting. As previously detailed, [29] enforces fairness by sub-sample selection and is in-processing. In contrast, we keep the whole sample and enforce fairness in a post-processing manner. The multi-class framework is also considered in [27]. However, the authors do not provide an explicit formulation of the optimal fair rule. Furthermore, their theoretical fairness guarantee is not distribution free. Finally, they only consider numerical experiments for binary classification. Our method definitely provides valuable benefits on all these aspects.

Outline of the paper. In the context of *exact* fairness, Section 2 defines DP fairness in the multi-class setting and explicit expression of the optimal fair classifier are provided. The corresponding data-driven procedure together with its statistical guarantees on risk and fairness are presented in Section 3. Section 4 extends the previous study to the case of *approximate fairness*. Section 5 details the algorithm implementation in the general approximate fairness setting, while numerical experiments are provided in Section 6.

2 Exact fairness in multi-class classification

This section focuses on exact fairness in multi-class classification. In particular, the optimal classifier for multi-class classification with *exact* DP constraint is established.

Let (X, S, Y) be a random tuple with distribution $\mathbb{P}$, where $X \in \mathcal{X}$ a subset of $\mathbb{R}^d$, $S \in \mathcal{S} := \{-1, 1\}$, and $Y \in [K] := \{1, \dots, K\}$ with K being a fixed number of classes. The distribution of the sensitive feature S is denoted by $(\pi_s)_{s\in\mathcal{S}}$, and we assume that $\min_{s\in\mathcal{S}} \pi_s > 0$. A classification rule g is a function mapping $\mathcal{X} \times \{-1, 1\}$ onto $[K]$, whose performance is evaluated through the misclassification risk

$$\mathcal{R}(g) := \mathbb{P}\left(g(X, S) \neq Y\right) \quad .$$

For $k \in [K]$, we denote $p_k(X, S) := \mathbb{P}\left(Y = k|X, S\right)$. Recall that a Bayes classifier minimizing the misclassification risk $\mathcal{R}(\cdot)$ over the set $\mathcal{G}$ of all classifiers is given by

$$g^*(x, s) \in \arg\max_k p_k(x, s) \ , \quad \text{for all } (x, s) \in \mathcal{X} \times \mathcal{S} \ .$$

2.1 Multi-class classification with demographic parity

We consider here DP constraint [5], that requires independence of the prediction function from the sensitive feature S. Let first extend this notion to multi-class classification in the case of hard (exact fairness) constraints

Definition 2.1 (Exact Demographic Parity)**.** *A classifier $g \in \mathcal{G}$ is* exactly *fair (denoted $g \in \mathcal{G}_{\text{fair}}$) with respect to the distribution $\mathbb{P}$ on $\mathcal{X} \times \mathcal{S} \times [K]$ if, for each $k \in [K]$,*

$$\mathbb{P}\left(g(X, S) = k|S = 1\right) = \mathbb{P}\left(g(X, S) = k|S = -1\right) \quad .$$

This definition naturally extends the DP constraint considered in binary classification [2, 7, 16, 21, 25]. When fairness comes into play, two important aspects of a classifier need to be assessed: the misclassification risk $\mathcal{R}(\cdot)$ and the unfairness, quantified as follows.

Definition 2.2 (Unfairness measure)**.** *The unfairness of a classifier $g \in \mathcal{G}$ is quantified by*

$$\mathcal{U}(g) := \max_{k\in[K]} \left|\mathbb{P}\left(g(X, S) = k|S = 1\right) - \mathbb{P}\left(g(X, S) = k|S = -1\right)\right| \quad .$$

Naturally, taking into account the definition above, a classifier g is exactly fair *if and only if $\mathcal{U}(g) = 0$.*

Alternative measures of unfairness could be considered. The maximum can for instance be replaced by a summation over k. While both measures have their advantages, picking the maximum simplifies fairness evaluation in empirical studies.

2.2 Optimal exactly fair classifier

This section provides an explicit formulation of the optimal fair classifiers *w.r.t.* the misclassification risk under DP constraint. An optimal exactly fair classifier g^*_{fair} solves

$$\min_{g \in \mathcal{G}_{\text{fair}}} \mathcal{R}(g) \ .$$

Obtaining an optimal fair classifier requires to properly balance the misclassification risk together with the fairness criterion. For this purpose, let consider the Lagrangian of the above problem and introduce for $\lambda = (\lambda_1, \ldots, \lambda_K) \in \mathcal{R}^K$,

$$\mathcal{R}_\lambda(g) := \mathcal{R}(g) + \sum_{k=1}^{K} \lambda_k \left[\mathbb{P}\left(g(X,S) = k | S = 1\right) - \mathbb{P}\left(g(X,S) = k | S = -1\right) \right] \ . \tag{1}$$

We call this measure *fair-risk* and detail below how minimizing this risk for a properly chosen λ gives the fair classifier g^*_{fair}. We require besides the following technical condition.

Assumption 2.3 (Continuity assumption)**.** *The mapping $t \mapsto \mathbb{P}\left(p_k(X,S) - p_j(X,S) \leq t | S = s\right)$ is assumed continuous, for any $k, j \in [K]$ and $s \in \mathcal{S}$.*

Assumption 2.3 implies that the distribution of the differences $p_k(X,S) - p_j(X,S)$ has no atoms. It is required to derive a closed expression of g^*_{fair}. Although it may sound unusual, it simplifies to the continuity of $t \mapsto \mathbb{P}\left(p_k(X,S) \leq t | S = s\right)$ considered in [8] for the binary case ($K = 2$). These conditions however describe different sets of distributions when $K \geq 3$. Assumption 2.3 is a tailored condition for the multi-class problem.

We are now in a position to provide a characterization of optimal fair classification.

Proposition 2.4. *Let Assumption 2.3 be satisfied and define*

$$\lambda^* \in \arg\min_{\lambda \in \mathbb{R}^K} \sum_{s \in \mathcal{S}} \mathbb{E}_{X|S=s} \left[\max_k \left(\pi_s p_k(X,s) - s\lambda_k \right) \right] \ .$$

Then, $g^*_{\text{fair}} \in \arg\min_{g \in \mathcal{G}_{\text{fair}}} \mathcal{R}(g)$ if and only if $g^*_{\text{fair}} \in \arg\min_{g \in \mathcal{G}} \mathcal{R}_{\lambda^*}(g)$.

In other words, the optimum of the risk $\mathcal{R}(g)$ over the class of fair classifiers also maximizes the fair-risk $\mathcal{R}_{\lambda^*}$. By construction, $\mathcal{R}_{\lambda^*}$ is a risk measure which efficiently balances both classification accuracy and unfairness. Proposition 2.4 directly implies that $\mathcal{R}_{\lambda^*}(g) \geq \mathcal{R}_{\lambda^*}(g^*_{\text{fair}}) = \mathcal{R}(g^*_{\text{fair}}) \geq 0$, for all $g \in \mathcal{G}$. Furthermore, Prop. 2.4 entails a closed form expression of optimal exactly fair classifiers, which is the bedrock of our procedure: any optimal fair classifier is simply maximizing scores, which are obtained by shifting the original conditional probabilities in a optimal manner.

Corollary 2.5. *Under Assumption 2.3, an optimal* exactly fair *classifier is characterized by*

$$g^*_{\text{fair}}(x,s) \in \arg\max_k \left(\pi_s p_k(x,s) - s\lambda^*_k \right), \ (x,s) \in \mathcal{X} \times \mathcal{S}.$$

Remark 2.6. *The binary setting corresponds to the case where $K = 2$ (with label space $\mathcal{Y} = \{0,1\}$). In this specific setting, the fairness constraint reduces to a single constraint and the optimal rule from Corollary 2.5 simplifies as*

$$g^*_{\text{fair}}(x,s) = \mathbb{1}_{\{p_1(x,s) \geq \frac{1}{2} + \frac{s\lambda^*}{2\pi_s}\}}, \quad (x,s) \in \mathcal{X} \times \mathcal{S} \ ,$$

where λ^* is solution in λ of

$$F_1\left(\frac{\lambda+\pi_1}{2\pi_1}\right) = F_{-1}\left(\frac{-\lambda+\pi_{-1}}{2\pi_{-1}}\right) \ ,$$

with $F_s(t) = \mathbb{P}\left(p_1(X,S) \leq t \mid S = s\right)$.

3 Data-driven procedure with statistical guarantees

We now provide a plug-in estimator for the optimal fair classifier g^*_{fair}. This algorithm enjoys strong theoretical guarantees in terms of both fairness and risk. In particular, Section 3.2 exhibits distribution-free exact fairness guarantees.

3.1 Plug-in estimator

The enhanced estimation procedure is in two-steps. We first build estimators of the conditional probabilities $(p_k)_k$. Then, the estimation of parameters λ^* and $(\pi_s)_{s\in\mathcal{S}}$ is considered.

More precisely, our data-driven procedure is semi-supervised as it relies on two independent datasets, one labeled and another unlabeled. The first *labeled* dataset $\mathcal{D}_n = (X_i, S_i, Y_i)_{i=1,\dots,n}$ contains *i.i.d.* samples from the distribution $\mathbb{P}$. It allows to train estimators $(\hat{p}_k)_k$ of the conditional probabilities $(p_k)_k$, *e.g.,* Random Forest, SVM, *etc.* The second *unlabeled* dataset $\mathcal{D}'_N$ contains N *i.i.d.* copies of (X, S). It is used to calibrate fairness at the right level and estimate in particular several quantities such as marginal distributions. Therefore, $\mathcal{D}'_N$ is split in the following way: the *i.i.d.* sample $(S_1, \dots, S_N)$ of sensitive features is used to compute empirical frequencies $(\hat{\pi}_s)_{s\in\mathcal{S}}$ as estimates of $(\pi_s)_{s\in\mathcal{S}}$ (recall that $\pi_s = \mathbb{P}(S = s)$). For $s \in \mathcal{S}$, the number of observations corresponding to $S = s$ is denoted N_s, so that $N_{-1} + N_1 = N$. The feature vectors in $\mathcal{D}'_N$ are denoted $X^s_1, \dots, X^s_{N_s}$ and consist of *i.i.d.* data from the distribution $\mathbb{P}_{X^s}$ of $X|S = s$. All samples are assumed independent.

Remark 3.1. *Classical datasets often only contain labeled samples. Then, our approach requires to split the data into two independent samples $\mathcal{D}_n$ and $\mathcal{D}'_N$, by removing labels in the latter. As illustrated in Appendix D, this splitting step is important to calibrate the right level of fairness.*

We now discuss an important aspect of our procedure. Ones the empirical conditional probabilities $\hat{p}_k(\cdot,\cdot)$ are trained, the theoretical analysis of the risk and the unfairness of the plug-in rule requires continuity conditions on the random variables $\hat{p}_k(X, S)$ (conditional on the learning sample, see Assumption 2.3). However, such property is automatically satisfied whenever perturbing $(\hat{p}_k)_k$ with a 'small' random noise. To be more specific, we introduce $\bar{p}_k(X, S, \zeta_k) := \hat{p}_k(X, S) + \zeta_k$, for a given uniform perturbation ζ_k on $[0, u]$. This perturbation improves the fairness calibration in both theory and practice. Without the pertubation, atoms may appear for the random variables $\hat{p}_k(X, S) - \hat{p}_j(X, S)$ and then no guarantee on the fairness (nor on the risk) can be established. On the other hand, its introduction does not deflate our theoretical study. In particular, our analysis, as in Theorem 3.3, takes the additional perturbation into account.

Let $(\zeta_k)_{k\in[K]}$ and $(\zeta^s_{k,i})$ be independent copies of a Uniform distribution on $[0, u]$. Because of this extra randomness, we call our fair algorithm $\hat{g}$ *randomized exactly fair classifier* and define it by plug-in as

$$\hat{g}(x, s) = \arg\max_{k\in[K]} \left(\hat{\pi}_s \bar{p}_k(x, s, \zeta_k) - s\hat{\lambda}_k\right) , \tag{2}$$

for all $(x, s) \in \mathcal{X} \times \mathcal{S}$, with $\hat{\lambda} \in \mathbb{R}^K$ given as

$$\hat{\lambda} \in \arg\min_{\lambda} \sum_{s \in \mathcal{S}} \frac{1}{N_s} \sum_{i=1}^{N_s} \left[\max_{k \in [K]} \left(\hat{\pi}_s \bar{p}_k(X_i^s, s, \zeta_{k,i}^s) - s\lambda_k \right) \right]. \tag{3}$$

Note that the construction of the plug-in rule $\hat{g}$ relies on (x, s) but also on the perturbations ζ_k and $\zeta_{k,i}^s$ for $k \in [K]$, $i \in N_s$ and $s \in \mathcal{S}$. This additional data points are easy to collect since they are *i.i.d.* uniform random variables.

3.2 Statistical guarantees

We are now in position to derive fairness and consistency guarantees of our plug-in procedure.

Universal exact fairness guarantee. We first focus on fairness assessment and prove that the plug-in estimator $\hat{g}$ is asymptotically exactly fair. The convergence rate on the unfairness to zero is parametric with the number of unlabeled data N. Notably, the fairness guarantee is distribution-free and holds for any estimators of the conditional probabilities.

Theorem 3.2. *There exists a constant $C > 0$ depending only on K and $\min_{s \in \mathcal{S}} \pi_s$, such that, for any estimators $\hat{p}_k$, we have*

$$\mathbb{E}\left[\mathcal{U}(\hat{g})\right] \leq \frac{C}{\sqrt{N}} \ .$$

This result illustrates a key feature of our post-processing approach. It makes (asymptotically) exactly fair any off-the-shelf (unconstrained) estimators of the conditional probabilities. This post-processing step is especially appealing when the cost of re-training an existing learning algorithm is high.

Consistency result. We now provide the consistency of $\hat{g}$ *w.r.t.* the misclassification risk. We define the L_1-norm in $\mathbb{R}^K$ between the estimator $\hat{\mathbf{p}} := (\hat{p}_1, \ldots, \hat{p}_K)$ and the vector of the conditional probabilities $\mathbf{p} := (p_1, \ldots, p_K)$ by $\|\hat{\mathbf{p}} - \mathbf{p}\|_1 = \sum_{k \in [K]} |\hat{p}_k(X, S) - p_k(X, S)|$.

Theorem 3.3. *Let Assumption 2.3 be satisfied, then,*

$$\mathbb{E}[\mathcal{R}_{\lambda^*}(\hat{g})] - \mathcal{R}_{\lambda^*}(g^*_{\text{fair}}) \leq C \left(\mathbb{E}\left[\|\hat{\mathbf{p}} - \mathbf{p}\|_1\right] + \sum_{s \in \mathcal{S}} \mathbb{E}\left[|\hat{\pi}_s - \pi_s|\right] + \mathbb{E}\left[\mathcal{U}(\hat{g})\right] + u \right) \ .$$

The above result highlights that the excess fair-risk of $\hat{g}$ depends on 1) the quality of the estimators of the conditional probabilities through its L_1-risk; 2) the quality of the estimators of $(\pi_s)_{s \in \mathcal{S}}$; 3) the unfairness of the classifier; and 4) the upper-bound u on the regularizing perturbations. In view of Theorem 3.2, $\hat{g}$ is then consistent *w.r.t.* the misclassification risk as soon as the estimator $\hat{\mathbf{p}}$ is consistent in L_1-norm.

Corollary 3.4. *If $\mathbb{E}\left[\|\hat{\mathbf{p}} - \mathbf{p}\|_1\right] \to 0$ and $u = u_n \to 0$ when $n \to \infty$, we have*

$$|\mathbb{E}[\mathcal{R}(\hat{g})] - \mathcal{R}(g^*_{\text{fair}})| \to 0, \qquad as \quad n, N \to \infty \ .$$

We emphasize that Theorem 3.2 and Corollary 3.4 directly imply that $\hat{g}$ performs asymptotically as well as g^*_{fair} in terms of both fairness and accuracy: under suitable conditions, we have $\mathbb{E}[\mathcal{R}(\hat{g})] \to \mathcal{R}(g^*_{\text{fair}})$ and $\mathbb{E}[\mathcal{U}(\hat{g})] \to 0$ as $n \to \infty$.

Remark 3.5. *The estimation of* $\mathbf{p}$ *is an important aspect of the procedure in order to get the consistency of* $\hat{g}$*. In Appendix C, we differ the study of an example of consistent learning algorithm based on ERM for which we derive a rate of convergence for the excess fair-risk in Theorem 3.3.*

4 Approximate fair multi-class classification

Approximate fairness, also called ε-fairness, is particularly popular from a practical perspective in the field of algorithmic fairness. Importantly, the user is allowed to relax the fairness constraint whenever relevant or needed. Such relaxation is crucial when strict fairness strongly deflates the accuracy of the method. Of course, such modularity has a cost: the solution can not be as fair as the exact fair one; Besides, the unfairness level becomes a parameter that has no clear interpretation. Without clear justification, some empirical rules exist such as the forth-firth that tolerates an unfairness of 0.2 [11, 14, 20].

In this section we extend the results of Sections 2-3 in the *approximate* fairness setting, without taking in consideration the issue of selection of the level ε of unfairness.

ε-demographic parity in multi-class setting. First, we extend Definition 2.1 to the context of ε-fairness.

Definition 4.1 (ε-Demographic parity (DP))**.** *Let* $\varepsilon \geq 0$*, we say that a classifier* $g \in \mathcal{G}$ *is* ε*-fair (and write* $g \in \mathcal{G}_{\varepsilon-\text{fair}}$*)* w.r.t. *the distribution* $\mathbb{P}$ *on* $\mathcal{X} \times \mathcal{S} \times [K]$ *if for each* $k \in [K]$

$$|\mathbb{P}\left(g(X,S)=k|S=1\right) - \mathbb{P}\left(g(X,S)=k|S=-1\right)| \leq \varepsilon \ .$$

When $\varepsilon = 0$, Definition 2.1 reduces to exact fairness. Analogously to the exact fairness setting, we need a formalism of ε fairness. To this end, we use again the measure of the unfairness $\mathcal{U}(\cdot)$ introduced in Definition 2.2.

Definition 4.2 (ε-fairness)**.** *A classifier* g *is* ε*-fair if and only if* $\mathcal{U}(g) \leq \varepsilon$*.*

Optimal fair classifier. Our goal is to derive an explicit formulation of the optimal ε-fair classifiers *w.r.t.* the misclassification risk, denoted by $g^*_{\varepsilon-\text{fair}}$, which is solution of

$$\min_{g \in \mathcal{G}_{\varepsilon-\text{fair}}} \mathcal{R}(g) \ .$$

Solving this problem shares similarities with the exact fairness case. However, deriving the optimal ε-fair classifier is trickier and requires different tools. Nevertheless, the first step remains to write the Lagrangian of the above problem: for $\lambda^{(1)} = (\lambda_1^{(1)}, \ldots, \lambda_K^{(1)}) \in \mathbb{R}_+^K$ and $\lambda^{(2)} = (\lambda_1^{(2)}, \ldots, \lambda_K^{(2)}) \in \mathbb{R}_+^K$ we define the *ε-fair-risk* as

$$\begin{aligned}
\mathcal{R}_{\lambda^{(1)},\lambda^{(2)}}(g) := \mathcal{R}(g) & \\
&+ \sum_{k=1}^{K} \lambda_k^{(1)} [\mathbb{P}\left(g(X,S)=k|S=1\right) \\
&\qquad - \mathbb{P}\left(g(X,S)=k|S=-1\right) - \varepsilon] \qquad (4) \\
&+ \sum_{k=1}^{K} \lambda_k^{(2)} [\mathbb{P}\left(g(X,S)=k|S=-1\right) \\
&\qquad - \mathbb{P}\left(g(X,S)=k|S=1\right) - \varepsilon] \ .
\end{aligned}$$

An analog of Proposition 2.4 follows as well as a complete characterization of the optimal ε-fair classifier.

Proposition 4.3. *Let $H : \mathbb{R}_+^{2K} \to \mathbb{R}$ be the function*

$$H(\lambda^{(1)}, \lambda^{(2)}) = \sum_{s\in\mathcal{S}} \mathbb{E}_{X|S=s}\left[\max_k \left(\pi_s p_k(X,s) - s(\lambda_k^{(1)} - \lambda_k^{(2)})\right)\right] + \varepsilon \sum_{k=1}^{K} (\lambda_k^{(1)} + \lambda_k^{(2)}) \ .$$

Let Assumption 2.3 be satisfied and define $\lambda^{*(1)}, \lambda^{*(2)} \in \mathbb{R}_+^{2K}$ by

$$(\lambda^{*(1)}, \lambda^{*(2)}) \in \arg\min_{(\lambda^{(1)},\lambda^{(2)})\in\mathbb{R}_+^{2K}} H(\lambda^{(1)}, \lambda^{(2)}) \ .$$

Then, $g^*_{\varepsilon-\text{fair}} \in \arg\min_{g\in\mathcal{G}_{\varepsilon-\text{fair}}} \mathcal{R}(g)$ if and only if $g^*_{\varepsilon-\text{fair}} \in \arg\min_{g\in\mathcal{G}} \mathcal{R}_{\lambda^{*(1)},\lambda^{*(2)}}(g)$.
In addition, for all $(x,s) \in \mathcal{X}\times\mathcal{S}$, we can rewrite

$$g^*_{\varepsilon-\text{fair}}(x,s) = \arg\max_{k\in[K]} \left(\pi_s p_k(x,s) - s(\lambda_k^{*(1)} - \lambda_k^{*(2)})\right).$$

Plug-in ε fair classifier. From now on, we strictly follow the methodology considered in the exact fairness setting. In particular, we derive an empirical counterpart of the classifier $g^*_{\varepsilon-\text{fair}}$ in a semi-supervised manner using the same datasets as in Section 3.1. The rule remains the same: estimate all unknown quantities and plug them into the expression of $g^*_{\varepsilon-\text{fair}}$. This allows to write the plug-in estimator

$$\hat{g}_\varepsilon(x,s) = \arg\max_{k\in[K]} \left(\hat{\pi}_s \bar{p}_k(x,s,\zeta_k) - s(\hat{\lambda}_k^{(1)} - \hat{\lambda}_k^{(2)})\right) \ , \tag{5}$$

for all $(x,s) \in \mathcal{X}\times\mathcal{S}$, where the couple $(\hat{\lambda}^{(1)}, \hat{\lambda}^{(2)})$ is minimizer over $\mathbb{R}_+^{2K}$ of $\hat{H}(\lambda^{(1)}, \lambda^{(2)})$ which is defined as

$$\hat{H}(\lambda^{(1)}, \lambda^{(2)}) = \sum_{s\in\mathcal{S}} \frac{1}{N_s} \sum_{i=1}^{N_s} \left[\max_k \left(\hat{\pi}_s \bar{p}_k(X_i^s, s, \zeta_{k,i}^s) - s(\lambda_k^{(1)} - \lambda_k^{(2)})\right)\right] + \varepsilon \sum_{k=1}^{K} (\lambda_k^{(1)} + \lambda_k^{(2)}) \ . \tag{6}$$

Notice that here again, we exploited the randomization trick that is still required to set the correct level of fairness. We conclude this section by a remark regarding the fairness and the risk of the ε-fair estimator.

Going through the proofs of Theorems 3.2 and 3.3, we notice that they can be adapted to the ε fairness setting *modulo* minor changes. The only part that must be taken with care concerns the sub-differential of the empirical objective function that brings into play an additional restriction on the parameter space, since the dual variables in the Lagrangian are positive. This being said, we can extend Theorems 3.2 and 3.3 to the ε setting case and show that:

i) **Distribution-free ε-fairness.** For any estimators $\hat{p}_k$, the estimator $\hat{g}_\varepsilon$ achieves the right fairness level, that is, $|\mathbb{E}[\mathcal{U}(\hat{g}_\varepsilon)] - \varepsilon| \leq \frac{C}{\sqrt{N}}$ for some positive constant C.

ii) **Consistency results.** If the preliminary estimator of the conditional probabilities is consistent in L_1-norm, that is, if $\mathbb{E}\left[\|\hat{\mathbf{p}} - \mathbf{p}\|_1\right] \to 0$ and if $u = u_n \to 0$ when $n \to \infty$, then $\mathbb{E}\left[\mathcal{R}(\hat{g}_\varepsilon)\right] \to \mathcal{R}(g^*_{\varepsilon-\text{fair}})$ as $n, N \to \infty$.

This provides strong theoretical guarantees for $\hat{g}_\varepsilon$ in terms of both fairness and risk. Our approach can specifically control the fairness of the algorithm and set it at a desired level.

5 Implementation of the algorithm

In the present section, we focus on the implementation of our algorithm that produces an ε-fairness classifier. While the implementation in the exact fairness setting might be improved using accelerated gradient descent, we do not develop it here and simply identify the exact fair algorithm as the approximate fair one whenever $\varepsilon = 0$.

The proposed approximate fair algorithm is defined in Eq. (5) and requires to solve an optimization problem in Eq. (6). In this section, we elaborate on the implementation–pseudo-code provided in Algorithm 1.

Algorithm 1 ε-fairness calibration

Input: ε parameter enabling the exact or approximate fairness, new data point (x, s), base estimators $(\bar{p}_k)_k$, unlabeled sample $\mathcal{D}'_N$, $(\zeta_k)_k$ and $(\zeta^s_{k,i})_{k,i,s}$ collection of i.i.d uniform perturbations in $[0, 10^{-5}]$

Step 0. Split $\mathcal{D}'_N$ and construct the samples $(S_1, \ldots, S_N)$ and $\{X^s_1, \ldots, X^s_{N_s}\}$, for $s \in \mathcal{S}$;

Step 1. Compute the empirical frequencies $(\hat{\pi}_s)_s$ based on $(S_1, \ldots, S_N)$;

Step 2. Compute $\hat{\lambda}^{(1)} = (\hat{\lambda}^{(1)}_1, \ldots, \hat{\lambda}^{(1)}_K)$ and $\hat{\lambda}^{(2)} = (\hat{\lambda}^{(2)}_1, \ldots, \hat{\lambda}^{(2)}_K)$ as a solution of Eq. (6);
`Sequential quadratic programming of Section 5 can be used for this step.`

Step 3. Compute $\hat{g}$ thanks to Eq. (5);

Output: ε-fair classification $\hat{g}(x, s)$ at point (x, s).

First of all, base estimators $(\bar{p}_k)_k$ are needed as input of the algorithm. In our numerical study, we consider Random Forest (RF), SVM, and logistic regression (reglog). However, we emphasize that for this step we can fit any off-the-shelf estimators by using the labeled dataset $\mathcal{D}_n$. In particular, already pre-trained efficient machine learning algorithms, whose retraining might be costly can be used. This is one of the main advantages of post-processing approaches as compared to in-processing ones. One might not forget the randomization in the definition of $\bar{p}_k$ that offers good theoretical properties for fairness calibration (see Section 3.1).

Once we have computed the $(\bar{p}_k)_k$, the fair classifier $\hat{g}$ relies on the estimators $\hat{\lambda}^{(1)}$ and $\hat{\lambda}^{(2)}$ computed in **Step 2.** of the algorithm. It requires solving the minimization problem given by Eq. (6). The corresponding objective function is convex but non-smooth due to the evaluation of the function *max* function. One classical way to regularize the objective function is to simply replace the hard-max by a soft-max Namely, for β a positive real number designating the temperature parameter

and $x \in \mathbb{R}^K$, we set

$$\text{softmax}(x) := \sum_{k=1}^{K} \sigma_\beta(x)_k \cdot x_k \ ,$$
$$\text{where} \quad \sigma_\beta(x)_k := \frac{\exp(x_k/\beta)}{\sum_{k=1}^{K} \exp(x_k/\beta)} \ .$$

Whenever $\beta \to 0$ the soft-max reduces to the classical max function. Problem (6) with the soft-max relaxation is regular enough to be solved by a constrained optimization method, such as sequential quadratic programming [15, 24]. Empirical study shows that $\beta = 0.005$ enables a good accuracy of our algorithm, without deviating too much from the original solution (with the max function).

Instead of regularizing the objective function, one can alternatively use sampling methods such as cross-entropy optimization [26] on the original objective function. Despite their precision, the downside of these algorithms is their computational complexity, whose growth with the problem dimension is much faster than the one of their smooth counterpart. For this reason, the regularization approach has been preferred in the following numerical study.

6 Numerical Evaluation

We now evaluate our method numerically[1]. Section 6.1 illustrates the efficiency of our ε-fairness algorithm on synthetic data, while experiments on various real datasets are provided in Section 6.2. Since, up to our knowledge, imposing fairness constraint in multi-class classification in a model-agnostic manner is not addressed in the literature we compare our method to the state-of-the-art approach proposed in [2] for binary classification.

6.1 Evaluation on synthetic data

Synthetic data. Let define the synthetic data (X, S, Y). Conditional on $Y = k$ with $k \in [K]$, features $X \in \mathbf{R}^d$ follows a Gaussian mixture of m components, while the sensitive feature $S \in \{-1, +1\}$ follows a Bernoulli *contamination* with parameter p and $p-1$ if $k \leq \lfloor K/2 \rfloor$ and $k > \lfloor K/2 \rfloor$ respectively:

$$\begin{aligned}
(X|Y=k) &\sim \frac{1}{m}\sum_{i=1}^{m} \mathcal{N}_d(c^k + \mu_i^k, I_d), \quad \text{for } k \in [K], \\
(S|Y=k) &\sim 2 \cdot \mathcal{B}(p) - 1, \qquad \text{if} \quad k \leq \lfloor K/2 \rfloor \,, \\
(S|Y=k) &\sim 2 \cdot \mathcal{B}(1-p) - 1, \qquad \text{if} \quad k > \lfloor K/2 \rfloor \,,
\end{aligned}$$

with $c^k \sim \mathcal{U}_d(-1, 1)$, and $\mu_1^k, \ldots, \mu_m^k \sim \mathcal{N}_d(0, I_d)$. Notably, this synthetic data structure enables to challenge different aspects of the algorithm. The parameter p measures the historical bias in the dataset. Specifically, the model becomes fair when $p = 0.5$ and completely unfair when $p \in \{0, 1\}$ (see Figure 1 for an illustration). As default parameters, we set $K = 6$, $p = 0.75$, $m = 10$ and $d = 20$.

[1]The source of our method can be found at `https://github.com/xxxxxx`.

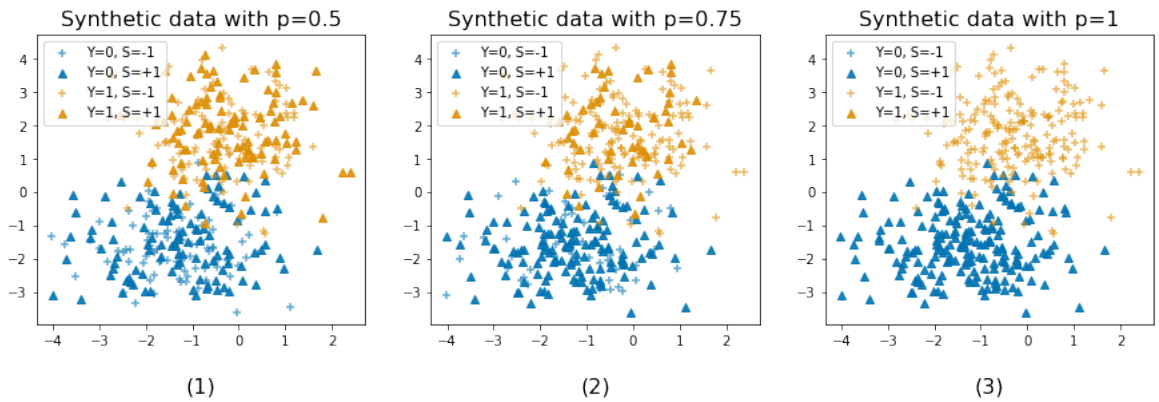

Figure 1: Example of synthetic data in binary case ($K = 2$) where $d = 2$ and $m = 1$. The level of unfairness is set as follows: *(1)* $p = 0.5$ (*e.g.* no unfairness); *(2)* $p = 0.75$ (*e.g.* unfair dataset); *(3)* $p = 1$ (*e.g.* highly unfair dataset).

Simulation scheme. We compare our method to the unfair approach. We set $u = 10^{-5}$ and the probabilities p_k are estimated by RF with default parameters in `scikit-learn`. For all experiments, we generate $n = 5000$ synthetic examples and split the data into three sets (60% training set, 20% hold-out set and 20% unlabelled set). The performance of a classifier g is evaluated by its empirical accuracy $\mathrm{Acc}(g)$ on the hold-out set. The fairness of g is measured on the hold-out set via the empirical counterpart of the unfairness measure $\mathcal{U}(g)$ given in Definition 2.2. We repeat each procedure 30 times in order to report the average performance (accuracy and unfairness) alongside its standard deviation on the hold-out set.

Fairness versus Accuracy. Fig. 2 illustrates the evolution of the performance (unfairness and accuracy) of the algorithm with respect to ε. Fig. 3-Left displays the fairness and accuracy of our algorithm for different levels of historical bias (quantified by p) in the dataset. The evolution of the performance in Fig. 2 behaves as expected: enforcing more fairness is counter-balanced by a weaker accuracy (see also in Fig. 3), therefore the trade-off between unfairness and accuracy can be controlled by the parameter ε. In particular, in case of exact fairness $\varepsilon = 0$, the gain in fairness is particularly salient and effective. By contrast, whenever $\varepsilon = 0.15$, the fair classifier becomes similar to the unfair method, meaning that the original unfairness of the problem is around $\varepsilon = 0.15$. From Fig. 3-Left, we additionally notice that: 1) the fairness efficiency of the algorithm is particularly significant for datasets with large historical bias ($p = 0.95$ or 0.99); 2) our method succeeds to reach the demanded unfairness level up to small approximation terms (see how the curves are vertical as soon as the bound on the unfairness is reached).

Fairness at the level of scores. Fig. 4 confirms our findings in Proposition 2.5: ε-fairness is enforced by shifting the conditional probabilities (*e.g.*, the exact fairness with $\varepsilon = 0$). When ε moves away from 0 (approximate fairness), the conditional probabilities translate a situation where the ε-fair classifier becomes more unfair. Here again we observe, but at the score distributions level, the matching between the unfair and the ε-fair classifiers whenever $\varepsilon = 0.15$.

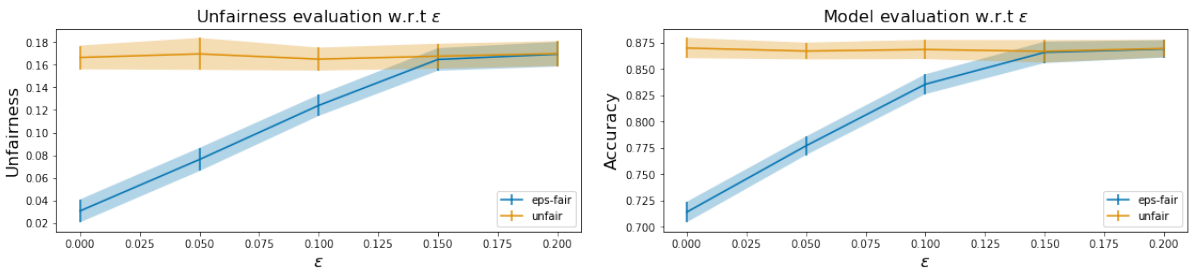

Figure 2: Performance of ε-fair and unfair classifiers in terms of accuracy and fairness. *Left*: evolution of the unfairness *w.r.t.* ε; *Right*: evolution of the accuracy *w.r.t.* ε. We report the means and standard deviations over 30 repetitions.

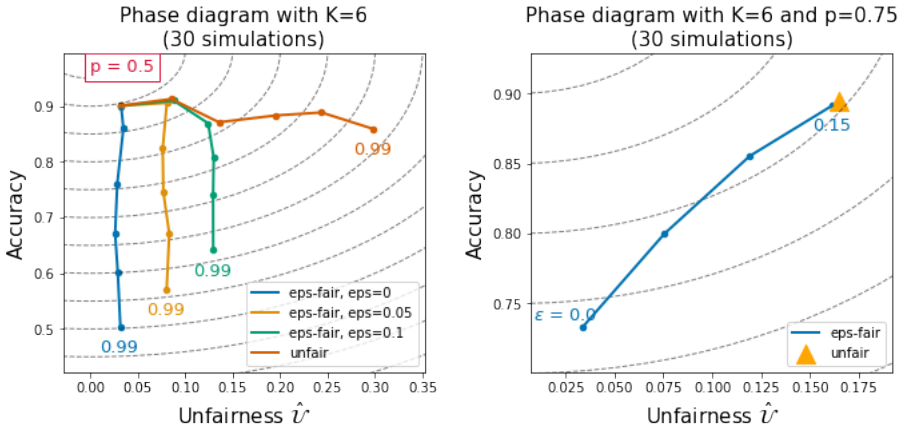

Figure 3: (Accuracy, Unfairness) phase diagrams *w.r.t. Left* the level of bias p; *Right* the accuracy-fairness trade-off parameter ε . Top-left corner gives the best trade-off.

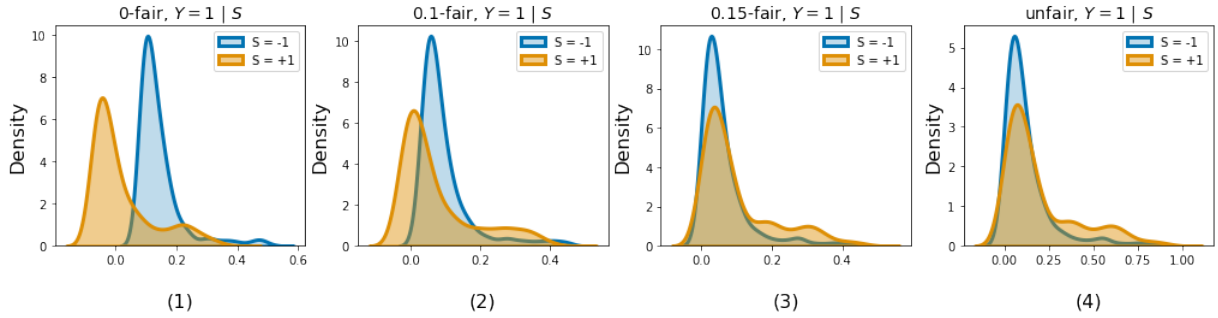

Figure 4: Empirical distribution of the score functions for the class $Y = 1$, conditional to the sensitive feature $S = \pm 1$. *(1)-(3)* ϵ-fairness with $\epsilon \in \{0, 0.1, 0.15\}$, *(4)* unfair.

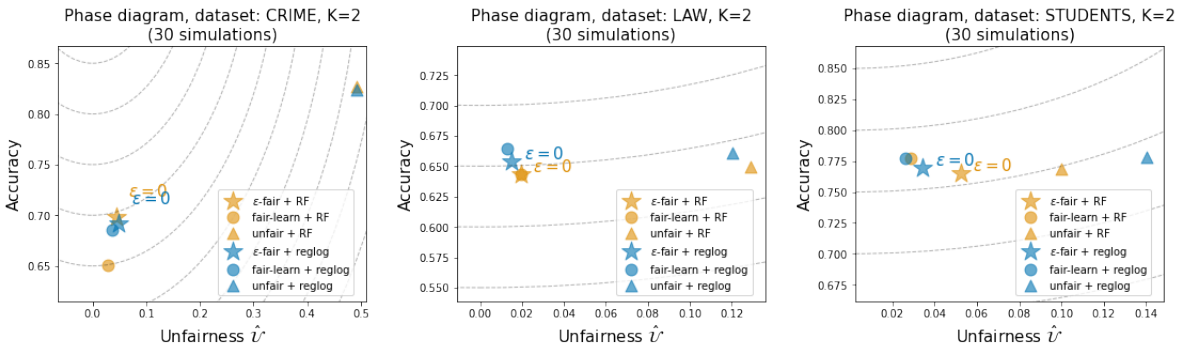

Figure 5: (Accuracy, Unfairness) phase diagrams that shows the performance of the methods. Top-left corner gives the best trade-off.

6.2 Application to real datasets

Methods. We compare our ε-fair method for both linear and non-linear multi-class classification. For linear models, we consider the one-versus-all logistic regression (reglog); for non-linear models, we use SVM model with Gaussian kernel (gaussSVC) and RF. Hyperparemeters are provided in Appx. D. Note that in multi-class setting, the accuracy of a random guess for a balanced dataset is around $1/K$.

Datasets. The performance of our method is evaluated on three benchmark datasets : CRIME, LAW and STUDENTS Hereafter, we provide a short description of these datasets.

- *Communities&Crime* (CRIME) dataset contains socio-economic, law enforcement, and crime data about communities in the US with 1994 examples. The task is to predict the number of violent crimes per 10^5 population which, we divide into $K = 5$ balanced classes based on equidistant quantiles. Following [6] the binary sensitive feature is the percentage of black population.
- *Law School Admissions* (LAW) dataset [28] presents national longitudinal bar passage data and has 20649 examples. The task is to predict a students GPA divided into $K = 3$ classes based on equidistant quantiles. The sensitive attribute is the race (white versus non-white).
- *Student Performance* (STUDENTS) dataset [12] is about student achievement in two Portuguese high schools. The task is to predict the number of grades passed (out of 3 grades *i.e.*, $K = 4$) based on student social and school related features. In binary case we predict whether the student passed the final grade. The sensitive attribute is the student age (≥ 18 vs. < 18).

Performance in binary case ($K = 2$). Before considering the multi-class setting, we analyze the relevancy of our proposal for binary classification.

In this context, the state-of-the-art is established by the in-processing approach in [2] that penalizes unfairness and will serve as baseline[2]. We focus the comparison between our approach and

[2]The method in [2] was developed for *Equality of Odds* but their code is also implemented for *Demographic Parity* see https://github.com/fairlearn/fairlearn

the state-of-the-art benchmark one to $\varepsilon = 0$ (the exact fairness problem) and illustrate in Fig 5 the performance of the methods on the LAW, CRIME and STUDENTS. While common belief suggests that in-processing methods outperform post-processing methods, it appears that our post-processing exact fairness approach is very efficient both in terms of accuracy and fairness. Indeed, the numerical experiments reveal that our method is competitive in several aspects: 1) Competitive unfairness. Overall, our exactly-fair algorithm achieves similar performance as the state-of-the-art benchmark one, when we consider reglog or RF. 2) Competitive accuracy. While on LAW and STUDENTS we achieve approximately the same accuracy, we achieve a better accuracy on CRIME when we consider RF (0.70 vs 0.65). 3) Time complexity. Since the method proposed in [2] is an in-processing algorithm, using the dedicated package, the running time of the baseline is much more higher than with our method.

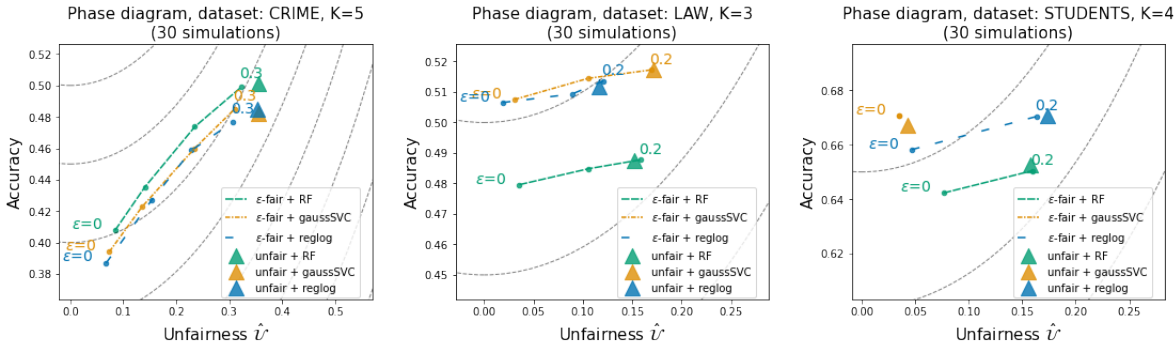

Figure 6: (Accuracy, Unfairness) phase diagrams that shows the evolution, *w.r.t.* the accuracy fairness trade-off parameter ε. Top-left corner gives the best trade-off.

Performance in multi-class case ($K \geq 3$). Numerical experiments in the multi-class setting are presented in Fig. 6. They confirm our observations on synthetic data. The performance of the ε-fairness algorithm gets closer to the unfair method as we relax the constraint on fairness. In addition, the results highlight the effectiveness of our method in enforcing fairness as ε decreases. In particular, the fairness calibration is close to the pre-specified level, regardless of the base algorithm (reglog, gaussSVC, or RF).

7 Conclusion

In the multi-class classification framework, we provide an optimal fair classification rule under DP constraint and derive misclassification and fairness guarantees of the associated plug-in fair classifier (see Alg. 1). We handle both exact and approximate fairness settings and show that our approach achieves distribution-free fairness and can be applied on top of any probabilistic base estimator. We illustrate the proficiency of our procedure on various synthetic and real datasets. In particular, our algorithm is efficient for enforcing a pre-specified level of fairness.

Calibrating the level of unfairness $\varepsilon \geq 0$ might be desired in some situations. A future direction of research is to describe a methodology that statistically justifies a data-driven calibration of this parameter in order to optimally compromise risk and unfairness.

References

[1] A. Agarwal, A. Beygelzimer, M. Dudík, J. Langford, and H. Wallach. A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

[2] A. Agarwal, M. Dudik, and Z. S. Wu. Fair regression: Quantitative definitions and reduction-based algorithms. In *International Conference on Machine Learning*, 2019.

[3] S. Barocas and A. Selbst. Big Data's Disparate Impact. *SSRN eLibrary*, 2014.

[4] S. Barocas, M. Hardt, and A. Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2018.

[5] T. Calders, F. Kamiran, and M. Pechenizkiy. Building classifiers with independency constraints. In *IEEE international conference on Data mining*, 2009.

[6] T. Calders, A. Karim, F. Kamiran, W. Ali, and X. Zhang. Controlling attribute effect in linear regression. In *IEEE International Conference on Data Mining*, 2013.

[7] S. Chiappa, R. Jiang, T. Stepleton, A. Pacchiano, H. Jiang, and J. Aslanides. A general approach to fairness with optimal transport. In *AAAI*, 2020.

[8] E. Chzhen, C. Denis, M. Hebiri, L. Oneto, and M. Pontil. Leveraging labeled and unlabeled data for consistent fair binary classification. In *Advances in Neural Information Processing Systems*, 2019.

[9] E. Chzhen, C. Denis, M. Hebiri, L. Oneto, and M. Pontil. Fair regression via plug-in estimator and recalibrationwith statistical guarantees. In *Advances in Neural Information Processing Systems*, 2020.

[10] E. Chzhen, C. Denis, M. Hebiri, L. Oneto, and M. Pontil. Fair regression with wasserstein barycenters. In *Advances in Neural Information Processing Systems*, 2020.

[11] B. Collins. Tackling unconscious bias in hiring practices: The plight of the rooney rule. 2007.

[12] Paulo Cortez and Alice Silva. Using data mining to predict secondary school student performance. *EUROSIS*, 01 2008.

[13] M. Donini, L. Oneto, S. Ben-David, J. S. Shawe-Taylor, and M. Pontil. Empirical risk minimization under fairness constraints. In *Neural Information Processing Systems*, 2018.

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

[15] Zhengqing Fu, Goulin Liu, and Lanlan Guo. Sequential quadratic programming method for nonlinear least squares estimation and its application. *Mathematical Problems in Engineering*, 2019.

[16] P. Gordaliza, E. Del Barrio, G. Fabrice, and J. M. Loubes. Obtaining fairness using optimal transport theory. In *International Conference on Machine Learning*, 2019.

[17] Thibaut Le Gouic, Jean-Michel Loubes, and Philippe Rigollet. Projection to fairness in statistical learning. *arXiv preprint arXiv:2005.11720*, 2020.

[18] S. Hajian, J. Domingo-Ferrer, and A. Martínez-Ballesté. Discrimination prevention in data mining for intrusion and crime detection. In *2011 IEEE Symposium on Computational Intelligence in Cyber Security (CICS)*, pages 47–54, 2011.

[19] M. Hardt, E. Price, and N. Srebro. Equality of opportunity in supervised learning. In *Neural Information Processing Systems*, 2016.

[20] Harry Holzer and David Holzer. Assessing affirmative action. *Journal of Economic Literature*, 38(3):483–568, 2000.

[21] R. Jiang, A. Pacchiano, T. Stepleton, H. Jiang, and S. Chiappa. Wasserstein fair classification. *arXiv preprint arXiv:1907.12059*, 2019.

[22] F. Kamiran, I. Zliobaite, and T. Calders. Quantifying explainable discrimination and removing illegal discrimination in automated decision making. *Knowl. Inf. Syst.*, 35(3):613–644, 2013.

[23] K. Lum and J. Johndrow. A statistical framework for fair predictive algorithms. *arXiv preprint arXiv:1610.08077*, 2016.

[24] Pu-yan Nie. Sequential penalty quadratic programming filter methods for nonlinear programming. *Nonlinear Analysis: Real World Applications*, 8(1):118–129, 2007.

[25] L. Oneto, M. Donini, and M. Pontil. General fair empirical risk minimization. *arXiv preprint arXiv:1901.10080*, 2019.

[26] Reuven Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and computing in applied probability*, 1(2):127–190, 1999.

[27] Shiv Kumar Tavker, Harish Guruprasad Ramaswamy, and Harikrishna Narasimhan. Consistent plug-in classifiers for complex objectives and constraints. In *Advances in Neural Information Processing Systems*, volume 33, pages 20366–20377, 2020.

[28] L. F. Wightman and H. Ramsey. *LSAC national longitudinal bar passage study.* Law School Admission Council, 1998.

[29] Q. Ye and W. Xie. Unbiased subdata selection for fair classification: A unified framework and scalable algorithms. *arXiv preprint arXiv:2012.12356*, 2020.

[30] M. B. Zafar, I. Valera, M. Gomez Rodriguez, and K. P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *International Conference on World Wide Web*, 2017.

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

[32] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning fair representations. In *International Conference on Machine Learning*, 2013.

[33] T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statistics*, 32, 2004.

In this section, we gather the proof of our results. In all the sequel, C denotes a generic constant, whose value may vary from line to line.

A Proof for exact fairness

This section is devoted to the proof of the results given in Section 2 and 3.

A.1 Proof for fair optimal rule

We begin with an auxiliary lemma, which provides an alternative useful representation of $\mathcal{R}_\lambda(g)$.

Lemma A.1. *The fair-risk of a classifier g with balancing parameter $\lambda = (\lambda_1, \dots, \lambda_K) \in \mathbb{R}^K$ rewrites:*

$$\mathcal{R}_\lambda(g) = \sum_{s\in\mathcal{S}} \mathbb{E}_{X|S=s}\left[\sum_{k=1}^{K} \left(\pi_s p_k(X,S) - s\lambda_k\right) \mathbb{1}_{\{g(X,S)\neq k\}}\right] \ . \tag{7}$$

Proof of Lemma A.1. Let $\lambda \in \mathbb{R}^K$ and recall the following definition of the fair-risk

$$\begin{aligned}\mathcal{R}_\lambda(g) =& \mathbb{P}\left(g(X,S)\neq Y\right) \\ &- \sum_{s\in\mathcal{S}}\sum_{k=1}^{K} s\lambda_k \mathbb{P}_{X|S=s}\left(g(X,S)\neq k\right) \ .\end{aligned}$$

We have the following decomposition

$$\begin{aligned}\mathbb{P}\left(g(X,S)\neq Y\right) &= \sum_{k=1}^{K} \mathbb{E}\left[\mathbb{1}_{\{g(X,S)\neq k\}}\mathbb{1}_{\{Y=k\}}\right] \\ &= \sum_{k=1}^{K}\sum_{s\in\mathcal{S}} \mathbb{E}\left[\mathbb{1}_{\{g(X,S)\neq k\}}\mathbb{1}_{\{S=s\}} p_k(X,S)\right] \\ &= \sum_{k=1}^{K}\sum_{s\in\mathcal{S}} \mathbb{E}_{X|S=s}\left[\mathbb{1}_{\{g(X,s)\neq k\}}\pi_s p_k(X,s)\right] \ ,\end{aligned}$$

which directly implies (7). □

Proof of Proposition 2.4. Recall that g^*_λ minimizes $\mathcal{R}_\lambda$ on $\mathcal{G}$. Besides, we deduce from Lemma A.1 that

$$\mathcal{R}_\lambda(g^*_\lambda) = 1 - \sum_{s\in\mathcal{S}} \mathbb{E}_{X|S=s}\left[\max_{k\in[K]}\left(\pi_s p_k(X,s) - s\lambda_k\right)\right] \ . \tag{8}$$

Hence, a maximizer λ^* in $\mathbb{R}^K$ of $\lambda \mapsto \mathcal{R}_\lambda(g^*_\lambda)$ takes the form

$$\lambda^* \in \arg\min_{\lambda\in\mathbb{R}^K} \sum_{s\in\mathcal{S}} \mathbb{E}_{X|S=s}\left[\max_{k\in[K]}\left(\pi_s p_k(X,s) - s\lambda_k\right)\right] \ .$$

The above criterion is convex in λ. Therefore, first order optimality conditions for the minimization over λ of the above criterion imply that, for each $k \in [K]$,

$$\begin{aligned}0 =& \sum_{s\in\mathcal{S}} s\mathbb{P}_{X|S=s}\left(\forall j\neq k\ \left(\pi_s p_k(X,s) - s\lambda^*_k\right) > \left(\pi_s p_j(X,s) - s\lambda^*_j\right)\right) \\ &+ su^s_k \mathbb{P}_{X|S=s}\left(\forall j\neq k\ \left(\pi_s p_k(X,s) - s\lambda^*_k\right) \geq \left(\pi_s p_j(X,s) - s\lambda^*_j\right),\ \exists j\neq k\ \left(\pi_s p_k(X,s) - s\lambda^*_k\right) = \left(\pi_s p_j(X,s) - s\lambda^*_j\right)\right)\end{aligned}$$

with $u_k^s \in [0,1]$ for all $k \in [K]$ and $s \in \mathcal{S}$. Thanks to Assumption 2.3, $p_k(X,s) - p_j(X,s)$ has no atoms for all $s \in \mathcal{S}$ and then the second part of the r.h.s. vanishes. Therefore for all $k \in [K]$

$$\mathbb{P}_{X|S=1}\left(g^*_{\lambda^*}(X,S) \neq k\right) = \mathbb{P}_{X|S=-1}\left(g^*_{\lambda^*}(X,S) \neq k\right) \ ,$$

meaning that the classifier $g^*_{\lambda^*}$ is fair. Furthermore, for any fair classifier $g \in \mathcal{G}_{\text{fair}}$, we observe that

$$\mathcal{R}(g^*_{\lambda^*}) = \mathcal{R}_{\lambda^*}(g^*_{\lambda^*}) \leq \mathcal{R}_{\lambda^*}(g) = \mathcal{R}(g),$$

so that $g^*_{\lambda^*}$ is also an optimal fair classifier.

Conversely, consider any optimal fair classifier $g^*_{\text{fair}} \in \mathcal{G}_{\text{fair}}$. Combining the fairness of g^*_{fair} with the optimality of λ^* over the family $(R_\lambda(g^*_\lambda))_{\lambda \in \mathbf{R}^K}$, we deduce

$$\mathcal{R}_{\lambda^*}(g^*_{\text{fair}}) = \mathcal{R}(g^*_{\text{fair}}) \leq \mathcal{R}_{\lambda^*}(g^*_{\lambda^*}) \leq \mathcal{R}_{\lambda^*}(g), \text{for any } g \in \mathcal{G}\,.$$

Hence any optimal fair classifier is a minimizer of $\mathcal{R}_{\lambda^*}$ over $\mathcal{G}$. □

Proof of Corollary 2.5. The proof follows directly from Lemma A.1 and Proposition 2.4. In particular, Eq. (8) implies that

$$g^*_\lambda \in \arg\min_g \mathcal{R}_{\lambda^*}(g),$$

is characterized by

$$g^*_{\lambda^*}(x,s) \in \arg\max_{k \in [K]} \left(\pi_s p_k(x,s) - s\lambda^*_k\right).$$

□

A.2 Proof of Consistency results

We start this section with two results, Lemmas A.2-A.3 that directly follow from similar arguments as in the proofs of Proposition A.2. and Lemma B.8 in [9] respectively. Their proofs are hence omitted.

Lemma A.2. *The parameter λ^*, and $\hat{\lambda}$ are bounded.*

Lemma A.3. *We have that, for each $s \in \mathcal{S}$ and $k \in [K]$,*

$$\mathbb{E}\left[\frac{1}{N_s}\sum_{i=1}^{N_s} \mathbb{1}\{\exists j \neq k, \hat{h}^s_k(X_i, \hat{\lambda}_k) = \hat{h}^s_j(X_i, \hat{\lambda}_j)\}\right] \leq \frac{C}{N_s} \ ,$$

where $\hat{h}^s_k : (x,\lambda) \mapsto \hat{\pi}_s \bar{p}_k(x,s) - s\lambda$.

Let us now consider the proofs of Theorem 3.2 and Theorem 3.3.

Proof of Theorem 3.2. As in Lemma A.3, we first introduce, for $s \in \mathcal{S}$ and $k \in [K]$,

$$\hat{h}^s_k : (x,\lambda) \mapsto \hat{\pi}_s \bar{p}_k(x,s) - s\lambda \ .$$

By construction, the estimator $\bar{p}_k(X,S)$ satisfies Assumption 2.3, therefore for all $s \in \mathcal{S}$ and $k \in [K]$

$$\mathbb{P}_{X|S=s}\left(\hat{g}(X,S) = k\right) = \mathbb{P}_{X|S=s}\left(\forall j \neq k, \hat{h}^s_k(X,\hat{\lambda}_k) > \hat{h}^s_j(X,\hat{\lambda}_j)\right) \ .$$

Now, let us make use of the optimality of $\hat{\lambda}$. We denote by $\hat{\mathbb{P}}_{X|S=s}$ the empirical measure on the data $\{X_1^s, \ldots, X_{N_s}^s\}$. Considering the first order optimality conditions for $\hat{\lambda}$, we can show that, for all $k \in [K]$ and $s \in \mathcal{S}$, there exists $\alpha_k^s \in [-1, 1]$ such that

$$s\hat{\mathbb{P}}_{X|S=s}\left(\forall j \neq k, \hat{h}_k^s(X, \hat{\lambda}_k) > \hat{h}_j^s(X, \hat{\lambda}_j)\right) + \alpha_k^s \hat{\mathbb{P}}_{X|S=s}\left(\forall j \neq k, \hat{h}_k^s(X, \hat{\lambda}_k) \geq \hat{h}_j^s(X, \hat{\lambda}_j),\ \exists j \neq k, \hat{h}_k^s(X, \hat{\lambda}_k) = \hat{h}_j^s(X, \hat{\lambda}_j)\right) = 0 \ .$$

From the above equation, we deduce that

$$\begin{aligned}\mathcal{U}(\hat{g}) &= \sum_{k=1}^{K} \left|\mathbb{P}_{X|S=1}\left(\hat{g}(X,S) = k\right) - \mathbb{P}_{X|S=-1}\left(\hat{g}(X,S) = k\right)\right| \\ &\leq \sum_{k=1}^{K}\sum_{s\in\mathcal{S}} \left|\left(\mathbb{P}_{X|S=s} - \hat{\mathbb{P}}_{X|S=s}\right)\left(\forall j \neq k,\ \hat{h}_k^s(X, \hat{\lambda}_k) > \hat{h}_j^s(X, \hat{\lambda}_j)\right)\right| \\ &\qquad + \sum_{k=1}^{K}\sum_{s\in\mathcal{S}} \hat{\mathbb{P}}_{X|S=s}\left(\exists j \neq k,\ \hat{h}_k^s(X, \hat{\lambda}_k) = \hat{h}_j^s(X, \hat{\lambda}_j)\right) \ .\end{aligned}$$

Observe that for all $k \in [K]$

$$\begin{aligned}&\left|\left(\mathbb{P}_{X|S=s} - \hat{\mathbb{P}}_{X|S=s}\right)\left(\forall j \neq k,\ \hat{h}_k^s(X, \hat{\lambda}_k) > \hat{h}_j^s(X, \hat{\lambda}_j)\right)\right| = \\ &\qquad \left|\left(\mathbb{P}_{X|S=s} - \hat{\mathbb{P}}_{X|S=s}\right)\left(\forall j \neq k,\ \bar{p}_k(X,s) - \bar{p}_j(X,s) \geq \frac{s(\hat{\lambda}_k - \hat{\lambda}_j)}{\hat{\pi}_s}\right)\right| \\ &\qquad\qquad \leq \sum_{j=1}^{K} \sup_{t\in\mathbb{R}} \left|\left(\mathbb{P}_{X|S=s} - \hat{\mathbb{P}}_{X|S=s}\right)\left(\bar{p}_k(X,s) - \bar{p}_j(X,s) \geq t\right)\right| \ .\end{aligned}$$

Therefore, from the Dvoretzky-Kiefer-Wolfowitz Inequality conditional on $\mathcal{D}_n$ and on $(S_1, \ldots, S_N)$, we deduce that, for each $s \in \mathcal{S}$ and $k \in [K]$

$$\mathbb{E}\left[\left|\left(\mathbb{P}_{X|S=s} - \hat{\mathbb{P}}_{X|S=s}\right)\left(\forall j \neq k, \hat{h}_k^s(X, \hat{\lambda}_k) > \hat{h}_j^s(X, \hat{\lambda}_j)\right)\right|\right] \leq C\sqrt{\frac{1}{N_s}} \ .$$

Applying Lemma A.3, we then get that, Conditional on $\mathcal{D}_n$ and on $(S_1, \ldots, S_N)$, we have that

$$\mathbb{E}\left[\mathcal{U}(\hat{g})\right] \leq C\sum_{s\in\mathcal{S}}\sqrt{\frac{1}{N_s}} \ .$$

Since N_s is a binomial random variable with parameter (π_s, N), we get

$$\mathbb{E}\left[\mathcal{U}(\hat{g})\right] \leq C\sqrt{\frac{1}{N}},$$

where C depends on K and $\min(\pi_{-1}, \pi_1)$. □

Proof of Theorem 3.3. The proof goes conditional on the training data. First, let us decompose *excess fair-risk* of the classifier $\hat{g}$ in a convenient way for our analysis

$$\mathcal{R}_{\lambda^*}(\hat{g}) - \mathcal{R}_{\lambda^*}(g^*_{\text{fair}}) = \left(\mathcal{R}_{\lambda^*}(\hat{g}) - \mathcal{R}_{\hat{\lambda}}(\hat{g})\right) + \left(\mathcal{R}_{\hat{\lambda}}(\hat{g}) - \mathcal{R}_{\lambda^*}(g^*_{\lambda^*})\right) \quad , \tag{9}$$

where we recall that $g^*_{\text{fair}} = g^*_{\lambda^*}$. We propose to deal with the two terms in r.h.s. of Equation (9) separately. According the the first term, we have

$$\begin{aligned}
\left(\mathcal{R}_{\lambda^*}(\hat{g}) - \mathcal{R}_{\hat{\lambda}}(\hat{g})\right) &= \sum_{s\in\mathcal{S}}\sum_{k=1}^{K} s\lambda^*_k \mathbb{P}_{X|S=s}\left(\hat{g}(X,S)\neq k\right) - \sum_{s\in\mathcal{S}}\sum_{k=1}^{K} s\hat{\lambda}_k \mathbb{P}_{X|S=s}\left(\hat{g}(X,S)\neq k\right) \\
&= \sum_{s\in\mathcal{S}}\sum_{k=1}^{K} s\left(\lambda^*_k - \hat{\lambda}_k\right)\mathbb{P}_{X|S=s}\left(\hat{g}(X,S)\neq k\right) \quad .
\end{aligned}$$

Since, for each $k \in [K]$, the parameters λ^*_k and $\hat{\lambda}_k$ are bounded (see Lemma A.2), we deduce that

$$\mathcal{R}_{\lambda^*}(\hat{g}) - \mathcal{R}_{\hat{\lambda}}(\hat{g}) \leq C\mathcal{U}(\hat{g}) \quad . \tag{10}$$

Then we have shown that the first term in the r.h.s. of Eq. (9) relies on the unfairness of the classifier $\hat{g}$. Now, let us consider the second term in r.h.s. of Equation (9). Our goal will be to show that this term mainly depends on the quality of the base estimators $\hat{p}_k$. Since λ^* is a maximizer of $\mathcal{R}_\lambda(g^*_\lambda)$ over λ, it is clear that, conditional on the data, $\mathcal{R}_{\lambda^*}(g^*_{\lambda^*}) \geq \mathcal{R}_{\hat{\lambda}}(g^*_{\hat{\lambda}})$. (The parameter $\hat{\lambda}$ is seen as fixed conditional on the data.) Therefore, we have

$$\mathcal{R}_{\hat{\lambda}}(\hat{g}) - \mathcal{R}_{\lambda^*}(g^*_{\lambda^*}) \leq \mathcal{R}_{\hat{\lambda}}(\hat{g}) - \mathcal{R}_{\hat{\lambda}}(g^*_{\hat{\lambda}}) \quad .$$

By introducing $\hat{g}^*_{\hat{\lambda}}$, we remove the estimation of λ^* from the study of $\mathcal{R}_{\hat{\lambda}}(\hat{g}) - \mathcal{R}_{\lambda^*}(g^*_{\lambda^*})$. At this point, it becomes clear that bounding this term does not relies on the unlabeled sample sizes N_s. Let us recall the definition of $g^*_{\hat{\lambda}}$: conditional on the data

$$g^*_{\hat{\lambda}} \in \arg\min_{g\in\mathcal{G}} \mathcal{R}_{\hat{\lambda}}(g) \quad .$$

Then using similar arguments as those leading to Eq. (8) implies that (see also Corollary 2.5)

$$g^*_{\hat{\lambda}} \in \arg\max_{k\in[K]} \left(\pi_s p_k(x,s) - s\hat{\lambda}_k\right) \quad .$$

As a consequence, using the writing of the fair-risk provided by Lemma A.1

$$\mathcal{R}_{\hat{\lambda}}(\hat{g}) - \mathcal{R}_{\hat{\lambda}}(g^*_{\hat{\lambda}}) = \sum_{s\in\mathcal{S}} \mathbb{E}_{X|S=s}\left[\max_{k\in[K]}\left(\pi_s p_k(X,S) - s\hat{\lambda}_k\right) - \sum_{k=1}^{K}\left(\pi_s p_k(X,S) - s\hat{\lambda}_k\right)\mathbb{1}_{\{\hat{g}(X,S)=k\}}\right] \quad .$$

Because of the indicator function, there is only one non-zero element in the inner sum. Then we observe that for each $s \in \mathcal{S}$

$$\begin{aligned}
&\left|\max_{k\in[K]}\left(\pi_s p_k(X,S) - s\hat{\lambda}_k\right) - \sum_{k=1}^{K}\left(\pi_s p_k(X,S) - s\hat{\lambda}_k\right)\mathbb{1}_{\{\hat{g}(X,S)=k\}}\right| \\
&\qquad \leq 2\max_{k\in[K]}\left|(\pi_s p_k(X,S) - s\hat{\lambda}_k) - (\hat{\pi}_s \bar{p}_k(X,S) - s\hat{\lambda}_k)\right| \\
&\qquad \leq 2\left(\max_{k\in[K]}|p_k(X,S) - \bar{p}_k(X,S)| + |\pi_s - \bar{\pi}_s|\right) \quad ,
\end{aligned}$$

where the last inequality is due to the fact that $\pi_s, \hat{\pi}_s, p_k$, and $\bar{p}_k$ are all in $[0,1]$. Therefore, recalling that $\bar{p}_k$ is a randomized version of $\hat{p}_k$ we can write

$$\mathcal{R}_{\hat{\lambda}}(\hat{g}) - \mathcal{R}_{\hat{\lambda}}(g^*_{\hat{\lambda}}) \leq C \left(\|\hat{\mathbf{p}} - \mathbf{p}\|_1 + \sum_{s \in \mathcal{S}} |\hat{\pi}_s - \pi_s| + u \right) \ ,$$

and obtain the bound

$$\mathcal{R}_{\hat{\lambda}}(\hat{g}) - \mathcal{R}_{\lambda^*}(g^*_{\lambda^*}) \leq C \left(\|\hat{\mathbf{p}} - \mathbf{p}\|_1 + \sum_{s \in \mathcal{S}} |\hat{\pi}_s - \pi_s| + u \right) \ .$$

In view of Equation (9), the above inequality together with Equation (10) yield the desired result. □

B Proof for approximate fairness

We begin with an auxiliary lemma, which provides an alternative useful representation of $\mathcal{R}_{\lambda^{(1)},\lambda^{(2)}}(g)$.

Lemma B.1. *The ε-fair-risk of a classifier g with balancing parameters $\lambda^{(1)} = (\lambda_1^{(1)}, \ldots, \lambda_K^{(1)}) \in \mathbb{R}_+^K, \lambda^{(2)} = (\lambda_1^{(2)}, \ldots, \lambda_K^{(2)}) \in \mathbb{R}_+^K$ reads as:*

$$\mathcal{R}_{\lambda^{(1)},\lambda^{(2)}}(g) = \sum_{s \in \mathcal{S}} \mathbb{E}_{X|S=s} \left[\sum_{k=1}^{K} \left(\pi_s p_k(X,S) - s(\lambda_k^{(1)} - \lambda_k^{(2)}) \right) \mathbb{1}_{\{g(X,S) \neq k\}} \right] - \varepsilon \sum_{k=1}^{K} (\lambda_k^{(1)} + \lambda_k^{(2)}). \quad (11)$$

Proof of Lemma B.1. Let $(\lambda^{(1)}, \lambda^{(2)}) \in \mathbb{R}_+^{2K}$ and recall the following definition of the ε-fair-risk

$$\mathcal{R}_{\lambda^{(1)},\lambda^{(2)}}(g) = \mathbb{P}\left(g(X,S) \neq Y\right) - \sum_{k=1}^{K} \sum_{s \in \mathcal{S}} s(\lambda_k^{(1)} - \lambda_k^{(2)}) \, \mathbb{E}_{X|S=s} \left[\mathbb{1}_{\{g(X,s) \neq k\}} \right] - \varepsilon \sum_{k=1}^{K} (\lambda_k^{(1)} + \lambda_k^{(2)})$$

The result in (11) directly follows from the following decomposition

$$\begin{aligned}
\mathbb{P}\left(g(X,S) \neq Y\right) &= \sum_{k=1}^{K} \mathbb{E}\left[\mathbb{1}_{\{g(X,S) \neq k\}} \mathbb{1}_{\{Y=k\}} \right] \\
&= \sum_{k=1}^{K} \sum_{s \in \mathcal{S}} \mathbb{E}\left[\mathbb{1}_{\{g(X,S) \neq k\}} \mathbb{1}_{\{S=s\}} p_k(X,S) \right] \\
&= \sum_{k=1}^{K} \sum_{s \in \mathcal{S}} \mathbb{E}_{X|S=s}\left[\mathbb{1}_{\{g(X,s) \neq k\}} \pi_s p_k(X,s) \right] \ .
\end{aligned}$$

□

Proof of Proposition 4.3. From Lemma B.1, we deduce that $g^*_{\lambda^{(1)},\lambda^{(2)}}$ should be defined for all $(x,s) \in \mathcal{X} \times \mathcal{S}$ as

$$g^*_{\lambda^{(1)},\lambda^{(1)}}(x,s) = \arg\max_{k \in [K]} \left(\pi_s p_k(X,S) - s(\lambda_k^{(1)} - \lambda_k^{(1)}) \right) \ ,$$

since it minimizes the risk $\mathcal{R}_{\lambda^{(1)},\lambda^{(2)}}$. Now we should maximize $\mathcal{R}_{\lambda^{(1)},\lambda^{(2)}}(g^*_{\lambda^{(1)},\lambda^{(2)}})$ in the dual variables. Notice that the ε-fair risk can be wirten as

$$\mathcal{R}_{\lambda^{(1)},\lambda^{(2)}}(g^*_{\lambda^{(1)},\lambda^{(2)}}) = 1 - \sum_{s\in\mathcal{S}} \mathbb{E}_{X|S=s}\left[\max_{k\in[K]}\left(\pi_s p_k(X,S) - s(\lambda_k^{(1)} - \lambda_k^{(2)})\right)\right] - \varepsilon\sum_{k=1}^{K}(\lambda_k^{(1)} + \lambda_k^{(2)}) \ .$$

Hence, a maximizer $(\lambda^{*(1)},\lambda^{*(2)})$ in $\mathbb{R}_+^{2K}$ of $(\lambda^{(1)},\lambda^{(2)}) \mapsto \mathcal{R}_{\lambda^{(1)},\lambda^{(2)}}(g^*_{\lambda^{(1)},\lambda^{(2)}})$ takes the form

$$(\lambda^{*(1)},\lambda^{*(2)}) \in \arg\min_{(\lambda^{(1)},\lambda^{(2)})\in\mathbb{R}_+^{2K}} \underbrace{\sum_{s\in\mathcal{S}} \mathbb{E}_{X|S=s}\left[\max_{k\in[K]}\left(\pi_s p_k(X,s) - s(\lambda_k^{(1)} - \lambda_k^{(2)})\right)\right] + \varepsilon\sum_{k=1}^{K}(\lambda_k^{(1)} + \lambda_k^{(2)})}_{H(\lambda^{(1)},\lambda^{(2)})} \ .$$

The rest of the proof consists in showing that such a calibration of the tuning parameters $\lambda^{(1)},\lambda^{(2)}$ implies that $g^*_{\lambda^{(1)},\lambda^{(2)}}$ is indeed ε-fair. Observe that

$$H(\lambda^{(1)},\lambda^{(2)}) \geq \varepsilon\sum_{k=1}^{K}(\lambda_k^{(1)} + \lambda_k^{(2)}) \ ,$$

and then $\lim_{\|(\lambda^{(1)},\lambda^{(2)})\|_2^2\to\infty} H(\lambda^{(1)},\lambda^{(2)}) = +\infty$. Moreover, this criterion is convex in $(\lambda^{(1)},\lambda^{(2)})$. Therefore the minimum $(\lambda^{*(1)},\lambda^{*(2)})$ exists. In particular, we can derive the first order optimality conditions of the above problem *w.r.t.* $\lambda^{(1)}$ which implies that for each $k\in[K]$,

$$\begin{aligned}
0 = &-\sum_{s\in\mathcal{S}} s\mathbb{P}_{X|S=s}\Big(\forall j\neq k\ (\pi_s p_k(X,s) - s(\lambda_k^{*(1)} - \lambda_k^{*(2)})) > (\pi_s p_j(X,s) - s(\lambda_j^{*(1)} - \lambda_j^{*(2)}))\Big) \\
&+ s\,u_k^s\,\mathbb{P}_{X|S=s}\Big(\forall j\neq k\ (\pi_s p_k(X,s) - s(\lambda_k^{*(1)} - \lambda_k^{*(2)})) \geq (\pi_s p_j(X,s) - s(\lambda_j^{*(1)} - \lambda_j^{*(2)})), \\
&\qquad\qquad \exists j\neq k\ (\pi_s p_k(X,s) - s(\lambda_k^{*(1)} - \lambda_k^{*(2)})) = (\pi_s p_j(X,s) - s(\lambda_j^{*(1)} - \lambda_j^{*(2)}))\Big) + \varepsilon \ ,
\end{aligned}$$

with $u_k^s\in[0,1]$ for all $k\in[K]$ and all $s\in\mathcal{S}$. Similarly the first order conditions *w.r.t.* $\lambda^{(1)}$ implies that for each $k\in[K]$,

$$\begin{aligned}
0 = &\sum_{s\in\mathcal{S}} s\mathbb{P}_{X|S=s}\Big(\forall j\neq k\ (\pi_s p_k(X,s) - s(\lambda_k^{*(1)} - \lambda_k^{*(2)})) > (\pi_s p_j(X,s) - s(\lambda_j^{*(1)} - \lambda_j^{*(2)}))\Big) \\
&+ s\,v_k^s\,\mathbb{P}_{X|S=s}\Big(\forall j\neq k\ (\pi_s p_k(X,s) - s(\lambda_k^{*(1)} - \lambda_k^{*(2)})) \geq (\pi_s p_j(X,s) - s(\lambda_j^{*(1)} - \lambda_j^{*(2)})), \\
&\qquad\qquad \exists j\neq k\ (\pi_s p_k(X,s) - s(\lambda_k^{*(1)} - \lambda_k^{*(2)})) = (\pi_s p_j(X,s) - s(\lambda_j^{*(1)} - \lambda_j^{*(2)}))\Big) + \varepsilon \ ,
\end{aligned}$$

with $v_k^s\in[0,1]$ for all $k\in[K]$ and $s\in\mathcal{S}$. Thanks to Assumption 2.3, $p_k(X,s) - p_j(X,s)$ has no atom for all $s\in\mathcal{S}$ and then the second part of the r.h.s. of both above equations vanish. Hence, the first order optimality conditions *w.r.t.* $\lambda_k^{*(1)}$ becomes

$$\mathbb{P}_{X|S=1}\left(g^*_{\lambda^{*(1)},\lambda^{*(2)}}(X,S)\neq k\right) - \mathbb{P}_{X|S=-1}\left(g^*_{\lambda^{*(1)},\lambda^{*(2)}}(X,S)\neq k\right) = \varepsilon \ ,$$

and the first order optimality conditions *w.r.t.* $\lambda_k^{*(2)}$ writes as

$$\mathbb{P}_{X|S=1}\left(g^*_{\lambda^{*(1)},\lambda^{*(2)}}(X,S)\neq k\right) - \mathbb{P}_{X|S=-1}\left(g^*_{\lambda^{*(1)},\lambda^{*(2)}}(X,S)\neq k\right) = -\varepsilon \ .$$

Therefore, we deduce the following constraint on $\lambda_k^{*(1)}, \lambda_k^{*(2)}$

$$\lambda_k^{*(1)}\lambda_k^{*(2)} = 0 \text{ and } \lambda_k^{*(1)} + \lambda_k^{*(2)} \geq 0 \ .$$

Hence, if $\lambda_k^{*(1)} + \lambda_k^{*(2)} > 0$,

$$\left|\mathbb{P}_{X|S=1}\left(g^*_{\lambda^{*(1)},\lambda^{*(2)}}(X,S) \neq k\right) - \mathbb{P}_{X|S=-1}\left(g^*_{\lambda^{*(1)},\lambda^{*(2)}}(X,S) \neq k\right)\right| = \varepsilon.$$

In the case where $\lambda_k^{*(1)} = \lambda_k^{*(2)} = 0$, we use Euler Inequality and deduce

$$\left|\mathbb{P}_{X|S=1}\left(g^*_{\lambda^{*(1)},\lambda^{*(2)}}(X,S) \neq k\right) - \mathbb{P}_{X|S=-1}\left(g^*_{\lambda^{*(1)},\lambda^{*(2)}}(X,S) \neq k\right)\right| \leq \varepsilon \ .$$

□

C Rates of convergence for ERM estimator

We have established in Section 3.2 theoretical guarantees on risk and fairness for our plug-in procedure, when using any off-the-shelf consistent estimator of the conditional probability. In this section, we study more close detail the classical setting where the conditional probabilities estimation step is provided by ERM and derive an explicit bound on the fair-risk of the resulting fair classifier,

For a given (measurable) score function $\mathbf{f}(\cdot) = (f_1(\cdot), \ldots, f_K(\cdot))$ mapping $\mathcal{X} \times \{-1, 1\}$ onto $\mathbb{R}^K$, we define the induced classification rule [3]

$$g_{\mathbf{f}}(\cdot) \in \arg\max_{k\in[K]} f_k(\cdot) \ .$$

For the sake of simplicity, we focus on the L_2-risk[4]

$$R_2(\mathbf{f}) := \mathbb{E}\left[\sum_{k=1}^{K}\left(Z_k - f_k(X,S)\right)^2\right] \ ,$$

where $Z_k := 2\mathbb{1}_{\{Y=k\}} - 1$.

The optimal score function $\mathbf{f}^*$ *w.r.t.* R_2 is denoted $\mathbf{f}^* := \arg\min_{\mathbf{f}} R_2(\mathbf{f})$, where the infimum is taken over all measurable functions that map $\mathcal{X} \times \{-1, 1\}$ onto $\mathbb{R}^K$. The optimum $\mathbf{f}^*$ satisfies the relation $f_k^*(X,S) = 2p_k(X,S) - 1$. In particular, Zhang's Lemma [33] implies that $\mathbb{E}\left[\mathcal{R}(g_{\mathbf{f}}) - \mathcal{R}(g^*)\right] \leq \left(\mathbb{E}\left[R_2(\mathbf{f}) - R_2(\mathbf{f}^*)\right]\right)^{1/2}$, for any score function $\mathbf{f}$. This inequality highlights the connection between the usual misclassification risk of $g_{\mathbf{f}}$ and the L_2-risk of the score function $\mathbf{f}$.

Let us now consider the empirical counterpart of the L_2-risk R_2, given for any function $\mathbf{f}$ by

$$\hat{R}_2(\mathbf{f}) := \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}\left(Z_k^i - f_k(X_i,S_i)\right)^2,$$

with $Z_k^i := 2\mathbb{1}_{\{Y_i=k\}} - 1$. The empirical risk minimizer $\hat{f}$ over a given convex set $\mathcal{F}$ of functions is given by

$$\hat{\mathbf{f}} \in \arg\min_{\mathbf{f}\in\mathcal{F}} \hat{R}_2(\mathbf{f}) \ .$$

[3] Whenever the maximum is reached at multiple indices, we set by convention $g_{\mathbf{f}}$ as the smallest index in $[K]$.

[4] Since the $0-1$ loss lacks convexity, we consider the square loss as a convex surrogate to avoid computational issues.

In view of the expression for the optimal score function $\mathbf{f}^*$, we naturally set $\hat{p}_k := \frac{\hat{f}_k+1}{2}$ as an estimator of the conditional probability p_k. Given $\hat{\mathbf{p}}(\cdot) = (\hat{p}_1(\cdot), \ldots, \hat{p}_K(\cdot))$, our plug-in procedure given in Eq. (2) provides an exactly fair classifier that we denote by $g_{\hat{\mathbf{f}}}$. According to the theoretical study conducted in Section 3.2, Theorem 3.2 ensures that the classifier $g_{\hat{\mathbf{f}}}$ is asymptotically exactly fair.

According to the analysis of the fair-risk of the classifier $g_{\hat{\mathbf{f}}}$, we invoke Theorem 3.3 complemented with the consistency of the empirical risk minimizer $\hat{\mathbf{f}}$ *w.r.t.* to the L_2-risk (due to Zhang's Lemma). In order to specify the precise rate of convergence of the fair-risk, we introduce additional assumptions on the class of functions $\mathcal{F}$.

Assumption C.1. *The set $\mathcal{F}$ satisfies the following:*

1. *There exists $B > 0$ s.t. $\|\mathbf{f}\|_\infty := \max_{k\in[K]} \sup_{x\in\mathcal{X}} |f_k(x)| \leq B$, for each $\mathbf{f} \in \mathcal{F}$;*
2. *For $\varepsilon > 0$, there exists an ε-net $\mathcal{F}_\varepsilon \subset \mathcal{F}$ w.r.t. $\|.\|_\infty$ and a positive constant $C_\mathcal{F}$ s.t. $\log(|\mathcal{F}_\varepsilon|) \leq C_\mathcal{F} \log(\varepsilon^{-1})$.*

Note that Assumption C.1 covers bounded parametric classes among others (for instance, linear or polynomial with bounded degree and bounded coefficients score functions). This structural assumption on the set of models $\mathcal{F}$ enables to control the rate of convergence of the excess fair-risk of $g_{\hat{\mathbf{f}}}$ the constructed fair classifier.

Proposition C.2. *Assume that $\mathbf{f}^* \in \mathcal{F}$ and $u = u_n = \frac{1}{n}$. If Assumptions 2.3 and C.1 hold, then*

$$\mathbb{E}\left[\mathcal{R}_{\lambda^*}(g_{\hat{\mathbf{f}}}) - \mathcal{R}_{\lambda^*}(g^*_{\text{fair}})\right] \leq C\left(\left(\frac{\log(n)}{n}\right)^{1/2} + N^{-1/2}\right) \ .$$

Under classical assumptions on the complexity of $\mathcal{F}$, Proposition C.2 induces in particular a parametric rate of convergence for the excess fair-risk of $g_{\hat{\mathbf{f}}}$.

Proof. From Theorems 3.2 and 3.3, we have that

$$\mathbb{E}\left[\mathcal{R}_{\lambda^*}(g_{\hat{\mathbf{f}}}) - \mathcal{R}_{\lambda^*}(g^*_{\text{fair}})\right] \ \leq \ \mathbb{E}\,\|\hat{\boldsymbol{p}} - \mathbf{p}\|_1 + \frac{1}{n} + \frac{C}{\sqrt{N}} \ .$$

Since

$$\mathbb{E}\,\|\hat{\boldsymbol{p}} - \mathbf{p}\|_1 \leq \frac{1}{2}\mathbb{E}\left\|\hat{\boldsymbol{f}} - \boldsymbol{f}^*\right\|_1 \leq \frac{1}{2}\sqrt{\sum_{k=1}^K \mathbb{E}\left[\left(f_k(X,S) - f_k^*(X,S)\right)^2\right]} \ ,$$

it remains to provide a control on the term $\sum_{k=1}^K \mathbb{E}\left[\left(f_k(X,S) - f_k^*(X,S)\right)^2\right]$. For this purpose, let us first prove that for each score function $\mathbf{f} \in \mathcal{F}$, the following holds

$$\sum_{k=1}^K \mathbb{E}\left[\left(f_k(X,S) - f_k^*(X,S)\right)^2\right] \leq 2\left(R_2(\mathbf{f}) - R_2(\mathbf{f}^*)\right) \ . \tag{12}$$

Indeed, we observe that

$$\frac{(Z_k - f_k)^2 + (Z_k - f_k^*)^2}{2} - \left(Z_k - \left(\frac{f_k + f_k^*}{2}\right)\right)^2 = \frac{(f_k - f_k^*)^2}{4} \ .$$

From this equality, we then deduce that

$$\frac{1}{4}\sum_{k=1}^{K}\mathbb{E}\left[(f_k(X,S)-f_k^*(X,S))^2\right]=\frac{1}{2}\left(R_2(\mathbf{f})+R_2(\mathbf{f}^*)\right)-R_2\left(\frac{\mathbf{f}+\mathbf{f}^*}{2}\right)\ .$$

Since R_2 is positive, we get Equation (12).

The next step of the proof is to bound $R_2(\hat{\mathbf{f}})-R_2(\mathbf{f}^*)$. We have by definition of $\hat{\mathbf{f}}$

$$R_2(\hat{\mathbf{f}})-R_2(\mathbf{f}^*)\leq R_2(\hat{\mathbf{f}})-R_2(\mathbf{f}^*)-2\left(\hat{R}_2(\hat{\mathbf{f}})-\hat{R}_2(\mathbf{f}^*)\right).$$

Furthermore from Assumption C.1 with $\varepsilon=1/n$, we have

$$R_2(\hat{\mathbf{f}})-R_2(\mathbf{f}^*)-2\left(\hat{R}_2(\hat{\mathbf{f}})-\hat{R}_2(\mathbf{f}^*)\right)\leq\frac{C}{n}+\sup_{f\in\mathcal{F}_{1/n}}\{R_2(\mathbf{f})-R_2(\mathbf{f}^*)\}-2\left(\hat{R}_2(\hat{\mathbf{f}})-\hat{R}_2(\mathbf{f}^*)\right)\ . \quad (13)$$

If we denote $\mathrm{Err}(\mathbf{f})=\mathrm{R}_2(\mathbf{f})-\mathrm{R}_2(\mathbf{f}^*)$ and $\widehat{\mathrm{Err}}(\mathbf{f})=\hat{R}_2(\hat{\mathbf{f}})-\hat{R}_2(\mathbf{f}^*)$, from Bernstein's Inequality together with Assumption C.1, we have, for all $t>0$ and $\mathbf{f}\in\mathcal{F}_\varepsilon$,

$$\begin{aligned}
&\mathbb{P}\left(\mathrm{Err}(\mathbf{f})-2\cdot\widehat{\mathrm{Err}}(\mathbf{f})\geq \mathrm{t}\right)\\
\leq\ &\mathbb{P}\left(2\left(\mathrm{Err}(\mathbf{f})-\widehat{\mathrm{Err}}(\mathbf{f})\right)\geq t+\mathrm{Err}(\mathbf{f})\right)\\
\leq\ &\exp\left(-\frac{\frac{n}{8}\cdot(t+\mathbb{E}[h(Z,\mathbf{f}(X,S))])^2}{\mathbb{E}[|h(Z,\mathbf{f}(X,S))|^2]+\frac{C}{3}\cdot(t+\mathbb{E}[h(Z,\mathbf{f}(X,S))])}\right)
\end{aligned}$$

where $h(Z,\mathbf{f}(X,S)):=\sum_{k=1}^{K}(|Z-f_k(X,S)|^2-|Z-f_k^*(X,S)|^2)$. Furthermore, observe that

$$\mathbb{E}[|h(Z,\mathbf{f}(X,S))|^2]\leq C\cdot\mathbb{E}[h(Z,\mathbf{f}(X,S))]\ ,$$

which, plugged in the previous inequality, directly provides

$$\mathbb{P}\left(\mathrm{Err}(\mathbf{f})-2\cdot\widehat{\mathrm{Err}}(\mathbf{f})\geq \mathrm{t}\right)\ \leq\ \exp\left(-Cnt\right)\ .$$

Hence, combining a union bound argument together with Assumption C.1 and (13), we compute

$$\mathbb{E}[R_2(\hat{\mathbf{f}})-R_2(\mathbf{f}^*)]\leq\frac{C}{n}+CC_{\mathcal{F}}\frac{\log(n)}{n}\ . \quad (14)$$

Plugging this inequality in (12) concludes the proof. □

D Numerical experiments

Hyperparameters. The hyperparamerers are set with default parameters in scikit-learn except the number of trees for RF which is set at 500.

Splitting the sample. Our theoretical study relies on the independence of the datasets $\mathcal{D}_n$ and $\mathcal{D}'_n$. Figure 7 illustrates the importance of such condition for the fairness but also the accuracy of our proposed *argmax-fair* method. Indeed, whenever splitting is not performed (left parts of plots), the fairness performance of the fair algorithm can even be worse than the unfair method. This emphasize that splitting is crucial and enables to avoid over-fitting on the training set.

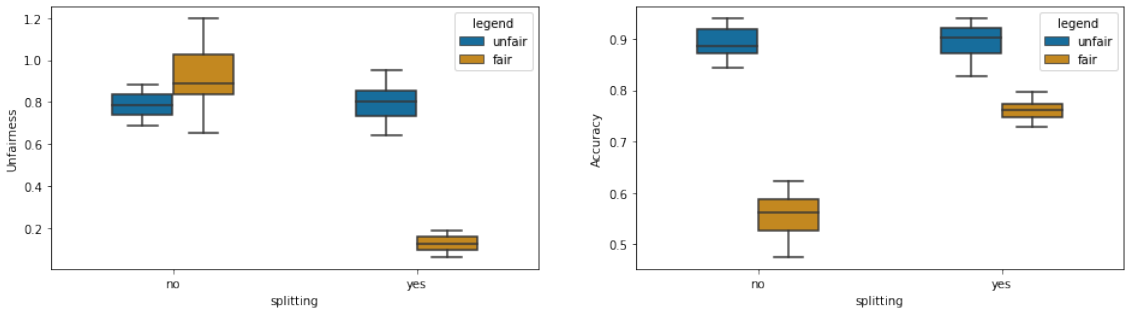

Figure 7: Empirical impact of data splitting on unfairness (Left – the lower the better) and accuracy (Right: accuracy – the higher the better). Boxplots generated over 30 repetitions with $p = 0.75$. Left: unfairness – the lower the better; Right: accuracy – the higher the better.