
“How Biased is Your Feature?”: Computing Fairness Influence Functions with Global Sensitivity Analysis

Bishwamittra Ghosh

School of Computing
National University of Singapore

Debabrota Basu

Équipe Scool, Univ. Lille, Inria, UMR 9189 - CRISTAL, CNRS, Centrale Lille, France

Kuldeep S. Meel

School of Computing
National University of Singapore

Abstract

Fairness in machine learning has attained significant focus due to the widespread application of machine learning in high-stake decision-making tasks. Unless regulated with a fairness objective, machine learning classifiers might demonstrate unfairness/bias towards certain demographic populations in the data. Thus, the quantification and mitigation of the bias induced by classifiers have become a central concern. In this paper, *we aim to quantify the influence of different features on the bias of a classifier*. To this end, we propose a framework of *Fairness Influence Function* (FIF), and compute it as a scaled difference of conditional variances in the classifier’s prediction. We also instantiate an algorithm, FairXplainer, that uses variance decomposition among the subset of features and a local regressor to compute FIFs accurately, while also capturing the intersectional effects of the features. Our experimental analysis validates that FairXplainer captures the influences of both individual features and higher-order feature interactions, estimates the bias more accurately than existing local explanation methods, and detects the increase/decrease in bias due to affirmative/punitive actions in the classifier.

1 Introduction

The last decades have witnessed a significant progress in machine learning with applications in high-stake decision making, such as college admission [30], recidivism prediction [43], job applications [1] etc. In such applications, the deployed machine learning classifier often demonstrate bias towards certain demographic groups involved in the data [13]. For example, a classifier deciding the eligibility of college admission may offer more admission to White-male candidates than to Black-female candidates—possibly because of the historical bias in the admission data, or the accuracy-centric learning objective of the classifier, or a combination of both [5, 24, 49]. Following such phenomena, multiple fairness metrics, such as *statistical parity*, *equalized odds*, *predictive parity* etc, have been proposed to quantify the bias of the classifier on a dataset. For example, if the classifier in college admission demonstrates a statistical parity of 0.6, it means that White-male candidates are offered admission 60% more than Black-female candidates [6, 15, 16].

Although fairness metrics globally quantify bias, they cannot detect or explain the sources of bias [3, 28, 35]. In order to identify the sources of bias and also the effect of affirmative/punitive

actions to alleviate/deteriorate bias, it is important to understand *which factors contribute how much to the bias of a classifier on a dataset*. To this end, we follow a feature-attribution approach to understand the sources of bias [3, 28], where we relate the *influences* of input features towards the resulting bias of the classifier. Particularly, we define and compute *Fairness Influence Function* (FIF) that quantifies the contribution of individual and subset of features to the resulting bias. FIFs do not only allow practitioners to identify the features to act up on but also to quantify the effect of various affirmative [8, 19, 23, 45–48] or punitive actions [21, 32, 42] on the resulting bias.

Our Contributions. The contribution of this paper is three-fold.

1. *Formalism:* We propose to compute individual and intersectional **Fairness Influence Functions** (FIFs) of features as a measure of contribution towards the bias of the classifier (Sec 4). The *intersectionality* [7] allows us to detect the higher order interactions among the features. In the formalism, we first axiomatize that to be a proper attribution of bias, the sum of FIFs of all subsets of the features is equal to the bias of the classifier (Axiom 1). Following that, we show that computing FIFs over subsets of features is equivalent to computing a scaled difference in the *conditional variances* (Theorem 1) of the classifier between the sensitive groups of interest. This allows us to connect global sensitivity analysis, a standard technique recommended by regulators to assess numerical models [14, 34], with computing feature influences leading to bias in the classifier’s output.

2. *Algorithmic:* We instantiate an algorithm, FairXplainer, for computing FIFs for subsets of features for a given classifier, a dataset, and any group fairness metrics, namely statistical parity, equalized odds, and predictive parity (Sec. 5). The key ideas are to import techniques from Global Sensitivity Analysis (GSA) [37] to decompose the variance of the prediction of the classifier among the subset of features and to use a local regression method [27] to compute FIFs.

3. *Experimental:* We experimentally validate that FairXplainer is significantly more accurate and can capture the intersectional or joint effect of the features on the bias induced by a classifier (Sec. 6). Improvement in accuracy and extension to intersectional FIFs are both absent in the existing works aimed to this problem [3, 28]. Also, FairXplainer enables us to detect the change in FIFs due to different fairness enhancing algorithms and fairness reducing attacks, which opens up new avenues to analyze the effects of these algorithms.

We illustrate the usefulness of our contributions via an example scenario proposed in [17].

Example 1.1. Following [17], we consider a classifier that decides an individual’s eligibility for health insurance based on non-sensitive features ‘fitness’ and ‘income’. ‘fitness’ and ‘income’ depend on a sensitive feature ‘age’ leading to two sensitive groups “young” (age < 40) and “elderly” (age ≥ 40), as highlighted in (Figure 1a–1b).

Case study 1: For each sensitive group, we generate 500 samples of (income, fitness) and train a decision tree (DT1), which predicts without explicitly using the sensitive feature ‘age’ (Figure 1c). Using standard off-the-shelf techniques, we can compute statistical parity as $\Pr[\hat{Y} = 1 | \text{age} < 40] - \Pr[\hat{Y} = 1 | \text{age} \geq 40] = 0.7 - 0.17 = 0.53$, and therefore DT1 is unfair towards “elderly”.

Applying techniques developed in this paper, we investigate the source of unfairness of DT1 by computing FIFs of features, where *positive numbers denote a reduction in fairness and negative numbers denotes fairness improvement*. In Figure 1e, fitness (FIF = 0.74), and the joint effect of fitness and income (FIF = 0.05) cause higher statistical parity, i.e. higher bias. This observation is invoked as DT1 predicts positively for the higher values of fitness (root node in DT1) and elderly individuals often possess lower fitness (Figure 1b). In contrast, the income of individuals (FIF = −0.33) decreases statistical parity, as elderly individuals have a higher income but lower fitness.

Case study 2: Since DT1 is unfair to the “elderly”, we learn another decision tree (DT2) by applying an affirmative action, where we decrease the threshold on income from 0.69 to 0.555 when the fitness is lower (green node in Figure 1d). This action allows more elderly individuals to receive insurance by including more people with lower fitness, and thus the statistical parity becomes $\Pr[\hat{Y} = 1 | \text{age} < 40] - \Pr[\hat{Y} = 1 | \text{age} \geq 40] = 0.71 - 0.7 = 0.01$. This is significantly less than the earlier statistical parity of 0.53. Again, applying techniques developed in this paper, we compute FIF of features in Figure 1f. Here, the FIF of income and fitness reflects the reduction in statistical parity as their influences almost nullify each other. Thus, FIF depicts the effects of different features and their combinations on the resultant bias incurred by the classifier.

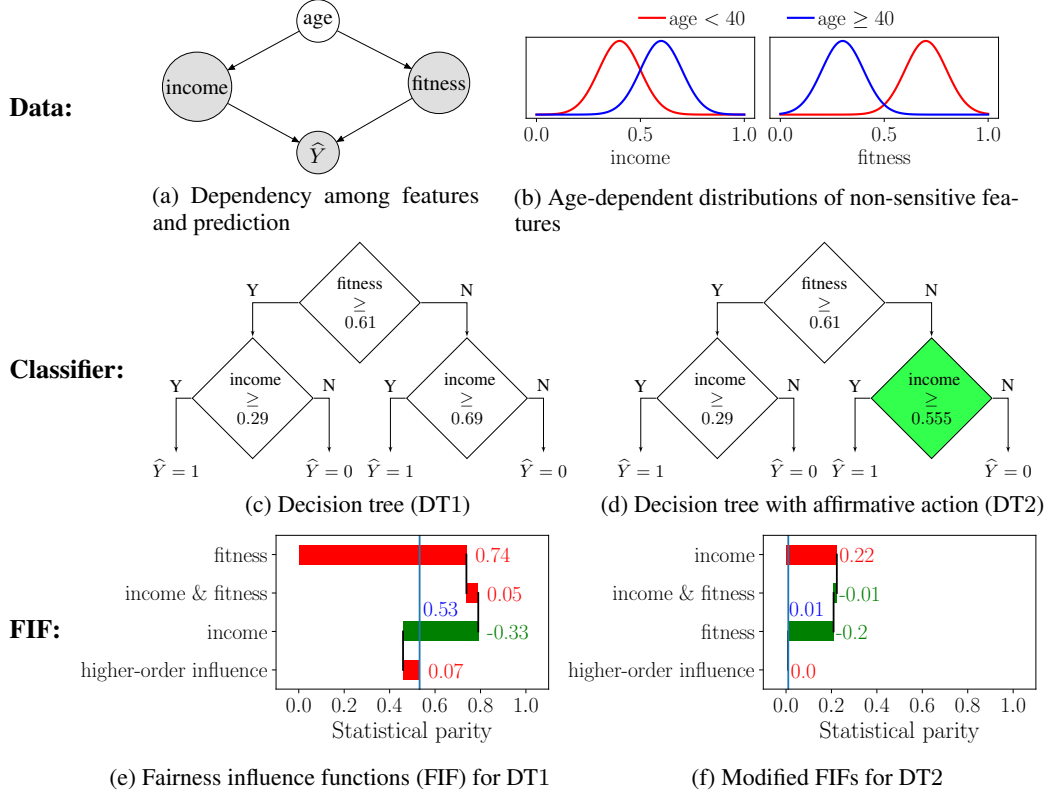


Figure 1: FIFs of input features to investigate the bias (i.e., statistical parity) of a decision tree classifier predicting the eligibility for health insurance using age-dependent features ‘fitness’ and ‘income’. An affirmative action reduces bias, and corresponding FIFs present the evidence.

2 Related Work

Recently, several studies apply *local explanation methods* of black-box prediction to explain sources of bias by feature-attribution [3, 28] and causal path decomposition [35]. Since our work adopts feature-attribution approach, it reveals two-fold limitations of existing methods: (i) *inaccuracy* in computing FIFs and (ii) *failing to compute intersectional* FIFs. Elaborately, FIF computation in [3, 28] is inaccurate because of applying local explainers such as SHAP [29] to compute global properties of the classifier such as group fairness. In addition, features are often correlated in practical fairness tasks, and computing only individual FIFs ignores the joint contribution of multiple features on the unfairness of the classifier. Also, these works provide empirical evaluations to justify the choices of SHAP-based tools for explaining fairness and does not consider the global nature of group fairness metrics. In this paper, we develop a formal framework to explain sources of unfairness in a classifier and also a novel methodology to address it. To the best of our knowledge, this is the first work to do the both. Among other related works, [4] link GSA measures such as Sobol and Cramér-von-Mises indices to different fairness metrics. While their approach relates the GSA of sensitive features on the resulting bias, we focus on applying GSA to all features to compute FIFs. Their approach only detects the presence or absence of bias, while we focus on decomposing bias as the sum of FIFs of all features. In another line of work, [9] and [18] compute feature-influence as the shift of bias from its original value by randomly intervening features. Their work is different from our axiomatic approach, where the sum of FIFs equals the bias.

3 Background: Fairness and Global Sensitivity Analysis

Before proceeding to the details of our contribution, we present the fundamentals of group fairness metrics as quantifiers of bias and global sensitivity analysis as a classical method of feature attribution.

3.1 Fairness in Machine Learning: Fairness Metrics

We consider¹ a dataset \mathbf{D} as a collection of n triples $\{(\mathbf{x}^{(i)}, \mathbf{a}^{(i)}, y^{(i)})\}_{i=1}^n$ generated from an underlying distribution \mathcal{D} . Each non-sensitive data point $\mathbf{x}^{(i)}$ consists of k features $\{\mathbf{x}_1^{(i)}, \dots, \mathbf{x}_k^{(i)}\}$. Each sensitive data point $\mathbf{a}^{(i)}$ consists of m categorical features $\{\mathbf{a}_1^{(i)}, \dots, \mathbf{a}_m^{(i)}\}$. $y^{(i)} \in \{0, 1\}$ is the binary class corresponding to $(\mathbf{x}^{(i)}, \mathbf{a}^{(i)})$. We use $(\mathbf{X}, \mathbf{A}, Y)$ to denote the random variables corresponding to $(\mathbf{x}, \mathbf{a}, y)$. We represent a binary classifier trained on the dataset \mathbf{D} as $\mathcal{M} : (\mathbf{X}, \mathbf{A}) \rightarrow \hat{Y}$. Here, $\hat{Y} \in \{0, 1\}$ is the class predicted for (\mathbf{X}, \mathbf{A}) . Given this setup, we discuss different fairness metrics to compute bias in the prediction of a classifier [15, 19, 33].

1. *Statistical Parity* (SP) [15]: Statistical parity belongs to *independence* measuring group fairness metrics, where the prediction \hat{Y} is statistically independent of sensitive features \mathbf{A} . The statistical parity of a classifier is measured as $f_{\text{SP}}(\mathcal{M}, \mathbf{D}) \triangleq \max_{\mathbf{a}} \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}] - \min_{\mathbf{a}} \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}]$, which is the difference between the maximum and minimum conditional probability of positive prediction the classifier for different sensitive groups.
2. *Equalized Odds* (EO) [19]: *Separation* measuring group fairness metrics such as equalized odds constrain that \hat{Y} is independent of \mathbf{A} given the ground class Y . Formally, for $Y \in \{0, 1\}$, equalized odds is $f_{\text{EO}}(\mathcal{M}, \mathbf{D}) \triangleq \max(\max_{\mathbf{a}} \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}, Y = 0] - \min_{\mathbf{a}} \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}, Y = 0], \max_{\mathbf{a}} \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}, Y = 1] - \min_{\mathbf{a}} \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}, Y = 1])$.
3. *Predictive Parity* (PP) [44]: *Sufficiency* measuring group fairness metrics such as predictive parity constrain that the ground class Y is independent of \mathbf{A} given the prediction \hat{Y} . Formally, $f_{\text{PP}}(\mathcal{M}, \mathbf{D}) \triangleq \max(\max_{\mathbf{a}} \Pr[Y = 1 | \mathbf{A} = \mathbf{a}, \hat{Y} = 0] - \min_{\mathbf{a}} \Pr[Y = 1 | \mathbf{A} = \mathbf{a}, \hat{Y} = 0], \max_{\mathbf{a}} \Pr[Y = 1 | \mathbf{A} = \mathbf{a}, \hat{Y} = 1] - \min_{\mathbf{a}} \Pr[Y = 1 | \mathbf{A} = \mathbf{a}, \hat{Y} = 1])$.

All of the aforementioned group fairness metrics depend on the difference between different conditional probabilities of positive prediction of a classifier. For all of these metrics, lower value of $f(\mathcal{M}, \mathbf{D})$ indicates higher fairness demonstrated by the classifier \mathcal{M} . We deploy these fairness metrics as the measures of bias of a classifier w.r.t a given dataset.

3.2 Global Sensitivity Analysis: Variance Decomposition

Global sensitivity analysis is a field that studies how the global uncertainty in the output of a function can be attributed to the different sources of uncertainties in the input while considering the whole input domain [37]. Sensitivity analysis is an essential component for quality assurance and impact assessment of models in EU [14], USA [34], and research communities [38]. *Variance-based sensitivity analysis* is a form of global sensitivity analysis, where variance is considered as the measure of uncertainty [40, 41]. To illustrate, let us consider a real-valued function $g(\mathbf{Z})$, where \mathbf{Z} is a vector of k input variables $\{Z_1, \dots, Z_k\}$. Now, we decompose $g(\mathbf{Z})$ among the subsets of inputs, such that:

$$\begin{aligned} g(\mathbf{Z}) &= g_0 + \sum_{i=1}^k g_{\{i\}}(Z_i) + \sum_{i < j}^k g_{\{i,j\}}(Z_i, Z_j) + \dots + g_{\{1,2,\dots,k\}}(Z_1, Z_2, \dots, Z_k) \\ &= g_0 + \sum_{\mathbf{S} \subseteq [k] \setminus \emptyset} g_{\mathbf{S}}(\mathbf{Z}_{\mathbf{S}}) \end{aligned} \quad (1)$$

In this decomposition, g_0 is a constant, $g_{\{i\}}$ is a function of Z_i , $g_{\{i,j\}}$ is a function of Z_i and Z_j , and so on. Here, $[k] \triangleq \{1, 2, \dots, k\}$ and \mathbf{S} is an ordered subset of $[k] \setminus \emptyset$. We denote $\mathbf{Z}_{\mathbf{S}} \triangleq \{Z_i | i \in \mathbf{S}\}$ as the input of $g_{\mathbf{S}}$, where $\mathbf{Z}_{\mathbf{S}}$ is a set of variables with indices belonging to \mathbf{S} . The standard condition of this decomposition is the orthogonality of each term in the right-hand side of Eq. (1) [40]. $g_{\mathbf{S}}(\mathbf{Z}_{\mathbf{S}})$ is the effect of varying all the features in $\mathbf{Z}_{\mathbf{S}}$ simultaneously. For $|\mathbf{S}| = 1$, it quantifies an individual variable's effect. For $|\mathbf{S}| > 1$, it quantifies the higher-order interactive effect of variables. Now, if we

¹We represent sets/vectors by bold letters, and the corresponding distributions by calligraphic letters. We express random variables in uppercase, and an assignment of a random variable in lowercase.

assume g to be square integrable, we obtain the decomposition of the variance of the output [40].

$$\text{Var}[g(\mathbf{Z})] = \sum_{i=1}^k V_{\{i\}} + \sum_{i < j}^k V_{\{i,j\}} + \cdots + V_{\{1,2,\dots,k\}} = \sum_{\mathbf{S} \subseteq [k] \setminus \emptyset} V_{\mathbf{S}} \quad (2)$$

where $V_{\{i\}}$ is the variance of $g_{\{i\}}$, $V_{\{i,j\}}$ is the variance of $g_{\{i,j\}}$ and so on. Formally, $V_{\mathbf{S}} \triangleq \text{Var}_{\mathbf{Z}_{\mathbf{S}}} [\mathbb{E}_{\mathbf{Z}/\mathbf{Z}_{\mathbf{S}}} [g(\mathbf{Z}) \mid \mathbf{Z}_{\mathbf{S}}]] - \sum_{\mathbf{S}' \subset \mathbf{S} \setminus \emptyset} V_{\mathbf{S}'}$. Here, \mathbf{S}' denotes all the ordered proper subsets of \mathbf{S} . This variance decomposition shows how the variance of $g(\mathbf{Z})$ can be decomposed into terms attributable to each input, as well as the interactive effects among them. Together all terms sum to the total variance of the model output. *We reduce the problem of computing FIFs of subsets of features into a variance decomposition problem.*

4 Fairness Influence Functions: Definition and Computation

In this section, we discuss our contribution of formalizing and computing Fairness Influence Functions (FIF) as a quantifier of the contribution of a subset of features to the resultant bias of a classifier on a dataset. The key idea in the formalization of FIFs is their additive formulation. The key idea in the computation of FIFs is to relate bias with the scaled difference of the conditional variance of positive prediction of the classifier over specific sensitive groups, and to apply variance decomposition to compute individual and intersectional FIFs.

FIF: An Axiomatic Formulation. We are given a binary classifier $\mathcal{M} : (\mathbf{X}, \mathbf{A}) \rightarrow \hat{Y}$, a dataset $\mathbf{D} = \{(\mathbf{x}^{(i)}, \mathbf{a}^{(i)}, y^{(i)})\}_{i=1}^n$, and a fairness metric $f(\mathcal{M}, \mathbf{D}) \in \mathbb{R}^{\geq 0}$. Our objective is to compute the influences of non-sensitive features on f . We compute influence of each subset of non-sensitive features $\mathbf{X}_{\mathbf{S}}$, where $\mathbf{S} = \{S_i \mid 1 \leq S_i \leq k\} \subseteq [k] \setminus \emptyset$ is a non-empty subset of indices.

Definition 1 (Fairness Influence Function). Fairness Influence Function (FIF) $w_{\mathbf{S}} : \mathbf{X}_{\mathbf{S}} \rightarrow \mathbb{R}$ measures the quantitative contribution of the subset of features $\mathbf{X}_{\mathbf{S}} \subseteq \mathbf{X}_{[k]}$ on the incurred bias $f(\mathcal{M}, \mathbf{D})$ of the classifier \mathcal{M} for dataset \mathbf{D} . We refer $w_{\mathbf{S}}$ as an *individual influence* when $|\mathbf{S}| = 1$ and an *inter-sectional influence* when $|\mathbf{S}| > 1$ —particularly, a second-order influence when $|\mathbf{S}| = 2$ and a higher-order influence when $3 \leq |\mathbf{S}| \leq k$.

Since FIF is meant to attribute the bias of the classifier to the subset of features, it is natural to choose a function such that the FIFs of all the subsets of features exactly adds up to the total bias. This leads to the following additivity axiom.

Axiom 1 (Additivity of influence). Let $f(\mathcal{M}, \mathbf{D}) \in \mathbb{R}^{\geq 0}$ be the bias of the classifier and $w_{\mathbf{S}} : \mathbf{X}_{\mathbf{S}} \rightarrow \mathbb{R}$ be the FIF of features $\mathbf{X}_{\mathbf{S}}$ on f . The additivity axiom states that the total influences of all possible subset of non-sensitive features is equal to the bias of the classifier.

$$f(\mathcal{M}, \mathbf{D}) = \sum_{\mathbf{S} \subseteq [k] \setminus \emptyset} w_{\mathbf{S}} \quad (3)$$

Computing FIF: Reduction to Variance Decomposition. Henceforth, our objective is to design an estimator of FIF $w_{\mathbf{S}}$ that satisfies Axiom 1. In the following, we discuss the closed form representation of $w_{\mathbf{S}}$ for statistical parity fairness metric and extend to other metrics in Section 5.

Connecting Statistical Parity and Conditional Variance. Let $\mathbf{a}_{\max} = \arg \max_{\mathbf{a}} \Pr[\hat{Y} = 1 \mid \mathbf{A} = \mathbf{a}]$ and $\mathbf{a}_{\min} = \arg \min_{\mathbf{a}} \Pr[\hat{Y} = 1 \mid \mathbf{A} = \mathbf{a}]$ be the most favored and the least favored sensitive group of the classifier, respectively, according to their conditional probability of positive prediction. Next, we consider a Bernoulli random variable $B_{\mathbf{a}}$ to denote the positive prediction of the classifier for the sensitive group $\mathbf{A} = \mathbf{a}$. Then, the probability of the random variable is $p_{\mathbf{a}} \triangleq \Pr[B_{\mathbf{a}} = 1] = \Pr[\hat{Y} = 1 \mid \mathbf{A} = \mathbf{a}]$, and variance is $V_{\mathbf{a}} = p_{\mathbf{a}}(1 - p_{\mathbf{a}})$. Therefore, considering two random variables $B_{\mathbf{a}_{\max}}$ and $B_{\mathbf{a}_{\min}}$, we state the statistical parity of the classifier as $p_{\mathbf{a}_{\max}} - p_{\mathbf{a}_{\min}} = (V_{\mathbf{a}_{\max}} - V_{\mathbf{a}_{\min}}) / (1 - (p_{\mathbf{a}_{\max}} + p_{\mathbf{a}_{\min}}))$, where $p_{\mathbf{a}_{\max}}$ and $p_{\mathbf{a}_{\min}}$ is the probability of positive prediction of the classifier for the group $\mathbf{A} = \mathbf{a}_{\max}$ and $\mathbf{A} = \mathbf{a}_{\min}$, respectively. Intuitively, the statistical parity of the classifier is the scaled (by $1/(1 - (p_{\mathbf{a}_{\max}} + p_{\mathbf{a}_{\min}}))$) difference of the conditional variance of the positive prediction of the classifier over the most and the least favored sensitive groups.

Example 4.1. In Example 1.1 (Figure 1c), the conditional probabilities of positive prediction of the decision tree for the most and least favored groups are $p_{\mathbf{a}_{\max}} = \Pr[\hat{Y} = 1 \mid \text{age} < 40] = 0.704$ and

$p_{a_{\min}} = \Pr[\hat{Y} = 1 | \text{age} \geq 40] = 0.172$, respectively. Thus, the statistical parity of the classifier is $p_{a_{\max}} - p_{a_{\min}} = 0.704 - 0.172 = 0.532$. Now, we compute conditional variances of positive prediction as $V_{a_{\max}} = \text{Var}[\hat{Y} = 1 | \text{age} < 40] = 0.208$ and $V_{a_{\min}} = \text{Var}[\hat{Y} = 1 | \text{age} \geq 40] = 0.142$. Thus, following our argument, we compute the scaled difference in conditional variances as $(0.208 - 0.142)/(1 - (0.702 + 0.172)) = 0.532$, which coincides with the statistical parity.

Now, we apply variance decomposition (Eq. (2)) for both the conditional variances $V_{a_{\max}}$ and $V_{a_{\min}}$ to compute the FIF of features, as stated in Theorem 1.

Theorem 1 (FIF as Difference of Conditional Variances). Let $V_{\mathbf{a}, \mathbf{S}} \triangleq \text{Var}_{\mathbf{X}_{\mathbf{S}}}[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}]$ be the decomposed conditional variance of the classifier’s positive prediction for the sensitive group $\mathbf{A} = \mathbf{a}$, where features $\mathbf{X}_{\mathbf{S}}$ are jointly varied. Then, we can express FIF of $\mathbf{X}_{\mathbf{S}}$ as

$$w_{\mathbf{S}} = \frac{V_{\mathbf{a}_{\max}, \mathbf{S}} - V_{\mathbf{a}_{\min}, \mathbf{S}}}{1 - (\Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\max}] + \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\min}])} \quad (4)$$

and FIFs defined by Equation (4) satisfy Axiom 1.

For brevity of space, we defer the detailed proof to Appendix B.

Consequences of Theorem 1. Here, we discuss the algorithmic consequences of Theorem 1 and probable issues that might appear in computation.

1. *Algorithm Design.* Expressing FIF in terms of the variance decomposition over subset of features allows us to import and extend well-studied techniques of global sensitivity analysis to perform FIF estimation accurately and at scale.
2. *Bias Amplifying and Eliminating Features.* The sign of a FIF $w_{\mathbf{S}}$ denotes whether the features $\mathbf{X}_{\mathbf{S}}$ are amplifying the bias of the classifier or eliminating it. When $w_{\mathbf{S}} > 0$, $\mathbf{X}_{\mathbf{S}}$ increases bias. When $w_{\mathbf{S}} < 0$, $\mathbf{X}_{\mathbf{S}}$ eliminates bias, and thus improves fairness. The features $\mathbf{X}_{\mathbf{S}}$ are neutral when $w_{\mathbf{S}} = 0$.
3. *Degenerate Cases:* If the classifier always makes positive prediction or never makes positive prediction, i.e. $\Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}] \in \{1, 0\}$ for any sensitive group $\mathbf{A} = \mathbf{a}$, the conditional variance $\text{Var}[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}] = 0$. When total variance is zero, the decomposed variances $V_{\mathbf{a}, \mathbf{S}}$ may become zero for each feature combination $\mathbf{X}_{\mathbf{S}}$, and Equation (4) cannot trivially compute fairness influence functions. Though such degenerate cases rarely occur for real-life data, we can avoid them by randomly perturbing at least one sample in the dataset to yield the opposite predicted class.
4. *Numerical Instability.* In Equation (4), the conditional probabilities of positive prediction of the classifier $\Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\max}]$ and $\Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\min}]$ are constant. When $\Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\max}] + \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\min}] = 1$, the denominator in $w_{\mathbf{S}}$ becomes 0 and hence, $w_{\mathbf{S}}$ is undefined. In this case, we add/subtract a small number in the denominator to avoid numerical instability of division though we have not encountered any such case in our experiments.

5 FairXplainer: An Algorithm to Compute Fairness Influence Functions

We propose an algorithm, FairXplainer, that leverages the variance decomposition of Eq. (4) to compute the Fairness Influence Functions (FIFs) of all the subset of features. FairXplainer has two algorithmic blocks: (i) local regression to decompose the classifier into component functions and (ii) computing the variance (or covariance) of each component. We describe the schematic of FairXplainer in Algorithm 1.

A Set-additive Representation of the Classifier. To apply variance decomposition (Eq. (2)), we learn a set-additive representation of the classifier (Eq. (1)). Let us denote the classifier \mathcal{M} conditioned on a sensitive group \mathbf{a} as $g_{\mathbf{a}}(\mathbf{X}) \triangleq \mathcal{M}(\mathbf{X}, \mathbf{A} = \mathbf{a})$. We express $g_{\mathbf{a}}$ as a set-additive model:

$$g_{\mathbf{a}}(\mathbf{X}) = g_{\mathbf{a},0} + \sum_{\mathbf{S} \subseteq [k] \setminus \emptyset, |\mathbf{S}| \leq \lambda} g_{\mathbf{a}, \mathbf{S}}(\mathbf{X}_{\mathbf{S}}) + \delta \quad (5)$$

Here, $g_{\mathbf{a},0}$ is a constant and $g_{\mathbf{a}, \mathbf{S}}$ is a *component function* of $g_{\mathbf{a}}$ taking $\mathbf{X}_{\mathbf{S}}$ as input, and δ is the approximation error. For computational tractability, we consider only components of *maximum order* λ . FairXplainer deploys backfitting algorithm for learning component functions in Eq. (5).

Algorithm 1 FairXplainer

Input: $\mathcal{M} : (\mathbf{X}, \mathbf{A}) \rightarrow \hat{Y}$, $\mathbf{D} = \{(\mathbf{x}^{(i)}, \mathbf{a}^{(i)}, y^{(i)})\}_{i=1}^n$, $f(\mathcal{M}, \mathbf{D}) \in \mathbb{R}^{\geq 0}$, λ
Output: w_S

- 1: $\mathbf{a}_{\max} = \arg \max_{\mathbf{a}} \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}]$, $\mathbf{a}_{\min} = \arg \min_{\mathbf{a}} \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}]$, $k \leftarrow |\mathbf{X}|$
- 2: **for** $\mathbf{a} \in \{\mathbf{a}_{\max}, \mathbf{a}_{\min}\}$ **do** ▷ Enumerate for specific sensitive groups
- 3: $g_{\mathbf{a}, \mathbf{S}}, g_{\mathbf{a}, 0} \leftarrow \text{LOCALREGRESSION}(\mathcal{M}(\mathbf{X}, \mathbf{A} = \mathbf{a}), \{\mathbf{x}^{(i)}\}_{i=1}^n, \lambda, k)$
- 4: $V_{\mathbf{a}, \mathbf{S}} \leftarrow \text{COVARIANCE}(g_{\mathbf{a}}, \{\mathbf{x}^{(i)}\}_{i=1}^n, g_{\mathbf{a}, \mathbf{S}}, g_{\mathbf{a}, 0})$
- 5: Compute w_S using $V_{\mathbf{a}_{\max}, \mathbf{S}}$ and $V_{\mathbf{a}_{\min}, \mathbf{S}}$ as in Equation (4) ▷ Theorem 1
- 6: **function** $\text{LOCALREGRESSION}(g_{\mathbf{a}}, \{\mathbf{x}^{(i)}\}_{i=1}^n, \lambda, k)$
- 7: **Initialize:** $g_{\mathbf{a}, 0} \leftarrow \text{MEAN}(\{g(\mathbf{x}^{(i)})\}_{i=1, \mathbf{a}^{(i)}=\mathbf{a}}^n)$, $\hat{g}_{\mathbf{a}, \mathbf{S}} \leftarrow 0$, $\forall \mathbf{S} \in [k] \setminus \emptyset, |\mathbf{S}| \leq \lambda$
- 8: **while** each $\hat{g}_{\mathbf{a}, \mathbf{S}}$ does not converge **do**
- 9: **for** each \mathbf{S} **do**
- 10: $\hat{g}_{\mathbf{a}, \mathbf{S}} \leftarrow \text{SMOOTH}[\{g_{\mathbf{a}}(\mathbf{x}^{(i)}) - g_{\mathbf{a}, 0} - \sum_{\mathbf{S}' \neq \mathbf{S}} \hat{g}_{\mathbf{a}, \mathbf{S}'}(\mathbf{x}_{\mathbf{S}}^{(i)})\}_{i=1, \mathbf{a}^{(i)}=\mathbf{a}}^n]$ ▷ Backfitting
- 11: $\hat{g}_{\mathbf{a}, \mathbf{S}} \leftarrow \hat{g}_{\mathbf{a}, \mathbf{S}} - \text{MEAN}(\{\hat{g}_{\mathbf{a}, \mathbf{S}}(\mathbf{x}_{\mathbf{S}}^{(i)})\}_{i=1, \mathbf{a}^{(i)}=\mathbf{a}}^n)$ ▷ Mean centering
- 12: **return** $g_{\mathbf{a}, \mathbf{S}}, g_{\mathbf{a}, 0}$
- 13: **function** $\text{COVARIANCE}(g_{\mathbf{a}}, \{\mathbf{x}^{(i)}\}_{i=1}^n, g_{\mathbf{a}, \mathbf{S}}, g_{\mathbf{a}, 0})$
- 14: **return** $\sum_{i=1, \mathbf{a}^{(i)}=\mathbf{a}}^n g_{\mathbf{a}, \mathbf{S}}(\mathbf{x}_{\mathbf{S}}^{(i)})(g_{\mathbf{a}}(\mathbf{x}^{(i)}) - g_{\mathbf{a}, 0})$

Local Regression with Backfitting. We perform local regression with backfitting algorithm to learn the component functions up to order λ (Line 6–12). Backfitting algorithm is an iterative algorithm, where in each iteration one component function, say $g_{\mathbf{a}, \mathbf{S}}$, is learned while keeping other component functions fixed. Specifically, $g_{\mathbf{a}, \mathbf{S}}$ is learned as a smoothed function of g and rest of the components $g_{\mathbf{a}, \mathbf{S}'}$, where $\mathbf{S}' \neq \mathbf{S}$ is a non-empty subset of $[k] \setminus \emptyset$. To keep every component function mean centered, backfitting requires to impose the constraints $g_{\mathbf{a}, 0} = \text{MEAN}(\{g(\mathbf{x}^{(i)})\}_{i=1, \mathbf{a}^{(i)}=\mathbf{a}}^n)$ (Line 7), which is the mean of $g_{\mathbf{a}}$ evaluated on samples belonging to the sensitive group $\mathbf{A} = \mathbf{a}$; and $\sum_{i=1, \mathbf{a}^{(i)}=\mathbf{a}}^n g_{\mathbf{a}, \mathbf{S}}(\mathbf{x}_{\mathbf{S}}^{(i)}) = 0$ (Line 11), where $\mathbf{x}_{\mathbf{S}}^{(i)}$ be the subset of feature values associated with feature indices \mathbf{S} for the i -th sample $\mathbf{x}^{(i)}$. These constraints assign the expectation of $g_{\mathbf{a}}$ on the constant term $g_{\mathbf{a}, 0}$ and the variance of $g_{\mathbf{a}}$ to the component functions.

While performing local regression, backfitting uses a smoothing operator [26] over the set of samples (Line 10). A smoothing operator, referred as SMOOTH , allows us to learn a global representation of a component function by smoothly interpolating the local curves obtained by local regression [26]. In this paper, we apply cubic spline smoothing [25] to learn each component function. Cubic spline is a piecewise polynomial of degree 3 with C^2 continuity. Hence, up to the second derivatives of each piecewise term are zero at the endpoints of intervals. We refer to Appendix C for further details.

Variance and Covariance Computation. Once each component function $g_{\mathbf{a}, \mathbf{S}}$ is learned with LOCALREGRESSION (Line 6–12), we compute variances of the component functions and their covariances with $g_{\mathbf{a}}$. Since each component function is mean centered (Line 11), we compute the variance of $g_{\mathbf{a}, \mathbf{S}}$ on the dataset as $\text{Var}[g_{\mathbf{a}, \mathbf{S}}] = \sum_{i=1, \mathbf{a}^{(i)}=\mathbf{a}}^n (g_{\mathbf{a}, \mathbf{S}}(\mathbf{x}_{\mathbf{S}}^{(i)}))^2$. Hence, variance captures the independent effect of $g_{\mathbf{a}, \mathbf{S}}$. Covariance is computed to account for the correlation among features \mathbf{X} . We compute the covariance of $g_{\mathbf{a}, \mathbf{S}}$ with $g_{\mathbf{a}}$ on the dataset as $\text{Cov}[g_{\mathbf{a}, \mathbf{S}}, g_{\mathbf{a}}] = \sum_{i=1, \mathbf{a}^{(i)}=\mathbf{a}}^n g_{\mathbf{a}, \mathbf{S}}(\mathbf{x}_{\mathbf{S}}^{(i)})(g_{\mathbf{a}}(\mathbf{x}^{(i)}) - g_{\mathbf{a}, 0})$. Here, $g_{\mathbf{a}}(\cdot) - g_{\mathbf{a}, 0}$ is the mean centered form of $g_{\mathbf{a}}$. Covariance of $g_{\mathbf{a}, \mathbf{S}}$ can be both positive and negative depending on whether the features $\mathbf{X}_{\mathbf{S}}$ are positively or negatively correlated with $g_{\mathbf{a}}$. Specifically, under the set additive model, we obtain $\text{Cov}[g_{\mathbf{a}, \mathbf{S}}, g_{\mathbf{a}}] = \text{Var}[g_{\mathbf{a}, \mathbf{S}}] + \text{Cov}[g_{\mathbf{a}, \mathbf{S}}, \sum_{\mathbf{S}' \neq \mathbf{S}} g_{\mathbf{a}, \mathbf{S}'}]$. Now, we use $V_{\mathbf{a}, \mathbf{S}} = \text{Cov}[g_{\mathbf{a}, \mathbf{S}}, g_{\mathbf{a}}]$ as the effective variance of $\mathbf{X}_{\mathbf{S}}$ for a given sensitive group $\mathbf{A} = \mathbf{a}$ (Line 13–14). In Line 1–5, we compute $V_{\mathbf{a}, \mathbf{S}}$ for the most and the least favored groups, and plug them in Theorem 1 to compute FIF of $\mathbf{X}_{\mathbf{S}}$.

Extension of FairXplainer to Equalized Odds and Predictive Parity. We extend FairXplainer in computing FIF of group fairness metrics beyond statistical parity, namely equalized odds and predictive parity. For equalized odds, we deploy FairXplainer twice, one for computing FIFs on a subset of samples in the dataset where $Y = 1$ and another on samples with $Y = 0$. Then, the maximum of the sum of FIFs between $Y = 1$ and $Y = 0$ quantifies the equalized odds of the classifier.

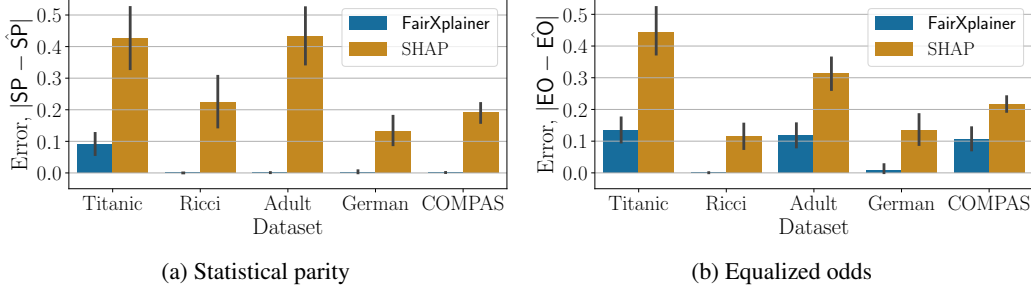


Figure 2: Comparing FairXplainer and SHAP on the estimation error of fairness metrics. Lower values on the Y-axis denote a better result. FairXplainer has significantly less error than SHAP.

To compute FIFs for predictive parity, we condition the dataset by the predicted class \hat{Y} and separate into two sub-datasets: $\hat{Y} = 1$ and $\hat{Y} = 0$. For each sub-dataset, we deploy FairXplainer by setting the ground-truth class Y as label. This contrasts the computation for statistical parity and equalized odds, where the predicted class \hat{Y} is considered as label. Finally, the maximum of the sum of FIFs between two sub-datasets for $\hat{Y} = 1$ and $\hat{Y} = 0$ quantifies the predictive parity of the classifier.

6 Empirical Performance Analysis

In this section, we perform an empirical evaluation of FairXplainer. In the following, we discuss the experimental setup, the objectives of experiments, and experimental results. We adjoin the source code and additional experimental results on the effects of maximum order λ , runtime, accuracy on different datasets and corresponding explanations (Appendix D) to the supplementary material.

Experimental Setup. We implement a prototype of FairXplainer in Python (version 3.8). In FairXplainer, we deploy SALib library [20] to compute FIFs leveraging techniques of global sensitivity analysis. In our experiments, we consider five widely studied datasets from fairness literature, namely COMPAS [2], Adult [12], German-credit [11], Ricci [31], and Titanic (<https://www.kaggle.com/c/titanic>). We deploy Scikit-learn [36] to learn different classifiers: Logistic Regression, Support Vector Machine, Decision Tree, and Neural Networks. In experiments, we specify FairXplainer to compute intersectional influences up to second order ($\lambda = 2$). We compare FairXplainer with Shapley-valued based FIF computational framework, referred as SHAP [28]. In addition, we deploy FairXplainer along with different fairness-enhancing [22] and fairness attack [42] algorithms, and analyze the effect of these algorithms on the FIFs and the resultant fairness metric. In the following, we discuss the objective of our empirical study.

1. How do **accuracies** of FairXplainer and SHAP compare for estimating FIFs and resulting bias?
2. What is the impact of **intersectional vs. individual FIFs** in tracing the sources of bias?
3. How do FIFs change while **applying** different fairness enhancing algorithms, i.e. **affirmative actions**, and fairness attacks, i.e. **punitive actions**?

In summary, FairXplainer *achieves significantly less estimation error than SHAP in all the datasets*. This shows the importance of adopting global sensitivity analysis to compute FIFs for group fairness metrics than local explanation approaches. Moreover, FairXplainer *effectively traces the sources of bias by computing intersectional FIFs, which earlier method like SHAP did not capture*. FairXplainer also detects the effects of the affirmative and punitive actions on the bias of a classifier and the corresponding tensions between different subsets of features.

Accurate Estimation of Bias. We compare FairXplainer with SHAP in estimating Statistical Parity (SP) and Equalized Odds (EO), where each metric is calculated by summing all FIFs (Axiom 1). We consider five datasets, each trained on different classifiers by applying five-fold cross-validation. We compute estimation error by taking the absolute difference between the exact and estimated value of a metric, and present results in Figure 2. In both metrics, FairXplainer demonstrates significantly less error than SHAP. For example, the mean error of FairXplainer in estimating SP on Titanic dataset is 0.1 vs. 0.42 for SHAP. In this context, the error of FairXplainer is often zero in most datasets and only insignificant (≤ 0.13) due to the degenerate cases (Section 4) with $\Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}] \in \{0, 1\}$.

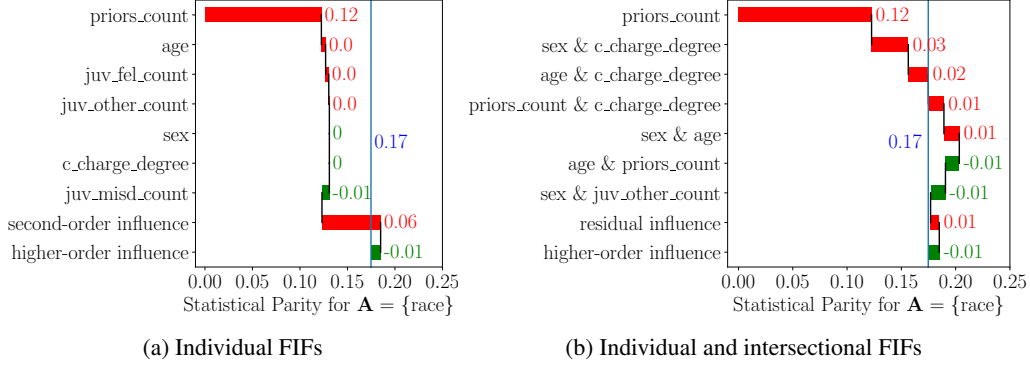


Figure 3: FIFs for COMPAS dataset on explaining statistical parity. Intersectional FIFs depict sources of bias in detail than individual FIFs.

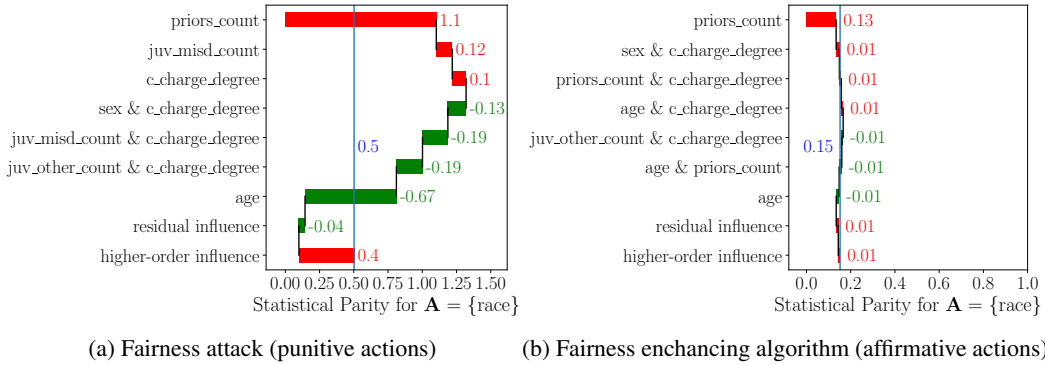


Figure 4: Effects of a fairness attack [42] and a fairness enhancing algorithm [22] on FIFs.

Therefore, *global sensitivity analysis based approach FairXplainer is significantly more accurate in computing FIFs and corresponding fairness metric than local explanation based approach SHAP.*

Individual vs. Intersectional FIFs. Now, we aim to understand the importance of intersectional FIFs over individual FIFs. We consider COMPAS dataset with $\text{race} = \{\text{Caucasian}, \text{non-Caucasian}\}$ as the sensitive feature, and a logistic regression classifier to predict whether a person will re-offend crimes within the next two years. Since the classifier only optimizes training error, it demonstrates statistical parity as 0.17, which means a non-Caucasian has 0.17 higher probability of re-offending crimes than a Caucasian. Next, we investigate the source of bias and present individual FIFs in Figure 3a, and both individual and intersectional FIFs in Figure 3b. In both figures, we show top seven influential FIFs sorted by absolute values, followed by residual higher-order FIFs. In Figure 3a, ‘priors count’ of an individual is dominating in increasing statistical parity (FIF = 0.12). Other non-sensitive features appear to have almost zero FIFs. However, the sum of second-order influences of all features increases statistical parity by 0.06, denoting that the data is highly correlated and presenting only individual FIFs does not trace the true sources of bias. For example, while both ‘sex’ and ‘age’ of persons have zero influence on bias (Figure 3a), these features together with ‘c charge degree’, ‘priors count’, and ‘juvenile other count’ contribute highly on statistical parity (sum of FIFs = $0.03 + 0.02 + 0.01 + 0.01 - 0.01 - 0.01 = 0.05$). *Therefore, intersectional influences together with individual influences lead to a clearer understanding on the source of bias of a classifier. Note that, unlike SHAP, FairXplainer is the only framework that computes beyond individual FIFs.*

Fairness affirmative/punitive actions. Continuing on the experiment in Figure 3, we evaluate the effect of fairness attack and enhancing algorithms on FIFs for COMPAS dataset in Figure 4. In Figure 3, statistical parity of the classifier is 0.17. Applying a data poisoning fairness attack [42] increases statistical parity to 0.5 and the data reweighing-based fairness-enhancing algorithm [22] decreases it to 0.15. We observe that in both cases, there are a subset of features decreasing bias while another subset of features increasing it, e.g. ‘age’ vs. ‘priors count’. Figure 4a suggests that the attack algorithm would be more successful if it could hide the influence of ‘age’ of a person in receiving discriminating prediction for re-offending crimes. Figure 4b suggests that the fairness enhancing

algorithm can improve by ameliorating the effect of ‘priors count’ further. Thus, FairXplainer provides a dissecting tool to undertake necessary steps to improve or worsen fairness of the classifier.

7 Conclusion

We propose the Fairness Influence Function (FIF) to quantify the contribution of input features on the resulting bias of a classifier for a given dataset. Relying on an additive axiom, we express group fairness metrics computed for sensitive groups as the sum of FIFs of all the subsets of non-sensitive features. To compute FIFs, we first prove existing group fairness metrics as the scaled difference of the conditional variances in the predictions of the classifier and then apply variance decomposition based on global sensitivity analysis. Finally, we propose FairXplainer, an algorithm for efficiently and accurately computing FIFs by deploying a local regression to learn a set-additive decomposition of the classifier. The experimental results show that FairXplainer estimates bias with significantly higher accuracy than the local explanation based approach SHAP. Also, FairXplainer computes both individual and intersectional FIFs unlike SHAP to yield a better understanding of the sources of bias. In future, we aim to develop algorithms that leverage FIFs to yield unbiased decisions.

References

- [1] Ifeoma Ajunwa, Sorelle Friedler, Carlos E Scheidegger, and Suresh Venkatasubramanian. Hiring by algorithm: predicting and preventing disparate impact. *Available at SSRN*, 2016. URL: <http://sorelle.friedler.net/papers/SSRN-id2746078.pdf>.
- [2] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias risk assessments in criminal sentencing. *ProPublica*, May, 23, 2016.
- [3] Tom Begley, Tobias Schwedes, Christopher Frye, and Ilya Feige. Explainability for fair machine learning. *arXiv preprint arXiv:2010.07389*, 2020.
- [4] Clément Bénése, Fabrice Gamboa, Jean-Michel Loubes, and Thibaut Boissin. Fairness seen as global sensitivity analysis. *arXiv preprint arXiv:2103.04613*, 2021.
- [5] Richard Berk. Accuracy and fairness for juvenile justice risk assessments. *Journal of Empirical Legal Studies*, 16(1):175–194, 2019.
- [6] Philippe Besse, Eustasio del Barrio, Paula Gordaliza, Jean-Michel Loubes, and Laurent Risser. A survey of bias in machine learning through the prism of statistical parity. *The American Statistician*, pages 1–11, 2021.
- [7] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018.
- [8] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems*, pages 3992–4001, 2017.
- [9] Anupam Datta, Shayak Sen, and Yair Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE symposium on security and privacy (SP)*, pages 598–617. IEEE, 2016.
- [10] Carl de Boor. Subroutine package for calculating with b-splines. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 1971.
- [11] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.
- [12] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL <http://archive.ics.uci.edu/ml>.
- [13] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

- [14] European Commission. Better regulation toolbox, 2021.
- [15] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 259–268, 2015.
- [16] Pratyush Garg, John Villasenor, and Virginia Foggo. Fairness metrics: A comparative analysis. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 3662–3666. IEEE, 2020.
- [17] Bishwamittra Ghosh, Debabrota Basu, and Kuldeep S. Meel. Justicia: A stochastic SAT approach to formally verify fairness. In *Proceedings of AAAI*, 2 2021.
- [18] Bishwamittra Ghosh, Debabrota Basu, and Kuldeep S. Meel. Algorithmic fairness verification with graphical models. In *Proceedings of AAAI*, 2 2022.
- [19] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.
- [20] Jon Herman and Will Usher. SALib: An open-source python library for sensitivity analysis. *The Journal of Open Source Software*, 2(9), jan 2017. doi: 10.21105/joss.00097. URL <https://doi.org/10.21105/joss.00097>.
- [21] Xinru Hua, Huanzhong Xu, Jose Blanchet, and Viet Nguyen. Human imperceptible attacks and applications to improve fairness. *arXiv preprint arXiv:2111.15603*, 2021.
- [22] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.
- [23] Faisal Kamiran, Asim Karim, and Xiangliang Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929. IEEE, 2012.
- [24] Frank J Landy, Janet L Barnes, and Kevin R Murphy. Correlates of perceived fairness and accuracy of performance evaluation. *Journal of Applied psychology*, 63(6):751, 1978.
- [25] Genyuan Li, Herschel Rabitz, Paul E Yelvington, Oluwayemisi O Oluwole, Fred Bacon, Charles E Kolb, and Jacqueline Schoendorf. Global sensitivity analysis for systems with independent and/or correlated inputs. *The journal of physical chemistry A*, 114(19):6022–6032, 2010.
- [26] Catherine Loader. Smoothing: local regression techniques. In *Handbook of computational statistics*, pages 571–596. Springer, 2012.
- [27] Clive Loader. *Local regression and likelihood*. Springer Science & Business Media, 2006.
- [28] Scott M Lundberg. Explaining quantitative measures of fairness. In *Fair & Responsible AI Workshop@ CHI2020*, 2020.
- [29] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, pages 4765–4774, 2017.
- [30] Barbara Martinez Neda, Yue Zeng, and Sergio Gago-Masague. Using machine learning in admissions: Reducing human and algorithmic bias in the selection process. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, pages 1323–1323, 2021.
- [31] Ann C McGinley. Ricci v. DeStefano: A masculinities theory analysis. *Harv. JL & Gender*, 33: 581, 2010.
- [32] Ninareh Mehrabi, Muhammad Naveed, Fred Morstatter, and Aram Galstyan. Exacerbating algorithmic bias through fairness attacks. *arXiv preprint arXiv:2012.08723*, 2020.
- [33] Razieh Nabi and Ilya Shpitser. Fair inference on outcomes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

- [34] Office of the Science Advisor, Council for Regulatory Environmental Modeling. Guidance on the development, evaluation, and application of environmental models, 2009. https://web.archive.org/web/20110426180258/http://www.epa.gov/CREM/library/cred_guidance_0309.pdf.
- [35] Weishen Pan, Sen Cui, Jiang Bian, Changshui Zhang, and Fei Wang. Explaining algorithmic fairness through fairness-aware causal path decomposition. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1287–1297, 2021.
- [36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [37] Andrea Saltelli, Marco Ratto, Terry Andres, Francesca Campolongo, Jessica Cariboni, Debora Gatelli, Michaela Saisana, and Stefano Tarantola. *Global sensitivity analysis: the primer*. John Wiley & Sons, 2008.
- [38] Andrea Saltelli, Gabriele Bammer, Isabelle Bruno, Erica Charters, Monica Di Fiore, Emmanuel Didier, Wendy Nelson Espeland, John Kay, Samuele Lo Piano, Deborah Mayo, et al. Five ways to ensure that models serve society: a manifesto, 2020.
- [39] Larry Schumaker. *Spline functions: basic theory*. Cambridge University Press, 2007.
- [40] Il’ya M Sobol’. On sensitivity estimation for nonlinear mathematical models. *Matematicheskoe modelirovanie*, 2(1):112–118, 1990.
- [41] Il’ya M Sobol’. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and computers in simulation*, 55(1-3):271–280, 2001.
- [42] David Solans, Battista Biggio, and Carlos Castillo. Poisoning attacks on algorithmic fairness. *arXiv preprint arXiv:2004.07401*, 2020.
- [43] Nikolaj Tollenaar and PGM Van der Heijden. Which method predicts recidivism best?: a comparison of statistical, machine learning and data mining predictive models. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 176(2):565–584, 2013.
- [44] Sahil Verma and Julia Rubin. Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*, pages 1–7. IEEE, 2018.
- [45] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333, 2013.
- [46] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018.
- [47] Junzhe Zhang and Elias Bareinboim. Fairness in decision-making—the causal explanation formula. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.
- [48] Wenbin Zhang and Eirini Ntoutsi. Faht: an adaptive fairness-aware decision tree classifier. *arXiv preprint arXiv:1907.07237*, 2019.
- [49] Indre Zliobaite. On the relation between accuracy and fairness in binary classification. *arXiv preprint arXiv:1505.05723*, 2015.

Appendix

A Societal Impact

In this paper, we quantify the influence of input features on the incurred bias/unfairness of the classifier on a dataset. This quantification facilitates our understanding of potential features or subsets of features attributing highly to the bias. This also allows us to understand the effect of the fairness enhancing algorithms in removing bias and their achievement/failure to do so. To the best of our understanding, this paper does not have any negative societal impact.

B FIF as Difference of Conditional Variances: Proof of Theorem 1

Theorem 1 (FIF as Difference of Conditional Variances). Let $V_{\mathbf{a},\mathbf{S}} \triangleq \text{Var}_{\mathbf{X}_{\mathbf{S}}}[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}]$ be the decomposed conditional variance of the classifier's positive prediction for the sensitive group $\mathbf{A} = \mathbf{a}$, where features $\mathbf{X}_{\mathbf{S}}$ are jointly varied. Let \mathbf{a}_{\max} and \mathbf{a}_{\min} be the most and the least favored groups, respectively. Then, FIF of $\mathbf{X}_{\mathbf{S}}$ corresponding to statistical parity is

$$w_{\mathbf{S}} = \frac{V_{\mathbf{a}_{\max},\mathbf{S}} - V_{\mathbf{a}_{\min},\mathbf{S}}}{1 - (\Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\max}] + \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\min}])}, \quad (4)$$

and FIFs defined by Equation (4) satisfy Axiom 1.

Proof. Let $B_{\mathbf{a}} \in \{0, 1\}$ be a Bernoulli random variable such that $B_{\mathbf{a}} = 1$ denotes the classifier predicting positive class $\hat{Y} = 1$ for an individual belonging to the sensitive group $\mathbf{A} = \mathbf{a}$, and $B_{\mathbf{a}} = 0$ denotes $\hat{Y} = 0$. Hence, if $p_{\mathbf{a}} \triangleq \Pr[B_{\mathbf{a}} = 1]$, then $p_{\mathbf{a}}$ is also the conditional probability of positive prediction of the classifier, $p_{\mathbf{a}} = \Pr[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}]$. The variance of $B_{\mathbf{a}}$ is computed as

$$\text{Var}[B_{\mathbf{a}}] = p_{\mathbf{a}}(1 - p_{\mathbf{a}}).$$

Now, we consider two Bernoulli random variables $B_{\mathbf{a}}$ and $B_{\mathbf{a}'}$ associated with two different sensitive groups $\mathbf{A} = \mathbf{a}$ and $\mathbf{A} = \mathbf{a}'$, respectively. The difference in variances between $B_{\mathbf{a}}$ and $B_{\mathbf{a}'}$ implies that

$$\begin{aligned} \text{Var}[B_{\mathbf{a}}] - \text{Var}[B_{\mathbf{a}'}] &= p_{\mathbf{a}}(1 - p_{\mathbf{a}}) - p_{\mathbf{a}'}(1 - p_{\mathbf{a}'}) \\ &= p_{\mathbf{a}} - p_{\mathbf{a}}^2 - p_{\mathbf{a}'} + p_{\mathbf{a}'}^2 \\ &= (p_{\mathbf{a}} - p_{\mathbf{a}'})(1 - (p_{\mathbf{a}} + p_{\mathbf{a}'})) \end{aligned}$$

After reorganization, we express the difference in probabilities $p_{\mathbf{a}} - p_{\mathbf{a}'}$ as

$$p_{\mathbf{a}} - p_{\mathbf{a}'} = \frac{\text{Var}[B_{\mathbf{a}}] - \text{Var}[B_{\mathbf{a}'}]}{1 - (p_{\mathbf{a}} + p_{\mathbf{a}'})} \quad (6)$$

By setting $\mathbf{a} = \mathbf{a}_{\max}$ and $\mathbf{a}' = \mathbf{a}_{\min}$ in Equation (6), we express the statistical parity of the classifier in terms of the scaled difference in the conditional variance of the positive prediction of the classifier.

$$\begin{aligned} f_{\text{SP}}(\mathcal{M}, \mathbf{D}) &\triangleq p_{\mathbf{a}_{\max}} - p_{\mathbf{a}_{\min}} = \frac{\text{Var}[B_{\mathbf{a}_{\max}}] - \text{Var}[B_{\mathbf{a}_{\min}}]}{1 - (p_{\mathbf{a}_{\max}} + p_{\mathbf{a}_{\min}})} \\ &= \frac{\text{Var}[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\max}] - \text{Var}[\hat{Y} = 1 | \mathbf{A} = \mathbf{a}_{\min}]}{1 - (p_{\mathbf{a}_{\max}} + p_{\mathbf{a}_{\min}})}. \end{aligned} \quad (7)$$

Next, we apply variance decomposition (Equation (2)) to estimate the influence of each $\mathbf{X}_{\mathbf{S}}$. From Equation (2) in Section 3, we obtain that the variance of $B_{\mathbf{a}}$ can be decomposed as

$$\text{Var}[B_{\mathbf{a}}] = \sum_{\mathbf{S} \subseteq [k] \setminus \emptyset} V_{\mathbf{a},\mathbf{S}},$$

where $V_{\mathbf{a},\mathbf{S}}$ denotes the decomposed variance of $B_{\mathbf{a}}$ w.r.t. the features $\mathbf{X}_{\mathbf{S}}$ conditioned on the sensitive group $\mathbf{A} = \mathbf{a}$.

Now, we apply variance decomposition on both $\text{Var}[B_{\mathbf{a}_{\max}}]$ and $\text{Var}[B_{\mathbf{a}_{\min}}]$ in Equation 7 to obtain

$$f_{\text{SP}}(\mathcal{M}, \mathbf{D}) = \frac{\text{Var}[B_{\mathbf{a}_{\max}}] - \text{Var}[B_{\mathbf{a}_{\min}}]}{1 - (p_{\mathbf{a}_{\max}} + p_{\mathbf{a}_{\min}})} = \sum_{\mathbf{S} \subseteq [k] \setminus \emptyset} \frac{V_{\mathbf{a}_{\max}, \mathbf{S}} - V_{\mathbf{a}_{\min}, \mathbf{S}}}{1 - (p_{\mathbf{a}_{\max}} + p_{\mathbf{a}_{\min}})} \quad (8)$$

Finally, we estimate the FIF of the subset of features $\mathbf{X}_{\mathbf{S}}$ as

$$w_{\mathbf{S}} = \frac{V_{\mathbf{a}_{\max}, \mathbf{S}} - V_{\mathbf{a}_{\min}, \mathbf{S}}}{1 - (p_{\mathbf{a}_{\max}} + p_{\mathbf{a}_{\min}})} \quad (9)$$

by separating the terms corresponding to $\mathbf{X}_{\mathbf{S}}$.

Following Equation (8), $w_{\mathbf{S}}$ satisfies Axiom 1 as $f_{\text{SP}}(\mathcal{M}, \mathbf{D}) = \sum_{\mathbf{S} \subseteq [k] \setminus \emptyset} w_{\mathbf{S}}$. \square

C A Smoothing Operator SMOOTH: Cubic Splines

In the LOCALREGRESSION module of FairXplainer (Line 6–12, Algorithm 1), we use a smoothing operator SMOOTH (Line 10). In our experiments, *we use cubic splines as the smoothing operator*. Here, we elucidate the technical details of cubic splines.

In interpolation problems, a B-spline of order n is traditionally used to smoothen the intersection of piecewise interpolators [39]. A B-spline of degree n is a piecewise polynomial of degree $n - 1$ defined over a variable X . Each piecewise term is computed on local points and is aggregated as a global curve smoothly fitting the data. The values of X where the polynomial pieces meet together are called knots, and are denoted by $\{\dots, t_0, t_1, t_2, \dots\}$.

Let $B_{r,n}(X)$ denote the basis function for a B-spline of order n , and r is the index of the knot vector. According to Carl de Boor [10], $B_{r,1}(X)$, for $n = 1$, is defined as

$$B_{r,1}(X) = \begin{cases} 0 & \text{if } X < t_r \text{ or } X \geq t_{r+1}, \\ 1 & \text{otherwise} \end{cases}$$

This definition satisfies $\sum_i B_{r,1}(X) = 1$. The higher order basis functions are defined recursively as

$$B_{r,n+1}(X) = p_{r,n}(X)B_{r,n}(X) + (1 - p_{r+1,n}(X))B_{r+1,n}(X),$$

where

$$p_{r,n}(X) = \begin{cases} \frac{X - t_r}{t_{r+n} - t_r} & \text{if } t_{r+n} \neq t_r, \\ 0 & \text{otherwise.} \end{cases}$$

In this paper, we consider cubic splines with the basis function $B_{r,4}(X)$ that constitutes a B-spline of degree 3. This polynomial has C^2 continuity, i.e. for each piecewise term, derivatives up to the second order are zero at the endpoints of each interval in the knot vector. We estimate component functions $g_{\mathbf{a},\mathbf{S}}$'s with the basis function $B_{r,4}(\mathbf{X})$ of cubic splines [25], as shown in Equation (10).

$$\begin{aligned} g_{\mathbf{a},\{i\}}(\mathbf{X}_{\{i\}}) &\approx \sum_{r=-1}^{m+1} \alpha_r^i B_{r,n}(\mathbf{X}_{\{i\}}) \\ g_{\mathbf{a},\{i,j\}}(\mathbf{X}_{\{i,j\}}) &\approx \sum_{p=-1}^{m+1} \sum_{q=-1}^{m+1} \beta_{pq}^{ij} B_p(\mathbf{X}_{\{i\}}) B_q(\mathbf{X}_{\{j\}}) \\ g_{\mathbf{a},\{i,j,k\}}(\mathbf{X}_{\{i,j,k\}}) &\approx \sum_{p=-1}^{m+1} \sum_{q=-1}^{m+1} \sum_{r=-1}^{m+1} \gamma_{pqr}^{ijk} B_p(\mathbf{X}_{\{i\}}) B_q(\mathbf{X}_{\{j\}}) B_r(\mathbf{X}_{\{k\}}) \end{aligned} \quad (10)$$

Here, m is the number of knots. We learn the coefficients α, β, γ using the backfitting algorithm (Line 6–12, Algorithm 1).

D Experimental Evaluations

D.1 Experimental Setup

We conduct experiments on Red Hat Enterprise Linux Server release 6.10 (Santiago) with E5 – 2690 v3 CPU and 8GB of RAM. Since our objective is to compute FIFs for any classifier, we do not perform any tuning of hyper-parameters during training. We use the default hyper-parameter choices available in Scikit-learn [36]. The only parameter that we choose in FairXplainer is the maximum order of intersectionality, namely λ . We present the ablation study w.r.t. λ in Appendix D.3.

D.2 Runtime of FairXplainer

In Figure 5, we report the runtime of FairXplainer in computing FIFs (with $\lambda = 2$) for different group fairness metrics on four classifiers and five datasets. The dimension of the dataset, measured by the number of samples and the number of features, is a key factor for runtime. For example, Adult dataset has the maximum number of samples ($= 26048$) compared to other datasets. In contrast, German dataset has the maximum features ($= 26$). Consequently, FairXplainer requires more runtime in both Adult and German dataset compared to others. However, *in all datasets, FIF computation takes at most 25 seconds, that demonstrates the efficiency of FairXplainer in practical fairness problems.*

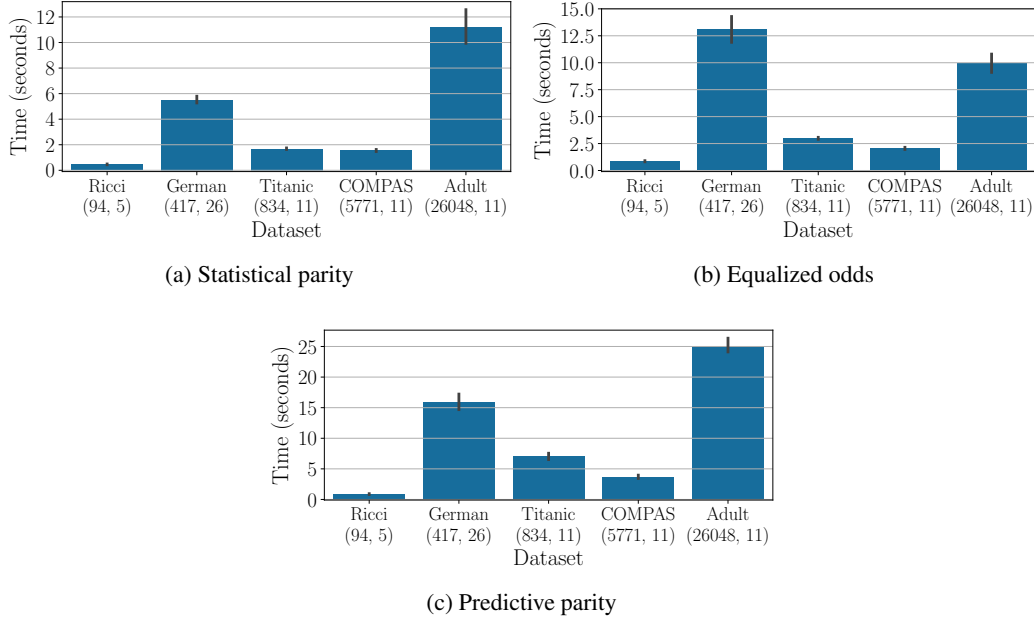


Figure 5: Runtime of FairXplainer (in seconds) for computing FIFs in different datasets. For each dataset, we mention its dimension as a tuple (# sample, # features) in the xticks. As the dimension increases, we observe higher runtime of FairXplainer.

D.3 Ablation Study: Effect of Maximum Order of Intersectionality λ

We test the effect of $\lambda \in \{1, 2, 3\}$ on the runtime of FairXplainer. For each λ , we consider 540 fairness instances (5 datasets with 27 different combinations of sensitive features \times 5-fold cross validation \times 4 classifiers). We plot the corresponding results in the cumulative distribution plot of Figure 6. Here, X -axis denotes the runtime (in seconds) and Y -axis denotes the total number of solved instances, i.e. a point (x, y) denotes that y number of instances are solved within x seconds.

As we increase from $\lambda = 1$ to $\lambda = 2$ and then to $\lambda = 3$, FairXplainer requires around 0.5 and 2 orders of magnitude more runtime, respectively, to solve the equal number of instances. This is due to the fact that with increase in λ , there is a combinatorial explosion in the number of component functions in the backfitting algorithm. Thus, *computing higher-order FIFs requires a higher runtime of FairXplainer.*

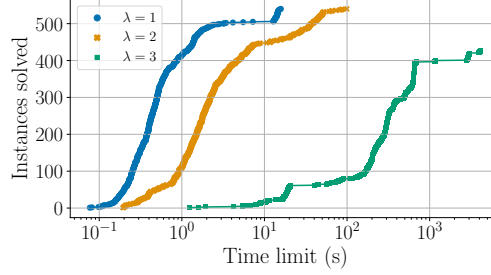


Figure 6: Effect on runtime while varying max-order λ in FairXplainer. Higher λ incurs more a computational effort, hereby solving less fairness instances within the same time limit.

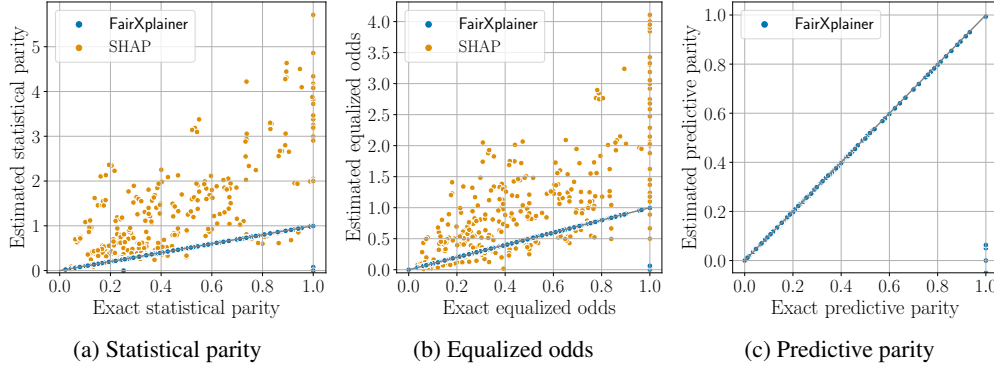


Figure 7: Comparison between the estimated and exact value of a fairness metric, computed as the sum of FIFs according to Axiom 1. Results are presented separately for FairXplainer and SHAP. Each dot represents the result for a fairness metric, carried over multiple datasets and classifiers. Ideally, if a dot lies on the line connecting $(0, 0)$ to $(1, 1)$, it denotes a correct fairness estimate. FairXplainer demonstrates significantly better accuracy of estimation than SHAP. For predictive parity metric, only FairXplainer allows associated FIFs computation.

D.4 Accuracy in Estimating Different Fairness Metrics

We compare and evaluate the accuracies of FairXplainer and SHAP in estimating a fairness metric, computed as the sum of FIFs (Axiom 1). We illustrate the results with a scatter-plot in Figure 7. In the plot, each dot represents the estimation of a fairness metric on a dataset and a classifier.

The majority of dots for FairXplainer are on the line connecting $(0, 0)$ to $(1, 1)$ in Figure 7. This implies that the estimation of FairXplainer is significantly accurate. Interestingly, error rarely occurs in FairXplainer due to the degenerate cases mentioned in Section 4. In contrast, SHAP often computes a metric inaccurately, thereby showing the drawback of applying a local explanation approach for FIF computation. *Thus, FairXplainer is significantly more accurate than SHAP in computing FIFs and the corresponding fairness metrics.*

D.5 FIF of Different Datasets

We deploy a neural network (3 hidden layers, each with 2 neurons, L2 penalty regularization term as 10^{-5} , a constant learning rate as 0.001) on different datasets, namely Adult, Ricci, and Titanic, and demonstrate the corresponding FIFs in Figures 8, 9, and 10, respectively. In all the figures, both individual and intersectional FIFs depict the sources of bias more clearly than individual FIFs alone, as argued in Section 6.

In Adult dataset, the classifier predicts whether an individual earns more than \$50k per year or not, where ‘race’ and ‘sex’ are sensitive features. We observe that the trained network is unfair and it demonstrates statistical parity as 0.23. As we analyze FIFs, *education number*, *age*, and *capital gain/loss* are key features responsible for the bias. Unlike COMPAS dataset (Figure 3b), there does not exist any feature or subset of features in Adult dataset that potentially reduces the bias.

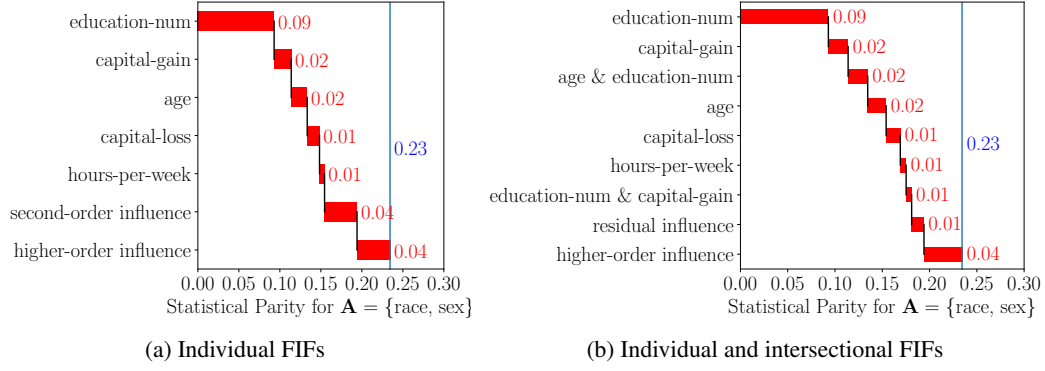


Figure 8: FIFs for Adult dataset on explaining statistical parity.

In Ricci dataset, the classification task is to predict whether a firefighter obtains promotion or not in New Haven, Connecticut administered exams. We consider ‘race’ as the sensitive feature for this experiment. We observed that the classifier has statistical parity of 0.3 based on the race of a person. In the computed FIFs, the *combined score* and *desired position* of a person increases bias, whereas written exam score reduces bias while interacting with other features. In addition, the higher order influences demonstrate a higher value (FIF = 0.26), meaning that the features are highly correlated and act in favor of bias.

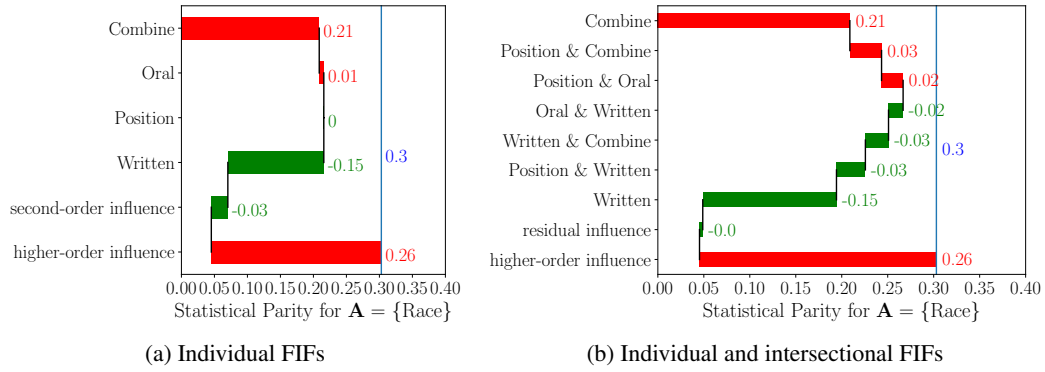


Figure 9: FIFs for Ricci dataset on explaining statistical parity.

In Titanic dataset, the neural network predicts whether a person survives the Titanic shipwreck or not. In this experiment, we consider the ‘sex’ of a person as a sensitive feature and observe that the classifier is highly unfair (statistical parity as 0.83) on the basis of ‘sex’. In the computed FIFs, most of the features except the *passenger class* and the *age* of a person increases the bias of the classifier.

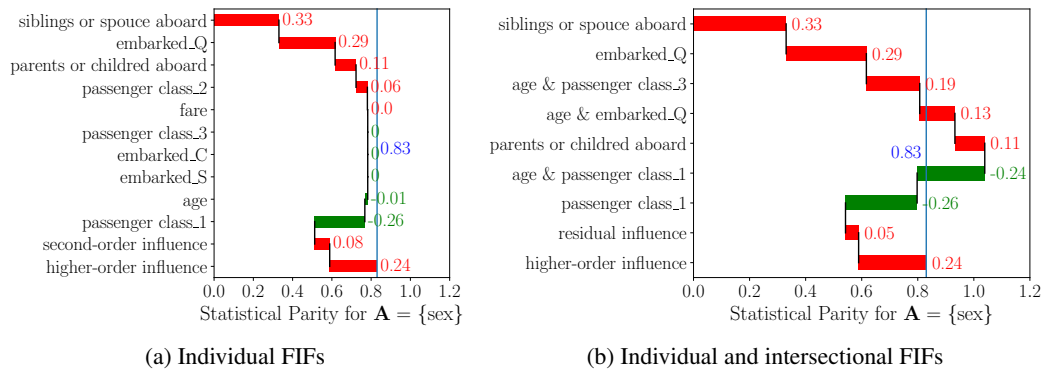


Figure 10: FIFs for Titanic dataset on explaining statistical parity.