

Procedural Fairness in Machine Learning

ZIMING WANG, Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation, Department of Computer Science and Engineering, Southern University of Science and Technology, China

CHANGWU HUANG*, School of AI and Liberal Arts, Beijing Normal-Hong Kong Baptist University, China

KE TANG*, Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation, Department of Computer Science and Engineering, Southern University of Science and Technology, China

XIN YAO, School of Data Science, Lingnan University, Hong Kong, China

Fairness in machine learning (ML) has garnered significant attention. However, current research has mainly concentrated on the distributive fairness of ML models, with limited focus on another dimension of fairness, i.e., procedural fairness. In this paper, we first define the procedural fairness of ML models by drawing from the established understanding of procedural fairness in philosophy and psychology fields, and then give formal definitions of individual and group procedural fairness. Based on the proposed definition, we further propose a novel metric to evaluate the group procedural fairness of ML models, called GPF_{FAE} , which utilizes a widely used explainable artificial intelligence technique, namely feature attribution explanation (FAE), to capture the decision process of ML models. We validate the effectiveness of GPF_{FAE} on a synthetic dataset and eight real-world datasets. Our experimental studies have revealed the relationship between procedural and distributive fairness of ML models. After validating the proposed metric for assessing the procedural fairness of ML models, we then propose a method for identifying the features that lead to the procedural unfairness of the model and propose two methods to improve procedural fairness based on the identified unfair features. Our experimental results demonstrate that we can accurately identify the features that lead to procedural unfairness in the ML model, and both of our proposed methods can significantly improve procedural fairness while also improving distributive fairness, with a slight sacrifice on the model performance.

JAIR Track: Fairness and Bias in AI

JAIR Associate Editor: Roberta Calegari

JAIR Reference Format:

Ziming Wang, Changwu Huang, Ke Tang, and Xin Yao. 2026. Procedural Fairness in Machine Learning. *Journal of Artificial Intelligence Research* 85, Article 20 (February 2026), 30 pages. DOI: [10.1613/jair.1.20498](https://doi.org/10.1613/jair.1.20498)

1 Introduction

As artificial intelligence (AI) is increasingly used in critical domains such as finance (Chen et al. 2016), hiring (L. Li et al. 2021), and criminal justice (Dressel and Farid 2018) to make consequential decisions affecting individuals, concerns about discrimination and fairness inevitably arise and have become the forefront of deliberations

*Corresponding authors

Authors' Contact Information: Ziming Wang, ORCID: [0000-0002-3118-8742](https://orcid.org/0000-0002-3118-8742), wangzm2021@mail.sustech.edu.cn, Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China; Changwu Huang, ORCID: [0000-0003-3685-2822](https://orcid.org/0000-0003-3685-2822), changwuhuang@bnu.edu.cn, School of AI and Liberal Arts, Beijing Normal-Hong Kong Baptist University, Zhuhai, China; Ke Tang, ORCID: [0000-0002-6236-2002](https://orcid.org/0000-0002-6236-2002), tangk3@sustech.edu.cn, Guangdong Provincial Key Laboratory of Brain-inspired Intelligent Computation, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China; Xin Yao, ORCID: [0000-0001-8837-4442](https://orcid.org/0000-0001-8837-4442), xinyao@ln.edu.hk, School of Data Science, Lingnan University, Hong Kong, China.



This work is licensed under a [Creative Commons Attribution International 4.0 License](https://creativecommons.org/licenses/by/4.0/).

© 2026 Copyright held by the owner/author(s).

DOI: [10.1613/jair.1.20498](https://doi.org/10.1613/jair.1.20498)

within the realm of AI ethics (Huang et al. 2023; J. Li et al. 2021). Generally, in the context of decision-making, fairness refers to the “*absence of any prejudice or favoritism toward an individual or a group based on their inherent or acquired characteristics*” (Mehrabi et al. 2021). This is a commonly accepted conception of fairness in AI, especially in fairness-aware machine learning (ML). The two well-known dimensions of fairness are distributive fairness (a.k.a outcome fairness) and procedural fairness (a.k.a process fairness) (Greenberg 1987; Morse et al. 2022). Among them, distributive fairness focuses on the fairness of outcomes resulting from a decision, while procedural fairness emphasizes fairness in the decision-making process (Grgić-Hlača et al. 2018; Morse et al. 2022). There has long been considerable discussion in the humanities community about the relationship and distinction between the two (Ambrose and Arnaud 2013; Folger 1987). Although distributive and procedural fairness mutually influence justice assessments, procedural fairness is considered to be the more robust assessment criterion (Bos et al. 2001; Morse et al. 2022; Thibaut and Walker 1975). Further, people are more willing to support an unfair outcome when they feel the process is fair (Morse et al. 2022; Van den Bos et al. 1998).

In the last decades, numerous studies have explored fairness in ML, and various quantitative fairness metrics (Fabris et al. 2023) and unfairness mitigation approaches have been proposed (Blandin and Kash 2023; Mehrabi et al. 2021; Pessach and Shmueli 2022). However, these efforts are primarily undertaken from the perspective of distributive fairness and do not adequately address procedural fairness in ML (Grgić-Hlača et al. 2018; Morse et al. 2022). Until now, relatively little attention has been paid to procedural fairness (Grgić-Hlača et al. 2018; Mehrabi et al. 2021; Morse et al. 2022; Zhao et al. 2023), and even a better definition of procedural fairness in ML models is lacking (Balkir et al. 2022).

Notably, in the philosophy and social sciences literature, procedural fairness has been well studied, emphasizing whether the procedures, rules, and processes used in decision-making are fair, transparent, and equitable (Leventhal 1980; Thibaut and Walker 1975). Inspired by this perspective, we extend and adapt these concepts into the context of ML. This adaptation ensures consistency with established procedural fairness concepts while focusing on the decision-making processes of ML models. Moreover, while major AI governance and standardization initiatives (e.g., the EU AI Act (European Union 2025), OECD AI Principles (OECD 2024)) emphasize fairness as a core principle, they generally do not explicitly distinguish between distributive and procedural fairness. In practice, most existing technical implementations and fairness metrics have focused predominantly on distributive outcomes. Our work addresses this imbalance by introducing a formal definition and a quantifiable metric for procedural fairness, offering a decision-process-oriented perspective that complements existing technical efforts and can be aligned with broader fairness objectives outlined in these governance frameworks.

To the best of our knowledge, the only work that has both defined and evaluated the procedural fairness of the ML model is Grgić-Hlača et al. 2018, in which the authors defined the procedural fairness of a model in terms of the inherent fairness of the features used in training the model. However, no equivalence can be drawn between the fairness of the features actually used and the procedural fairness of the ML model. For example, an ML model obtained by training with unfair features (e.g., sensitive features) does not imply that its decision-making process is unfair, as verified by our experimental results on the COMPAS dataset in Section 3 as a counterexample. The key problem with this is that their definition solely perceives the procedural fairness of the ML model based on the input features, without considering whether the decision-making process or logic behind the model’s predictions is fair or not. According to the concept of procedural fairness (Morse et al. 2022), the decision-making process is indeed the core element and cornerstone in perceiving procedural fairness. From this perspective, it is evident that this definition does not accurately capture the key essence of procedural fairness. Furthermore, as they themselves stated (Grgić-Hlača et al. 2018), they measured procedural fairness by “*relying on humans’ moral judgments or instincts of humans about the fairness of using input features in a decision-making context.*” While incorporating human perspectives is essential in defining fairness, such evaluation methods come with significant practical challenges: they are costly to implement, difficult to scale to new tasks or datasets, and may yield inconsistent results across different populations or settings. In summary, the urgency of research around procedural fairness

(including its conceptual definition, evaluation methods, and related aspects) is evident. However, within the overall field of AI fairness, the study of procedural fairness is currently in its infancy, highlighting the need for more and enhanced research efforts.

This motivates us to focus our efforts on procedural fairness in ML. In this paper, we present a systematic approach to defining, evaluating, and improving procedural fairness in ML models. Building on prior understanding from the social sciences, we first provide a clear definition of procedural fairness in the context of ML and formally define individual and group procedural fairness. Then, based on the proposed definition, a novel quantitative metric to assess group procedural fairness is proposed based on feature attribution explanation (FAE) (Guidotti et al. 2018), which is a popular and well-studied explainable AI (XAI) method that explains decisions of ML models by calculating the attribution of each input feature (i.e., the importance score of each input feature) (Bhatt, Weller, et al. 2021; Wang, Huang, Y. Li, et al. 2024; Wang, Huang, and Yao 2024, 2023). Our proposed procedural fairness metric can provide insight into the decision process of the model through the FAE method and evaluate the group procedural fairness of the model by assessing the overall distribution difference of FAE explanations between similar data points of two groups. To evaluate the effectiveness of our proposed definitions and methods, nine different datasets are used in our experimental studies. The relationship between procedural fairness and distributive fairness of the ML model is further analyzed. Finally, in the case where the model exhibits procedural unfairness, we have developed an approach that traces the source of procedural unfairness by identifying features that are significantly different in the FAE explanations of the two groups, and propose two mitigation methods to improve procedural fairness based on the identified unfair features. Our experimental results on the nine datasets show that our methods are indeed able to find the features that lead to procedural unfairness accurately, and that both mitigation methods significantly improve procedural fairness with a slight impact on model performance, while also improving distributive fairness at the same time.

The main contributions of this paper include the following:

- (1) We provide a more precise and more comprehensive definition of procedural fairness for ML models by drawing on existing research in the humanities, and further define individual and group procedural fairness formally.
- (2) We propose a quantitative metric based on FAE to assess the group procedural fairness based on the proposed definition, and validate the metric through experimental studies. Furthermore, we analyze the relationship between procedural fairness and distributive fairness of the ML model.
- (3) We propose a detection method to identify the sources of unfairness. For a procedural unfairness ML model, the features that lead to procedural unfairness are found by detecting features that are significantly different in the FAE explanations. Then, based on the identified unfair features, we propose two approaches to mitigate procedural unfairness. Experimental results on nine datasets show that our proposed detection method can accurately identify the features contributing to procedural unfairness, and both proposed enhancement methods can effectively improve procedural fairness without significantly compromising the model's performance, while simultaneously improving distributive fairness.

The rest of this paper is organized as follows. Section 2 presents related work. Section 3 defines the procedural fairness of the ML model, and then proposes a metric to assess group procedural fairness and evaluates its effectiveness. Section 4 proposes a method to identify the features that lead to procedural unfairness in the model, and two methods to improve procedural fairness are proposed and experimentally studied. Section 5 concludes the paper and gives future research directions.

2 Related Work

In this section, we briefly review the background and related work on fairness in ML. We first introduce the definitions and measurements of distributive fairness, which have been extensively studied in the literature. Then,

we review existing work on the procedural fairness of ML models, which has received less attention. Finally, we present related work that has applied the FAE approach to fairness.

2.1 Related Work on Distributive Fairness in ML

In recent years, fairness in ML has attracted increasing attention (Pessach and Shmueli 2022), and researchers have proposed various definitions and metrics to measure the distributive fairness of ML models, which can be categorized into individual and group distributive fairness (Mehrabi et al. 2021).

Individual distributive fairness refers to ML models that make similar predictions for similar individuals (Mehrabi et al. 2021), and its formal definition is described below.

- **Individual Fairness (IF)** (Benussi et al. 2022; Dwork et al. 2012): this measure requires that similar individuals should be treated similarly (Pessach and Shmueli 2022). Similarity can be defined with respect to a particular task (Dwork et al. 2012). *IF* can be described as:

$$IF = |\hat{y}^{(i)} - \hat{y}^{(j)}|; \quad s.t. \quad d(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \leq \varepsilon, \quad (1)$$

where $\mathbf{x}^{(\cdot)}$ denotes an individual, $\hat{y}^{(\cdot)}$ refers to the corresponding label predicted by the ML model for the individual, $d(\cdot, \cdot)$ is a distance metric between two individuals, and ε is a threshold to control the similarity between individuals.

Group distributive fairness refers to ML models that treat different groups equally (Mehrabi et al. 2021), and there are three common metrics, namely Demographic Parity (DP) (Calders and Verwer 2010; Dwork et al. 2012), Equal Opportunity (EO) (Hardt et al. 2016), and Equalized Odds (EOD) (Hardt et al. 2016).

- **Demographic Parity (DP)** (Calders and Verwer 2010; Dwork et al. 2012): this measure requires that the difference between the positive prediction rates of different sensitive groups should be as small as possible. It is commonly referred to as statistical parity. *DP* can be described as:

$$DP = |P(\hat{y} = 1 | s = s_1) - P(\hat{y} = 1 | s = s_2)|, \quad (2)$$

where \hat{y} is the predicted labels of the classifier, s represents the sensitive attribute (e.g., sex, race), and s_1 and s_2 denote two different groups associated with the sensitive attribute.

- **Equal Opportunity (EO)** (Hardt et al. 2016): this measure requires that the difference between the true-positive rates of different sensitive groups should be as small as possible. *EO* can be described as:

$$EO = |P(\hat{y} = 1 | s = s_1, y = 1) - P(\hat{y} = 1 | s = s_2, y = 1)|, \quad (3)$$

where y is the true label, and the definitions of the other symbols are the same as for the *DP* metric.

- **Equalized Odds (EOD)** (Hardt et al. 2016): this measure requires that both the differences between the false-positive rates and the true-positive rates of different sensitive groups should be as small as possible. *EOD* can be described as:

$$EOD = \frac{1}{2} \times (|P(\hat{y} = 1 | s = s_1, y = 0) - P(\hat{y} = 1 | s = s_2, y = 0)| + |P(\hat{y} = 1 | s = s_1, y = 1) - P(\hat{y} = 1 | s = s_2, y = 1)|). \quad (4)$$

To improve the distributive fairness of ML models, researchers have proposed various pre-processing (Kamiran and Calders 2012; Zemel et al. 2013), in-processing (B. H. Zhang et al. 2018; Q. Zhang et al. 2021, 2022), and post-processing (Hardt et al. 2016) mechanisms. Although these methods are valuable in mitigating bias in prediction outcomes, existing fairness metrics and mitigation methods mainly focus on distributive fairness while neglecting procedural fairness (Zhao et al. 2023).

2.2 Related Work on Procedural Fairness in ML

Currently, there are only some survey papers that discuss procedural fairness in ML (Green and Hu 2018; Guidotti et al. 2018; Mehrabi et al. 2021; Morse et al. 2022; Pessach and Shmueli 2022; Z. Tang et al. 2023), and many of them point out that there is little attention paid to procedural fairness (Grgić-Hlača et al. 2018; Mehrabi et al. 2021; Morse et al. 2022; Zhao et al. 2023). A notable exception and a very important work is Grgić-Hlača et al. 2018, which defined the procedural fairness of the ML model by the fairness of the features it uses, i.e., “We consider a classifier C trained using a subset of features F from a set \bar{F} of all possible features. We define the process fairness of C to be the fraction of all users who consider the use of every feature in F to be fair”. However, as mentioned before, this definition assesses the procedural fairness of a model based only on the inherent fairness of its input features, without considering the decision-making process or logic inside the model. Additionally, their evaluation method relies on human assessments, which are costly to conduct and difficult to scale to new tasks or datasets. Moreover, their definition equates the procedural fairness of ML models with the fairness of the features used. However, a model that uses unfair features (e.g., sensitive features) does not imply that its decision process is unfair. This is illustrated by our experiments on the COMPAS dataset in Section 3 as a counterexample. As pointed out by Balkir et al. 2022, procedural fairness lacks a better definition as well as precise quantitative metrics similar to those that have been developed for distributive fairness.

In this study, we establish the definition of procedural fairness of the ML model by considering the consistency of decision logic across individuals or groups. That is, our proposed procedural fairness definition captures the fairness of the model’s decision process, unlike distributive fairness, which considers the equality of the model’s prediction (output) between different individuals or groups.

2.3 Related Work on Applying FAE to Fairness

XAI technologies attempt to provide understandable explanations from different perspectives for humans to gain insight into the decisions of AI systems (Arrieta et al. 2020; Guidotti et al. 2018; Wang, Huang, and Yao 2024). One kind of such techniques, known as feature attribution explanation (FAE) method, explains the predictions of an ML model by computing the attribution of each input feature (i.e., the importance of each input feature) (Bhatt, Weller, et al. 2021; Wang, Huang, Y. Li, et al. 2024), which helps to identify the most significant features influencing the model’s predictions, thus providing valuable insights into the decision-making procedure and rationales behind the ML model’s decision (Zhao et al. 2023).

In this paper, we primarily employ SHAP (Lundberg and Lee 2017), a popular perturbation-based FAE method, to gain insight into the decision process of the ML models. Its advantage lies in its model-agnostic nature, as it does not require internal model information and can be applied to explain any ML model. To verify the robustness and generality of our approach, we further incorporate two additional FAE techniques: gradient*input (GI) (Shrikumar et al. 2016) and integrated gradients (IG) (Sundararajan et al. 2017), which are representative gradient-based explanation methods. Their advantage lies in their extremely high computational efficiency. By comparing results across multiple FAE methods, we ensure that our fairness evaluations are not overly reliant on a specific explanation technique and are consistent across diverse explanation perspectives.

Although FAE is the most widely used XAI technique (Arrieta et al. 2020; Bhatt, Xiang, et al. 2020), and much of the literature points out that XAI plays an important role in achieving ML fairness (Abdollahi and Nasraoui 2018; Balkir et al. 2022), only a few works apply the FAE methods to fairness (Dai et al. 2022; Zhao et al. 2023). Among them, Begley et al. 2020 and Pan et al. 2021 used FAE methods to attribute distributive fairness metrics to capture the contribution of each feature to an unfair decision result. Dimanov et al. 2020 modified the ML model to reduce the importance score of the sensitive feature obtained by FAE and found that the model was still unfair, thus pointing out that direct observation of the importance score of the sensitive feature obtained by FAE does not reliably reveal the fairness of the model. The most relevant research to us is the work of Zhao et al. 2023, which

utilized FAE to capture insights into the model's decision process and to identify bias in a procedural-oriented manner. However, their focus is on the differences between the quality of explanations obtained by different groups, i.e., the fairness of the explanations rather than the procedural fairness of the model's predictions.

In short, although many researchers have pointed out that XAI could improve the fairness of ML models (Abdollahi and Nasraoui 2018; Balkir et al. 2022), there is no existing work that utilizes FAE (or other XAI techniques) to assess the fairness of the ML model. In this paper, we evaluate and improve the procedural fairness of the ML model based on the FAE method.

3 New Definition and Measurement of Procedural Fairness

In this section, we first introduce the relevant notations used in this paper. Then, we provide a definition of procedural fairness for the ML model and propose an FAE-based measurement to assess the group procedural fairness. Subsequently, the effectiveness of the proposed procedural fairness metric is validated through extensive experiments. Furthermore, the relationship between procedural fairness and distributive fairness is analyzed. Lastly, we point out some limitations of our proposed procedural fairness measurement and offer promising solutions.

3.1 Notation

We consider the binary classification problem that aims to learn a mapping function between input feature vectors $\mathbf{x} \in \mathbb{R}^d$ and class labels $y \in \{0, 1\}$ based on a given dataset $\mathcal{D} = (\mathbf{X}, \mathbf{Y}) = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^m$, where $\mathbf{x}^{(i)} = [x_1^{(i)}, \dots, x_d^{(i)}] \in \mathbb{R}^d$ are feature vectors and $y^{(i)} \in \{0, 1\}$ are their corresponding labels. This task is often achieved by finding a model or classifier $f: \mathbf{x} \mapsto y$ based on the dataset \mathcal{D} so that given a feature vector \mathbf{x} with unknown label y , the classifier can predict its label $\hat{y} = f(\mathbf{x})$. Also, each data point \mathbf{x} has an associated sensitive attribute s (e.g., sex, race) that indicates the group membership of an individual. Actually, there can be multiple sensitive attributes, but in this paper, we consider, without loss of generality, a single sensitive attribute case (e.g., the gender of each user $s = \{male, female\}$) and use s_1 and s_2 to denote two different groups associated with the sensitive attribute. That is, each data point $(\mathbf{x}^{(i)}, y^{(i)}) \in \mathcal{D}$ has an associated sensitive feature value $s^{(i)} \in \{s_1, s_2\}$. Correspondingly, the subsets or groups of dataset \mathcal{D} with values $s = s_1$ and $s = s_2$ are denoted as $\mathcal{D}_1 = (\mathbf{X}_1, \mathbf{Y}_1) = \{(\mathbf{x}^{(i)}, y^{(i)}) \in \mathcal{D} | s^{(i)} = s_1\}$ and $\mathcal{D}_2 = (\mathbf{X}_2, \mathbf{Y}_2) = \{(\mathbf{x}^{(i)}, y^{(i)}) \in \mathcal{D} | s^{(i)} = s_2\}$, respectively.

For a model $f: \mathbf{x} \mapsto \hat{y}$, the symbol \mapsto denotes the decision-making process or mapping process of the ML model f . However, for the vast majority of ML models, the mapping process is opaque, making it difficult to measure and assess its procedural fairness. Therefore, to formally define procedural fairness, we seek to establish a representation, denoted as Φ , that portrays the decision-making process of the ML model. Unlike the mapping process \mapsto , Φ should be understandable, quantifiable, and comparable, allowing us to quantify and assess the procedural fairness of the ML model. Additionally, we use g to represent a local FAE function which takes a model f and an explained data point $\mathbf{x}^{(i)}$ as inputs and returns explanations (i.e., feature importance scores) $\mathbf{e}^{(i)} = g(f, \mathbf{x}^{(i)}) \in \mathbb{R}^d$, where $e_j^{(i)}$ (i.e., $g(f, \mathbf{x}^{(i)})_j$) is the importance score of the feature $x_j^{(i)}$ for the model's prediction $f(\mathbf{x}^{(i)})$.

3.2 Definition of Procedural Fairness

In this subsection, we establish a definition of procedural fairness in ML models by drawing on research on procedural fairness in philosophy (Leventhal 1980; Thibaut and Walker 1975) and fairness concepts in the field of ML (Mehrabi et al. 2021). Specifically, we focus on the decision logic of ML models and extend the notion of ML fairness to encompass procedural elements, and further formally define individual and group procedural fairness.

Definition 1 The procedural fairness of the ML model is defined as the internal decision process or logic of the model without any prejudice or preference for individuals or groups due to their inherent or acquired characteristics.

Definition 2 The individual procedural fairness of the ML model is defined as that two similar data points $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$ should have similar decision processes or logic:

$$d_{\Phi}(\Phi(\mathbf{x}^{(i)}), \Phi(\mathbf{x}^{(j)})) \approx 0 \text{ s.t. } d_{\mathbf{x}}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \approx 0, \quad (5)$$

where d_{Φ} and $d_{\mathbf{x}}$ are used to measure the similarity of two decision processes and two data points, respectively.

Definition 3 The group procedural fairness of the ML model is defined as that similar data points in two groups $\mathbf{x}^{(i)} \in \mathbf{X}_1$ and $\mathbf{x}^{(j)} \in \mathbf{X}_2$ should have similar decision processes or logic:

$$d_{\Phi}(\Phi(\mathbf{x}^{(i)}), \Phi(\mathbf{x}^{(j)})) \approx 0 \text{ s.t. } \mathbf{x}^{(i)} \in \mathbf{X}_1, \mathbf{x}^{(j)} \in \mathbf{X}_2 \text{ and } d_{\mathbf{x}}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \approx 0. \quad (6)$$

Consistent with individual distributive fairness, similarity d_{Φ} and $d_{\mathbf{x}}$ in Definitions 2 and 3 are often defined with respect to a particular task. The biggest challenge is to find a representation Φ that can be used to portray the ML models' decision logic or process.

As discussed in Section 2.2, the difference between our definition of procedural fairness and Grgić-Hlača *et al.* 2018 is that they defined procedural fairness in terms of the fairness of the features (based on human judgment) used by an ML model. Instead, we define the model's procedural fairness based on the model's decision logic or process. We further formally define procedural fairness for individuals and groups, capturing fairness at the individual and group levels, respectively.

3.3 Measurement of Procedural Fairness

Despite the definition of procedural fairness given, as mentioned before, the biggest challenge in assessing the procedural fairness of ML models is how to portray the ML models' decision logic or process Φ . Fortunately, XAI techniques can provide insight into the decision logic within the ML model. In particular, Zhao *et al.* 2023 used FAE techniques to capture the model's decision process for identifying biases and showed its effectiveness. Therefore, we also use FAE techniques to capture the model's decision process Φ . Specifically, given a data point $\mathbf{x}^{(i)}$, its decision logic $\Phi(\mathbf{x}^{(i)})$ is portrayed as its FAE result $\mathbf{e}^{(i)} = g(f, \mathbf{x}^{(i)})$. Then we propose a quantitative FAE-based metric for assessing the group procedural fairness, called FAE-based Group Procedural Fairness (GPF_{FAE}). GPF_{FAE} metric can be regarded as an instantiation of Definition 3, which is defined as:

$$\begin{aligned} GPF_{FAE} &= d_{\Phi}(\mathbf{E}_1, \mathbf{E}_2); \\ \mathbf{E}_1 &= \{\mathbf{e}^{(i)} | \mathbf{e}^{(i)} = g(f, \mathbf{x}^{(i)}), \mathbf{x}^{(i)} \in \mathbf{X}'_1\}, \\ \mathbf{E}_2 &= \{\mathbf{e}^{(j)} | \mathbf{e}^{(j)} = g(f, \mathbf{x}^{(j)}), \mathbf{x}^{(j)} \in \mathbf{X}'_2\}, \end{aligned} \quad (7)$$

where $d_{\Phi}(\cdot, \cdot)$ is a measurement of the distance between two sets of FAE explanation results \mathbf{E}_1 and \mathbf{E}_2 , \mathbf{X}'_1 and \mathbf{X}'_2 are sets of n data points from \mathbf{X}_1 and \mathbf{X}_2 , respectively, representing n pairs of similar data points in the two groups, which are generated as shown in Algorithm 1.

The datasets \mathbf{X}'_1 and \mathbf{X}'_2 used to evaluate the procedural fairness metric GPF_{FAE} should be representative of each group involved in the dataset. To ensure this, \mathbf{X}'_1 and \mathbf{X}'_2 are created through two distinct phases. Given the two subsets of each group \mathbf{X}_1 and \mathbf{X}_2 from the dataset \mathbf{X} , the total number of selected data points n , and a similarity or distance metric $d_{\mathbf{x}}(\cdot, \cdot)$ between individual data points, datasets \mathbf{X}'_1 and \mathbf{X}'_2 with the size of n are created as follows: in the first phase, for the first half of the dataset size n , a data point $\mathbf{x}^{(i)}$ is randomly chosen from \mathbf{X}_1 (Line 3). Subsequently, the data point $\mathbf{x}^{(j)} \in \mathbf{X}_2$ that has the smallest distance to $\mathbf{x}^{(i)}$ measured by $d_{\mathbf{x}}$ is selected (Line 4). These two selected data points are then added to \mathbf{X}'_1 and \mathbf{X}'_2 , respectively (Line 5). For the

Algorithm 1 Selecting datasets for GPF_{FAE} .

Input: Two subsets X_1 and X_2 of the dataset X , the total number of selected data points n , the data point similarity (or distance) metric d_x .

Output: The datasets X'_1 and X'_2 for evaluating GPF_{FAE} .

- 1: $X'_1 = \emptyset, X'_2 = \emptyset$.
- 2: **for** $k = 1, \dots, \lfloor \frac{n}{2} \rfloor$ **do**
- 3: Randomly select a data point $x^{(i)}$ from X_1 .
- 4: Find the $x^{(j)} \in X_2$ with the smallest $d_x(x^{(i)}, x^{(j)})$.
- 5: $X'_1 = X'_1 \cup x^{(i)}, X'_2 = X'_2 \cup x^{(j)}$.
- 6: **end for**
- 7: **for** $k = \lfloor \frac{n}{2} \rfloor + 1, \dots, n$ **do**
- 8: Randomly select a data point $x^{(j)}$ from X_2 .
- 9: Find the $x^{(i)} \in X_1$ with the smallest $d_x(x^{(j)}, x^{(i)})$.
- 10: $X'_1 = X'_1 \cup x^{(i)}, X'_2 = X'_2 \cup x^{(j)}$.
- 11: **end for**
- 12: **return** the datasets X'_1 and X'_2 .

remaining half of the dataset size n , a data point $x^{(j)}$ is randomly selected from X_2 (Line 8). Following this, the data point $x^{(i)} \in X_1$ with the smallest distance $d_x(x^{(j)}, x^{(i)})$ is chosen (Line 9). The selected pair of data points is then collected in X'_1 and X'_2 , respectively (Line 10). In this way, there are n data points in each of X'_1 and X'_2 with the minimum distance between each pair of them.

In this paper, we use the maximum mean discrepancy (MMD) with an exponential kernel function as d_Φ to measure the distributional differences between two explanation sets E_1 and E_2 , and the p -value obtained by performing a permutation test on the generated kernel matrix is used as the final evaluation result. If the p -value is larger than a threshold, we consider the model meets the procedural fairness requirement; otherwise, the model is deemed procedurally unfair. The Euclidean distance is used as the distance metric d_x for identifying the most similar data points. The ML models employed in this study are artificial neural networks (ANNs), although our method is applicable to any other ML models.

In short, the GPF_{FAE} quantifies the disparity in FAE explanation results between similar data points belonging to two different groups. The pipeline of using our proposed GPF_{FAE} to measure the group procedural fairness of the ML model is shown in Fig. 1.

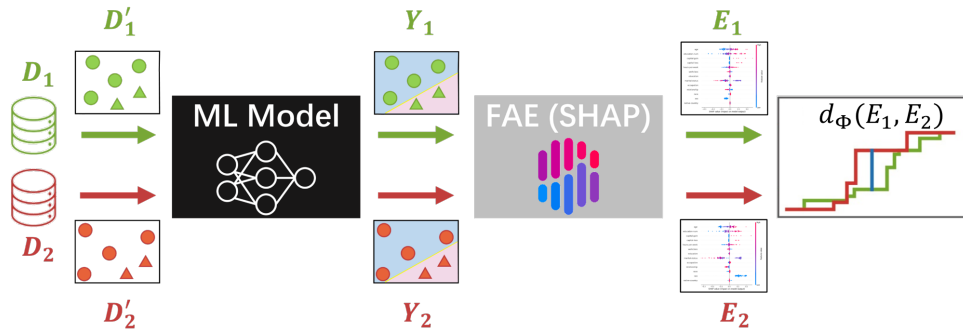


Fig. 1. Pipeline of using GPF_{FAE} metric to evaluate group procedural fairness.

To simplify our discussions, we refer to an ML model that meets distributive fairness as a distributive-fair model, and a model that does not meet distributive fairness criteria as a distributive-unfair model. Similarly, an ML model that is in line with our defined group procedural fairness metric is denoted as a procedural-fair model, and a model that is not in line with procedural fairness criteria is called a procedural-unfair model. These terms will be used throughout the remainder of this paper.

3.4 Experimental Evaluation of the Proposed Procedural Fairness Metric

In this subsection, we extensively validate and evaluate our proposed metric, GPF_{FAE} , through a comprehensive set of experimental studies. We begin by presenting the relevant experimental setup, providing details of the methodology employed for our evaluation. Next, we validate the effectiveness of our proposed GPF_{FAE} metric on various procedural-fair and -unfair models, thereby demonstrating its capability to accurately distinguish procedural-fair and -unfair models. Subsequently, we further investigate the relationship between GPF_{FAE} and the degree of procedural fairness, exploring how our metric captures and quantifies the level of procedural fairness or unfairness of an ML model. We also examine the impact of using different FAE methods on the evaluation results and computational efficiency of the GPF_{FAE} metric, verifying the reliability and robustness of the proposed metric. Furthermore, we explore the relationship between procedural fairness and distributive fairness of the ML model, revealing their interactions and potential implications. Additionally, we analyze the choice of the hyper-parameter n in the GPF_{FAE} metric, considering its impact on the metric's performance and accuracy. Finally, we critically discuss the limitations associated with our proposed metric, ensuring a comprehensive understanding of its capabilities and potential limitations, and provide promising solutions. The source code of this work is available at <https://github.com/oddwang/GPF-FAE>.

3.4.1 Experimental Setup. Here we describe the relevant settings in our experiments, including the dataset used, the model used, and the relevant parameter settings.

Datasets. In this paper, we conducted experiments on a synthetic dataset and eight real-world datasets that are widely used in fair ML (Le Quy et al. 2022). For the synthetic dataset, we referred to the generation scheme of Jones et al. 2020, which has 10,000 data points, including two non-sensitive features x_1 and x_2 , one sensitive feature x_s , one proxy feature x_p for sensitive feature x_s , and one label class y . Among them, $x_1 \sim N(0, 1)$, $x_2 \sim N(0, 1)$, while for the sensitive feature x_s , 6000 data points are set to 1 and 4000 data points are set to 0 (representing the advantaged and disadvantaged groups, respectively). The proxy feature $x_p \sim N(x_s, 0.1)$, and the label class $y = [\frac{1}{1+e^{-t}} + 0.5]$, where $t = w_0 + w_1x_1 + w_2x_2 + w_3x_s + w_4x_p + N(0, 1)$. In this paper, $w_0, w_1, w_2, w_3,$ and w_4 were taken as $-0.2, 1.5, 0.5, 0.5,$ and 0.5 , respectively. The relationship graph between features and class label in the synthetic dataset is shown in Fig. 2. In addition, eight real-world datasets widely used in ML fairness (Le Quy et al. 2022), namely *Adult* (Dua and Graff 2017), *Dutch* (Van der Laan 2000), *LSAT* (Wightman 1998), *COMPAS* (Angwin et al. 2016), *German* (Dua and Graff 2017), *KDD* (Dua and Graff 2017), *Bank* (Dua and Graff 2017), and *Default* (Dua and Graff 2017), are also used in our experimental study. Table 1 summarizes these datasets. Each dataset is preprocessed in the same way: label encoding of categorical features, and then normalizing all features by Z-score. In addition, each dataset was randomly divided into training and test sets with a ratio of 4:1, denoted as \mathcal{D}_{train} and \mathcal{D}_{test} , respectively.

Parameter Setting. In our experiments, a two-layer artificial neural network (ANN) was trained with the ReLU activation function, Adam optimizer, and binary cross-entropy (BCE) loss function on each dataset. Each ANN model was fully connected to the hidden layer with 32 nodes, except for the *German* and *KDD* datasets, which had a high number of features, and we set the hidden layer with 64 nodes. The number of iterations and the learning rate were set to 300 and 0.01, respectively.

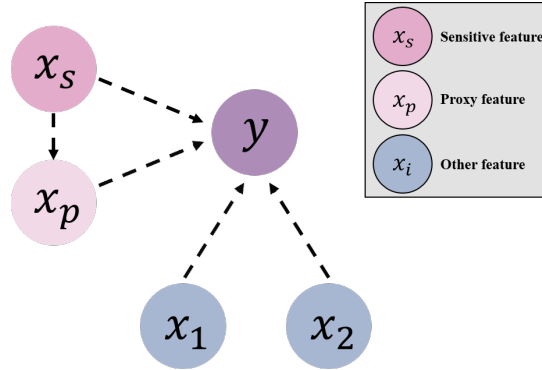


Fig. 2. Relationship graph for the synthetic dataset. The dotted arrows indicate influence relationships, e.g., $x_1 \rightarrow y$ means that the value of x_1 affects y .

Table 1. The eight real-world datasets and one synthetic dataset used in the study. “ $|\mathcal{D}|$ ”, “ $|X|$ ”, “ DP ”, and “ S ” denote the number of data points, the number of features, the DP value of the dataset itself, and the sensitive attribute under consideration, respectively.

Dataset	$ \mathcal{D} $	$ X $	DP	S	Disadvantaged Group	Advantaged Group
Adult	48,842	14	0.195	Sex	Female	Male
Dutch	60,420	11	0.298	Sex	Female	Male
LSAT	20,798	11	0.198	Race	Non-White	White
COMAPS	6172	7	0.132	Race	African-American	Others
German	1000	20	0.075	Sex	Female	Male
KDD	284,556	36	0.076	Sex	Female	Male
Bank	45,211	16	0.048	Marital	Married	Single
Default	30,000	23	0.034	Sex	Female	Male
Synthetic	10,000	4	0.199	x_s	0	1

The MMD significance level was set to 0.05, which means that there is a significant difference between the explanation distribution of the two groups when $GPF_{FAE} \leq 0.05$, i.e., the model is procedural-unfair, and conversely, the model is procedural-fair. In addition, in our experiments, we took $n = 100$ (the selection of parameter n was chosen based on our experimentation and will be discussed in Section 3.4.6), i.e., we selected 100 pairs of similar data points from X_1 and X_2 . Specifically, 50 data points were randomly selected from X_1 of the test set, and then the 50 data points that are most similar to each of them were found from X_2 of the test set and put them into X'_1 and X'_2 , respectively. We repeated this process in reverse, and finally, there were 100 data points in each of X'_1 and X'_2 with the minimum distance between each pair of them.

We used three distributive fairness metrics, DP , EO , and EOD , implemented in the open-source Python algorithmic fairness toolkit AI Fairness 360 (AIF360) (Bellamy et al. 2019) to explore the relationship between procedural and distributive fairness in ML. Referring to the criteria of AIF360 (Bellamy et al. 2019), we used 0.10 as the threshold of the group distributive fairness metrics (Stevens et al. 2020). That is, when the DP , EO , and EOD metrics are below 0.10, the model is regarded as distributive-fair; otherwise, the model is considered as

distributive-unfair. Each experiment case was performed 10 times in independent runs using different random number seeds.

3.4.2 Evaluation on Procedural-Fair and Procedural-Unfair Models. In this part, we evaluate the proposed metric GPF_{FAE} by constructing a procedural-fair model and a procedural-unfair model on each dataset, respectively, so that we can examine whether the proposed metric GPF_{FAE} accurately identifies whether an ML model is procedural-fair or procedural-unfair.

We first specify how to construct procedural-fair and -unfair models, respectively. To obtain the procedural-fair model, we only used “fair features” to train the model. The underlying assumption is consistent with Grgić-Hlača *et al.* 2018 that the models trained with fair features are procedural-fair. On the synthetic dataset, we used two features x_1 and x_2 that are unrelated to the sensitive attribute. On each real-world dataset, since there is no “ground truth” or definitive criteria to guide the selection of “fair features”, we calculated the correlation of each feature with the considered sensitive attribute using the Pearson correlation coefficient. We considered the features with a correlation coefficient below 0.10 as “fair features” for training the model (except on the *COMPAS* and *LSAT* datasets, where we used 0.20 as the threshold due to the high correlation coefficient of each feature with the sensitive attribute).

We ensured that the model is significantly procedural-unfair by examining the explanation results obtained by the FAE method on the sensitive attribute. Specifically, we ensured that the model is a procedural-unfair model when different values of the sensitive attribute, such as male and female, have opposite impacts on the model’s decisions. Taking the *Adult* dataset as an example, as shown in Fig. 3, the male (red points) and female (blue points) groups have positive and negative impacts on the final decision, respectively, indicating that the model is procedural-unfair in favor of the male group. However, it is important to note that this condition is sufficient but not necessary, as there may be proxy attributes at play that do not guarantee procedural fairness, even when different values of the sensitive attribute have the same impact on the decision.

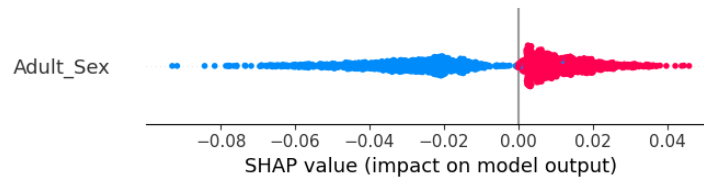


Fig. 3. The “sex” feature on the *Adult* dataset is used as an example to illustrate the criteria for constructing a significantly procedural-unfair model. Each red and blue point represents a data point from the male and female groups, respectively.

We constructed procedural-unfair models on each dataset in two steps. First, for the four inherently fairer datasets, *German*, *KDD*, *Bank*, and *Default*, with DP values less than 0.10 (as shown in Table 1), we randomly resampled the data points labeled as “1” in the advantaged group until their dataset achieves a DP value greater than 0.10, and called them *Unfair German*, *Unfair KDD*, *Unfair Bank*, and *Unfair Default*.

Second, on datasets except *COMPAS* and *LSAT*, we used BCE as the loss function to obtain a procedural-unfair model that meets the requirements. On the *COMPAS* and *LSAT* datasets, we used “ $BCE + \lambda \times DP$ ” (λ is slightly less than 0) as the loss function to obtain the procedural-unfair model. This is because we were surprised to find that on both datasets, the models obtained by training directly with BCE as the loss function are not significant procedural-unfair models, as they do not satisfy the conditions required by our procedural-unfair model, although the datasets themselves are unfair (this case will be discussed further below). Eventually, the parameter λ on the *COMPAS* and *LSAT* are set to -0.1 and -0.05, respectively. The selection criterion of parameter λ was decided

according to whether the FAE explanation results on the sensitive attribute of the trained model satisfy our requirements for the constructed procedural unfairness model. Obviously, the other normal parameter values that can satisfy the requirements are equally feasible.

Finally, we present the FAE explanation results of the procedural-unfair model constructed on each dataset on the sensitive attribute, as shown in Fig. 4. It demonstrates that the procedural-unfair models we constructed on the nine datasets all satisfy our requirement that they all significantly favor the advantaged group. It is worth noting that the distribution of explanations for the sensitive attribute in the *COMPAS* dataset in Fig. 4 is the opposite of the other datasets. This is because the task of the *COMPAS* dataset is to assess whether a criminal will re-offend, and contrary to the other tasks, predicting a positive category (i.e., will have recidivism) is instead disadvantageous, and thus the distribution of its explanations is exactly the opposite, which precisely means that the model favors the advantaged group.

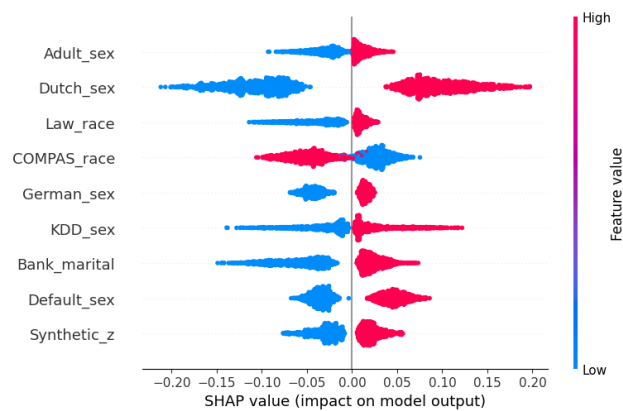


Fig. 4. FAE explanation results obtained for the sensitive attribute on each constructed procedural-unfair model, where red and blue represent the advantaged and disadvantaged groups, respectively.

The evaluation results of the GPF_{FAE} metric on the procedural-fair and -unfair models constructed on the nine datasets are shown in Tables 2 and 3, respectively. We also present the evaluation results of the three distributive fairness metrics, DP , EO , and EOD , to analyze the relationship between procedural fairness and distributive fairness afterward.

Tables 2 and 3 show that our GPF_{FAE} metric accurately distinguishes between procedural-fair and -unfair models. On the constructed procedural-fair models, the GPF_{FAE} values are close to or at 1.0 on all datasets except on the *German* dataset, which proves that GPF_{FAE} considers the distribution of the explanations of the two groups to be very consistent, i.e., the decision process of the model is very fair. In contrast, the GPF_{FAE} values on the *German* dataset are relatively small and will be discussed further in Section 3.4.7. On the other hand, on all the constructed procedural-unfair models, the GPF_{FAE} metric values are close to or reach 0.0, which means that the GPF_{FAE} metric considers the decision-making process of the two groups to be significantly different, i.e., it can accurately assess that the model's decisions are significantly unfair.

In addition, as previously noted, we performed a slight back-optimization of the DP metric on the *COMPAS* and *LSAT* datasets to construct significant procedural-unfair models. This was due to the unexpected observation that the decision process of the models trained using BCE as the loss function on these two inherently unfair datasets did not exhibit a clear bias towards a particular group, especially on the *COMPAS* dataset, as shown in Table 4. To provide further insights, we present the explanation results of 100 data points each for the advantaged

Table 2. The evaluation results of GPF_{FAE} , DP , EO , and EOD metrics on each dataset under the constructed procedural-fair models. Among them, DP , EO , and EOD metrics considered to be distributive-unfair are underlined. The \uparrow means that the larger the metric, the better, and \downarrow means the opposite.

Dataset	Feature Number	$GPF_{FAE} \uparrow$	$DP \downarrow$	$EO \downarrow$	$EOD \downarrow$
Adult	10	0.943	0.079	0.032	0.019
Dutch	9	1.000	<u>0.140</u>	<u>0.102</u>	0.051
LSAT	5	0.990	0.003	0.002	0.003
COMPAS	3	1.000	<u>0.182</u>	<u>0.162</u>	<u>0.149</u>
German	15	0.525	0.047	0.073	0.069
KDD	17	1.000	0.018	0.023	0.012
Bank	12	0.994	0.013	0.041	0.018
Default	22	0.993	0.022	0.020	0.014
Synthetic	2	1.000	0.015	0.095	<u>0.119</u>

Table 3. The evaluation results of GPF_{FAE} , DP , EO , and EOD metrics on each dataset under the procedural-unfair models. Among them, DP , EO , and EOD metrics considered to be distributive-fair are underlined. The \uparrow means that the larger the metric, the better, and \downarrow means the opposite.

Dataset	Loss Function	$GPF_{FAE} \uparrow$	$DP \downarrow$	$EO \downarrow$	$EOD \downarrow$
Adult	BCE	0.012	0.180	<u>0.099</u>	<u>0.089</u>
Dutch	BCE	0.000	0.355	0.104	0.167
LSAT	$BCE - 0.05 \times DP$	0.001	0.267	0.157	0.319
COMPAS	$BCE - 0.1 \times DP$	0.001	0.367	0.365	0.326
Unfair German	BCE	0.000	0.117	<u>0.063</u>	<u>0.093</u>
Unfair KDD	BCE	0.000	0.104	0.370	0.202
Unfair Bank	BCE	0.001	0.150	0.285	0.163
Unfair Default	BCE	0.000	0.126	0.144	0.100
Synthetic	BCE	0.000	0.251	0.111	0.126

and disadvantaged groups on one of the independent runs on the *COMPAS* dataset, as shown in Fig. 5. On that independent run, the GPF_{FAE} metric value is 0.690.

Table 4. The evaluation results of GPF_{FAE} , DP , EO , and EOD metrics on models obtained directly using BCE as a loss function for the *COMPAS* and *LSAT* datasets. The \uparrow means that the larger the metric, the better, and \downarrow means the opposite.

Dataset	Dataset $DP \downarrow$	$GPF_{FAE} \uparrow$	$DP \downarrow$	$EO \downarrow$	$EOD \downarrow$
COMPAS	0.132	0.619	0.239	0.214	0.193
LSAT	0.198	0.422	0.201	0.104	0.240

As we can see from Table 4, although the inherent unfairness of the dataset and the model’s distributive fairness metrics indicate significant unfairness, the model’s decision process is relatively unbiased. Fig. 5 also confirms the conclusion that the distribution of the explained results for the advantaged and disadvantaged groups is very similar, and it can be seen that there is no preference for a particular group on the sensitive attribute “race”,

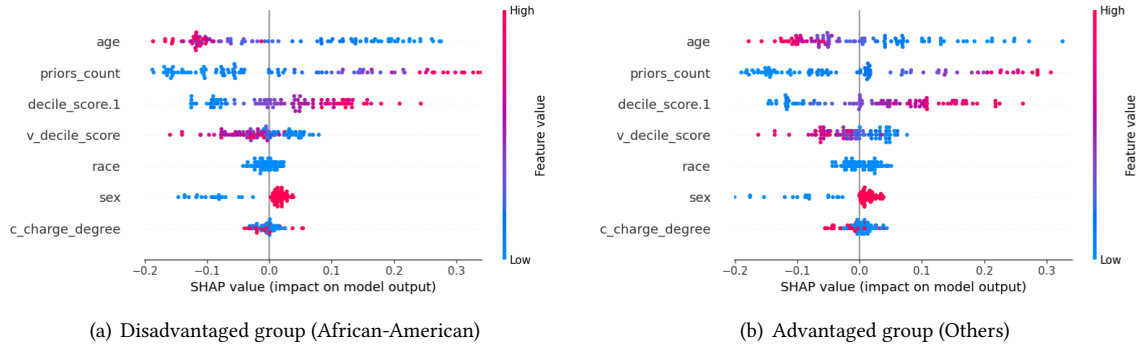


Fig. 5. Explanation results for disadvantaged and advantaged groups on the *COMPAS* dataset.

unlike the distribution of the sensitive attribute in the procedural-unfair model constructed on the *COMPAS* dataset, which can be seen in Fig. 4. This is also a counterexample to Grgić-Hlača *et al.* 2018’s definition of procedural fairness, showing that a model is modeled using intrinsically unfair features does not mean that it is a procedural-unfair model. Furthermore, this result is intriguing because the *COMPAS* dataset has been used as a typical example to study fairness, and while it exhibits significant unfairness in terms of decision results, the decision process of the model obtained does not have a clear bias towards any certain group. In addition, for problems like the *COMPAS* dataset that involve law and justice, we should pursue procedural fairness in addition to distributive fairness.

3.4.3 Evaluating the Relationship between GPF_{FAE} Metric and the Degree of Group Procedural Fairness. So far, we have been evaluating our proposed metric GPF_{FAE} by constructing various procedural-fair and -unfair models on nine datasets. However, the preceding sections demonstrate more that GPF_{FAE} can correctly detect whether a model is procedural-fair or not rather than evaluate the degree of procedural fairness of the model. In this part, we aim to more thoroughly evaluate the relationship between GPF_{FAE} and procedural fairness, specifically by investigating whether the value of GPF_{FAE} can correctly reflect and assess the degree of procedural fairness of a model.

However, it is difficult to construct various models with different degrees of procedural fairness/unfairness, especially on black-box models like ANNs. Therefore, instead of using ANNs, we use logistic regression (LR) models. On each dataset, we first constructed an LR model utilizing the features used to construct the procedural-fair model in Section 3.4.2 and the sensitive attribute under consideration. Then, we manually control the weights w_s of the LR model on the sensitive attribute, starting from 0.0 and increasing gradually. For the LR model, the weight w_s determines the degree of influence of the sensitive attribute in the decision-making process. Obviously, as the value of w_s gradually increases, the decision process of the model becomes increasingly unfair, and the trend of GPF_{FAE} is observed accordingly. The upper bounds of w_s values are taken as 0.4, 0.8, 0.5, 0.2, 0.25, 0.35, 0.5, 0.3, and 5 for different datasets. We collected 50 parameters uniformly between 0.0 and the upper bounds to generate models with varying degrees of procedural unfairness. Taking the *Adult* dataset as an example, we set w_s to be 50 different parameters between $[0.0, 0.4]$ one by one. Finally, we normalized the parameters w_s on each dataset to facilitate the presentation of the experimental results, as shown in Fig. 6.

As we can see in Fig. 6, on each dataset, with the increase in the value of w_s , i.e., the unfairness degree of the model’s decision process increases, the value of the GPF_{FAE} metric becomes progressively smaller. This illustrates that our proposed metric GPF_{FAE} is able to assess the degree of the model’s procedural fairness correctly.

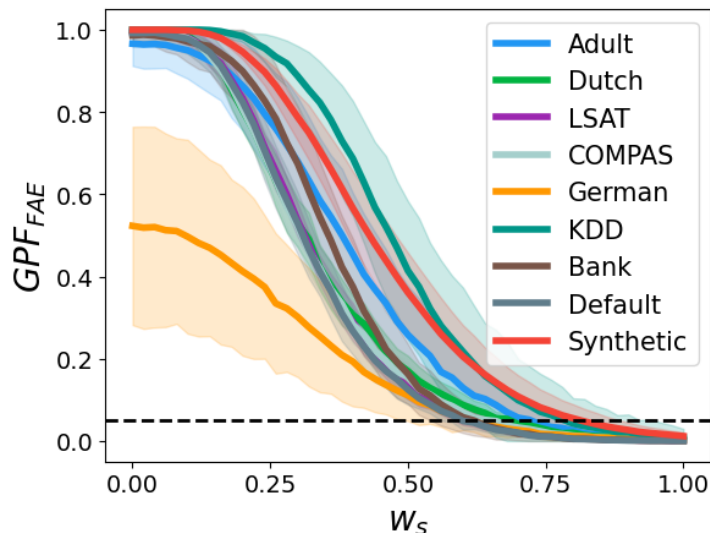


Fig. 6. Trend of GPF_{FAE} metric values as w_s increases. Figures show mean values over 10 random runs with a $1 - \sigma$ error-bar.

3.4.4 Applicability of Different FAE Methods within the GPF_{FAE} Metric. In this subsection, we explore whether FAE methods beyond SHAP can be employed to assess the procedural fairness of ML models using the proposed GPF_{FAE} metric. Specifically, we apply GPF_{FAE} with multiple FAE methods to LR models exhibiting varying degrees of procedural fairness constructed in Section 3.4.3. The considered FAE methods include the perturbation-based SHAP method (Lundberg and Lee 2017), as well as two gradient-based methods: GI (Shrikumar et al. 2016) and IG (Sundararajan et al. 2017). To quantify the consistency between these methods, we calculated the Pearson correlation coefficient between the GPF_{FAE} metric scores obtained by any two of the three methods and reported their average values. Their evaluation results are shown in Fig. 7.

Fig. 7 illustrates that the results produced by the different FAE methods within the GPF_{FAE} framework are highly positively correlated, with Pearson correlation coefficients close to 1.0 across all datasets. This strong consistency suggests that the GPF_{FAE} metric is robust across various FAE methods and is not restricted to SHAP.

We also measured the runtime of GPF_{FAE} when evaluated on the constructed LR models using each FAE method, and the results are shown in Table 5. All experiments were conducted on a Linux server with an Intel(R) Xeon(R) Platinum 8358 CPU @ 2.60 GHz processor and 512 GB of memory. To ensure consistency, all experiments were executed using a single CPU core in single-threaded mode.

As can be seen from Table 5, the perturbation-based SHAP method incurs significantly higher computational cost compared to the gradient-based methods. Nonetheless, SHAP offers the advantage of being model-agnostic, making it suitable for explaining arbitrary ML models. In contrast, if the model under evaluation supports gradient computation, gradient-based FAE methods such as GI or IG can be employed to enhance evaluation efficiency significantly. For consistency, we continue to use SHAP as the default explanation method in the subsequent sections of this paper.

3.4.5 Relationship between Procedural and Distributive Fairness. In the following, we discuss the relationship between procedural and distributive fairness in the ML model. From Tables 2 and 3, we can see that procedural and distributive fairness coincide in many cases, i.e., when the process is fair, the distributive fairness metrics are also fair, and vice versa. However, some procedural-fair models are considered as distributive-unfair and some

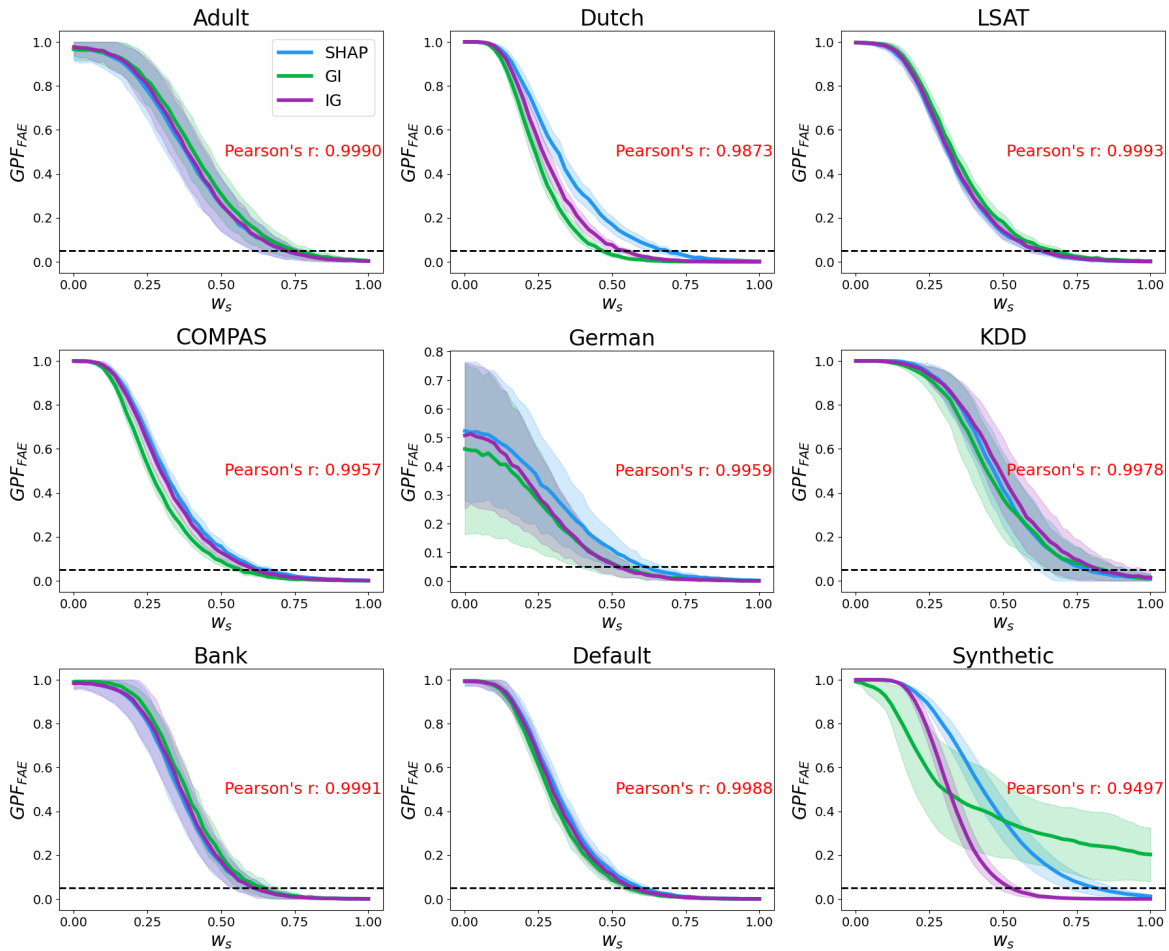


Fig. 7. The trend of GPF_{FAE} metric values using different FAE methods as w_s increases across various datasets. Figures show mean values over 10 random runs with a $1 - \sigma$ error-bar.

procedural-unfair models are considered to be distributive-fair. Such a situation can be observed in Table 4 and Fig. 5, where the decision process is relatively fair, despite the significant unfairness in the three distributive fairness metrics.

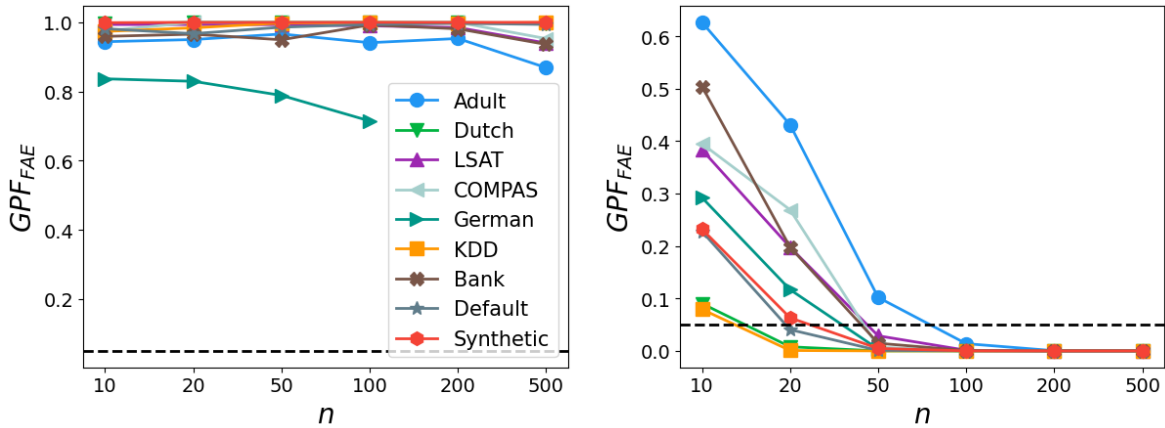
Thus, the relationship between procedural and distributive fairness in ML models is complex. They are sometimes coincidental with each other, but sometimes there are trade-offs and conflicts. Our findings here are consistent with the exploration of the relationship between the two in the humanities community (Ambrose and Arnaud 2013). For a more systematic exploration and in-depth analysis regarding the nuanced interactions and potential trade-offs between these two dimensions of fairness, one may refer to our follow-up work (Wang, Huang, K. Tang, et al. 2025).

3.4.6 Study of the Parameter n in the GPF_{FAE} Metric. The parameter n in the GPF_{FAE} metric determines how many pairs of similar data points are selected. In fact, ideally, we could compute the exact value of our proposed GPF_{FAE}

Table 5. Average runtime (in seconds) of computing the GPF_{FAE} metric on the constructed LR model using different FAE method.

Dataset	SHAP	GI	IG
Adult	128.84	0.22	0.25
Dutch	69.11	0.18	0.15
LSAT	9.36	0.13	0.15
COMPAS	3.38	0.10	0.11
German	134.07	0.15	0.11
KDD	142.83	0.98	1.06
Bank	127.03	0.18	0.17
Default	145.44	0.15	0.14
Synthetic	2.36	0.11	0.15
Mean	84.71	0.24	0.25

metric on the entire dataset (i.e., $n = \text{dataset size}$), but this would also result in longer computation times, so a trade-off is necessary. To choose the appropriate parameter n , we examined the impact of different values of parameter n . We took values of n as 10, 20, 50, 100, 200, and 500, respectively, and observed the corresponding changes in the evaluation results of the GPF_{FAE} metric on the procedural-fair and -unfair models constructed in Section 3.4.2, as shown in Fig. 8 (for the *German* dataset, n can only take a maximum of 100 due to dataset size limitations). The results indicate that for the procedural-fair models, the evaluation results remain relatively stable across different values of n . In contrast, for the procedural-unfair models, the GPF_{FAE} metric gradually detects the unfairness of the decision process of the model correctly and stabilizes as the parameter n increases. However, it is worth noting that a larger value of n implies more data points to be explained and a higher computational cost. In light of the trade-off between performance and efficiency, we choose $n = 100$ in our paper.

(a) GPF_{FAE} under different n on procedural fairness model(b) GPF_{FAE} under different n on procedural unfairness modelFig. 8. The trend of GPF_{FAE} metric with parameter n on the constructed procedural-fair and -unfair models.

3.4.7 Limitations. Although our proposed metric GPF_{FAE} can accurately measure the procedural fairness of the model, it is limited by the requirement of obtaining n pairs of similar data points with different values of the sensitive attribute. This may be challenging in situations where the dataset size is small or the data point distribution is very sparse. This can affect the final evaluation results. Intuitively, when the identified n pairs of data points are not similar, their decision logics will naturally not be similar, and the GPF_{FAE} metric values will be small, even though the model is procedurally fair.

This phenomenon is especially noticeable in the *German* dataset, which is the least data-rich dataset in the experiments of this paper, with only 1000 data, including 800 training data and 200 test data. Therefore, finding 100 pairs of similar data points in the 200 test data is obviously a challenge. As a result, the data points may not be similar enough to each other. The experimental results in Table 2 show that the GPF_{FAE} metric values are close to or reach 1.0 for the constructed procedural-fair model evaluated on all other datasets, which means that the GPF_{FAE} metric considers the constructed model to be very procedural fairness. However, on the *German* dataset, the GPF_{FAE} metric value is only 0.525. Similar observations can be made in other experiments, such as in Fig. 6 when exploring the relationship between the GPF_{FAE} metric and procedural fairness, where the GPF_{FAE} metric is close to 1.0 on all other datasets when w_s is 0.0, but is less than 0.60 on the *German* dataset with a very large variance. A similar phenomenon is also observed in Fig. 8(a), where the GPF_{FAE} metric value on the *German* dataset is significantly smaller than the other datasets and is not stable.

To demonstrate that this is due to the small size of the dataset causing the selected n pairs of data points are not similar, for the procedural-fair model constructed on the *German* dataset, we compare two approaches of selecting similar data points: one using the test set only (i.e., $N = 200$), and the other using the entire dataset (i.e., $N = 1000$). The changes in the GPF_{FAE} metric and the average distance $\overline{d_x}$ between the 100 pairs of similar data points are shown in Table 6.

Table 6. On the *German* dataset, the effect of selecting only similar data points from the test set ($N = 200$) and from the entire data set ($N = 1000$) on the GPF_{FAE} metric and the average distance $\overline{d_x}$ between similar data points on the constructed procedural-fair model.

Dataset	Feature Number	N	GPF_{FAE}	$\overline{d_x}$
German	15	200	0.525	2.772
		1000	0.708	2.331

The results from Table 6 demonstrate that the selected data point pairs are more similar to each other and the GPF_{FAE} metric values increase when selecting similar data points from the whole dataset instead of only the test set. This further proves that it is not that the constructed model is not fair enough, but that the data point size of the dataset is too small to find n pairs of similar data points, which affects the final evaluation results.

To further explore the effect of the degree of similarity of the selected data points on the GPF_{FAE} metric, we observe the trend of the average distance $\overline{d_x}$ between similar data points and the GPF_{FAE} metric on the constructed procedural-fair model on the *Dutch* dataset by adjusting the size N of the selected similar data point set. At different values of N , we all ensure a 50/50 data point size for the advantaged and disadvantaged groups. The *Dutch* dataset is chosen because it has a sufficient number of data points, and the GPF_{FAE} metric value can reach 1.0 on the constructed procedural-fair model with a moderate number of features. The results are shown in Fig 9.

We can see that when the value of N is small, the average distance $\overline{d_x}$ is large and the value of the GPF_{FAE} metric is small. And as the value of N increases, the average distance $\overline{d_x}$ between the selected similar data point pairs steadily decreases, while the value of the GPF_{FAE} indicator increases steadily until it stabilizes at about $N = 1200$. However, this value may vary with different datasets and dimensions. In general, when the size of the

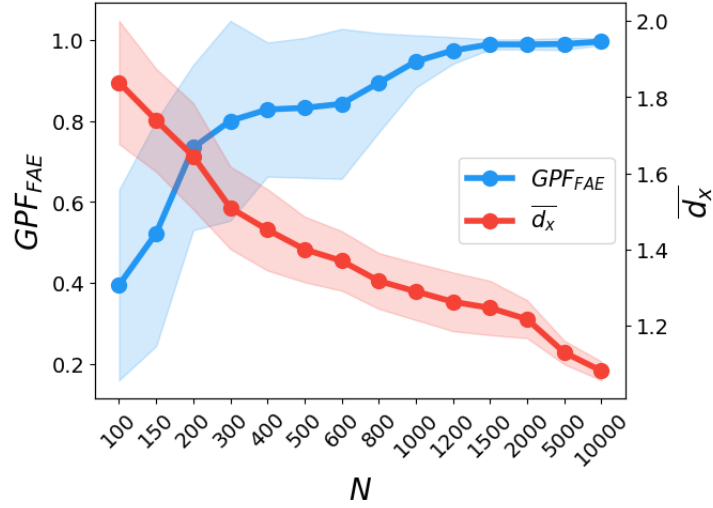


Fig. 9. The effect of selecting similar data points from different sizes of data N on the average distance \overline{d}_x between similar data points and the GPF_{FAE} metric on the procedural-fair model constructed for the *Dutch* dataset. Figures show mean values over 10 random runs with a $1 - \sigma$ error-bar.

dataset is small, the evaluation results may not be accurate enough due to the difficulty of obtaining similar data points. Therefore, to ensure reliable assessment results, it is recommended to select similar data points from at least 1000 data points, thus ensuring sufficient similarity between the selected data point pairs.

3.4.8 Counterfactual Data Generation for Sparse Datasets. In the previous subsection, we observed that the proposed GPF_{FAE} metric struggles to accurately assess procedural fairness when it is difficult to identify sufficiently similar data points, particularly in small-scale or sparse datasets. This limitation poses challenges to the metric’s applicability in such scenarios. To address this issue, we propose a counterfactual data generation strategy: when direct matching fails due to data sparsity, we synthetically generate similar data points to serve as counterfactuals.

Specifically, given a data point $\mathbf{x}^{(i)} \in \mathcal{D}_1$ to be matched, we first train a kernel density estimation (KDE) model on the other group \mathcal{D}_2 . Then, we sample k candidate points $\{\tilde{\mathbf{x}}_j\}_{j=1}^k \sim \text{KDE}$, and select the generated point with the smallest Euclidean distance to $\mathbf{x}^{(i)}$ as its counterfactual $\mathbf{x}_{cf}^{(i)}$:

$$\mathbf{x}_{cf}^{(i)} = \arg \min_{\tilde{\mathbf{x}}_j \sim \text{KDE}} \|\tilde{\mathbf{x}}_j - \mathbf{x}^{(i)}\|_2. \quad (8)$$

This process is repeated for all n data points to be matched, ensuring that each is paired with the most similar candidate among the k synthetically generated points generated by KDE in the opposite group. We evaluated this approach on the *German* dataset using LR models with varying degrees of procedural fairness constructed in Section 3.4.3. Additionally, we investigated the effect of different values of k , ranging from 100 to 2000. The results are shown in Fig. 10 below.

As shown in Fig. 10, on the *German* dataset, the counterfactual data generation strategy yields more accurate assessments of procedural fairness compared to direct matching with real data points. Additionally, increasing the number of generated candidates k leads to modest improvements in evaluation accuracy, but the difference

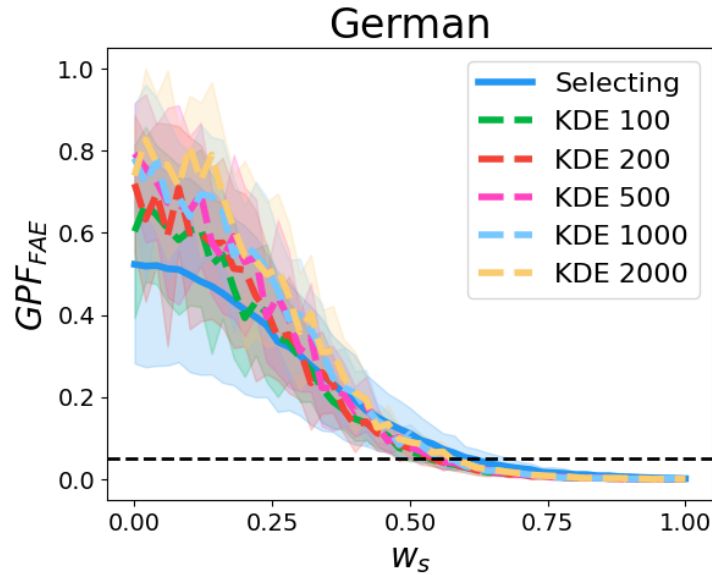


Fig. 10. The trend of the GPF_{FAE} metric value as w_s increases under different data point matching methods on the *German* dataset. Figures show mean values over 10 random runs with a $1 - \sigma$ error-bar.

is not significant. These findings demonstrate that counterfactual data generation is a promising direction for extending the applicability of GPF_{FAE} to data-sparse environments.

Nonetheless, we emphasize that such generative methods should be viewed as a complementary strategy rather than a replacement for similarity-based matching. In practice, when the dataset is sufficiently large (e.g., more than 1000 samples, as discussed in Section 3.4.7), directly identifying real, similar data points remains the most reliable approach, as it avoids assumptions or potential artifacts introduced by synthetic generation. Future research could explore more advanced or task-specific counterfactual generation methods, such as those based on conditional generative models or diffusion processes, to further enhance evaluation reliability.

4 Methods for Mitigating Procedural Unfairness in ML

In this section, we focus on mitigating procedural unfairness in ML models. Given a procedural-unfair ML model, we introduce a method to identify the sources that lead to its procedural unfairness, i.e., to find which features contribute to procedural unfairness (unfair features). By identifying these unfair features, we gain insights into the underlying causes of procedural unfairness. Then, we propose two distinct methods to mitigate the procedural fairness of the ML model based on the identified unfair features. The experimental results show that our proposed methods significantly improve the procedural fairness of models.

4.1 Identify Features that Lead to Procedural Unfairness

In Section 3.3, we evaluate the procedural fairness of the model by assessing the degree of difference between the distributions of the two explanation sets E_1 and E_2 . In this subsection, given a procedural-unfair ML model, we aim to identify the features that lead to its procedural unfairness, called unfair features (UFs). To detect these unfair features, we assess whether the difference in each feature (that is, the importance score for each input

feature) of the two explanation sets E_1 and E_2 is less than a predefined threshold, which is defined as

$$UFs = \{i | d_\Phi(E_{1,i}, E_{2,i}) \leq \beta; i = \{1, 2, \dots, d\}, \quad (9)$$

where $E_{1,i}$ and $E_{2,i}$ denote the explanation results on the i -th feature in the explanation sets E_1 and E_2 , respectively. β is a threshold. Consistent with Section 3.3, we also used the MMD with an exponential kernel function as d_Φ to measure the difference between $E_{1,i}$ and $E_{2,i}$, and the p -value obtained by performing a permutation test on the generated kernel matrix was used as the final evaluation result. If the p -value is less than the threshold β , the feature is considered as a UF . Similar to the experiments in Section 3.4, the threshold β was set to 0.05. However, users may set the threshold differently based on their specific requirements for the degree of procedural fairness.

In this subsection, we conduct experiments to identify UFs on the nine procedural-unfair models constructed in Section 3.4.2 on the nine datasets. The average number of UFs detected and the average runtime of the detection from 10 independent runs on each dataset are listed in Table 7.

Table 7. The average number and standard deviation of UFs detected and the average runtime in 10 independent runs on each dataset.

Dataset	Loss Function	Number of UFs	Runtime (in seconds)
Adult	BCE	2.80 ± 0.92	254.26
Dutch	BCE	2.10 ± 1.29	241.16
LSAT	$BCE - 0.05 \times DP$	1.50 ± 0.97	242.46
COMPAS	$BCE - 0.1 \times DP$	1.40 ± 0.52	18.38
Unfair German	BCE	3.10 ± 1.37	262.64
Unfair KDD	BCE	5.50 ± 2.72	279.74
Unfair Bank	BCE	1.40 ± 0.70	257.56
Unfair Default	BCE	1.00 ± 0.00	263.44
Synthetic	BCE	2.00 ± 0.00	4.02

According to our experiments, for the *Synthetic* dataset, our method was able to accurately identify the sensitive attribute x_s and its proxy attribute x_p as UFs in each independent run. On the *Default* dataset, only the sensitive attribute “sex” was detected as UF each time, and this result is similar to the result of selecting fair features by Pearson correlation coefficient in Section 3.4.2, where *Default* is the only dataset where the correlation coefficients of all other features and sensitive attributes are below the threshold 0.10. However, on most of the datasets, there is some fluctuation in the UFs detected in different independent runs. For example, on the *Adult* dataset, “sex” and “relationship” are the two unfair features detected each time, and it is correct that “sex” is exactly the sensitive attribute under consideration, while “relationship” includes about 11,000 data with the value of “husband” or “wife”, which implies strong gender information, and can be regarded as a proxy attribute of “sex”. And features such as “hours-per-week” and “race” are sometimes seen as unfair features.

It is precisely because the features that cause procedural unfairness differ across models that our model-based approach is significant in identifying UFs . Our proposed approach can effectively identify UFs that result in inconsistent decision-making processes between different groups and hence enable targeted interventions for improving fairness in ML models.

4.2 Method 1: Retraining the Model by Eliminating UFs

The first method to improve the procedural fairness of the model is straightforward, i.e., directly removing the detected UFs and retraining the model. We evaluate the effectiveness of this method in three aspects: before and

after removing the detected UFs (1) the changes in the GPF_{FAE} metric to evaluate changes in procedural fairness; (2) the changes in the DP , EO , and EOD metrics to evaluate changes in distributive fairness; and (3) the changes in model accuracy to evaluate changes in overall model performance.

Fig. 11 presents the changes in GPF_{FAE} , DP , EO , and EOD metrics of the model before and after removing the detected UFs . The results demonstrate that the procedural fairness of the model significantly improves after removing the detected UFs on each dataset, especially on the *Dutch*, *COMPAS*, *KDD*, *Bank*, *Default*, and *Synthetic* datasets, where the GPF_{FAE} metric is able to reach or approach 1.0. In addition, there is also a considerable improvement in the distributive fairness metrics, with many of the datasets being able to fall below the distributive fairness threshold of 0.10, i.e., they can be considered distributive-fair models. This result not only validates the effectiveness of our proposed method in improving the fairness of the model but also demonstrates the accuracy of the detected UFs .

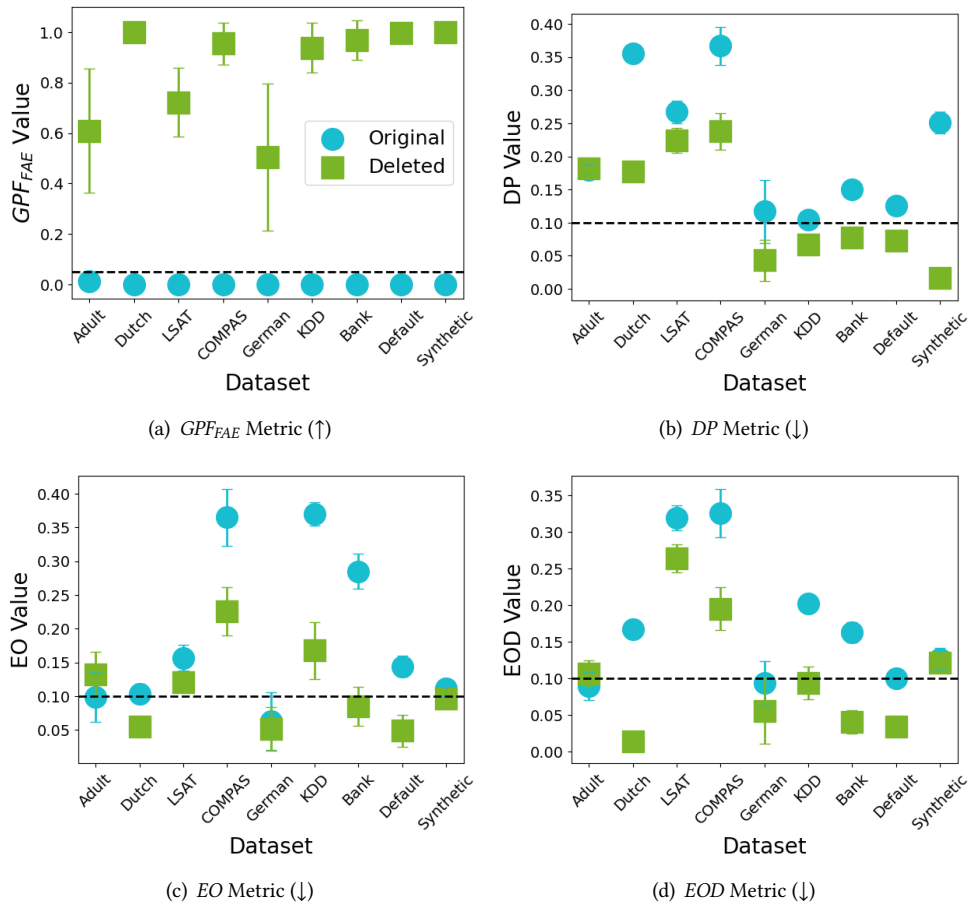


Fig. 11. Changes in GPF_{FAE} , DP , EO , and EOD metrics before and after removing the detected UFs . Error bars indicate standard deviations. The \uparrow means that the larger the metric, the better, and \downarrow means the opposite.

In addition, we show the changes in model accuracy on the test set before and after removing the detected *UFs*, as shown in Table 8 below. We also include the average retraining time in the table. As can be seen, the removal of detected *UFs* has the most significant impact on the accuracy of the model for the *Synthetic* dataset. This is expected since two features were removed from the *Synthetic* dataset with only four features in total, and the average accuracy decline over the nine datasets was not significant, at 0.8%. Additionally, the retraining time cost is low, averaging 3.48 seconds.

Table 8. Changes in model accuracy on the test set before and after removing the detected *UFs* and the average retraining time for the model

. “ Δ ” is the difference between the two.

Dataset	Original Acc	Deleted Acc	Δ	Deleted Training Time (in seconds)
Adult	85.1%±0.3%	84.9%±0.3%	0.2%±0.2%	3.76
Dutch	83.7%±0.4%	82.0%±0.5%	1.7%±0.3%	4.42
LSAT	89.9%±0.3%	89.9%±0.3%	0.0%±0.1%	2.44
COMPAS	68.1%±0.9%	68.0%±1.1%	0.1%±1.1%	1.15
Unfair German	79.7%±2.6%	78.0%±1.8%	1.7%±1.2%	1.76
Unfair KDD	94.0%±0.1%	93.3%±0.3%	0.7%±0.3%	9.70
Unfair Bank	86.2%±0.4%	86.1%±0.3%	0.1%±0.3%	3.93
Unfair Default	79.7%±0.4%	79.5%±0.3%	0.2%±0.2%	2.82
Synthetic	83.1%±0.5%	81.1%±0.7%	2.0%±0.5%	1.35
Mean	83.3%±0.7%	82.5%±0.6%	0.8%±0.5%	3.48

Overall, by directly removing the detected *UFs*, retraining the model is able to obtain a procedural-fair model, and the distributive fairness of the model is also improved, while the accuracy decrease of the model is only about 0.8% on average.

4.3 Method 2: Modifying the Model by Reducing the Impacts of *UFs*

Although the method of directly removing the detected *UFs* is simple and straightforward, the biggest problem is that it requires retraining the whole model. This may result in the decision logic of the retrained model differing significantly from the original model, i.e., not faithful to the original model, which may be undesirable.

Rather than retraining the whole model, we prefer to improve fairness by adapting to the existing model, which typically allows its decision logic to be more faithful to the original model. Among them, Dimanov *et al.* 2020 proposed a method for modifying the existing model. For an already trained model f_{θ} parameterized by θ (in this paper, it is equivalent to f in the previous section), they find a modified model f'_{θ} with the following two properties:

1. *Model similarity*: the model has similar performance before and after modifying

$$\forall i, f'_{\theta}(\mathbf{x}^{(i)}) \approx f_{\theta}(\mathbf{x}^{(i)}). \quad (10)$$

2. *Low sensitive attribute importance score*: the importance score provided by the explanation function g on the sensitive attribute s (j -th feature) decreases significantly after modifying

$$\forall i, |g(f'_{\theta}, \mathbf{x}^{(i)})_j| \ll |g(f_{\theta}, \mathbf{x}^{(i)})_j|. \quad (11)$$

Straightforwardly, Dimanov *et al.* 2020 reduce the impact of the sensitive attribute on the decision-making by modifying the model, while attempting to maintain the model's performance.

However, the purpose of Dimanov *et al.* 2020 is not to improve the fairness of the model. Instead, they found that after modifying the model, although the FAE obtained low importance scores for the sensitive attribute (tending towards 0.0), it was still a (distributive) unfairness model by metrics such as *DP*, *EO*, and *EOD*. This leads the authors to conclude that the (distributive) fairness of the model cannot be evaluated by the importance score of the sensitive attribute obtained by FAE.

On the one hand, due to the presence of proxy attributes, we agree that it is difficult to assess the fairness of a model by evaluating the importance score of the sensitive attribute alone. However, we believe that this approach contributes to the procedural fairness of the model because of its ability to effectively reduce (or even eliminate) the influence of the sensitive attribute on the decision-making process.

However, as mentioned before, it is not enough to deal with just the sensitive attribute because of the presence of the proxy attribute. Therefore, in this paper, we present an improved method to modify the existing model by building upon the approach proposed by Dimanov *et al.* 2020. Instead of only modifying the sensitive attribute, we modified all the detected *UFs*, thereby reducing the impact of all the detected *UFs* on the model decisions, and ultimately improving the procedural fairness of the model. Specifically, for the trained model f_θ , we modified it by optimizing an additional penalty term, called the explanation loss ζ , weighted by the hyper-parameter α , and normalized over all m training data points:

$$\begin{aligned}\mathcal{L}' &= \mathcal{L} + \alpha \times \zeta, \\ \zeta &= \sum_{k=1}^{|UFs|} \frac{1}{m} \times L^p\left(\left|\frac{\partial \mathcal{L}}{x_k^{(1)}}\right|, \left|\frac{\partial \mathcal{L}}{x_k^{(2)}}\right|, \dots, \left|\frac{\partial \mathcal{L}}{x_k^{(m)}}\right|\right),\end{aligned}\quad (12)$$

where \mathcal{L} is the cross-entropy loss of the model, and we used L^1 norm to be consistent with Dimanov *et al.* 2020. The pseudo-code is shown in Algorithm 2, where we took the number of iterations $\tau = 200$, which is sufficient for convergence. In our experiments, we took explanation loss weight $\alpha = 15$ (the selection of parameter α was chosen based on our experimentation and will be discussed later).

Algorithm 2 Modifying the existing model to improve procedural fairness.

Input: Original trained model f_θ , unfair features *UFs*, input matrix $X \in \mathbb{R}^{m \times d}$ with corresponding labels $Y \in \mathbb{R}^{m \times 1}$, iteration number τ , and explanation loss weight α

Output: The modified model f'_θ

- 1: $i = 0$
- 2: **while** $i < \tau$ **do**
- 3: Calculate the cross entropy loss \mathcal{L} with respect to f_θ
- 4: Calculate the explanation loss ζ

$$\zeta = \sum_{k=1}^{|UFs|} \frac{1}{m} \times L^p\left(\left|\frac{\partial \mathcal{L}}{x_k^{(1)}}\right|, \left|\frac{\partial \mathcal{L}}{x_k^{(2)}}\right|, \dots, \left|\frac{\partial \mathcal{L}}{x_k^{(m)}}\right|\right)$$

- 5: Calculate the total loss $\mathcal{L}' = \mathcal{L} + \alpha \times \zeta$
 - 6: Update model parameters with $\nabla_\theta \mathcal{L}'$ using Adam
 - 7: $i = i + 1$
 - 8: **end while**
 - 9: **return** the modified model f_θ as f'_θ
-

Straightforwardly, we improved the procedural fairness of the model by reducing the impact of all the detected *UFs* on the model decisions. Similarly, we evaluated the performance of the method in terms of three aspects of the model’s procedural fairness, distributive fairness, and changes in model performance before and after modification.

The changes in GPF_{FAE} , DP , EO , and EOD metrics before and after modifying the model are shown in Fig. 12. We can see that the method significantly improves the procedural fairness of the model, while the distributive fairness of the model is also improved to some extent.

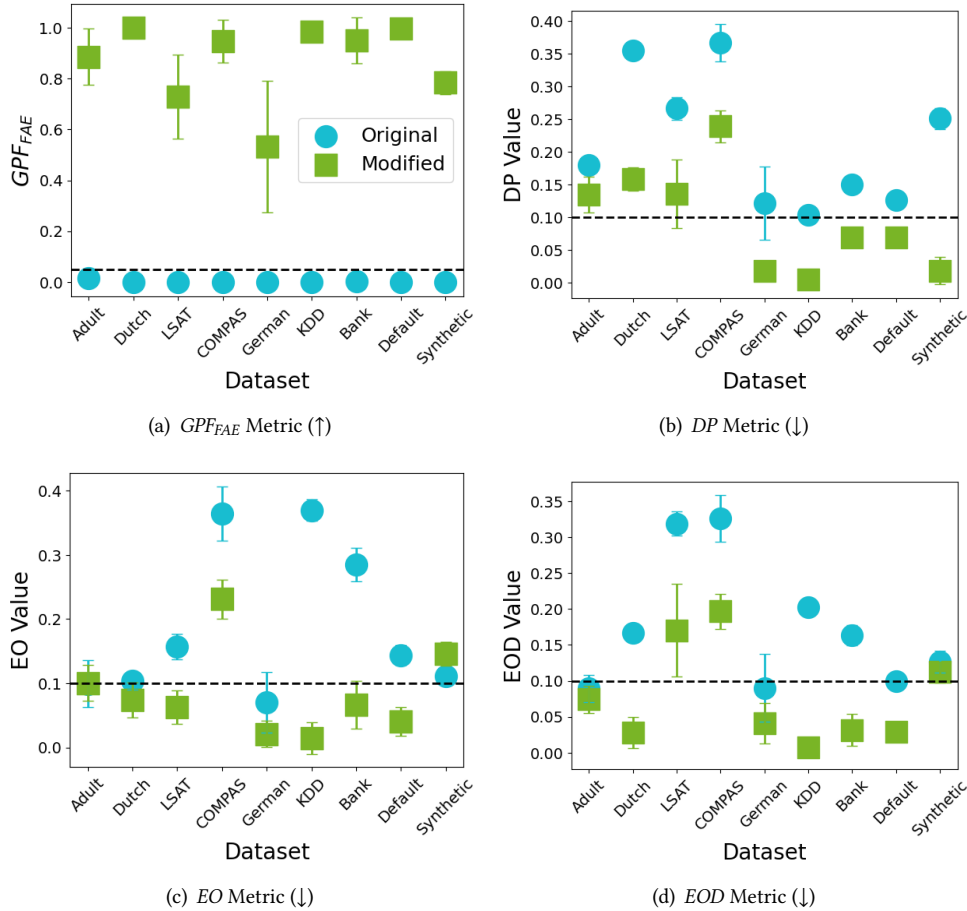


Fig. 12. Changes in GPF_{FAE} , DP , EO , and EOD metrics before and after modifying the detected *UFs*. Error bars indicate standard deviations. The \uparrow means that the larger the metric, the better, and \downarrow means the opposite.

The comparison of model accuracy before and after modifying the existing model is shown in Table 9. The table also lists the average time required to modify the model. It illustrates that the modification method has a greater impact on model performance compared to the retraining method, with an average model accuracy loss of 1.8% (1.3% on the eight real-world datasets). In addition, it runs longer, but is still fast, averaging 24.97 seconds.

Table 9. Changes in model accuracy on the test set before and after modifying the detected UFs and the average time for modifying the model. “ Δ ” is the difference between the two.

Dataset	Original Acc	Perturbed Acc	Δ	Perturbed Training Time (in seconds)
Adult	85.1%±0.2%	84.0%±0.7%	1.1%±0.5%	30.11
Dutch	83.7%±0.4%	80.7%±0.6%	3.0%±0.6%	41.17
LSAT	89.9%±0.3%	89.9%±0.4%	0.0%±0.3%	13.90
COMPAS	68.1%±0.9%	67.9%±1.5%	0.3%±1.2%	1.31
Unfair German	79.7%±2.6%	76.7%±3.0%	3.0%±2.3%	1.51
Unfair KDD	94.0%±0.1%	91.9%±0.4%	2.1%±0.5%	89.73
Unfair Bank	86.2%±0.4%	85.4%±0.7%	0.8%±0.6%	24.58
Unfair Default	79.7%±0.4%	79.5%±0.4%	0.2%±0.1%	21.20
Synthetic	83.1%±0.5%	77.6%±0.8%	5.5%±0.7%	1.25
Mean	83.3%±0.6%	81.5%±0.9%	1.8%±0.8%	24.97

Finally, we discuss the impact of different explanation loss weights α . We investigated the trends of the explanation loss ζ and decrease in accuracy of the modified model under different explanation loss weights α (taken as 0.1, 1, 5, 10, 15, 20, 50, 100, respectively), as shown in Fig. 13. The results indicate that, on the one hand, when the value of α is small, the explanation loss ζ is still large despite the low decrease of the model accuracy, i.e., it does not reduce the influence of UFs on decision making; on the other hand, when α takes a large value, it has a disastrous effect on the performance of the model. When α takes values between [10, 20], the results exhibit good stability, and we can achieve a better trade-off between these two goals, i.e., we can reduce the explanation loss ζ while affecting the model performance in a smaller way. In the experiments of this paper, we set $\alpha = 15$.

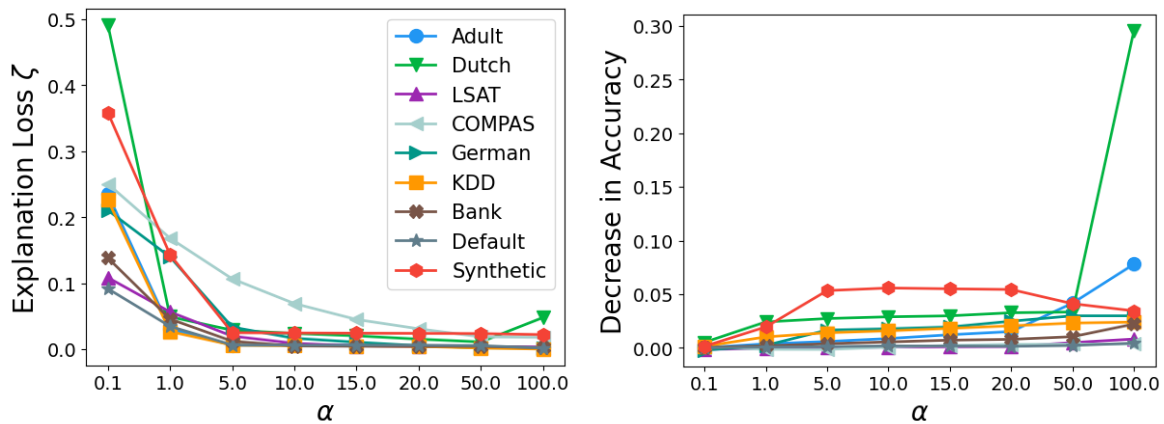
Fig. 13. Trends in the explanation loss ζ and decrease in accuracy of the modified model with different explanation loss weights α .

Fig. 13 also demonstrates that there is a conflict between model performance and procedural fairness, and that users have the flexibility to control the α value to make a trade-off between model performance and procedural fairness.

4.4 Similarities and Differences between the Two Methods

The two approaches we proposed to improve the procedural fairness of the model are very similar. They both work by reducing or removing detected *UFs* so that they no longer influence the model’s decisions. They are both able to improve the procedural fairness of the model to a large extent, while the distributive fairness of the model is also improved.

The primary difference between the two is that, compared to modifying the existing model, retraining after removing the *UFs* tends to be less detrimental to the model’s performance. However, since the modification approach is modified from the obtained model, the decision logic tends to be more “faithful” to the original model, i.e., the decision logic of the two is more similar. To visualize this, we projected the decision boundaries of the original model, the modified model, and the retrained model on the *Unfair Default* dataset in 2D PCA projected space, as shown in Fig. 14. We can see that the left side of the decision boundary of the retrained model has changed significantly compared to the original model, while the modified model is very similar to the original model. In addition, compared to the retraining approach, the modification approach allows for more flexible and fine-grained trade-offs between model performance and procedural fairness by controlling the parameter α .

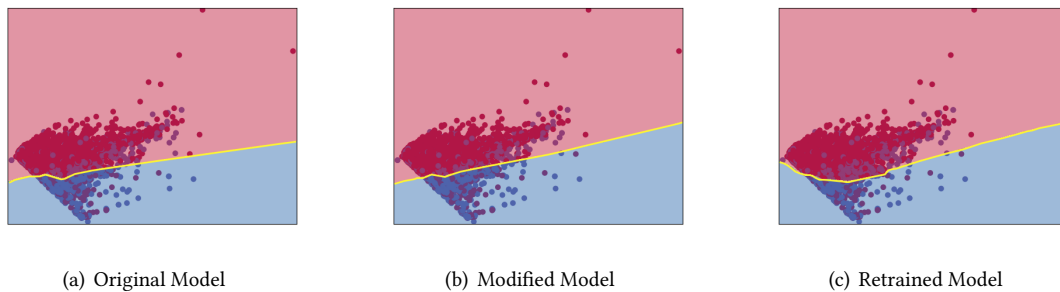


Fig. 14. Decision boundaries for the original, modified and retrained models visualized on the *Unfair Default* dataset in 2D reduced input space (dimensionality reduction by PCA). The red and blue backgrounds indicate negative and positive predictions, respectively. Each point represents a 2D projection of each data point in the dataset, and its color indicates its true label.

5 Conclusion

In this paper, we first proposed a definition of procedural fairness in ML. Then, we introduce a novel group procedural fairness metric based on the FAE approach, called GPF_{FAE} . Through comprehensive experimentation on nine datasets, we demonstrated the efficacy of GPF_{FAE} in accurately assessing the procedural fairness of the ML model. After assessing the procedural fairness of the model, for a procedural-unfair model, we proposed a method to identify the features that lead to the procedural unfairness (i.e., unfair features) and proposed two methods to improve the procedural fairness of the model based on the detected unfair features. Experiments on nine datasets indicate that our method can accurately detect the features that lead to the procedural unfairness of the model, and both proposed mitigation methods can significantly improve the procedural fairness of the model

while also increasing the distributive fairness, with only a slight decrease in the model’s performance. Overall, the work presented in this paper significantly advances the understanding and methodologies for enhancing the procedural fairness of ML models.

In future research, there are several aspects worthy of further investigation:

- (1) **Enhancing Counterfactual Matching in Sparse Datasets:** As discussed in Section 3.4.7, GPF_{FAE} metric requires sampling n pairs of similar data points with different sensitive attribute values. In sparse datasets, such matching can be unreliable. We proposed an initial solution using KDE-based counterfactual generation to address this limitation. Future research could explore more advanced or data-efficient sample generation techniques to further improve the reliability of fairness assessment, or develop alternative methodologies that do not depend on the availability of similar data points at all.
- (2) **Integrating Procedural Fairness into Training:** It would be worthwhile to explore the utilization of procedural fairness metrics to guide the training of ML models, so as to improve the procedural fairness of the models during the training process.
- (3) **Extending to Individual Procedural Fairness:** Although this paper gives the definition of individual and group procedural fairness of ML models, it only studies the group procedural fairness in depth. Future work should explore how to quantify and improve individual procedural fairness in ML models.
- (4) **Validating Generalizability and Real-World Applicability:** Since GPF_{FAE} is a newly proposed metric, its generalizability and practical value across diverse scenarios still require further empirical validation. In future work, we plan to systematically evaluate the adaptability and robustness of GPF_{FAE} across various sensitive attributes and multi-group fairness settings, as well as to explore its potential as a reference metric aligned with emerging AI regulatory and standardization frameworks.
- (5) **Adapting GPF_{FAE} for Sequential and Time-Series Tasks:** Applying GPF_{FAE} (or its variants) to time series or sequential decision-making tasks is a promising area for future expansion. Such scenarios involve time dependency and require adaptive adjustments to attribution techniques and fairness definitions.
- (6) **Balancing Fairness and Model Performance:** Prior research has underscored the existence of conflicts between the model performance and the distributive fairness (Caton and Haas 2024; Friedler et al. 2019; Speicher et al. 2018). In this paper, we reveal that there is also a conflict between procedural fairness and model performance. Therefore, how to trade-off and comprehensively consider the model performance, procedural fairness, and distributive fairness is a challenging and desirable task for future research.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62250710682), the Guangdong Provincial Key Laboratory (Grant No. 2020B121201001), the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (Grant No. 2017ZT07X386), and an internal grant of Lingnan University.

References

- B. Abdollahi and O. Nasraoui. 2018. “Transparency in fair machine learning: The case of explainable recommender systems.” In: *Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent*. Springer, 21–35.
- M. L. Ambrose and A. Arnaud. 2013. “Are procedural justice and distributive justice conceptually distinct?” In: *Handbook of Organizational Justice*. Psychology Press, 59–84.
- J. Angwin, J. Larson, S. Mattu, and L. Kirchner. 2016. “Machine bias.” In: *Ethics of Data and Analytics*. Auerbach Publications, 254–264.
- A. B. Arrieta et al.. 2020. “Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI.” *Information Fusion*, 58, 82–115.
- E. Balkir, S. Kiritchenko, I. Nejadgholi, and K. C. Fraser. 2022. “Challenges in applying explainability methods to improve the fairness of NLP models.” In: *Proceedings of the 2nd Workshop on Trustworthy Natural Language Processing (TrustNLP 2022)*, 80–92.
- T. Begley, T. Schwedes, C. Frye, and I. Feige. 2020. “Explainability for fair machine learning.” *arXiv preprint arXiv:2010.07389*.

- R. K. Bellamy et al.. 2019. "AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias." *IBM Journal of Research and Development*, 63, 4/5, 4–1.
- E. Benussi, A. Patane', M. Wicker, L. Laurenti, and M. Kwiatkowska. 2022. "Individual fairness guarantees for neural networks." In: *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 651–658.
- U. Bhatt, A. Weller, and J. M. Moura. 2021. "Evaluating and aggregating feature-based model explanations." In: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 3016–3022.
- U. Bhatt, A. Xiang, et al.. 2020. "Explainable machine learning in deployment." In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 648–657.
- J. Blandin and I. A. Kash. 2023. "Generalizing group fairness in machine learning via utilities." *Journal of Artificial Intelligence Research*, 78, 747–780.
- V. d. Bos, E. A. Lind, and H. A. M. Wilke. 2001. "The psychology of procedural and distributive justice viewed from the perspective of fairness heuristic theory." *Justice in the Workplace: From Theory to Practice*, 2, 49–66.
- T. Calders and S. Verwer. 2010. "Three naive bayes approaches for discrimination-free classification." *Data Mining and Knowledge Discovery*, 21, 277–292.
- S. Caton and C. Haas. 2024. "Fairness in machine learning: A survey." *ACM Comput. Surv.*, 56, 7, 1–38.
- N. Chen, B. Ribeiro, and A. Chen. 2016. "Financial credit risk assessment: A recent review." *Artificial Intelligence Review*, 45, 1–23.
- J. Dai, S. Upadhyay, U. Aivodji, S. H. Bach, and H. Lakkaraju. 2022. "Fairness via explanation quality: Evaluating disparities in the quality of post hoc explanations." In: *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, 203–214.
- B. Dimanov, U. Bhatt, M. Jamnik, and A. Weller. 2020. "You shouldn't trust me: Learning models which conceal unfairness from multiple explanation methods." In: *European Conference on Artificial Intelligence*, 2473–2480.
- J. Dressel and H. Farid. 2018. "The accuracy, fairness, and limits of predicting recidivism." *Science Advances*, 4, 1, eaao5580.
- D. Dua and C. Graff. 2017. *UCI Machine Learning Repository*. Available via <http://archive.ics.uci.edu/ml>, Accessed 28 Jul 2025. (2017).
- C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. 2012. "Fairness through awareness." In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 214–226.
- European Union. 2025. *AI Act (Regulation (EU) 2024/1689)*. Available via <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32024R1689>, Accessed 22 Jul 2025. (2025).
- A. Fabris, A. Esuli, A. Moreo, and F. Sebastiani. 2023. "Measuring fairness under unawareness of sensitive attributes: A quantification-based approach." *Journal of Artificial Intelligence Research*, 76, 1117–1180.
- R. Folger. 1987. "Distributive and procedural justice in the workplace." *Social Justice Research*, 1, 143–159.
- S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, S. Choudhary, E. P. Hamilton, and D. Roth. 2019. "A comparative study of fairness-enhancing interventions in machine learning." In: *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 329–338.
- B. Green and L. Hu. 2018. "The myth in the methodology: Towards a recontextualization of fairness in machine learning." In: *Proceedings of the Machine Learning: The Debates Workshop at the 35th International Conference on Machine Learning*.
- J. Greenberg. 1987. "A taxonomy of organizational justice theories." *Academy of Management Review*, 12, 1, 9–22.
- N. Grgić-Hlača, M. B. Zafar, K. P. Gummadi, and A. Weller. 2018. "Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning." In: *Proceedings of the AAAI Conference on Artificial Intelligence* 1. Vol. 32, 51–60.
- R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi. 2018. "A survey of methods for explaining black box models." *ACM Computing Surveys (CSUR)*, 51, 5, 1–42.
- M. Hardt, E. Price, and N. Srebro. 2016. "Equality of opportunity in supervised learning." *Advances in Neural Information Processing Systems*, 29, 3315–3323.
- C. Huang, Z. Zhang, B. Mao, and X. Yao. 2023. "An Overview of Artificial Intelligence Ethics." *IEEE Transactions on Artificial Intelligence*, 4, 4, 799–819.
- G. P. Jones, J. M. Hickey, P. G. Di Stefano, C. Dhanjal, L. C. Stoddart, and V. Vasileiou. 2020. "Metrics and methods for a systematic comparison of fairness-aware machine learning algorithms." *arXiv preprint arXiv:2010.03986*.
- F. Kamiran and T. Calders. 2012. "Data preprocessing techniques for classification without discrimination." *Knowledge and Information Systems*, 33, 1, 1–33.
- T. Le Quy, A. Roy, V. Iosifidis, W. Zhang, and E. Ntoutsi. 2022. "A survey on datasets for fairness-aware machine learning." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 12, 3, e1452.
- G. S. Leventhal. 1980. "What should be done with equity theory? New approaches to the study of fairness in social relationships." In: *Social Exchange: Advances in Theory and Research*. Springer, 27–55.
- J. Li, B. Mao, Z. Liang, Z. Zhang, Q. Lin, and X. Yao. 2021. "Trust and trustworthiness: What they are and how to achieve them." In: *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. IEEE, 711–717.

- L. Li, T. Lassiter, J. Oh, and M. K. Lee. 2021. "Algorithmic hiring in practice: Recruiter and HR professional's perspectives on AI use in hiring." In: *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society (AIES '21)*. Association for Computing Machinery, Virtual Event, USA, 166–176. ISBN: 9781450384735.
- S. M. Lundberg and S.-I. Lee. 2017. "A unified approach to interpreting model predictions." *Advances in Neural Information Processing Systems*, 30, 4765–4774.
- N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. 2021. "A survey on bias and fairness in machine learning." *ACM Computing Surveys (CSUR)*, 54, 6, 1–35.
- L. Morse, M. H. M. Teodorescu, Y. Awwad, and G. C. Kane. 2022. "Do the ends justify the means? Variation in the distributive and procedural fairness of machine learning algorithms." *Journal of Business Ethics*, 181, 4, 1083–1095.
- OECD. 2024. *OECD AI principles*. Available via <https://www.oecd.org/en/topics/ai-principles.html>, Accessed 27 Jul 2025. (2024).
- W. Pan, S. Cui, J. Bian, C. Zhang, and F. Wang. 2021. "Explaining algorithmic fairness through fairness-aware causal path decomposition." In: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 1287–1297.
- D. Pessach and E. Shmueli. 2022. "A review on fairness in machine learning." *ACM Computing Surveys (CSUR)*, 55, 3, 1–44.
- A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje. 2016. "Not just a black box: Learning important features through propagating activation differences." *arXiv preprint arXiv:1605.01713*.
- T. Speicher, H. Heidari, N. Grgic-Hlaca, K. P. Gummadi, A. Singla, A. Weller, and M. B. Zafar. 2018. "A unified approach to quantifying algorithmic unfairness: Measuring individual & group unfairness via inequality indices." In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2239–2248.
- A. Stevens, P. Deruyck, Z. Van Veldhoven, and J. Vanthienen. 2020. "Explainability and fairness in machine learning: Improve fair end-to-end lending for kiva." In: *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 1241–1248.
- M. Sundararajan, A. Taly, and Q. Yan. 2017. "Axiomatic attribution for deep networks." In: *International Conference on Machine Learning*, 3319–3328.
- Z. Tang, J. Zhang, and K. Zhang. 2023. "What-is and how-to for fairness in machine learning: A survey, reflection, and perspective." *ACM Computing Surveys*, 55, 13s, 1–37.
- J. W. Thibaut and L. Walker. 1975. *Procedural justice: A psychological analysis*. Hillsdale, NJ: Erlbaum.
- K. Van den Bos, H. A. Wilke, and E. A. Lind. 1998. "When do we need procedural fairness? The role of trust in authority." *Journal of Personality and Social Psychology*, 75, 6, 1449–1458.
- P. Van der Laan. 2000. "The 2001 census in the netherlands." In: *Conference The Census of Population*.
- Z. Wang, C. Huang, Y. Li, and X. Yao. 2024. "Multi-objective feature attribution explanation for explainable machine learning." *ACM Trans. Evol. Learn. Optim.*, 4, 1, 1–32.
- Z. Wang, C. Huang, K. Tang, and X. Yao. 2025. "Procedural fairness and its relationship with distributive fairness in machine learning." *arXiv preprint arXiv:2501.06753*.
- Z. Wang, C. Huang, and X. Yao. 2024. "A roadmap of explainable artificial intelligence: Explain to whom, when, what and how?" *ACM Trans. Auton. Adapt. Syst.*, 19, 4, 1–40.
- Z. Wang, C. Huang, and X. Yao. 2023. "Feature attribution explanation to detect harmful dataset shift." In: *2023 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.
- L. F. Wightman. 1998. "LSAC national longitudinal bar passage study. LSAC research report series."
- R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. 2013. "Learning fair representations." In: *International Conference on Machine Learning*. PMLR, 325–333.
- B. H. Zhang, B. Lemoine, and M. Mitchell. 2018. "Mitigating unwanted biases with adversarial learning." In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 335–340.
- Q. Zhang, J. Liu, Z. Zhang, J. Wen, B. Mao, and X. Yao. 2021. "Fairer machine learning through multi-objective evolutionary learning." In: *International Conference on Artificial Neural Networks*, 111–123.
- Q. Zhang, J. Liu, Z. Zhang, J. Wen, B. Mao, and X. Yao. 2022. "Mitigating unfairness via evolutionary multiobjective ensemble learning." *IEEE Transactions on Evolutionary Computation*, 27, 4, 848–862.
- Y. Zhao, Y. Wang, and T. Derr. 2023. "Fairness and explainability: Bridging the gap towards fair model explanations." In: *Proceedings of the AAAI Conference on Artificial Intelligence* 9. Vol. 37, 11363–11371.

Received 22 September 2025; accepted 10 January 2026