Towards Fair Classifiers Without Sensitive Attributes: Exploring Biases in Related Features

Tianxiang Zhao[†], Enyan Dai[†], Kai Shu[‡], Suhang Wang[†]
[†]College of Information Sciences and Technology, The Pennsylvania State University, USA
[‡]Department of Computer Science, College of Computing, Illinois Institute of Technology, USA
{tkz5084,emd5759,szw494}@psu.edu,kshu@iit.edu

ABSTRACT

Despite the rapid development and great success of machine learning models, extensive studies have exposed their disadvantage of inheriting latent discrimination and societal bias from the training data. This phenomenon hinders their adoption on high-stake applications. Thus, many efforts have been taken for developing fair machine learning models. Most of them require that sensitive attributes are available during training to learn fair models. However, in many real-world applications, it is usually infeasible to obtain the sensitive attributes due to privacy or legal issues, which challenges existing fair-ensuring strategies. Though the sensitive attribute of each data sample is unknown, we observe that there are usually some non-sensitive features in the training data that are highly correlated with sensitive attributes, which can be used to alleviate the bias. Therefore, in this paper, we study a novel problem of exploring features that are highly correlated with sensitive attributes for learning fair and accurate classifiers. We theoretically show that by minimizing the correlation between these related features and model prediction, we can learn a fair classifier. Based on this motivation, we propose a novel framework which simultaneously uses these related features for accurate prediction and enforces fairness. In addition, the model can dynamically adjust the regularization weight of each related feature to balance its contribution on model classification and fairness. Experimental results on real-world datasets demonstrate the effectiveness of the proposed model for learning fair models with high classification accuracy.

CCS CONCEPTS

• **Computing methodologies** → *Regularization*; Neural networks; *Machine learning*.

KEYWORDS

Fairness; Social mining; Data learning

1 INTRODUCTION

With the great improvement in performance, modern machine learning models are becoming increasingly popular and are widely used in decision-making systems such as medical diagnosis [2] and credit scoring [8]. Despite their great successes, extensive studies [13, 26, 34] have revealed that training data may include patterns of previous discrimination and societal bias. Machine learning models trained on such data can inherit the bias on sensitive attributes such as ages, genders, skin color, and regions [3, 9, 15]. For example, a study found strong unfairness exists in a *Criminal Prediction* system used to assess a criminal defendant's likelihood of becoming a recidivist [17]. The system shows a strong bias towards people with color, tending to predict them as recidivist even when they are not. Thus, hidden biases in a machine learning model could cause severe fairness problems, which raises concerns on their real-world applications, especially in high-stake scenarios.

Various efforts [11, 18, 28, 35] have been taken to address the fairness issue of current machine learning models. For example, [11, 19] pre-process the data to remove discrimination in training. [9, 35] design special regularization terms to ensure that the prediction output is insensitive w.r.t sensitive attributes. And [15, 27] post-process prediction results on instances of unfair classes. Despite their superior performance, all the aforementioned approaches require that sensitive attributes are available for removing bias. However, for many real-world applications, it is difficult to obtain sensitive attributes of each data sample due to various reasons such as privacy and legal issues, or difficulties in data collection [5, 21].

Tackling fairness issue without sensitive attributes available is challenging as we lack supervision to preprocess the training data, regularize the model or post-process the predictions. There are only very few initial efforts on learning fair classifiers without sensitive attributes [5, 21, 33]. Yan et al. [33] use a clustering algorithm to form pseudo groups to approximate real protected groups. Lahoti et al. [21] propose to use an auxiliary module to find computationally-identifiable regions where model under-performs, and optimize this worst-case performance. However, these works are often found to be ineffective in achieving fairness with demographics [21]. In addition, the groups or regions found by these approaches may not be related to the sensitive attribute we want to be fair with. For example, we might want the model to be fair on *gender*; while the clustering algorithm gives groups of *race*. Thus, more efforts need to be taken to address the important and challenging problem of learning fair models without sensitive attributes.

Though the sensitive attribute of each data sample is unknown, we observe that *there are usually some non-sensitive features in the training data that are highly correlated with sensitive attributes, which can be used to alleviate the bias*. Previous works [5, 17] observed that unfairness persists even when sensitive attributes are not used as input, which indicate that biases are embedded in some non-sensitive features used for training models. These non-sensitive features are highly correlated with sensitive attributes, which makes the model biased. We call such features as *Related Features*. These correlations arise from various reasons, such as biases in data collection, or interplay of an underlying physiological difference with socially determined role perception [4]. For example, Vogel and Porter [30] find that there exist striking differences in age distributions across racial/ethnic groups in US prisons. The Hispanic and black populations have a larger portion of individuals at younger ages, hence age is correlated with race in this field. In practice, common sense and prior domain knowledge can help to identify the related features given that we want to have a fair model on certain sensitive attributes. In addition, for different sensitive attribute such as *race* or *gender*, we can specify different sets of related features. With these related features identified, we would be able to alleviate the fairness issue. One straightforward way is to discard related features for training a fair model. However, it will also discard important information for classification. Thus, though promising, it remains an open question of how to effectively utilize related features to learn fair models with high classification accuracy.

Therefore, in this paper, we study a novel problem of exploring related features for learning fair and accurate classifiers without sensitive attributes. In essence, we are faced with three challenges: (i) how to utilize these related features to achieve fairness; (ii) how to achieve an optimal trade-off between accuracy and fairness; (iii) when given related feature sets contains misidentified features or are incomplete, how to adjust the usage of them. In an attempt to solve these challenges, we propose a novel framework F̲airness with R̲elated F̲eatures (FairRF). Instead of simply discarding related features, the basic idea of FairRF is to use the related features as both features for training the classifier and as pseudo sensitive attributes to regularize the behavior of it, which help to learn fair and accurate classifiers. We theoretically show that regularizing the model using related features can achieve fairness on sensitive attribute. Furthermore, to balance the classification accuracy and model fairness, and cope with the case when identified related attributes are inaccurate and noisy, FairRF can automatically learn the importance weight of each related feature for regularization in the model. The main contributions of the paper are as follows:

- We study a novel problem of exploring related features to learn fair classifiers without sensitive attributes;
- We theoretically show that by adopting related features to regularize the model, we can learn fairer classifier;
- We propose a novel framework FairRF which can simultaneously utilize the related features to learn fair classifiers and adjust the importance weights of each related feature; and
- We conduct extensive experiments on real-world datasets to demonstrate the effectiveness of the proposed method for fair classifiers with high classification accuracy.

2 RELATED WORK

To address the concerns of fairness in machine learning models, a number of fairness approaches are proposed. They can be generally split into three categories: (i) individual fairness [9, 20, 22, 36], which requires the model to give similar prediction to similar individuals; (ii) group fairness [9, 15, 39], which aims to treat the groups with different protected sensitive attributes equally; (iii) Max-Min fairness [16, 21, 38], which tries to maximize the minimum expected utility across groups. We focus on group fairness in this work.

Extensive works have been conducted to for group fairness-aware machine learning [3, 9, 15, 21, 23, 36, 39]. Based on the stage of applying fairness in training, these algorithms can be generally split into three categories: pre-processing approaches [19, 32, 39], in-processing approaches [35, 37], and post-processing approaches [15, 27]. Pre-processing approaches modify the training data to reduce the historical discrimination in the dataset. For instance, the bias could be eliminated by correcting labels [18, 39], revising attributes [11, 19], generating non-discriminatory data [28, 32], and obtaining fair representations [3, 6, 10, 23, 24, 36]. In-processing approaches revise the training of the state-of-the-art models to achieve fairness. More specifically, they apply fairness constraints or design a objective function considering the fairness of predictions [9, 35, 37]. Finally, the post-processing approaches directly change the predictive labels of trained models to obtain fair predictions [15, 27].

Despite their ability in alleviating the bias issues, aforementioned methods generally require the sensitive attributes of each data sample available to achieve fairness; while for many real-world applications, it is difficult to collect sensitive attributes of subjects due to various reasons such as privacy issues, legal problems and regulatory restrictions. The lacking of sensitive attributes of training data challenges the aforementioned methods [3]. Investigating fair models without sensitive attributes is important and challenging, and it is still in its early stage. There are only a few works on this direction [16, 21, 33]. One branch of approaches [16, 21] investigates fairness without demographics via solving a Max-Min problem. For instance, Lahoti et al. [21] proposes adversarial reweighted learning that leverages the notion of computationally-identifiable errors to achieve Rawlsian Max-Min fairness without sensitive attributes. However, these methods are only effective for achieving Max-Min fairness. The other branch [7, 33] addresses this missing sensitive attribute scenario via providing pseudo group splits. For instance, Yan et al. [33] pre-processes the data via clustering and uses obtained groups as the proxy. However, the conformity between obtained groups from these approaches and real protected groups are highly dependent on data distribution.

The proposed FairRF is inherently different from the aforementioned approaches: (i) We study a novel problem of exploring features that are highly related to the unseen sensitive ones for learning fair and accurate classifiers. Obtaining these features requires just a little prior domain knowledge, and it prevents the difficulty and instability of previous approaches in detecting protected groups [21, 33]; and (ii) We theoretically show that by regularizing the model prediction with the related features that are highly corrected with sensitive attributes, we can learn a fair model w.r.t the sensitive attribute. In addition, our experimental results show that the given related feature set can be incomplete or noisy.

3 PROBLEM DEFINITION

Throughout this paper, matrices are written as boldface capital letters and vectors are denoted as boldface lowercase letters. For an arbitrary matrix $\mathbf{M} \in \mathbb{R}^{n \times m}$, M_{ij} denotes the (i, j)-th entry of $\mathbf{M}$ while $\mathbf{m}_i$ and $\mathbf{m}^j$ mean the i-th row and j-th column of $\mathbf{M}$, respectively. Capital letters in calligraphic math font such as $\mathcal{P}$ are used to denote sets or cost function.

Let $\mathbf{X} \in \mathbb{R}^{n \times m}$ be the data matrix with each row $\mathbf{x}_i \in \mathbb{R}^{1 \times m}$ as an m-dimensional data instance. We use $\mathcal{F} = \{f_1, \dots, f_m\}$ to denote the m features and $\mathbf{x}^1, \dots, \mathbf{x}^m$ are the corresponding feature vectors, where $\mathbf{x}^j$ is the j-th column of $\mathbf{X}$. Let $\mathbf{y} \in \mathbb{R}^n$ be the label vector, where the i-th element of $\mathbf{y}$, i.e., y_i, is the label of $\mathbf{x}_i$. Following existing work on fair machine learning models [21], we focus on binary classification problem, i.e., $y_i \in \{0, 1\}$. Given $\mathbf{X}$ and $\mathbf{y}$, we aim to train a fair classifier with good classification performance.

Extensive studies [17, 21] have revealed that historical data may include previous discrimination and societal bias on sensitive attribute S such as ages, genders, skin color, and regions. Though sensitive attributes S are not used as features, i.e., $S \notin \mathcal{F}$, a subset of none-sensitive features $\mathcal{F}_s \in \mathcal{F}$ are highly correlated with sensitive attributes, making machine learning models trained on such data inherit the bias. For example, in dataset containing US criminal records [17], racial information is taken as sensitive. Although it is unseen, trained model could still be unfair as distribution of racial groups population may be leaked from the distribution of ages [30].

In many real-world applications, sensitive attributes of data samples are unavailable due to various reasons such as difficulty in data collection, security or privacy issues. It challenges existing fair machine leaning approaches that require sensitive attributes of data samples for fair models. Though sensitive attribute of each data sample is unknown, since the bias is caused by the subset of features $\mathcal{F}_S$ that are highly correlated with S, $\mathcal{F}_S$ can provide alternative supervision to learn fair models. Therefore, we aim to explore the utilization of $\mathcal{F}_S$ to help learn more fair model meanwhile maintain high classification performance. The problem is defined as:

Problem Definition Given the data matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$, with corresponding labels $\mathbf{y} \in \mathbb{R}^n$, and a predefined feature subset $\mathcal{F}_S \in \mathcal{F}$, where each $f_i \in \mathcal{F}$ called related feature which highly correlates with the unobserved protected attribute S, e.g., race or gender, learn a classifier that maintains high accuracy and is fair on S.

Note that we assume $\mathcal{F}_S \in \mathcal{F}$ is given from domain knowledge or experts. In practice, $\mathcal{F}_S$ can be incomplete and noisy. We design FairRF that is able to re-weight each $f_i \in \mathcal{F}_S$, so that it has the potential of remaining effective, as shown in experiments.

4 PRELIMINARY THEORETICAL ANALYSIS

In this paper, we adopt Pearson correlation coefficient to measure the correlation between two variables, defined as below:

Definition 1 (Pearson Correlation Coefficient). Pearson correlation coefficient measures the linear correlation between two random variables X and Y as:

$$\rho_{X,Y} = \frac{\mathbb{E}[(X - \mu_X) \cdot (Y - \mu_Y)]}{\sigma_X \cdot \sigma_Y}, \qquad (1)$$

where μ_X is the mean and σ_X is the standard deviation of X.

Next, we will show a theorem on the propagation property of Pearson correlation coefficient, which justifies our motivation of using $\mathcal{F}_S$ to regularize model predictions in the case of absent S. Below, we first present a rule depicting the relation of three included angles in space, which is the basis of our proof.

LEMMA 1. *Given a unit sphere centered at origin O, A, B and C are three points on the surface of the sphere. Assume that the angle $AOB = \theta_1$ and the angle $BOC = \theta_2$, then the cosine value of angle AOC is within: $[cos(\theta_1 + \theta_2), cos(\theta_1 - \theta_2)]$.*

PROOF. From Spherical law of cosines [12], we can know that:

$$cos\theta_3 = cos\theta_1 cos\theta_2 + sin\theta_1 sin\theta_2 cosB', \qquad (2)$$

where B' corresponds to the angle opposites B in spherical triangle ABC. As all angles are in the scale $[0, \pi]$, we can directly induce:

$$cos\theta_3 \geq cos\theta_1 cos\theta_2 - sin\theta_1 sin\theta_2 = cos(\theta_1 + \theta_2)$$
$$cos\theta_3 \leq cos\theta_1 cos\theta_2 + sin\theta_1 sin\theta_2 = cos(\theta_1 - \theta_2), \qquad (3)$$

which completes the proof. □

Next, we will show the relationship between Pearson correlation coefficient and cosine similarity of two variables.

LEMMA 2. *Given two random variables X, Y, Pearson correlation coefficient between them can be calculated as the cosine distance between $\mathbf{x}'$ and $\mathbf{y}'$, where $\mathbf{x}'$ is an infinite-length vector constructed by sampling z-score value of X, i.e., $x_i' = \frac{X_i - \mu_X}{\sigma_X}$ and X_i is the i-th sample. Similarly, $y_i' = \frac{y_i - \mu_Y}{\sigma_Y}$.*

PROOF. This can be easily proven by re-writing the form of Pearson correlation coefficient as:

$$\rho_{X,Y} = \frac{\mathbb{E}[(X - \mu_X) \cdot (Y - \mu_y)]}{\sigma_X \cdot \sigma_Y} = \lim_{n \to \infty} \sum_{i=1}^{n} \frac{(X_i - \mu_X) \cdot (Y_i - \mu_y)}{\sigma_X \cdot \sigma_Y}$$
$$= \lim_{n \to \infty} \sum_{i=1}^{n} x_i' \cdot y_i' = cos(\mathbf{x}', \mathbf{y}'), \qquad (4)$$

which completes the proof. □

With these preparations, we can now turn to our main theorem:

THEOREM 1. *Given three random variables $\{X, Y, Z\}$, with correlation coefficient $\rho_{X,Y} = cos\alpha$ and $\rho_{Y,Z} = cos\beta$, $\alpha, \beta \in [0, \pi]$, then $\rho_{X,Z}$ is within $[cos(\alpha + \beta), cos(\alpha - \beta)]$.*

PROOF. The proof can be developed via the following steps:

(1) Cosine similarity between $\mathbf{x}'$ and $\mathbf{y}'$ shows the cosine value of included angle between them. Hence, based on Lemma 2, we can learn that the cosine of angle between $\mathbf{x}'$ and $\mathbf{y}'$ is $cos(\alpha)$ and that of angle between $\mathbf{y}'$ and $\mathbf{z}'$ is $cos(\beta)$ from the given correlation coefficients.
(2) $\mathbf{x}', \mathbf{y}', \mathbf{z}'$ can be taken as line OA, OB, OC in Lemma 1 respectively. Hence, utilizing Lemma 1, we could induce that the cosine value of angle γ between $\mathbf{x}$ and $\mathbf{z}$ should fall within the scale $[cos(\alpha + \beta), cos(\alpha - \beta)]$.
(3) Finally, based on Lemma2, we can map the cosine value of angle γ back into correlation coefficient between X and Z.

After these steps, we can obtain that $\rho_{X,Z} = cos(\mathbf{x}', \mathbf{z}') \in [cos(\alpha + \beta), cos(\alpha - \beta)]$ and finish the proof. □
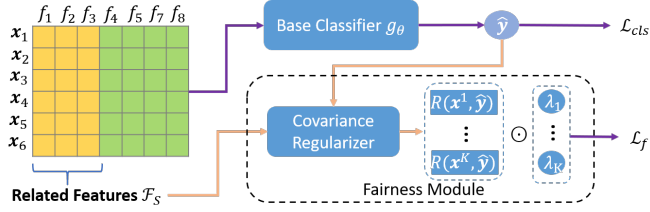
Figure 1: An illustration of the proposed framework FairRF. In Fairness Constraint block, λ_i controls the importance of regularization on i-th feature of $\mathcal{F}_S$. λ is dynamically updated, reducing prior domain knowledge required.

Basing on theorem 1, we can show how the constraint of correlation scale is propagated from $\mathcal{F}_S$ to S in Theorem 2, which theoretically proves our idea.

THEOREM 2. *Let f and S represent an input feature and sensitive attribute, respectively. Let $\hat{y}$ denotes the variable of model's prediction. Assume that f is highly correlated with S, i.e., $\rho_{f,S}$ is larger than a positive constant $\cos\alpha$. If the model is trained to make $\rho_{f,\hat{y}}$ near 0, i.e, within $[\cos(\frac{1}{2}\pi + \delta), \cos(\frac{1}{2}\pi - \delta)]$, where δ is close to 0, then $\rho_{S,\hat{y}}$ would be within $[\cos(\frac{1}{2}\pi + \delta + \alpha), \cos(\frac{1}{2}\pi - \delta - \alpha)]$.*

Theorem 2 can be easily proved based on Theorem 1. From it, we can see that when $\cos\alpha \approx 1$ and $\delta \approx 0$, $\rho_{S,\hat{y}}$ would also approximate 0. In this way, the prediction would be insensitive towards S, achieving fairness w.r.t sensitive attribute S.

We can extend Theorem 2 to the case of utilizing multiple related features simultaneously. For a set of related features $\mathcal{F}_S = \{f_1, f_2, ..., f_K\}$, assume their correlation coefficient with S in the form of $\{\cos\alpha_1, \cos\alpha_2, ..., \cos\alpha_K\}$, and with $\hat{Y}$ in the range of $[\cos(\frac{1}{2}\pi + \delta), \cos(\frac{1}{2}\pi - \delta)]$. Then $\rho_{S,\hat{Y}}$ would fall upon the intersections of their resulting value space, which can be written as:

$$\rho_{S,\hat{Y}} \in [\cos(\frac{1}{2}\pi + \delta + \alpha_{min}), \cos(\frac{1}{2}\pi - \delta - \alpha_{min})]. \quad (5)$$

where α_{min} is the smallest value in $\{\alpha_1, \alpha_2, ..., \alpha_K\}$. Note that this range is usually not tight, and high divergence within $\mathcal{F}_S$ would often restrict the range of $\rho_{S,\hat{Y}}$ more.

5 METHODOLOGY

In this section, we present the details of the proposed framework FairRF. The basic idea is using the regularization on correlated features $\mathcal{F}_S$ as the surrogate fairness objective. With the motivation theoretically justified in Sec 4, an illustration of FairRF is shown in Figure 1. It is composed of three parts: (i) a base classifier $g_\theta(\cdot)$ which predicts its label $\hat{y}_i$ given data sample $\mathbf{x}_i$; (ii) a covariance regularizer which constrains correlation between $\mathcal{F}_S$ and $\hat{y}$ to achieve fairness; and (iii) an importance learning module which adjusts importance score λ_j of each related feature $f_j \in \mathcal{F}_S$. Next, we introduce each component in detail.

5.1 Base Classifier

The proposed FairRF is flexible to use various classifiers as backbone such as neural networks, logistic regression and SVM. Without loss of generality, we use $g_\theta(\cdot)$ to denote the base classifier, where θ is the set of parameters of the base classifier. Following existing

work on fairness [21], we consider binary classification. We leave the extension to multi-class classification as future work. For a data sample $\mathbf{x}_i$, the predicted probability of $\mathbf{x}_i$ having label 1 is

$$\hat{y}_i = g_\theta(\mathbf{x}_i) \quad (6)$$

Then the binary cross entropy loss for training the classifier $g_\theta(\cdot)$ can be written as

$$\min_\theta \mathcal{L}_{cls} = \sum_{i=1}^{n} -y_i \log \hat{y}_i - (1 - y_i) \log(1 - \hat{y}_i) \quad (7)$$

where $y_i \in \{0, 1\}$ is the label of $\mathbf{x}_i$.

Generally, the well trained model is good at classification. However, as shown in previous studies [3, 39], the obtained model could make unfair predictions because spurious correlation may exist in the training data between sensitive attributes and labels due to societal bias. Though various efforts have been taken to mitigate the bias [9, 15, 35], most of them require knowing the sensitive attributes. With the sensitive attributes unknown, to learn fair models, we propose to regularize the predictions using the related features $\mathcal{F}_S$ that are highly correlated with S, which will be introduced next.

5.2 Exploring Related Features for Fairness

If the sensitive attribute s_i of each data sample $\mathbf{x}_i$ is known, we can adopt s_i to achieve fairness of the classification model by making the prediction independent of the sensitive attributes [9, 35]. Let $\mathbf{s} \in \mathbb{R}^{n \times 1}$ be the sensitive attribute vector with the i-th element of $\mathbf{s}$, i.e., s_i, as the sensitive attribute of $\mathbf{x}_i$. Similarly, let $\hat{\mathbf{y}} \in \mathbb{R}^{n \times 1}$ be the predictions with the i-th element being the prediction for $\mathbf{x}_i$. Following the design in [7, 35], the pursuit of non-dependence between prediction $\hat{y}$ and sensitive attribute $\mathbf{s}$ can be achieved through minimizing the correlation score between them, which can be mathematically written as:

$$\min_\theta \mathcal{R}(\mathbf{s}, \hat{\mathbf{y}}) = \left| \sum_{i=1}^{n} (s_i - \mu_s)(\hat{y}_i - \mu_{\hat{y}}) \right| \quad (8)$$

where μ_s and $\mu_{\hat{y}}$ are the mean of $\mathbf{s}$ and $\hat{\mathbf{y}}$, respectively. Note that we set constraints directly on the correlation score instead of correlation coefficient, but it can be seen from Eq.1 that it only differs from correlation coefficient by a constant multiplier $\sigma_\mathbf{s} \cdot \sigma_{\hat{y}}$. Constraining the scale of this regularization term, $\mathbf{s}$ and $\hat{y}$ would be encouraged to have no statistical correlation with each other.

However, as sensitive attribute $\mathbf{s}$ is unavailable in our problem, directly adopting the above regularization is impossible. Fortunately, from Theorem 2, we can see that if we have a set of non-sensitive features $\mathcal{F}_s$, with each feature $f_j \in \mathcal{F}_s$, i.e., $\mathbf{x}^j$, having high correlation with $\mathbf{s}$, then reducing the correlation between $\mathbf{x}^j$ with $\hat{\mathbf{y}}$ can indirectly reduce the correlation between $\mathbf{s}$ and $\hat{\mathbf{y}}$, which helps to achieve fairness, even though $\mathbf{s}$ is unknown. Hence, in FairRF, we apply correlation regularization on each feature $f_j \in \mathcal{F}_S$, in the purpose of making trained model fair towards S. Without loss of generality, let the set of features in $\mathcal{F}_S$ be $\{f_1, ..., f_K\}$, where $1 \leq K < m$. The regularization term is written as

$$\min_\theta \mathcal{R}_{related} = \sum_{j=1}^{K} \lambda_j \cdot \mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}}), \quad (9)$$

where λ_j is the weight for regularizing correlation coefficient between $\mathbf{x}^j$ and $\hat{\mathbf{y}}$. $\mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}})$ is given as

$$\mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}}) = \Big| \sum_{i=1}^{n} (X_{ij} - \mu_{x^j})(\hat{y}_i - \mu_{\hat{y}}) \Big| \tag{10}$$

where μ_{x^j} is the mean of $\mathbf{x}^j$.

Generally, if the correlation between $\mathbf{x}^j$ and $\mathbf{s}$ is large, we would prefer large λ_j to enforce $\mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}})$ to be close to 0, which can better reduce the correlation between $\mathbf{s}$ and $\hat{\mathbf{y}}$, resulting in a more fair classifier. If the correlation between $\mathbf{x}^j$ and $\mathbf{s}$ is not that large, a small λ_j is preferred because under such case, making $\mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}})$ close to 0 doesn't help much in making $\mathbf{s}$ and $\hat{\mathbf{y}}$ independent, but may instead introduce large noise in label prediction. Domain knowledge would be helpful in setting λ_j.

5.3 Learning Importance of Related Features

One limitation of this approach is the requirement of pre-defined $\boldsymbol{\lambda}$. This information provides prior knowledge and is important for the success of the proposed proxy regularization. However, in real-world applications, it is difficult to get accurate values, and $\mathcal{F}_S$ could be inaccurate. In addition, λ_j *is also important in balancing the contribution of f_j in model prediction and fairness.* Larger λ_j will result in the independence between $\mathbf{x}^j$ and $\hat{\mathbf{y}}$, making f_j contributes little in model prediction. Hence, in this section, we propose to learn $\boldsymbol{\lambda}$, allowing the model to automatically adjust its value.

Specifically, before learning, each related weight λ_j is initialized to a pre-defined value λ_j^0, which serves as an inaccurate estimation of its importance. Then, during training, the value of $\boldsymbol{\lambda}$ will be optimized along with model parameters iteratively. As no other information is available, we update $\boldsymbol{\lambda}$ by minimizing the total regularization loss, based on the intuition that an ideal surrogate correlation regularization should be achieved without causing significant performance drop. We limit the range of $\boldsymbol{\lambda}$ as $[0, 1]$, and the full optimization objective function can be written as follows:

$$\min_{\boldsymbol{\theta}, \boldsymbol{\lambda}} \; \mathcal{L}_{cls} + \eta \cdot \sum_{j=1}^{K} \lambda_j \cdot \mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}}) \quad \text{s.t.} \quad \lambda_j \geq 0, \forall f_j \in \mathcal{F}_S; \; \sum_{j=1}^{K} \lambda_j = 1 \tag{11}$$

where η sets the weights of regularization term, and $\boldsymbol{\theta}$ is the set of parameters of the classifier.

Eq.(11) can lead to a trivial solution, i.e., to minimize the cost function, it tends to set λ_j corresponding to the smallest $\mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}})$ to 1 and others to 0. To alleviate this issue, we add $\|\boldsymbol{\lambda}\|_2^2$ to penalize λ_j being close to 1. Thus, The final objective function of FairRF is

$$\min_{\boldsymbol{\theta}, \boldsymbol{\lambda}} \; \mathcal{L}_{cls} + \eta \cdot \sum_{j=1}^{K} \lambda_j \cdot \mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}}) + \beta \|\boldsymbol{\lambda}\|_2^2$$
$$\text{s.t.} \quad \lambda_j \geq 0, \forall f_j \in \mathcal{F}_S; \quad \sum_{j=1}^{K} \lambda_j = 1 \tag{12}$$

where β is used to control the contribution of $\|\boldsymbol{\lambda}\|_2^2$.

6 OPTIMIZATION ALGORITHM

The objective function in Eq.(12) is constrained optimization, which is difficult to be optimized directly. We take the alternating direction optimization [14] strategy to update θ and $\boldsymbol{\lambda}$ iteratively. The basic idea is to update one variable with the other one fixed at each step, which can ease the optimization process. Next, we give the details.

Update θ. To optimize θ, we fix $\boldsymbol{\lambda}$ and remove terms that are irrelevant to θ, which arrives at

$$\min_{\theta} \; \mathcal{L}_{cls} + \eta \sum_{j=1}^{K} \lambda_j \cdot \mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}}) \tag{13}$$

This is a non-constrained cost function, and we can directly apply gradient descent to learn θ.

UPDATE $\boldsymbol{\lambda}$. Then, given θ at the current step, $\boldsymbol{\lambda}$ can be obtained through solving the following optimization problem:

$$\boldsymbol{\lambda} = \arg\min_{\boldsymbol{\lambda}} \sum_{j=1}^{K} \lambda_j \cdot \mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}}) + \beta \|\boldsymbol{\lambda}\|_2^2,$$
$$\text{s.t.} \; -\lambda_j \leq 0, \forall f_j \in \mathcal{F}_S; \quad \sum_{j=1}^{K} \lambda_j - 1 = 0 \tag{14}$$

It is a convex primal problem, and strong duality holds as it follows *Slater's condition*. For simplicity of notation, we use $\mathcal{R}_j$ to represent $\mathcal{R}(\mathbf{x}^j, \hat{\mathbf{y}})$. Then, we can solve this problem using Karush-Kuhn-Tucker(KKT) [25] conditions as:

$$\begin{cases} \mathcal{R}_j + 2\beta \cdot \lambda_j - u_j + v = 0, \; \forall j; \; \textit{(stationary)} \\ u_j \cdot \lambda_j = 0, \; \forall j; \; \textit{(complementary slackness)} \\ \lambda_j \geq 0 \quad \forall j; \quad \sum_{j=1}^{K} \lambda_j = 1; \textit{(primal feasibility)} \\ u_j \geq 0 \quad \forall j. \end{cases} \tag{15}$$

In the above equation, $\boldsymbol{u}$ and v are Lagrange multipliers. From the stationary condition, we can get:

$$\lambda_j = \frac{u_j - v - \mathcal{R}_j}{2 \cdot \beta}, \quad j = 1, \ldots, K \tag{16}$$

Eliminating $\boldsymbol{u}$ using complementary slackness, we have:

$$\begin{cases} \lambda_j = 0, & \text{if } u_j = v + \mathcal{R}_j \geq 0; \\ \lambda_j = \frac{-v - \mathcal{R}_j}{2 \cdot \beta}, & \text{if } u_j = 0; \\ \lambda_j \geq 0 \quad \forall j; \quad \sum_j^K \lambda_j = 1 \end{cases} \tag{17}$$

From this condition, we know that $\lambda_j = \max\{0, \frac{-v - \mathcal{R}_j}{2 \cdot \beta}\}$. Since $\sum_{j=1}^{K} \lambda_j = 1$, v can be computed via solving the following equation:

$$\sum_{j=1}^{K} \max\{0, -v - \mathcal{R}_j\} = 2\beta. \tag{18}$$

Solving the above equation can be done as follows: we first rank $\mathcal{R}_j$ in descending order as $\mathcal{R}'_j$, i.e., $\mathcal{R}'_{j-1} \geq \mathcal{R}'_j$. Assume that v is within $[-\mathcal{R}'_{l-1}, -\mathcal{R}'_l]$, then the above equation is reduced to

$$\sum_{j=l}^{K} -v - \mathcal{R}'_j = 2 \cdot \beta \tag{19}$$

Then, we have

$$v = -\frac{2 \cdot \beta + \sum_{j=l}^{K} \mathcal{R}'_j}{K - l + 1} \tag{20}$$

If $v = -\frac{2 \cdot \beta + \sum_{j=l}^{K} \mathcal{R}'_j}{K - l + 1} \in [-\mathcal{R}'_{l-1}, -\mathcal{R}'_l]$, it is a valid solution; otherwise, it is invalid. We do this for every interval and find v. With v learned, we can calculate $\boldsymbol{\lambda}$ as:

$$\lambda_j = \max\{0, \frac{-v - \mathcal{R}_j}{2 \cdot \beta}\} \tag{21}$$

Training Algorithm. With the updating rules above, the full pipeline of the training algorithm for FairRF can be summarized in Algorithm 1 in the supplementary material.

7 EXPERIMENT

In this section, we conduct experiments to evaluate the effectiveness of the proposed FairRF in terms of both fairness and classification performance when sensitive attributes are unavailable. In particular, we aim to answer the following research questions:

- **RQ1** Can the proposed FairRF achieve fairness without sensitive attributes while maintain high accuracy?
- **RQ2** How would FairRF perform when the provided $\mathcal{F}_S$ contains misidentified related features or is incomplete?
- **RQ3** How would different choices of hyper-parameters influence the performance of FairRF?

7.1 Datasets

We conduct experiments on three publicly available benchmark datasets, including Adult [1], COMPAS [17] and LSAC [31].

- **ADULT**[1]: It contains $45,221$ records of personal yearly income, with binary label indicating if the yearly salary is over or under $\$50K$. Gender is considered as sensitive attribute. and we select age, relation and marital status as $\mathcal{F}_s$.
- **COMPAS**[2]: This dataset assesses the possibility of recidivism within a certain future, containing $11,750$ criminal records collected in US.The race of each defendant is the sensitive attribute. In constructing $\mathcal{F}_s$, score, decile text and sex are selected.
- **LSAC**[3]: It contains $65,307$ admissions data from 25 law schools in US over the 2005, 2006, and 2007 admission cycles. Labels indicate whether each candidate successfully pass the bar exam or not, and their gender information is considered as sensitive. For this dataset, we use race, year and residence as $\mathcal{F}_s$.

We make the train:eval:test splits as $5 : 2 : 3$. *Note that for all three datasets, features in $\mathcal{F}_s$ are selected following existing analysis or prior domain knowledge.* For example, in COMPAS, biases towards race have been found to exist in score and decile text [17]. The correlation between race and gender is also from reports by U.S. Bureau of Justice Statistics(BJS). Since race is the sensitive attribute of the dataset, we include score, decile text and gender in $\mathcal{F}_S$.

7.2 Experimental Settings

7.2.1 Baselines. To evaluate the effectiveness of FairRF, we first compare it with the vanilla model and sensitive-attribute-aware model, which can be treated as the lower and upper bound of our model's performance:

- **Vanilla model**: It directly uses the base classifier without any regularization terms. It is used to show the performance without fairness-assuring algorithm taken.
- **ConstrainS**: In this baseline, we assume that the sensitive attribute of each data sample is known. We add the correlation regularization between sensitive attribute vector $\mathbf{s}$ and model output $\hat{\mathbf{y}}$, i.e., $\mathcal{R}(\mathbf{s}, \hat{\mathbf{y}})$. It sets a reference point for the performance of the proposed framework. Note that for all the other baselines and our model, $\mathbf{s}$ is unknown.

We also include following representative approaches in fair learning without sensitive attributes as baselines:

Table 1: Comparison of different approaches on ADULT.

Methods	ACC	Δ_{EO}	Δ_{DP}
Vanilla	0.856 ± 0.001	0.046 ± 0.006	0.089 ± 0.005
ConstrainS	0.845 ± 0.002	0.040 ± 0.004	0.058 ± 0.003
ARL	0.861 ± 0.003	0.034 ± 0.012	0.141 ± 0.008
KSMOTE	0.560 ± 0.002	0.141 ± 0.031	0.120 ± 0.022
RemoveR	0.801 ± 0.010	0.124 ± 0.004	0.071 ± 0.002
FairRF	0.832 ± 0.001	$\mathbf{0.025} \pm 0.009$	$\mathbf{0.066} \pm 0.004$

Table 2: Comparison of different approaches on COMPAS

Methods	ACC	Δ_{EO}	Δ_{DP}
Vanilla	0.681 ± 0.004	0.242 ± 0.021	0.171 ± 0.015
ConstrainS	0.674 ± 0.002	0.154 ± 0.032	0.122 ± 0.031
ARL	0.672 ± 0.023	0.197 ± 0.042	0.286 ± 0.033
KSMOTE	0.601 ± 0.021	0.203 ± 0.042	0.151 ± 0.023
RemoveR	0.595 ± 0.024	0.205 ± 0.049	0.185 ± 0.024
FairRF	0.661 ± 0.009	$\mathbf{0.166} \pm 0.022$	$\mathbf{0.143} \pm 0.021$

Table 3: Comparison of different approaches on LSAC.

Methods	ACC	Δ_{EO}	Δ_{DP}
Vanilla	0.805 ± 0.001	0.042 ± 0.007	0.016 ± 0.004
ConstrainS	0.801 ± 0.001	0.014 ± 0.007	0.004 ± 0.002
ARL	0.811 ± 0.005	0.029 ± 0.029	0.022 ± 0.013
KSMOTE	0.722 ± 0.012	0.028 ± 0.062	0.012 ± 0.041
RemoveR	0.763 ± 0.002	0.037 ± 0.024	0.015 ± 0.006
FairRF	0.796 ± 0.002	$\mathbf{0.023} \pm 0.008$	$\mathbf{0.007} \pm 0.004$

- **KSMOTE** [33]: It performs clustering to obtain pseudo groups, and use them as substitute. The model is regularized to be fair with respect to those pseudo groups.
- **RemoveR**: This method directly removes all candidate related features, i.e., $\mathcal{F}_S$. We design this baseline in order to validate the benefits of our proposed method in regularizing related features.
- **ARL** [21] It follows Rawlsian principle of Max-Min welfare for distributive justice. It optimizes model's performance through re-weighting regions detected by an adversarial model.

Note that the fairness formulation of ARL is different from the group fairness we focus on. ARL [21] is inefficient in obtaining demographic fairness by design, which is also verified by our experiments. Although not working on the same fairness definition, we still include it as one baseline for completeness of the experiment.

7.2.2 Configurations. For KSMOTE, we directly use the code provided by [33]. For all other approaches, we implement a multi-layer perceptron (MLP) network with three layers as the backbone classifier. The two hidden dimensions are 64 and 32. Adam optimizer is adopted to train the model, with initial learning rate as 0.001.

7.2.3 Evaluation Metrics. To measure the fairness, following existing work on fair models [29, 33], we adopt two widely used evaluation metrics, i.e., equal opportunity and demographic parity, which are defined as follows:

Equal Opportunity [26] Equal opportunity requires that the probability of positive instances with arbitrary protected attributes i, j being assigned to a positive outcome are equal:

$$\mathbb{E}(\hat{y} \mid S = i, y = 1) = \mathbb{E}(\hat{y} \mid S = j, y = 1), \qquad (22)$$

where $\hat{y}$ is the output of model g_θ, representing the probability of being predicted as positive. In the experiments, we report difference in equal opportunity(Δ_{EO}):

$$\Delta_{EO} = |\mathbb{E}(\hat{y} \mid S = i, y = 1) - \mathbb{E}(\hat{y} \mid S = j, y = 1)| \qquad (23)$$

Demographic Parity [26] Demographic parity requires the behavior of prediction model to be fair on different sensitive groups. Concretely, it requires that the positive rate across sensitive attributes are equal:

$$\mathbb{E}(\hat{y} \mid S = i) = \mathbb{E}(\hat{y} \mid S = j), \forall i, j \qquad (24)$$

Similarly, in the experiment, we report the difference in demographic parity(Δ_{DP}):

$$\Delta_{DP} = |\mathbb{E}(\hat{y} \mid S = i) - \mathbb{E}(\hat{y} \mid S = j)| \qquad (25)$$

Equal opportunity and demographic parity measure the fairness from different perspectives. Equal opportunity requires similar performance across protected groups, while demographic parity is more focused on fair demographics. The smaller Δ_{EO} and Δ_{DP} are, the more fair a model is. Furthermore, to measure the classification performance, **accuracy** (ACC) is also reported.

7.3 Classification Performance Comparison

To answer **RQ1**, we fix the base classifier as MLP and conduct classification on all three datasets. For all the baselines, the hyperparameters are tuned via grid search on the validation dataset. In particular, for FairRF, β is set to 0.5 on ADULT, 0.8 on COMPAS, and 1.0 on LSAC. η is set as 0.15 for COMPAS and 0.3 for other two datasets. More details on the hyperparameters sensitivity will be discussed in Sec 7.5. Each experiment is conducted 5 times and the average performance in terms of accuracy, Δ_{EO} and Δ_{DP} with standard deviation are reported in Table 1, Table 2 and Table 3. From the tables, we make the following observations:

- Constraining related features can help the model to perform fairer on sensitive groups. For example, compared with vanilla approach in which no fair-learning techniques are applied, FairRF shows a clear improvement w.r.t Equal Opportunity and Demographic Parity across all three datasets;
- FairRF improves the fairness without causing significant performance drop, and works stably. No pre-computed clusters are required, and it does not involve training an adversarial model, hence FairRF can get results with less deviation compared to ARL and KSMOTE;
- Compared with baselines without sensitive attribute, FairRF is effective for both two fairness metrics; while other approaches such as ARL is able to improve on "equal opportunity", but the performance would drop w.r.t "demographic parity". This is because FairRF is able to learn λ_j to balance the fairness and accuracy.

7.4 Impact of the Quality of $\mathcal{F}_S$ on FairRF

In this section, we conduct experiment to investigate the impact of the quality of $\mathcal{F}_S$ on the performance of FairRF to answer **RQ2**. In particular, we consider the following variants of FairRF:

- **Random**: We randomly select five sets of $\mathcal{F}_S$ with the same number of attributes as FairRF. Average results are reported. We use it to show the influence of prior knowledge.

Table 4: Comparison of different strategies in selecting related features on ADULT.

Methods	ACC	Δ_{EO}	Δ_{DP}
Vanilla	0.856 ± 0.001	0.046 ± 0.006	0.089 ± 0.005
Random	0.830 ± 0.001	0.041 ± 0.012	0.057 ± 0.007
Top-1	0.830 ± 0.002	0.029 ± 0.008	0.067 ± 0.002
ConstrainAll	0.835 ± 0.001	0.035 ± 0.005	0.068 ± 0.003
Noisy	0.834 ± 0.002	0.030 ± 0.011	0.068 ± 0.006
Fix-λ	0.822 ± 0.002	0.065 ± 0.007	0.057 ± 0.004
FairRF	0.832 ± 0.001	$\mathbf{0.025} \pm 0.009$	0.066 ± 0.004

Table 5: Comparison of different strategies in selecting related features on COMPAS.

Methods	ACC	Δ_{EO}	Δ_{DP}
Vanilla	0.681 ± 0.004	0.242 ± 0.021	0.171 ± 0.015
Random	0.637 ± 0.006	0.226 ± 0.028	0.161 ± 0.016
Top-1	0.648 ± 0.007	0.183 ± 0.016	0.164 ± 0.013
ConstrainAll	0.651 ± 0.004	0.235 ± 0.012	0.168 ± 0.008
Noisy	0.653 ± 0.006	0.219 ± 0.023	0.154 ± 0.019
Fix-λ	0.631 ± 0.011	0.256 ± 0.025	0.159 ± 0.018
FairRF	0.661 ± 0.009	$\mathbf{0.166} \pm 0.022$	$\mathbf{0.143} \pm 0.021$

- **Fix-λ**: The same λ_i is adopted for all related features, and its value is not automatically updated during training. Selected related features are exactly the same as those chosen in FairRF.
- **Top-1**: It uses only the most-effective related features. We test all candidates and select the one that achieves highest performance when used as related feature, and report its performance.
- **ConstrainAll**: It includes all features in $\mathcal{F}_S$, i.e., all features are treated as related features. This is used to show if noisy features are included or no prior knowledge about related feature is given, FairRF can still work. We also learn λ for this variant.
- **Noisy**: Its contains features randomly sampled from both $\mathcal{F}_S$ and non-related attributes. In implementation, we randomly replace one attribute in $\mathcal{F}_S$ with non-related ones.

For all these baselines, hyper-parameters are found via grid search, and experiments are conducted for 5 times randomly. From Table 4 and 5, we can make following observations:

- FairRF can still bring improvements when $\mathcal{F}_S$ is inaccurate. The variant Noisy is shown to be effective across ADULT and COMPAS datasets.
- In the extreme case that no prior knowledge is available, FairRF still has potentials on fairness metrics compared with vanilla model, as shown by Random and ConstrainAll. It again shows that FairRF can cope with little domain knowledge scenario.
- FairRF benefits from automatically learning the importance of each given related attribute. Compared with Fix-λ, FairRF shows a much stronger fairness in terms of equal opportunity, and achieves better accuracy at the same time.
- FairRF shows a moderate improvement compared with Top-1. However, Top-1 requires careful selection of the most effective related feature, while FairRF can achieve better performance with less prior domain knowledge;

Due to space limitation, we only report the results on ADULT and COMPAS, but similar observations can be made on LSAC.
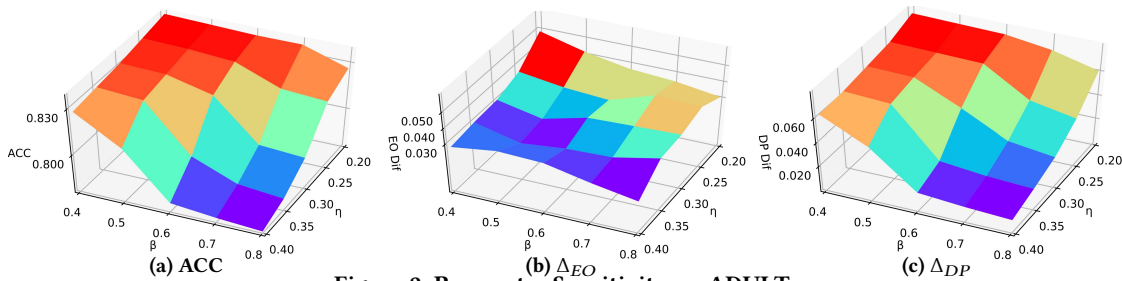
(a) ACC   (b) Δ_{EO} (c) Δ_{DP}

Figure 2: Parameter Sensitivity on ADULT

Table 6: Examples of learned λ on a set of selected related attributes. $\rho_{\cdot,Y}$ represents its correlation with class label, and $\rho_{\cdot,S}$ is the correlation with sensitive attributes S.

ADULT				COMPAS			
Attr	$\rho_{\cdot,Y}$	$\rho_{\cdot,S}$	λ	Attr	$\rho_{\cdot,Y}$	$\rho_{\cdot,S}$	λ
Age	0.09	0.05	0.51	Sex	0.11	0.07	0.27
Workclass	0.11	0.14	0.49	Score	0.31	0.27	0.00
Relation	0.41	0.58	0.00	Decile	0.25	0.24	0.21
Education	0.18	0.06	0.00	Duration	0.02	0.30	0.52

7.5 Parameter Sensitivity Analysis

In this subsection, we analyze the sensitivity of FairRF on hyper-parameters η and β. η controls the importance of coefficient regularization term, and β can adjust the distribution of learned λ. We vary η as $\{0.2, 0.25, 0.3, 0.35, 0.4\}$ and β as $\{0.4, 0.5, 0.6, 0.7, 0.8\}$. Other settings are the same as FairRF. This experiment is performed on ADULT, with results shown in Figure 2. From the figure, we can observe that: (i) Larger η will achieve fairer predictions, but may also cause severe drop in accuracy when it is larger than some thresholds; (ii) Generally, smaller β requires larger η to achieve fairness. Small β allows learned λ to be sparse. As a result, a large portion of coefficient regularization term could be enforced on less-discriminative attributes that are less-related at the same time; and (iii) β encourages learned λ to be uniform, resulting a faster drop in accuracy when η goes large. These observations could help to find suitable hyper-parameter choices in other applications.

7.6 Case Study on λ

In this subsection, we conduct case studies to analyze the behavior of FairRF in learning λ, i.e., the weights of related attributes. Specifically, we calculate the ground-truth correlation between the sensitive attribute S and others are computed, and a set of attributes with varying range of correlation coefficient magnitudes are selected as $\mathcal{F}_S$. η and β are set using grid search to make sure that fairness is obtained without significant drop in accuracy. We report the distribution of learned λ. Results on ADULT and COMPAS are shown in Table 6. From the result, we can observe

- FairRF tends to assign higher weight to features that have high correlation with S but small correlation with Y. For example, the correlation of "Duration" with label is 0.02 and with S is 0.30, FairRF assigns 0.52 to the feature. This is because such features have little effect on model accuracy but introduce a lot of bias. Assigning a large weight can help achieve fairness with marginal affects on performance;

- On the contrary, when a feature f_j has high correlation with Y, FairRF tends to assign smaller number to λ_j even if the correlation of the feature with S is large. For example, FairRF assigns 0 to "Relation". This is because when a feature has high correlation with label, it is important for model prediction. A large weight on fairness regularizer will significantly reduce the accuracy.

These observations further demonstrate that by learning λ, FairRF can balance the accuracy and fairness.

7.7 Flexibility of FairRF to Various Backbones

In the above experiment, we fix the base classifier as MLP. In this section, we investigate if FairRF can also benefit various classifiers to achieve fairness while maintaining high accuracy when the sensitive attributes are unknown. Specifically, we also adopt two other widely-used classifiers as the base classifiers of FairRF, i.e., Linear Regression (LR) and Support Vector Machine (SVM). *The details of experimental setting and results are given in Supplementary Material.* For both models, we find that FairRF only scarifies a little bit of accuracy while significantly improves the fairness. For example, by adding FairRF to LR , Δ_{EO} drops by 58.5% while the accuracy only drops by 2%.

8 CONCLUSION

In this paper, we study a novel and challenging problem of exploring related features for learning fair and accurate classifiers without knowing the sensitive attribute of each data sample. We propose a new framework FairRF which utilizes the related features as pseudo sensitive attribute to regularize the model prediction. Our theoretical analysis shows that if the related features are highly correlated with the sensitive attribute, by minimizing the correlation between the related features and model's prediction, we can learn a fair classifier with respect to the sensitive attribute. Since we lack the prior knowledge of the importance of each related feature, we design a mechanism for the model to automatically learn the importance weight of each feature to trade-off their contribution on classification accuracy and fairness. Experiments on real-world datasets show that the proposed approach is able to achieve more fair performance compared to existing approaches while maintain high classification accuracy when no sensitive attributes are known.

9 ACKNOWLEDGEMENT

REFERENCES

[1] Arthur Asuncion and David Newman. 2007. UCI machine learning repository.
[2] Mihalj Bakator and Dragica Radosav. 2018. Deep learning and medical diagnosis: A review of literature. *Multimodal Technologies and Interaction* 2, 3 (2018), 47.
[3] Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. 2017. Data decisions and theoretical implications when adversarially learning fair representations. *arXiv preprint arXiv:1707.00075* (2017).
[4] David D Celentano, Martha S Linet, and Walter F Stewart. 1990. Gender differences in the experience of headache. *Social science & medicine* 30, 12 (1990), 1289–1295.
[5] Amanda Coston, Karthikeyan Natesan Ramamurthy, Dennis Wei, Kush R Varshney, Skyler Speakman, Zairah Mustahsan, and Supriyo Chakraborty. 2019. Fair transfer learning with missing protected attributes. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. 91–98.
[6] Elliot Creager, David Madras, Jörn-Henrik Jacobsen, Marissa A Weis, Kevin Swersky, Toniann Pitassi, and Richard Zemel. 2019. Flexibly fair representation learning by disentanglement. *arXiv preprint arXiv:1906.02589* (2019).
[7] Enyan Dai and Suhang Wang. 2021. Say No to the Discrimination: Learning Fair Graph Neural Networks with Limited Sensitive Attribute Information. *WSDM* (2021).
[8] Xolani Dastile, Turgay Celik, and Moshe Potsane. 2020. Statistical and machine learning models in credit scoring: A systematic literature survey. *Applied Soft Computing* 91 (2020), 106263.
[9] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *ITCS*. 214–226.
[10] Harrison Edwards and Amos Storkey. 2015. Censoring representations with an adversary. *arXiv preprint arXiv:1511.05897* (2015).
[11] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *SIGKDD*. 259–268.
[12] Walter Gellert, M Hellwich, H Kästner, and H Küstner. 2012. *The VNR concise encyclopedia of mathematics*. Springer Science & Business Media.
[13] Milena A Gianfrancesco, Suzanne Tamang, Jinoos Yazdany, and Gabriela Schmajuk. 2018. Potential biases in machine learning algorithms using electronic health record data. *JAMA internal medicine* 178, 11 (2018), 1544–1547.
[14] Tom Goldstein, Brendan O'Donoghue, Simon Setzer, and Richard Baraniuk. 2014. Fast alternating direction optimization methods. *SIAM Journal on Imaging Sciences* 7, 3 (2014), 1588–1623.
[15] Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. In *NeurIPS*. 3315–3323.
[16] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. 2018. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*. PMLR, 1929–1938.
[17] Surya Mattu Julia Angwin, Jeff Larson and Lauren Kirchner. 2016. Machine bias: There's software used across the country to predict future criminals and it's biased against blacks. *ProPublica* (2016).
[18] Faisal Kamiran and Toon Calders. 2009. Classifying without discriminating. In *ICCC*. IEEE, 1–6.
[19] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *KAIS* 33, 1 (2012), 1–33.
[20] Jian Kang, Jingrui He, Ross Maciejewski, and Hanghang Tong. 2020. InFoRM: Individual Fairness on Graph Mining. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 379–389.
[21] Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed H Chi. 2020. Fairness without demographics through adversarially reweighted learning. *arXiv preprint arXiv:2006.13114* (2020).
[22] Preethi Lahoti, Krishna P Gummadi, and Gerhard Weikum. 2019. Operationalizing individual fairness with pairwise fair representations. *arXiv preprint arXiv:1907.01439* (2019).
[23] Francesco Locatello, Gabriele Abbati, Thomas Rainforth, Stefan Bauer, Bernhard Schölkopf, and Olivier Bachem. 2019. On the fairness of disentangled representations. In *NeurIPS*. 14584–14597.
[24] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. 2015. The variational fair autoencoder. *arXiv preprint arXiv:1511.00830* (2015).
[25] Olvi L Mangasarian. 1994. *Nonlinear programming*. SIAM.
[26] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2019. A survey on bias and fairness in machine learning. *arXiv preprint arXiv:1908.09635* (2019).
[27] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. 2017. On fairness and calibration. In *NeurIPS*. 5680–5689.
[28] Prasanna Sattigeri, Samuel C Hoffman, Vijil Chenthamarakshan, and Kush R Varshney. 2019. Fairness GAN: Generating datasets with fairness properties using a generative adversarial network. *IBM Journal of Research and Development* 63, 4/5 (2019), 3–1.
[29] Sahil Verma and Julia Rubin. 2018. Fairness definitions explained. In *2018 ieee/acm international workshop on software fairness (fairware)*. IEEE, 1–7.
[30] Matt Vogel and Lauren C Porter. 2016. Toward a demographic understanding of incarceration disparities: Race, ethnicity, and age structure. *Journal of quantitative criminology* 32, 4 (2016), 515–530.
[31] Linda F Wightman. 1998. LSAC National Longitudinal Bar Passage Study. LSAC Research Report Series. (1998).
[32] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. 2018. Fairgan: Fairness-aware generative adversarial networks. In *Big Data*. IEEE, 570–575.
[33] Shen Yan, Hsien-te Kao, and Emilio Ferrara. 2020. Fair Class Balancing: Enhancing Model Fairness without Observing Sensitive Attributes. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 1715–1724.
[34] Adrienne Yapo and Joseph Weiss. 2018. Ethical implications of bias in machine learning. In *Proceedings of the 51st Hawaii International Conference on System Sciences*.
[35] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. 2015. Fairness constraints: Mechanisms for fair classification. *arXiv preprint arXiv:1507.05259* (2015).
[36] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *ICML*. 325–333.
[37] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. 2018. Mitigating unwanted biases with adversarial learning. In *AIES*. 335–340.
[38] Chongjie Zhang and Julie A Shah. 2014. Fairness in multi-agent sequential decision-making. (2014).
[39] Lu Zhang, Yongkai Wu, and Xintao Wu. 2017. Achieving non-discrimination in data release. In *SIGKDD*. 1335–1344.

Table 7: Effectiveness of FairRF with various base classifiers on ADULT dataset.

Method	ACC	Δ_{EO}	Δ_{DP}
LR	0.832 ± 0.004	0.053 ± 0.003	0.125 ± 0.005
FairRF(LR)	0.815 ± 0.008	0.022 ± 0.009	0.072 ± 0.014
SVM	0.775 ± 0.013	0.083 ± 0.008	0.117 ± 0.013
FairRF(SVM)	0.775 ± 0.015	0.031 ± 0.017	0.056 ± 0.024
MLP	0.856 ± 0.001	0.046 ± 0.006	0.089 ± 0.005
FairRF(MLP)	0.832 ± 0.001	0.025 ± 0.009	0.066 ± 0.004

Table 8: Evaluate effectiveness of FairRF on different base classifiers on COMPAS.

Method	ACC	Δ_{EO}	Δ_{DP}
LR	0.678 ± 0.002	0.215 ± 0.033	0.198 ± 0.026
FairRF(LR)	0.671 ± 0.001	0.201 ± 0.011	0.146 ± 0.008
SVM	0.664 ± 0.013	0.241 ± 0.006	0.151 ± 0.008
FairRF(SVM)	0.661 ± 0.008	0.162 ± 0.008	0.134 ± 0.013
MLP	0.681 ± 0.004	0.242 ± 0.021	0.171 ± 0.015
FairRF(MLP)	0.661 ± 0.009	0.166 ± 0.022	0.143 ± 0.021

Algorithm 1 Training Algorithm of FairRF

Input: $\mathbf{X} \in \mathbb{R}^{n \times m}, \mathbf{y} \in \mathbb{R}^{n \times 1}, \mathcal{F}_S$
Output: classifier parameters $\boldsymbol{\theta}$
1: Randomly initialize $\boldsymbol{\theta}$; Initialize all entries in $\boldsymbol{\lambda}$ as $\frac{1}{K}$;
2: **for** batch in (X, Y) **do**
3:     Update $\boldsymbol{\theta}$ based on classification loss of current batch;
4: **end for**
5: **while** Not Converged **do**
6:     **for** step in MODEL_TRAIN_STEP **do**
7:         Update $\boldsymbol{\theta}$ based on Equation 11;
8:     **end for**
9:     **if** Require learning weight **then**
10:        Obtain $\mathcal{R}_j$ for each related feature $\mathbf{x}^j$;
11:        Calculate v and $\boldsymbol{\lambda}$ based on Eq.(18) to Eq.(21);
12:     **end if**
13: **end while**
14: **return** Trained classifier $\boldsymbol{\theta}$.

A TRAINING ALGORITHM

With the updating rules introduced in Section 6, the full pipeline of the training algorithm for FairRF can be summarized in Algorithm 1. Before adding the regularization, we first pre-train the model to converge at a good start point in line 3 in order to prevent correlation constraint from providing noisy signals. Then, from line 5 to line 13, we fine-tune the model to be fair w.r.t related features. If not refining related weights, $\boldsymbol{\lambda}$ will stay fixed. Otherwise, it will be updated iteratively with parameter θ, as shown from line 9 to 12.

B IMPLEMENTATION ON DIFFERENT BASE MODEL

In the experiments of main paper, we fix the base classifier as MLP. In this section, we present the incorporation of FairRF into various machine learning models to achieve fairness while maintain high accuracy when the sensitive attributes are unknown. Specifically, in addition to MLP, we also adopt two other widely-used classifiers as the base classifiers of FairRF, i.e., Linear Regression (LR) and Support Vector Machine (SVM). We implement both of them in a gradient-based manner. so that parameters can be optimized alternatively with the regularization term on related features, as in Algorithm 1.

Concretely, we tune the hyperparameters on the validation set. η is fixed to 0.4, and β is set to 0.4 and 0.6 for LR and SVM, respectively. Adam optimizer is adopted to train them, with the initial learning rate as 0.001. Each experiment is conducted for 5 times, and average results on ADULT and COMPAS are reported in Table 7 and 8, respectively.

From the table, we observe that

- Compared with the base classifiers, integrating FairRF makes the accuracy drops a little bit, which is in consistent with observations in other work on fair models [33] as the fairness regularizer usually drops the accuracy. However, the accuracy decrease is marginal. For example, for LR, the accuracy only drops by 2%, which shows that we are still able to maintain high accuracy;
- Though the accuracy drops a little bit, the fairness in terms of Δ_{EO} and Δ_{DP} on three models improves significantly, even though the sensitive attributes are not observed. For instance, for LR, with the FairRF framework, Δ_{EO} drops by 58.5% while the accuracy only drops by 2%. In other words, we scarify a little bit of accuracy while significantly improves the fairness.

These observations show that FairRF can benefit various machine learning models to achieve fairness while maintaining high accuracy when the sensitive attributes are unknown