Fair Classification Under Strict Unawareness

Haoyu Wang * Hengtong Zhang† Yaqing Wang‡ Jing Gao §

Abstract

Despite the wide adoption of classification algorithms in many fields, their predictions may hurt the benefit of some people due to the ubiquitous bias over sensitive features, such as race, gender and age. To avoid biased predictions, extensive research efforts have been devoted to training fair classification models under a variety of fairness definitions. However, we observe that recent fair classification methods may still make their predictions based on sensitive features implicitly under existing fairness definitions because the non-sensitive features these models rely on still have the capabilities of predicting the values of sensitive features. To overcome this limitation, we introduce a new fairness definition named "Fairness Through Strict Unawareness" for deep neural networks (DNN), which emphasizes the unpredictability of the sensitive features by the fair classification model. Accordingly, we proposed a bi-level optimization-based approach that prevents the encoded features of a DNN classifier to rely on any sensitive information (explicitly or implicitly). We show that the proposed framework satisfies the fairness under strict unawareness condition while still maintains its prediction accuracy. Experimental results on two benchmark datasets also support this claim. Results show that the proposed framework can significantly degrade the models' ability of inferring sensitive features without sacrificing its general predictive capability.

Keywords: Fairness, Classification

1 Introduction

Deep neural networks have demonstrated their success in classification tasks, but when applying these models to applications that involve people, such as criminal justice, talent recruiting and revenue forecast, biased decisions may be made as the model predictions are usually based on sensitive features, such as race, gender, and age. The machine learning community has recognized this issue and devoted their efforts towards developing fairness-aware classification algorithms. To evaluate the fairness degree of different models, various fairness definitions have been proposed, including *Fairness Through Unawareness* [13, 17, 21], *Statistical Parity* [9, 17] and *Equal Opportunity* [14]. Classification algorithms that satisfy these definitions have been proposed towards the goal of making fair predictions.

Despite these efforts, existing fair classification approaches still cannot ensure that the classifier output does not rely on any sensitive features. Roughly, existing work falls into the following two categories: 1) The sensitive features are removed from the training set, and this type of algorithms satisfy *Fairness Through Unawareness*. For example, in [11], sensitive features like "gender" are removed during the training process. Even though they are not explicitly involved in training the model, one may still infer the gender of a user via other features such as a user's height, weight and shoe size, and thus the gender information is still encoded implicitly in the model. 2) Other work incorporated regularizers of conditional probability or covariance [25, 12, 24] to enforce *Statistical Parity* and *Equal Opportunity* of the learnt deep learning models. However, there is no guarantee that the classification model does not encode any sensitive information. It is still very likely that one may predict sensitive features via its final layer of the deep neural networks.

Motivated by these limitations, we proposed a new fairness definition, referred to as *Fairness under Strict Unawareness*, which emphasizes on the model's unawareness of any sensitive information either explicitly or implicitly. For a deep neural network model, the classification output is dependent on the last hidden layer of the network, which can be regarded as the encoded representation of original data for classification. The deep neural network models that satisfy this definition should not involve any sensitive information in the encoded representations that are used to make output predictions. In other words, based on the encoded representations, the values of sensitive features should be unpredictable under this definition.

Based on this fairness definition, we designed a new classification framework that guarantees the inaccessibility of sensitive information without sacrificing the prediction accuracy of original classification task. The framework involves two classification tasks: 1) the original classification task, which is implemented by a deep neural network and denoted as D, and 2) the task of

*University at Buffalo, hwang79@buffalo.edu
†University at Buffalo, hengtong@buffalo.edu
‡Purdue University, wang5075@purdue.edu
§Purdue University, jinggao@purdue.edu

predicting sensitive information based on encoded representations (denoted as F). The objective is to derive a classifier F with high accuracy, and at the same time its encoded representation leads to a sensitive information predictor G with low accuracy. We formulate this objective as a bi-level optimization task. The upper-level solves for D and F that optimize the two aforementioned objectives simultaneously, and the lower-level derives the optimal classifier F based on the encoded representations of D. We proposed an effective solution to this bi-level optimization problem and theoretically analyzed the guaranteed performance of D on original classification task and the limited capability of F on sensitive information prediction.

To evaluate the effectiveness of the proposed approach, we conduct experiments on two benchmark datasets, i.e., Adult and German Credit datasets. Compared with state-of-the-art baselines, the proposed approach can effectively filter out sensitive information while maintain the performance of original classification task. Results show that the proposed approach is also able to achieve better fairness with respect to other fairness metrics, such as statistical parity and equal opportunity.

2 Related work

Fairness in machine learning has become a hot topic, and recently extensive efforts have been devoted to the development of fairness aware approaches. In this section, we review related work that are most relevant to our work, which fall into the following two categories, fair classification and fair representation.

As classification is one of the fundamental tasks in machine learning, how to integrate fairness into classification models has attracted much attention. Existing work can be grouped based on the adopted fairness definitions: 1) Most of the work [12, 15, 16, 22] aims to satisfy *Statistical Parity* when training the classification models. Specifically, [4] proposed three kinds of regularizers to enforce that instances with different sensitive feature values from the same class receive similar predictions. In [25], the authors aimed to optimize the classification accuracy under fairness constraint to comply with disparate treatment criterion. [1] proposed general frameworks for fair classification which are applicable to arbitrary Lipschitz continuous losses. 2) Recent work [14] proposed new fairness measures, including equality of opportunity and equalized odds fairness, based on which an optimal equalized odds threshold predictor was derived to meet these criteria in classification task. The work in the aforementioned two categories focus on single-model classification, and recently people also investigated fairness in multi-task classification. In [28], a popular rank-based non-parametric independence test is designed to achieve fair multi-task classification. Despite these efforts, existing fairness-aware classification methods still cannot guarantee that sensitive information is excluded from the classification process. Besides, these algorithms are often based on the trade off between accuracy and fairness, and to satisfy the fairness condition, they have to sacrifice the accuracy of the classification models. In this paper, we proposed new fairness definition and classification framework to address these limitations.

The objective of fair representation work is to transform original data into a new representation (usually low-dimensional) such that the representation is independent of sensitive features. Most of the work focuses on the derivation of fair representation from unlabeled data [18, 7, 27, 10, 20]. Some work deals with labeled data and incorporates the penalty on incorrect classification based on the representation into the objective [23, 26, 19, 2, 6, 5]. However, as the main objective is to ensure fair representation, the performance of classification based on this representation is typically degraded. Our work differs from this category of studies because we aim to derive an accurate classifier whose model does not encode sensitive information. In addition, the fair representation work requires the knowledge of actual sensitive feature value for each instance. In contrast, our proposed approach only needs some high-level statistical information of sensitive feature, such as the gender ratio. Therefore, our work can be applied to privacy-preserving scenarios when sensitive features are withheld from the datasets.

3 Methodology

In this section, we first review preliminaries for deep neural networks and existing fairness definitions. We then introduce the proposed fairness definition and the classification framework, followed by theoretical analysis of the framework.

3.1 Notations and Fairness Definitions

Throughout the paper, we use $\boldsymbol{X}$ with d rows and n columns to denote the input features without sensitive information, where d is the feature dimensionality and n is the number of training instances. We use $\boldsymbol{y} \in \{-1, 1\}^n$ to represent class labels, and use $\boldsymbol{z} \in \{0, 1\}^n$ to represent a binary sensitive feature.

Based on these notations, we present existing definitions of fairness for classification as follows:

DEFINITION 3.1. *(Statistical Parity). A binary predictor $\hat{y}$ satisfies statistical parity if $P(\hat{y}|z = 0) = P(\hat{y}|z = 1)$. It means that the likelihood of the predictor outcome*

should be the same regardless of the sensitive feature value.

DEFINITION 3.2. *(Equal Opportunity). A binary predictor $\hat{y}$ satisfies equal opportunity if $P(\hat{y}=1|z=0,y=1)=P(\hat{y}=1|z=1,y=1)$. It means that the probability of a person belonging to a positive class having a positive outcome should be the same regardless of the sensitive feature value.*

DEFINITION 3.3. *(Fairness Through Unawareness). An algorithm is fair if none of the sensitive features $\boldsymbol{z}$ is explicitly used in the model training and prediction process.*

3.2 Preliminaries Before delving into the details of the proposed fairness framework, we first introduce the deep neural network we adopt for classification, which is denoted by $D(\cdot)$. The architecture of the neural network is defined as:

$$\boldsymbol{H}_{i+1} = \tanh(\boldsymbol{W}_i^T \boldsymbol{H}_i), i = 0, 1, \cdots, n-1 \tag{3.1}$$

$$\boldsymbol{o} = \text{sigmoid}(\boldsymbol{w}^T \boldsymbol{H}_n), \tag{3.2}$$

where $\boldsymbol{H}_i$ is the output of the i-th layer with $\boldsymbol{H}_0 = \boldsymbol{X}$. Among them, $\boldsymbol{H}_n$ is the last hidden layer. As this layer is directly connected to the output layer, it is regarded as the encoded representation of the original data for classification. $\{\boldsymbol{W}_i\}$ and $\boldsymbol{w}$ are the weights of the intermediate and the final layer, respectively. $\boldsymbol{o}$ is the final output. We use binary Cross-Entropy loss to predict the label, without loss of generality:

$$\begin{aligned} &\mathcal{L}_c(\boldsymbol{X}, \boldsymbol{y}; \{\boldsymbol{W}_i\}, \boldsymbol{w}) \\ =& -\sum_{i=1}^{n}(y_i \log(o_i) + (1-y_i)\log(1-o_i)) \\ &+ \frac{\lambda}{2}(\sum_{i=0}^{n-1} ||\boldsymbol{W}_i||_F^2 + ||\boldsymbol{w}||_2^2) \end{aligned} \tag{3.3}$$

This classification model takes a training set $(\boldsymbol{X}_i, y_i)_{i=1}^n$ as input and finds parameters $\{\boldsymbol{W}_i\}$ and $\boldsymbol{w}$ that minimizes $\mathcal{L}_c$. Then the model outputs the predicted label y for an instance X based on model parameter. Note that the training and prediction of this classification model do not involve any sensitive feature z.

3.3 A DNN-based Fair Classification Framework Although existing definitions discussed in Section 3.1 strive to capture the fairness degree of classification models, they are unable to judge whether a model relies on sensitive features in the prediction even if sensitive features are not explicitly involved in the training process. Motivated by this important aspect of fairness, we propose a new fairness definition for classification as follows:

DEFINITION 3.4. *(Fairness Through Strict Unawareness). A DNN satisfies Fairness Through Unawareness if its derived encoded representations can not be used to predict sensitive features accurately. Formally, given the encoded representation of a DNN $\boldsymbol{H}_n$ and another trained classifier $F(\cdot)$ for predicting sensitive feature z, if $F(\boldsymbol{H}_n) = P(\boldsymbol{z})$, then the DNN satisfies Fairness Through Strict Unawareness.*

The essence of this definition is to guarantee that a classifier trained on the encoded representations of a DNN classifier can only make random prediction of sensitive feature value, i.e., its predicted probability always equals to the prior distribution of that feature.

Based on this definition, we propose an effective DNN classification framework that not only excludes the implicit usage of sensitive information but also maintains high accuracy on the original classification task. We introduce a bi-level optimization problem which combines the two objectives, i.e., classification and fairness.

Ideally, the encoded representations of a "perfect" fair DNN under the definition of *Fairness Through Strict Unawareness* do not encode any sensitive information and a predictor based on such representations can only lead to a random guessing of sensitive feature value $\boldsymbol{z}$[1]. In other words, for a predictor, the probability of predicting z (the sensitive feature) is independent of $\boldsymbol{H}_n$ (the encoded representations, which are usually the last hidden layer of DNN):

$$P(\boldsymbol{z}|\boldsymbol{H}_n) = P(\boldsymbol{z}), \tag{3.4}$$

In particular, when the sensitive feature is binary, the prior distribution of the sensitive feature can be captured as follows: $P(z_i = 1) = \frac{n_0}{n}$ and $P(z_i = 0) = 1 - \frac{n_0}{n}$, where n_0 is the number of instances whose sensitive feature value is 0. Ideally, the posterior distribution based on H_n should not deviate from the prior. Namely,

$$P(z_i = 1|(\boldsymbol{H}_n)_i) = \frac{n_0}{n}, P(z_i = 0|(\boldsymbol{H}_n)_i) = 1 - \frac{n_0}{n}, \tag{3.5}$$

where $(\boldsymbol{H}_n)_i$ is the i-th column of $\boldsymbol{H}_n$.

To achieve this goal, we proposed a bi-level optimization framework that involves a DNN classification model D for original task (i.e., the prediction of class label y)

[1] As mentioned, we merely discuss the case of a binary sensitive feature in this paper.

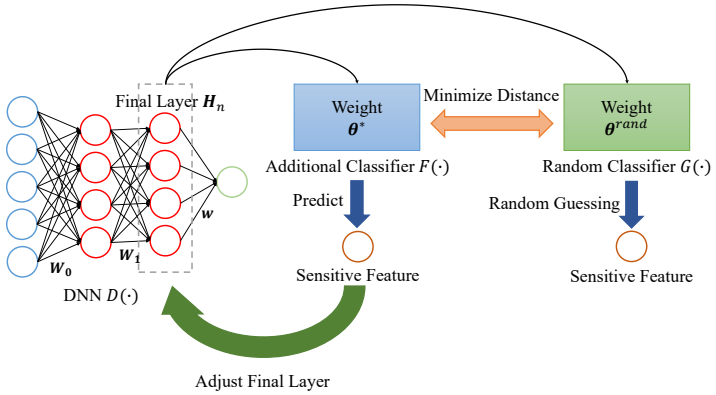


Figure 1: The framework of our proposed method. In the figure, the light blue rectangle represents the weight of the additional classifier to predict sensitive feature using final layer of DNN as input. The light green rectangle represents the weight of the random classifier to predict sensitive feature.

and a classifier F on top of DNN's encoded representations for sensitive feature prediction (i.e., the prediction of z). With this framework, we can ensure that the DNN model satisfies the Fairness under Strict Unawareness definition in Eqn. 3.5. We name the framework as **B**ilevel optimization-based **F**air **DNN** (BFDNN for short). The lower level of the BFDNN trains a classifier F to predict z with the encoded representations of D as the input. At the upper level, D is trained for the original classification task whose encoded representations serve as the input to F, at the same time F is enforced to be as close to a random classifier as possible. Intuitively, this optimization framework guarantees the good performance of the original classification task and also ensures that the encoded representations of the trained deep neural network model cannot be used to make meaningful predictions on sensitive features. Formally, the bi-level optimization framework can be formulated as:

$$\begin{aligned} &\min_{\boldsymbol{\theta}^*,\{\boldsymbol{W}_i\},\boldsymbol{w}} \mathcal{L}_c(\boldsymbol{X},\boldsymbol{y};\{\boldsymbol{W}_i\},\boldsymbol{w}) + \beta D(\boldsymbol{\theta}^*,\boldsymbol{\theta}^{rand}) \\ &\text{s.t.} \boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_g(\boldsymbol{H}_n;\boldsymbol{\theta}). \end{aligned} \tag{3.6}$$

In this objective function, $\mathcal{L}_c(\boldsymbol{X},\boldsymbol{y};\{\boldsymbol{W}_i\},\boldsymbol{w})$ is the classification loss of the DNN model on the original task. $D(\boldsymbol{\theta}^*,\boldsymbol{\theta}^{rand})$ measures the difference between $\boldsymbol{\theta}^*$ and $\boldsymbol{\theta}^{rand}$, and is set to be $||\boldsymbol{\theta}^* - \boldsymbol{\theta}^{rand}||_1$ in this paper. $\mathcal{L}_g$ is defined as:

$$\begin{aligned} \mathcal{L}_g = -\frac{1}{n}\sum_{i=1}^{n}(z_i \log(F((\boldsymbol{H}_n)_i;\boldsymbol{\theta})) \\ + (1-z_i)\log(1-F((\boldsymbol{H}_n)_i;\boldsymbol{\theta}))), \end{aligned} \tag{3.7}$$

and

$$\begin{aligned} \boldsymbol{\theta}^{rand} &= \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{rand} \\ &= \arg\min_{\boldsymbol{\theta}} -\frac{1}{n}\sum_{i=1}^{n}(\frac{n_0}{n}\log(G((\boldsymbol{H}_n)_i;\boldsymbol{\theta})) \\ &\quad + (1-\frac{n_0}{n})\log(1-G((\boldsymbol{H}_n)_i;\boldsymbol{\theta}))). \end{aligned} \tag{3.8}$$

Here, $\boldsymbol{\theta}^{rand}$ is the parameter of a random classifier $G(\cdot)$, which makes prediction only based on prior distribution $P(z)$. This parameter can be inferred by minimizing he Kullback-Leibler Divergence (D_{KL}) between $P(z)$ and $G(\boldsymbol{H}_n;\boldsymbol{\theta})$. Specifically, as a random classifier, its output $G(\boldsymbol{H}_n;\boldsymbol{\theta})$ should be close to the distribution $P(z)$. Therefore, we need to minimize the KL distance between $P(z)$ and $G(\boldsymbol{H}_n;\boldsymbol{\theta})$, which can be calculated as follows.

$$\begin{aligned} D_{KL}(P(z)||G(\boldsymbol{H}_n;\boldsymbol{\theta})) &= \sum_{i=1}^{n} P(z_i)\log\frac{P(z_i)}{G((\boldsymbol{H}_n)_i;\boldsymbol{\theta})} \\ &= -\sum_{i=1}^{n} P(z_i)\log G((\boldsymbol{H}_n)_i;\boldsymbol{\theta}) \\ &\quad + \underbrace{\sum_{i=1}^{n} P(z_i)\log P(z_i)}_{\text{constant}}. \end{aligned} \tag{3.9}$$

By removing the constant term, we derive the objective function (i.e., Eqn 3.8) to obtain the random classifier $G(\cdot)$. The parameter inferred by minimizing this objective function is $\boldsymbol{\theta}^{rand}$. Note that $\boldsymbol{\theta}^{rand}$ is pre-trained and fixed during the optimization process of the overall framework.

As one can see, $D(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{rand})$ measures the distance between $\boldsymbol{\theta}^*$ and $\boldsymbol{\theta}^{rand}$. Hence, the parameter $\boldsymbol{\theta}^*$ will be trained to be close to $\boldsymbol{\theta}^{rand}$ after the optimization. Thus, the last hidden layer $\boldsymbol{H}_n$ of the trained DNN will be adjusted to fit the random classifier. As a result, sensitive information cannot be predicted based on $\boldsymbol{H}_n$ of the DNN classifier $F(\cdot)$, and thus $F(\cdot)$ satisfies the Fairness Through Strict Unawareness definition (i.e., Definition-4).

Figure 1 shows the overall framework. As can be seen, the training of the DNN classifier $D(\cdot)$ will be influenced by the additional classifier $F(\cdot)$ that predicts on the sensitive feature. By enforcing $F(\cdot)$ to be close to a random classifier, we adjust $D(\cdot)$ to satisfy the Fairness Under Strict Unawareness condition. Though the choice of classifier $F(\cdot)$ could be flexible, in this paper, we choose logistic regression for $F(\boldsymbol{Q}; \boldsymbol{\theta})$ because it has the following advantages: (1) logistic regression is stable and robust, which helps to properly evaluate the accuracy of sensitive feature prediction, and (2) logistic regression's convex property enables subsequent analysis and derivation.

Eqn 3.6 is a standard bi-level optimization problem, which is NP hard in general. As we adopt logistic regression for $F(\cdot)$ (a convex model), the problem can be reduced to the following single-level constrained optimization problem according to the Karush-Kuhn-Tucker (KKT) conditions of the lower level problem [3]:

$$\min_{\boldsymbol{\theta}, \{\boldsymbol{W}_i\}, \boldsymbol{w}} \mathcal{L}_c(\boldsymbol{X}, \boldsymbol{y}; \{\boldsymbol{W}_i\}, \boldsymbol{w}) + \beta ||\boldsymbol{\theta} - \boldsymbol{\theta}^{rand}||_1$$
$$\text{s.t.} \partial_{\boldsymbol{\theta}} \mathcal{L}_g(\boldsymbol{H}_n; \boldsymbol{\theta}) = \boldsymbol{0} \tag{3.10}$$

To optimize the single-level constrained problem, we set the constraint as a regularizer in the loss function, which can be formulated as

$$\min_{\boldsymbol{\theta}, \{\boldsymbol{W}_i\}, \boldsymbol{w}} \mathcal{L}_c(\boldsymbol{X}, \boldsymbol{y}; \{\boldsymbol{W}_i\}, \boldsymbol{w}) + \beta ||\boldsymbol{\theta} - \boldsymbol{\theta}^{rand}||_1$$
$$+\gamma ||\partial_{\boldsymbol{\theta}} \mathcal{L}_g(\boldsymbol{H}_n; \boldsymbol{\theta})||_1 \tag{3.11}$$

This objective function can be optimized by any gradient-based optimizer directly.

3.3.1 Theoretical Considerations In this section, we demonstrate the theoretical guarantee of the proposed framework from the following two perspectives:

1. The output of classifier $F(\cdot)$ should be close to that of random guessing $G(\cdot)$ under the proposed framework. Specifically, we show that the distance between the corresponding parameters of the two classifiers (i.e., $\boldsymbol{\theta}^*$ and $\boldsymbol{\theta}^{rand}$) is bounded, and in turn, the distance between prediction of $F(\cdot)$ and random classifier $G(\cdot)$ can also be bounded.
2. The performance of classifier $D(\cdot)$ (i.e., the DNN classification model for original task) does not change much. Specifically, we show that the distance between the last hidden layer of the original DNN model and that of the proposed BFDNN framework is bounded.

Theorem 3.1 demonstrates the first perspective, i.e., the bound between the output of $F(\cdot)$ and the random classifier $G(\cdot)$ on the task of sensitive feature prediction. The proof to this theorem is provided in the supplementary.

THEOREM 3.1. *Suppose* $||(\boldsymbol{H}_n)_i||_1 \leq \epsilon_1$ *and* $||\boldsymbol{\theta}^* - \boldsymbol{\theta}^{rand}||_1 \leq \epsilon_2$. *Then the difference between the prediction output by classifier* $F(\cdot)$ *and random classifier* $G(\cdot)$ *is bounded by* $\frac{1}{4}\epsilon_1\epsilon_2$.

Proof. The predict score of $F(\cdot)$ is $\sigma((\boldsymbol{H}_n)_i^T \boldsymbol{\theta}^*)$, and the predict score of random model is $\sigma((\boldsymbol{H}_n)_i^T \boldsymbol{\theta}^{rand})$. Because $||\sigma^{'}(x)|| \leq \frac{1}{4}$, the sigmoid function is $\frac{1}{4}$-Lipschitz continuous: $||\sigma(x_1) - \sigma(x_2)||_1 \leq \frac{1}{4}||x_1 - x_2||_1$. Therefore, we have

$$\begin{aligned} &||\sigma((\boldsymbol{H}_n)_i^T \boldsymbol{\theta}^*) - \sigma((\boldsymbol{H}_n)_i^T \boldsymbol{\theta}^{rand})||_1 \\ \leq &\frac{1}{4}||(\boldsymbol{H}_n)_i||_1 ||\boldsymbol{\theta}^* - \boldsymbol{\theta}^{rand}||_1 \leq \frac{1}{4}\epsilon_1\epsilon_2 \end{aligned}$$

□

Then Lemma 3.1 and Theorem 3.2 below demonstrate the second perspective, i.e., the difference in H_n between the original DNN and the proposed BFDNN is bounded. To simplify the proof, we assume that the DNN only has one hidden layer and uses linear activation in the hidden layer. Both proofs are provided in the supplementary.

LEMMA 3.1. *Suppose* $||\boldsymbol{X}||_1 \leq \varepsilon_0$, $||\boldsymbol{W}_0^f||_1 \leq \varepsilon_1$, *and* $||\boldsymbol{\theta}||_1 \leq \varepsilon_2$. *Then* $||\partial_{\boldsymbol{W}_0^f} ||\partial_{\boldsymbol{\theta}} \mathcal{L}_g||_1||_1$ *is bound by* $\frac{4\varepsilon_0 + \varepsilon_0^2 \varepsilon_1 \varepsilon_2}{4n}$, *where* $\boldsymbol{W}_0^f$ *is the parameter of BFDNN.*

Proof. Because

$$\begin{aligned} &\partial_{\boldsymbol{W}_0^f} ||\partial_{\boldsymbol{\theta}} l_g||_1 = \frac{1}{n}(t_0 \text{sign}((t_1 - \boldsymbol{y})^T \boldsymbol{X}^T \boldsymbol{W}_0^f) \\ &+ \boldsymbol{X}(t_1 \odot (1 - t_1) \odot (\boldsymbol{X}^T (\boldsymbol{W}_0^f)^{k-1} \text{sign}(((\boldsymbol{W}_0^f)^{k-1})^T t_0)))\boldsymbol{\theta}^T) \end{aligned}$$

where $t_0 = \boldsymbol{X}(t_1 - \boldsymbol{y})$, $t_1 = \sigma(\boldsymbol{X}^T \boldsymbol{W}_0^f \boldsymbol{\theta})$. Thus, we have

$$\begin{aligned} ||\partial_{\boldsymbol{W}_0^f} ||\partial_{\boldsymbol{\theta}} l_g||_1||_1 &\leq \frac{1}{n}(||\boldsymbol{X}||_1 + \frac{1}{4}||\boldsymbol{X}||_1^2 ||\boldsymbol{W}||_1 ||\boldsymbol{\theta}||_1) \\ &\leq \frac{4\varepsilon_0 + \varepsilon_0^2 \varepsilon_1 \varepsilon_2}{4n} \end{aligned}$$

□

THEOREM 3.2. *In the condition of Lemma 3.1, after k times iterations, the gap between the original DNN $\mathcal{L}_c$'s representation $(\boldsymbol{H}_n^c)^k$ and model BFDNN $\mathcal{L}_f = \mathcal{L}_c + \beta||\boldsymbol{\theta} - \boldsymbol{\theta}^*||_1 + \gamma||\partial_{\boldsymbol{\theta}}\mathcal{L}_g(\boldsymbol{H}_n; \boldsymbol{\theta})||_1$'s representation $(\boldsymbol{H}_n^f)^k$ is bounded by $\frac{\varepsilon_0\eta\gamma}{L}(1-(1-L)^k)\frac{4\varepsilon_0+\varepsilon_0^2\varepsilon_1\varepsilon_2}{4n}$, where L is the Lipschitz coefficient of the model and $(\cdot)^k$ represents the parameter after k times iterations.*

Proof. Suppose we use a one hidden layer deep neural network. And to simplify the analysis, suppose we use linear layer as hidden layers and use sigmoid on the output layer. Consider the gap between $(\boldsymbol{W}_0^c)^k$ and $(\boldsymbol{W}_0^f)^k$

$$\begin{aligned}||(\boldsymbol{W}_0^c)^k - (\boldsymbol{W}_0^f)^k||_1 = ||(\boldsymbol{W}_0^c)^{k-1} - \eta\partial_{(\boldsymbol{W}_0^c)^{k-1}}\mathcal{L}_c) \\ - (\boldsymbol{W}_0^f)^{k-1} - \eta\partial_{(\boldsymbol{W}_0^f)^{k-1}}\mathcal{L}_f||_1\end{aligned}$$

Rewrite it and we have

$$\begin{aligned}||(\boldsymbol{W}_0^c)^k - (\boldsymbol{W}_0^f)^k||_1 &= ||((\boldsymbol{W}_0^c)^{k-1} - \eta\partial_{(\boldsymbol{W}_0^c)^{k-1}}\mathcal{L}_c) \\ &\quad - ((\boldsymbol{W}_0^f)^{k-1} - \eta\partial_{(\boldsymbol{W}_0^f)^{k-1}}\mathcal{L}_c) \\ &\quad - \eta\gamma\partial_{(\boldsymbol{W}_0^f)^{k-1}}\mathcal{L}_g||_1 \\ &\leq ||((\boldsymbol{W}_0^c)^{k-1} - \eta\partial_{(\boldsymbol{W}_0^c)^{k-1}}\mathcal{L}_c \\ &\quad - ((\boldsymbol{W}_0^f)^{k-1} - \eta\partial_{(\boldsymbol{W}_0^f)^{k-1}}\mathcal{L}_c)||_1 \\ &\quad + ||\eta\gamma\partial_{(\boldsymbol{W}_0^f)^{k-1}}||\partial_{\boldsymbol{\theta}}\mathcal{L}_g||_1||_1\end{aligned}$$

Because we use linear and sigmoid function as activation function, $\partial_{(\boldsymbol{W}_0^f)^{k-1}}\mathcal{L}_c$ is L-Lipschitz continuous with $L < 1$. Therefore, we have

$$\begin{aligned}||(\boldsymbol{W}_0^c)^k - (\boldsymbol{W}_0^f)^k||_1 &\leq (1-L)||(\boldsymbol{W}_0^c)^{k-1} - (\boldsymbol{W}_0^f)^{k-1}||_1 \\ &\quad + ||\eta\gamma\partial_{(\boldsymbol{W}_0^f)^{k-1}}||\partial_{\boldsymbol{\theta}}\mathcal{L}_g||_1||_1 \\ &\leq (1-L)||(\boldsymbol{W}_0^c)^{k-1} - (\boldsymbol{W}_0^f)^{k-1}||_1 \\ &\quad + \eta\gamma\frac{4\varepsilon_0 + \varepsilon_0^2\varepsilon_1\varepsilon_2}{4n}\end{aligned}$$

Let $\rho = \gamma\eta\frac{4\varepsilon_0+\varepsilon_0^2\varepsilon_1\varepsilon_2}{4n}$, and we have

$$\begin{aligned}&||(\boldsymbol{W}_0^c)^k - (\boldsymbol{W}_0^f)^k||_1 - \frac{\rho}{L} \\ \leq&(1-L)(||(\boldsymbol{W}_0^c)^{k-1} - (\boldsymbol{W}_0^f)^{k-1}||_1 - \frac{\rho}{L}) \\ \leq&(1-L)^k(||(\boldsymbol{W}_0^c)^0 - (\boldsymbol{W}_0^f)^0||_1 - \frac{\rho}{L})\end{aligned}$$

If we use the same initialization, then it can be simplified as:

$$||(\boldsymbol{W}_0^c)^k - (\boldsymbol{W}_0^f)^k||_1 \leq \frac{\rho}{L}(1-(1-L)^k)$$

Therefore, we have

$$||(\boldsymbol{H}_n^c)^k - (\boldsymbol{H}_n^f)^k||_1 \leq \frac{\rho}{L}(1-(1-L)^k)\varepsilon_0$$

□

4 Experiment

In this section, we evaluate the proposed BFDNN framework with the aim of answering the following questions:

- Q1 Does the proposed framework filter the sensitive information from the original deep neural network model?
- Q2 Does the proposed framework maintain the performance on the original classification task?
- Q3 Does the proposed framework improve fairness with respect to traditional definitions of fairness, such as Statistical Parity and Equal Opportunity?
- Q4 Is the proposed framework able to converge quickly?

We compare the proposed framework with the following baseline methods including state-of-the-art fair classification and fair representation methods. We also include a variant of the proposed BFDNN framework for ablation studies.

- DNN, a simple one hidden layer DNN for classification without any fairness contraints;
- IRAT [23], a state-of-the-art fair representation method, which realizes the independence between representation and sensitive features in a VAE framework;
- FLR [4], a fair classification approach which outputs similar predictions for instances in the same class with different sensitive feature values;
- MFC [25], another popular fairness classification approach which optimizes the classification accuracy under fairness constraint to comply with disparate treatment criterion;
- MFDNN, a variant of the proposed BFDNN framework. Its objective is to optimize the following multitask loss function: $\mathcal{L}_{Multi} = \alpha\mathcal{L}_c + (1-\alpha)\mathcal{L}_{rand}$, where $\mathcal{L}_c$ is the loss of original DNN classification and $\mathcal{L}_{rand}$ measures the KL-divergence between a classifier defined on the last hidden layer of DNN and a random classifier. MFDNN just trades off $\mathcal{L}_c$ and $\mathcal{L}_{rand}$, but BFDNN can get the optimal solution via its bi-level optimization framework.

4.1 Datasets and Experiment Settings We investigate the effectiveness of the proposed framework on two public benchmark datasets: Adult [8] and German Credit data [8], which are widely used in fairness research. The statistic information of the two datasets is summarized in the supplementary.

To evaluate the proposed framework and baselines, we adopt the following metrics:

Table 1: The performance of all the methods on original classification task shown by F1 and AUC measures (the higher the better), and the performance on sensitive feature prediction shown by F1(s) and AUC(s) (the lower the better). We use the bold and underline to denote the best and second best performance, respectively.

	Adult				German			
	F1 ↑	AUC ↑	F1(s) ↓	AUC(s) ↓	F1 ↑	AUC ↑	F1(s) ↓	AUC(s) ↓
DNN	0.6658	<u>0.9030</u>	0.6459	0.8572	<u>0.8189</u>	0.7284	<u>0.7960</u>	0.6524
IRAT	0.5514	0.8523	**0.4668**	**0.7492**	**0.8423**	**0.7597**	0.8316	0.6730
CFFR	0.6418	0.8864	0.6694	0.8766	0.8017	0.6812	0.8084	0.6456
MFC	0.6151	0.8748	0.6605	0.8651	0.7850	0.6025	0.8120	<u>0.6225</u>
MFDNN	**0.6699**	**0.9059**	0.5976	0.8283	0.8116	<u>0.7409</u>	0.8000	0.6344
BFDNN	<u>0.6679</u>	0.9025	<u>0.4933</u>	<u>0.7600</u>	0.8042	0.7367	**0.7304**	**0.6140**

- To evaluate the classification performance, we use F1 score and AUC. We evaluate the classification performance on both the original task and the sensitive feature prediction task. For the former, the higher F1 score and AUC the better, and for the latter, the lower scores indicate better fairness.
- The statistical parity score (SP) is defined as $\frac{\max(\frac{1}{n_0}\sum_{z_i=0}\hat{y}_i, \frac{1}{n_1}\sum_{z_i=1}\hat{y}_i)}{\min(\frac{1}{n_0}\sum_{z_i=0}\hat{y}_i, \frac{1}{n_1}\sum_{z_i=1}\hat{y}_i)}$, where $n_0 = \text{card}\{i|z_i = 0\}$, $n_1 = \text{card}\{i|z_i = 1\}$, and card defines the cardinality of a set.
- For Equal Opportunity, we define metrics conditioned on positive class and negative class respectively, named Positive Equal Opportunity Score (PEO) and Negative Equal Opportunity Score (NEO). Specifically, PEO is defined as $\frac{\max(\frac{1}{n_0}\sum_{z_i=0,y_i=1}\hat{y}_i, \frac{1}{n_1}\sum_{z_i=1,y_i=1}\hat{y}_i)}{\min(\frac{1}{n_0}\sum_{z_i=0,y_i=1}\hat{y}_i, \frac{1}{n_1}\sum_{z_i=1,y_i=1}\hat{y}_i)}$, where $n_0 = \text{card}\{i|z_i = 0, y_i = 1\}$, $n_1 = \text{card}\{i|z_i = 1, y_i = 1\}$, and NEO is defined as $\frac{\max(\frac{1}{n_0}\sum_{z_i=0,y_i=0}1-\hat{y}_i, \frac{1}{n_1}\sum_{z_i=1,y_i=0}1-\hat{y}_i)}{\min(\frac{1}{n_0}\sum_{z_i=0,y_i=0}1-\hat{y}_i, \frac{1}{n_1}\sum_{z_i=1,y_i=0}1-\hat{y}_i)}$, where $n_0 = \text{card}\{i|z_i = 0, y_i = 0\}$, $n_1 = \text{card}\{i|z_i = 1, y_i = 0\}$.

SP, PEO and NEO are in the range of $[1, +\infty)$. The lower, the better fairness.

4.1.1 Implementation Details We show the detail of data pre-processing and implementation infromation in the supplementary[2].

4.1.2 Sensitive Feature Prediction In this section, we will answer Q1. Table 1 shows BFDNN on the two datasets. Because this is an imbalanced classification problem, we use F1 score and AUC to evaluate its performance. According to Table 1, we have the following findings:

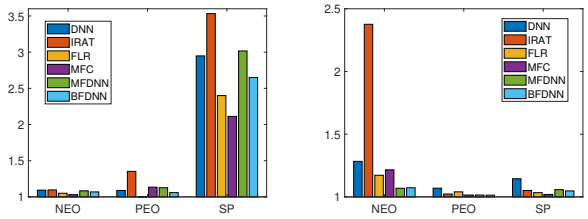
First, *The proposed BFDNN can filter sensitive information from DNN effectively.* According to Table 1, BFDNN can remarkably degrade the performance on sensitive feature prediction compared with DNN. BFDNN has 23.6% and 11.3% reduction with respect to F1 and AUC measures respectively on Adult dataset, and has 8.2% and 6.9% reduction with respect to F1 and AUC measures respectively on German dataset. Compared with MFDNN, BFDNN also has superior performance, particularly on Adult dataset. We also show the visualization of the final hidden layer of DNN and BFDNN in Figure 4. It is clear that the instances with different sensitive feature values are interwoven in Figure 4(b), which demonstrates the proposed BFDNN's capability of maintaining fairness.

Second, *the baseline methods including IRAT, CFFR, MFC and MFDNN cannot guarantee the filtering of sensitive information.* This can be observed from Table 1. In particular, CFFR and MFC cannot lower the prediction accuracy on sensitive feature prediction, which indicates that they still implicitly adopt sensitive information in their classification model. As for IRAT and MFDNN, their performance is not stable–each performs well on one of the two datasets (Adult or German) but not on the other. Especially, as a variant of the proposed approach, MFDNN cannot guarantee the exclusion of sensitive information from the model, and this confirms the necessity of the proposed bi-level optimization framework.

4.1.3 Original Classification Task In this section, we will answer Q2. F1 score and AUC in Table 1 demonstrate the performance of all the methods with respect to the prediction of the class label $\boldsymbol{y}$. We have the following findings:

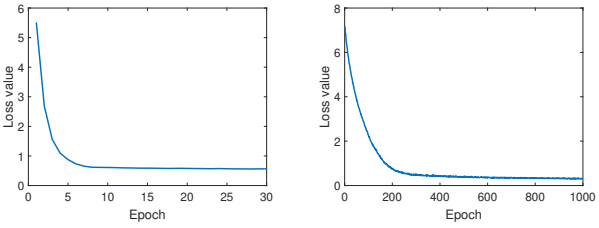
First, *the proposed BFDNN can still make accurate predictions on the original classification task.* As can be seen in Table1, BFDNN can achieve similar or even

[2]https://drive.google.com/file/d/1qTJAPz6nhynb3-QyduJ2pv4tWvf6um9i/view?usp=sharing



(a) Adult (b) German Credit

Figure 2: The performance of all methods with respect to fairness metrics on two datasets.



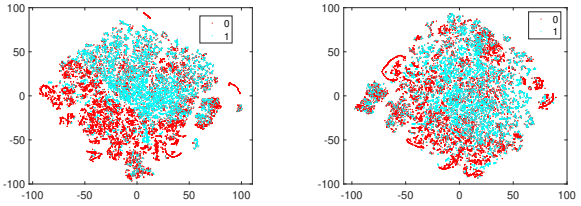
(a) Adult (b) German Credit

Figure 3: The convergence curve of BFDNN on two datasets.

higher F1 and AUC values compared with DNN, which shows that the proposed framework can preserve most of the discriminative information without relying on sensitive information. Thus, the proposed framework can effectively maintain original model's accuracy and exclude sensitive information.

Second, *IRAT, CFFR, and MFC degrade the classification performance significantly.* For example, the classification measures of IRAT on the original classification task drop significantly. On Adult dataset, the F1 score is only 0.5514 (17.2% reduction compared with DNN). As for CFFR, its AUC is only 0.6812 (6.5% reduction compared with DNN) on German Credit dataset. Similarly, the gap in the classification measures between MFC and original DNN is quite large. Such results show that even though some fairness-aware classification approaches improve fairness measures, but the improvement is at the cost of degrading original classification performance.

4.2 Existing Fairness Metrics In this section, we will answer Q3. Fig 2 shows the performance of all the methods with respect to existing fairness metrics SP, PEO and NEO on the two datasets. We can observe the following that the proposed framework can also improve upon traditional fairness metrics, including Statistical Parity and Equal Opportunity. This can be observed based on the superiority of BFDNN over DNN and MFDNN. According to Fig 2, on two datasets, BFDNN significantly outperforms DNN with respect to fairness metrics including NEO, PEO and SP. Although MFDNN



(a) DNN for classification (b) Proposed BFDNN

Figure 4: The visualization of final hidden layer of models for classification on Adult dataset via t-SNE. Sub-figure 4(a) is the visualization of original DNN and sub-figure 4(b) is the visualization of the proposed BFDNN.

improves fairness in some cases (e.g. NEO on Adult dataset, SP on two datasets), it still cannot achieve satisfactory fairness results based on the other metrics.

4.3 Convergence In this section, we will answer Q4. To test the convergence of BFDNN, we conduct the experiment on Adult and German Credit datasets by recording the values of the loss function at different epochs. Fig 3 shows the convergence of BFDNN on the two datasets. As can be seen, the proposed BFDNN converged quickly on both datasets.

5 Conclusions

Motivated by the limitations of existing fair classification algorithms, we proposed a new fairness definition that measures the capability of an algorithm in filtering out sensitive information from the classification model. We then proposed an effective bi-level optimization framework that not only satisfies the defined Fairness Under Strict Unawareness condition but also maintains the performance on the original classification task. This is achieved by optimizing original classification and enforcing sensitive feature predictor to be close to a random classifier. By this framework, sensitive information is excluded from the model but only highly discriminative and insensitive information is kept. Theoretically, we proved that gap between sensitive feature predictor and random classifier is bounded. In addition, the difference between the last hidden layer of original DNN and that of the proposed BFDNN is also bounded. Experimental results on Adult and German credit datasets demonstrate that the proposed framework significantly improves fairness in terms of the proposed definition as well as existing fairness metrics while maintains high classification accuracy on the original task.

Acknowledgement

This work is supported in part by the US National Science Foundation under grant NSF-IIS 1956017. Any

opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] A. Agarwal, M. Dudík, and Z. S. Wu, *Fair regression: Quantitative definitions and reduction-based algorithms*, arXiv preprint arXiv:1905.12843, (2019).

[2] A. Amini, A. P. Soleimany, W. Schwarting, S. N. Bhatia, and D. Rus, *Uncovering and mitigating algorithmic bias through learned latent structure*, in Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, 2019, pp. 289–295.

[3] J. F. Bard, *Practical bilevel optimization: algorithms and applications*, vol. 30, Springer Science & Business Media, 2013.

[4] R. Berk, H. Heidari, S. Jabbari, M. Joseph, M. Kearns, J. Morgenstern, S. Neel, and A. Roth, *A convex framework for fair regression*, arXiv preprint arXiv:1706.02409, (2017).

[5] A. Beutel, J. Chen, Z. Zhao, and E. H. Chi, *Data decisions and theoretical implications when adversarially learning fair representations*, arXiv preprint arXiv:1707.00075, (2017).

[6] A. J. Bose and W. Hamilton, *Compositional fairness constraints for graph embeddings*, arXiv preprint arXiv:1905.10674, (2019).

[7] E. Creager, D. Madras, J.-H. Jacobsen, M. A. Weis, K. Swersky, T. Pitassi, and R. Zemel, *Flexibly fair representation learning by disentanglement*, arXiv preprint arXiv:1906.02589, (2019).

[8] D. Dua and E. Karra Taniskidou, *Uci machine learning repository [http://archive. ics. uci. edu/ml]. irvine, ca: University of california*, School of Information and Computer Science, (2017).

[9] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, *Fairness through awareness*, in Proceedings of the 3rd innovations in theoretical computer science conference, 2012, pp. 214–226.

[10] H. Edwards and A. Storkey, *Censoring representations with an adversary*, arXiv preprint arXiv:1511.05897, (2015).

[11] P. Gajane and M. Pechenizkiy, *On formalizing fairness in prediction with machine learning*, arXiv preprint arXiv:1710.03184, (2017).

[12] N. Goel, M. Yaghini, and B. Faltings, *Non-discriminatory machine learning through convex fairness criteria*, in Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[13] N. Grgic-Hlaca, M. B. Zafar, K. P. Gummadi, and A. Weller, *The case for process fairness in learning: Feature selection for fair decision making*, in NIPS Symposium on Machine Learning and the Law, vol. 1, 2016, p. 2.

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

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

[16] E. Krasanakis, E. Spyromitros-Xioufis, S. Papadopoulos, and Y. Kompatsiaris, *Adaptive sensitive reweighting to mitigate bias in fairness-aware classification*, in Proceedings of the 2018 World Wide Web Conference, 2018, pp. 853–862.

[17] M. J. Kusner, J. Loftus, C. Russell, and R. Silva, *Counterfactual fairness*, in Advances in Neural Information Processing Systems, 2017, pp. 4066–4076.

[18] F. Locatello, G. Abbati, T. Rainforth, S. Bauer, B. Schölkopf, and O. Bachem, *On the fairness of disentangled representations*, in Advances in Neural Information Processing Systems, 2019, pp. 14584–14597.

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

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

[21] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, *A survey on bias and fairness in machine learning*, arXiv preprint arXiv:1908.09635, (2019).

[22] A. K. Menon and R. C. Williamson, *The cost of fairness in binary classification*, in Conference on Fairness, Accountability and Transparency, 2018, pp. 107–118.

[23] D. Moyer, S. Gao, R. Brekelmans, A. Galstyan, and G. Ver Steeg, *Invariant representations without adversarial training*, in Advances in Neural Information Processing Systems, 2018, pp. 9084–9093.

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

[25] M. B. Zafar, I. Valera, M. G. Rodriguez, and K. P. Gummadi, *Fairness constraints: Mechanisms for fair classification*, arXiv preprint arXiv:1507.05259, (2015).

[26] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, *Learning fair representations*, in International Conference on Machine Learning, 2013, pp. 325–333.

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

[28] C. Zhao and F. Chen, *Rank-based multi-task learning for fair regression*, in 2019 IEEE International Conference on Data Mining (ICDM), IEEE, 2019, pp. 916–925.