Knowledge and Information Systems (2023) 65:543–570
https://doi.org/10.1007/s10115-022-01738-w

REGULAR PAPER

Fairness in graph-based semi-supervised learning

Tao Zhang[1] · Tianqing Zhu[1] · Mengde Han[1] · Fengwen Chen[2] · Jing Li[2] · Wanlei Zhou[3] · Philip S Yu[4]

Received: 28 October 2021 / Revised: 23 July 2022 / Accepted: 31 July 2022 /
Published online: 1 October 2022

Abstract
Machine learning is widely deployed in society, unleashing its power in a wide range of applications owing to the advent of big data. One emerging problem faced by machine learning is the discrimination from data, and such discrimination is reflected in the eventual decisions made by the algorithms. Recent study has proved that increasing the size of training (labeled) data will promote the fairness criteria with model performance being maintained. In this work, we aim to explore a more general case where quantities of unlabeled data are provided, indeed leading to a new form of learning paradigm, namely fair semi-supervised learning. Taking the popularity of graph-based approaches in semi-supervised learning, we study this problem both on conventional label propagation method and graph neural networks, where various fairness criteria can be flexibly integrated. Our developed algorithms are proved to be non-trivial extensions to the existing supervised models with fairness constraints. Extensive

✉ Tianqing Zhu
Tianqing.Zhu@uts.edu.au

Tao Zhang
Tao.Zhang-3@student.uts.edu.au

Mengde Han
Mengde.Han@student.uts.edu.au

Fengwen Chen
Fengwen.Chen@student.uts.edu.aum

Jing Li
jing.li-20@student.uts.edu.au

Wanlei Zhou
wlzhou@cityu.edu.mo

Philip S Yu
psyu@uic.edu

1 Centre for Cyber Security and Privacy, School of Computer Science, University of Technology Sydney, Sydney, NSW, Australia

2 Centre for Artificial Intelligence, University of Technology Sydney, Sydney, NSW, Australia

3 Institute of Data Science, City University of Macau, Macau, China

4 Department of Computer Science, University of Illinois at Chicago, Chicago, Illinois, USA

experiments on real-world datasets exhibit that our methods achieve a better trade-off between classification accuracy and fairness than the compared baselines.

Keywords Fairness · Discrimination · Machine learning · Semi-supervised learning

1 Introduction

Machine learning algorithms, as useful decision-making tools, are widely used in the society. These algorithms are often assumed to be paragons of objectivity. However, many studies show that the decisions made by these models can be biased against certain groups of people. For example, Abid et al. observed that large-scale language models capture undesirable racial bias [1] and Vigdor et al. [2] reported gender bias in credit ranking of Apple card. These events prove that discrimination can arise from machine learning, and one of the most important discrimination sources is from data, including data collection (imbalanced training set) and data preparation (biased content in the training set) [3]. Given the widespread use of machine learning to support decisions over loan allocations, insurance coverage, and many other basic precursors to equity, fairness in machine learning has become a significantly important issue [4]. Thus, how to design big data enabled machine learning algorithms that treat all groups equally is critical.

In recent years, many fairness metrics have been proposed to define what is fairness in machine learning. Popular fairness metrics include statistical fairness [5, 6], individual fairness [7–10] and causal fairness [11, 12]. Meanwhile, many algorithms have been developed to address fairness issues for both supervised learning settings [6, 13, 14] and unsupervised settings [15–18]. Generally, these studies have focused on two key issues: how to formalize the concept of fairness in the context of machine learning tasks, and how to design efficient algorithms that strike a desirable trade-off between accuracy and fairness. What is lacking is the research that considers semi-supervised learning (SSL) scenarios.

In real-world machine learning tasks, a large amount of data used for training is necessary, and is often a combination of labeled and unlabeled data. Therefore, fair SSL is a vital area of development. Like the other learning settings, achieving a balance between accuracy and fairness is a key issue. According to [19], increasing the size of the training set can create a better trade-off. This finding sparked an idea over whether the trade-off might be improved via unlabeled data. Unlabeled data is abundant in era of big data and, if it could be used as training data, we may be able to make a better compromise between fairness and accuracy. To achieve this goal, two challenges are ahead of us: (1) how to achieve fair learning from both labeled and unlabeled data; and (2) how to give labels for unlabeled data to ensure that the learning is towards a fair direction.

To solve these challenges, we propose two approaches to improve the trade-off with unlabeled data in graph-based SSL which is one of the most prominent methods in SSL. Graph-based SSL first constructs a graph, where nodes represent all samples, and weighted edges reflect the similarity between a pair of nodes. Then the label information of unlabeled samples can be inferred from the graphs based on the manifold assumption. Graph-based SSL mainly includes two lines, graph-based regularization [20–22] and graph neural networks (GNNs) [20, 23], and thus we design two approaches to achieve fairness in these two lines.

Graph-based SSL shares an assumption that smoothness (e.g., the labels of adjacent nodes are likely to be the same) should present in the local and global graph structure [22]. Regularization methods are used to smooth the predictions or feature representations over local

neighborhoods. Our first approach, fair semi-supervised margin classifiers (FSMC), is formulated as an optimization problem, where the objective function includes a loss for both the classifier and label propagation, and fairness constraints over labeled and unlabeled data. Classification loss is to optimize the accuracy of training result; label propagation loss is to optimize the label predictions on unlabeled data; the fairness constraint is to lead optimization towards to a fairness direction. The optimization includes two steps. In the first step, fairness constraints enforce weights update towards a fair direction. This step can be solved by a convex problem and convex-concave programming when disparate impact and disparate mistreatment are used as fairness metrics respectively. In the second step, updated weights further direct labels assigned to unlabeled data in a fair direction by label propagation. Labels for unlabeled data can be calculated in a closed form. In this way, labeled and unlabeled data are used to achieve a better trade-off between accuracy and fairness.

GNNs have been widely used in supervised learning or semi-supervised learning tasks, such as convolutional GNNs and recurrent GNNs [23]. In SSL, GNNs aim to classify the data in a graph using a small subset of labeled data and all the data features. A large number of unlabeled data added to model training is able to help the utilization of structural and feature information of all data, and thus improves the classification accuracy. Our second approach, fair graph neural networks (FGNN), is built with GNNs, where the loss function includes classification loss and fairness loss. Classification loss optimizes the classification accuracy over all labeled data, and fairness loss enforces fairness over labeled data and unlabeled data. GNN models combine graph structures and features, and our method allows GNN models to distribute gradient information from the classification loss and fairness loss. Thus, fair representations of nodes with labeled and unlabeled data can be learned to achieve the ideal trade-off between accuracy and fairness.

With the aim of achieving fair graph-based SSL, the contributions of this paper are as follows.

- First, we conduct the study of algorithmic fairness in the setting of graph-based SSL, including graph-based regularizations and graph neural networks. These approaches enable the use of unlabeled data to achieve a better trade-off between fairness and accuracy.
- Second, we propose algorithms to solve optimization problems when disparate impact and disparate mistreatment are integrated as fairness metrics in the graph-based regularization.
- Third, we consider different cases of fairness constraints on labeled and unlabeled data. This helps us understand the impact of unlabeled data on model fairness, and how to control the fairness level in practice.
- Forth, we conduct extensive experiments to validate the effectiveness of our proposed methods.

The rest of this paper is organized as follows. The preliminaries is given in Section 2. The first proposed method FSMC is given in Sect. 3, and the second proposed method FGNN is given in Sect. 4. The experiments are set out in Sect. 5. The related work appears in Sect. 6, with the conclusion in Sect. 7.

2 Preliminaries

2.1 Notations

Let $X = \{x_1, \ldots, x_k\}^T \in \mathbb{R}^{k \times v}$ denote the training data matrix, where k is the number of data point and v is the number of unprotected attributes; $\mathbf{z} = \{z_1, \ldots, z_k\} \in \{0, 1\}^k$ denotes the protected attribute, e.g., gender or race. Labeled dataset is denoted as $\mathcal{D}_l = \{x_i, z_i, y_{l,i}\}_{i=1}^{k_l}$ with k_l data points, and $\mathbf{y_l} = \{y_{l,1}, \ldots, y_{l,k_l}\}^T \in \{0, 1\}^{k_l}$ is the label for the labeled dataset. Unlabeled dataset is denoted as $\mathcal{D}_u = \{x_i, z_i\}_{i=1}^{k_u}$ with k_u data points, and $\mathbf{y_u} = \{y_{u,1}, \ldots, y_{u,k_u}\}^T \in \{0, 1\}^{k_u}$ is the predicted labels for the unlabeled dataset.

Given the whole dataset, an adjacency matrix is denoted as $A = \theta_{ij} \in \mathbb{R}^{k \times k}, \forall i, j \in 1, \ldots, k, (k = k_l + k_u)$, where θ_{ij} is the weight to evaluate the relationship of two data points. The degree matrix D is constructed as a diagonal matrix whose i-th diagonal element is $d_{ii} = \sum_{j=1}^{k} \theta_{ij}$. We use L to denote Laplacian matrix, calculated as $L = D - A$. Our objective is to learn a classification model $f(\cdot)$ with the model parameters $\mathbf{w}$ (or W) and $\mathbf{y_u}$ over discriminatory datasets $\mathcal{D}_l$ and $\mathcal{D}_u$ that delivers high accuracy with low discrimination.

2.2 Fairness metrics

In our framework, we have applied disparate impact and disparate mistreatment as the fairness metrics [6, 24].

2.2.1 Disparate impact

A classification model does not suffer disparate impact if,

$$\Pr(\hat{y} = y \mid z = 1) = \Pr(\hat{y} = y \mid z = 0) \tag{1}$$

where $\hat{y}$ is the predicted label. When the rate of positive predictions is the same for both groups $z = 1$ and $z = 0$, then there is no disparate impact.

2.2.2 Disparate mistreatment

A binary classifier will not suffer disparate mistreatment if the misclassification rate of different groups with different values of sensitive feature z is the same. Here, three different kind of disparate mistreatments are adopted to evaluate the discrimination as follows,

- Overall misclassification rate (OMR):

$$Pr(\hat{y} \neq y \mid z = 1) = Pr(\hat{y} \neq y \mid z = 0) \tag{2}$$

- False positive rate (FPR):

$$Pr(\hat{y} \neq y \mid z = 1, y = 0) = Pr(\hat{y} \neq y \mid z = 0, y = 0) \tag{3}$$

- False negative rate (FNR):

$$Pr(\hat{y} \neq y \mid z = 1, y = 1) = Pr(\hat{y} \neq y \mid z = 0, y = 1) \tag{4}$$

In most cases, a classifier suffers discrimination in terms of disparate impact or disparate mistreatment. The discrimination level is defined as the differences in rates between different groups.

Definition 1 (Discrimination level) Let γ_z denote the probability of positive predictions of group z on a model f training with a dataset D in terms of a fairness metric. The discrimination level $\Gamma(\hat{y})$ on a model f training with a dataset D is measured by the difference between groups:

$$\Gamma(\hat{y}) = \gamma_0(\hat{y}) - \gamma_1(\hat{y}). \tag{5}$$

Take disparate impact as an example, we have $\gamma_1 = Pr(\hat{y} = 1 \mid z = 1)$, and the discrimination level is $\Gamma(\hat{y}) =\mid Pr(\hat{y} = 1 \mid z = 1) - Pr(\hat{y} = 1 \mid z = 0) \mid$.

2.3 Fairness constraints

Many fairness constraints [6, 24, 25] have been proposed to enforce various fairness metrics, such as disparate impact and disparate mistreatment, and these fairness constraints can be used in our framework. The basic idea to design fairness constraints is that using the covariance between the users' sensitive attributes and the signed distance between the feature vectors restricts the correlation between sensitive attributes and classification results. This can be described as,

$$\begin{aligned} \text{Cov}\left(\mathbf{z}, \mathbf{g_w}\right) &= \mathbb{E}\left[\left(\mathbf{z} - \bar{z}\right)\left(\mathbf{g_w} - \bar{\mathbf{g_w}}\right)\right] \\ &\approx \frac{1}{k}\mathbf{g_w}\left(\mathbf{z} - \bar{z}\right) \end{aligned} \tag{6}$$

where $\mathbf{g_w^T} \in \mathbb{R}^k$ is a vector that denotes the signed distance between the feature vectors and the decision boundary of a classifier. $\mathbf{z}$ denotes the vector of the protected attribute, and $\bar{z}$ denotes the mean value of the protected attribute. The details of obtaining Eq.(6) can be found in [6]. The form of $\mathbf{g_w}$ is different in fairness metrics, and we list them in the following,

- Disparate impact

$$\mathbf{g_w} = \mathbf{w^T} X \tag{7}$$

- Overall misclassification rate

$$\mathbf{g_w} = \min\left(0, \mathbf{y}^T\mathbf{y}\mathbf{w}^T X\right) \tag{8}$$

- False positive rate

$$\mathbf{g_w} = \min\left(0, \frac{\mathbf{1} - \mathbf{y}^T}{2}\mathbf{y}\mathbf{w}^T X\right) \tag{9}$$

- False negative rate

$$\mathbf{g_w} = \min\left(0, \frac{\mathbf{1} + \mathbf{y}^T}{2}\mathbf{y}\mathbf{w}^T X\right) \tag{10}$$

2.4 Graph-based semi-supervised learning

2.4.1 Graph-base regularization

In graph-based regularization, the goal is searching for a function f on the graph. f has to satisfy two criteria simultaneously: (1) it should be as close to the given labels as possible, and (2) it should be smooth on the entire constructed graph. Graph stores the geometric

structure in the data (such as similarity or proximity) and use this structure as a regularizer to infer labels of unlabeled data. Generally, the graph-based regularization methods adopt the following objective function,

$$\mathcal{J} = \mathcal{J}_{\mathcal{C}} + \alpha \mathcal{J}_{\mathcal{L}} \tag{11}$$

where $\mathcal{J}_{\mathcal{C}}$ is the classification loss; α is a balancing parameter; $\mathcal{J}_{\mathcal{L}}$ is a graph-based regularizer. Different methods can have different variants of the regularizer. In our paper, we consider Laplacian regularizer as it is the most common used regularizer, which is calculated by,

$$\mathcal{J}_{\mathcal{L}} = \sum_{i,j} \theta_{ij} ||f(x_i) - f(x_j)||^2. \tag{12}$$

Here, θ_{ij} is a graph-based weight. The edges in the graph between each pair of data points i and j is weighted. The closer the two points are in Euclidean space d_{ij}, the greater the weight θ_{ij}. In this paper, we chose a Gaussian similarity function to calculate the weights, given as follows:

$$\theta_{ij} = \exp\left(-\frac{d_{ij}^2}{\sigma^2}\right) = \exp\left(-\frac{\sum_d \left(x_i^d - x_j^d\right)^2}{\sigma^2}\right) \tag{13}$$

where σ is a length scale parameter. This parameter has an impact on the graph structure; hence, the value of σ needs to be selected carefully [21].

2.4.2 GNN-based SSL

Another method that has received a lot of attention recently is GNNs [23, 26]. The main idea is that the representation vector of the node can contain information from the structure of the graph, and also on any associated feature information. A graph neural network aggregates the neighboring nodes' features into a hidden representation for a central node. This aggregation operation can also be imposed on the hidden representation to form a deeper neural network. In general, for node i, a single aggregation operation can be represented as follows,

$$H^{l+1} = \upsilon\left(\Phi W^l H^l\right) \tag{14}$$

where H^l is the hidden representation of l-th layer; W is the trainable weight matrix in the layer l; υ is the activation function; Φ denotes the rule of how to aggregate neighboring information. Predictions of each node is given on top of the hidden representation of the last layer.

3 Fairness constraints in SSL on margin classifiers

In this section, we first present the proposed framework in Sect. 3.1. Then fairness metrics of disparate impact and disparate mistreatment in logistic regression are analyzed in Sect. 3.2, and finally a discussion is given in Sect. 3.3.

3.1 The proposed framework

We formulate the framework of fair SSL as following, including the classification loss, the label propagation loss and fairness constraints.

$$\min_{\mathbf{w},\mathbf{y_u}} \mathcal{J}_{\mathcal{C}}(\mathbf{w}, \mathbf{y_u}) + \alpha \mathcal{J}_{\mathcal{L}}(\mathbf{y_u}) \qquad s.t.s(\mathbf{w}) \leq c \tag{15}$$

where $\mathcal{J}_{\mathcal{C}}$ is the classification loss between predicted labels and true labels; $\mathcal{J}_{\mathcal{L}}$ is the loss of label propagation from labeled data to unlabeled data; α is a parameter to balance the loss; $s(\mathbf{w})$ is the expression of fairness constraints; and c is a threshold.

3.1.1 Classification loss

A classification loss function evaluates how well a specific algorithm models the given dataset. When different algorithms are used to train datasets, such as logistic regression or neural networks, a corresponding loss function is applied to evaluate the accuracy of the model.

3.1.2 Label propagation loss

According to [22], when Laplacian regularizer is used, the label propagation loss for $\mathcal{J}_{\mathcal{L}}$ through SSL can be expressed as,

$$\mathcal{J}_{\mathcal{L}} = \min_{\mathbf{y_u}} \mathrm{Tr}(\mathbf{y}^T L \mathbf{y}) \tag{16}$$

where Tr denotes the trace, and the vector $\mathbf{y} = [\mathbf{y}_l; \mathbf{y}_u] \in \mathbb{R}^k$ includes labels of labeled and unlabeled data.

3.1.3 Fairness constraints

Adding fairness constraints is a useful method to enforce fair learning with in-processing methods. In SSL, labeled data and unlabeled data have different impacts on discrimination because of two reasons: (1) predicting labels for unlabeled data will bring noise to the labels; (2) labeled data and unlabeled data may have different data distributions. Therefore, the discrimination inherently in unlabeled data is different from the discrimination in labeled data. For these reasons, we impose fairness constraints on labeled and unlabeled data to measure discrimination to see the disparate impact of fairness constraints on labeled and unlabeled data. We consider four cases of fairness constraints enforced on the training data:

- 1. Labeled constraint: The fairness constraint is on labeled data.
- 2. Unlabeled constraint: The fairness constraint is on unlabeled data.
- 3. Combined constraint: The fairness constraint is on labeled data and unlabeled data separately.
- 4. Mixed constraint: The fairness constraint is on labeled and unlabeled data together.

3.2 Fair SSL of logistic regression

In this section, we propose algorithms to solve the optimization problem (15) with a binary logistic regression (LR) classifier. (Other margin classifiers can also be applied in our method, and we give another example of support vector machines in the supplemental material.) The

classifier is subjected to the fairness metric of disparate impact with mixed labeled and unlabeled data. The objective function of LR is defined as,

$$\mathcal{J}_{\mathcal{C}}^{LR} = -\ln(\mathbf{p})\mathbf{y} - \ln(\mathbf{1}-\mathbf{p})(\mathbf{1}-\mathbf{y}) \tag{17}$$

where $\mathbf{p} = \frac{1}{1+e^{-\mathbf{w}^T X}}$ is the probability distribution of mapping X to the class label $\mathbf{y}$; $\mathbf{1}$ denotes a column vector with all its elements being 1. Given the logistic regression loss, the label propagation loss and the fairness metric, the optimized problem (15) adopts the form,

$$\begin{aligned} &\min_{\mathbf{w},\mathbf{y_u}} -\ln(\mathbf{p})\mathbf{y} - \ln(\mathbf{1}-\mathbf{p})(\mathbf{1}-\mathbf{y}) + \alpha Tr(\mathbf{y}^T L\mathbf{y}) \\ &s.t. \mid \frac{1}{k}\mathbf{g_w}\,(\mathbf{z}-\bar{z}) \mid \leq c \end{aligned} \tag{18}$$

3.2.1 Disparate impact

First, we solve the optimization problem with disparate impact as the fairness metric. The optimization of problem (18) includes two parts: learning the weights $\mathbf{w}$ and predicted labels of unlabeled data $\mathbf{y_u}$. The basic idea of solution is that because of the fairness constraint, the weight $\mathbf{w}$ is updated towards a fair direction, and using the updated $\mathbf{w}$ to update $\mathbf{y_u}$ also ensures that $\mathbf{y_u}$ is directed towards fairness. The problem is solved by updating $\mathbf{w}$ and $\mathbf{y_u}$ iteratively as follows.

Solving w when $\mathbf{y_u}$ is fixed, the problem (18) becomes

$$\begin{aligned} &\min_{\mathbf{w}} -\ln(\mathbf{p})\mathbf{y} - \ln(\mathbf{1}-\mathbf{p})(\mathbf{1}-\mathbf{y}) \\ &s.t. \mid \frac{1}{k}\mathbf{w}^T X\,(\mathbf{z}-\bar{z}) \mid \leq c \end{aligned} \tag{19}$$

Note that problem (19) is a convex problem that can be written as a regularized optimization problem by moving fairness constraints to the objective function. The optimal $\mathbf{w}^*$ can then be calculated by using Karush-Kuhn-Tucker (KKT) conditions.

Solving $\mathbf{y_u}$ when w is fixed, the problem (18) becomes

$$\min_{\mathbf{y_u}} -\ln(\mathbf{p})\mathbf{y} - \ln(\mathbf{1}-\mathbf{p})(\mathbf{1}-\mathbf{y}) + \alpha Tr(\mathbf{y}^T L\mathbf{y}) \tag{20}$$

Given that problem (20) is also a convex problem, the optimal $\mathbf{y_u}$ can be obtained from the deviation of $\mathbf{y_u}$ in problem (20). In order to calculate $\mathbf{y_u}$ conveniently, we split Laplacian matrix L into four blocks after the l-th row and the l-th column: $L = \begin{bmatrix} L_{ll} & L_{lu} \\ L_{ul} & L_{uu} \end{bmatrix}$. The deviation of Eq.(20) is then calculated w.r.t. $\mathbf{y_u}$ and setting to zero, we have

$$\alpha(2\mathbf{y_u}L_{uu} + L_{ul}\mathbf{y_l} + (\mathbf{y_l}L_{lu})^T) - [(ln(\mathbf{p}))^T + (ln(\mathbf{1}-\mathbf{p}))^T] = 0 \tag{21}$$

Note that L is a symmetric matrix and, after simplification, the closed updated form of $\mathbf{y_u}$ can be derived from

$$\mathbf{y_u} = -L_{uu}^{-1}(L_{ul}\mathbf{y_l} + \frac{1}{2\alpha}[(ln(\mathbf{p}))^T + (ln(\mathbf{1}-\mathbf{p}))^T]) \tag{22}$$

Note that the computed optimal $\mathbf{y_u}$ is decimals, and it cannot be used to update $\mathbf{w}$ directly because only integers are allowed to optimize $\mathbf{w}$ in the next update. Due to this, we need to

convert $\mathbf{y_u}$ from decimals to integers to update $\mathbf{w}$. Before using $\mathbf{y_u}$ to update the next $\mathbf{w}$, the value of $y_{u,i} \in \mathbf{y_u}, i = 1, \ldots, k_u$ is set to,

$$y_{u,i} = \begin{cases} 1, & y_{u,i} \geq \xi \\ 0, & y_{u,i} < \xi \end{cases} \tag{23}$$

where ξ is the threshold that determines the classification result. Then, the optimization problem (18) can be solved by optimizing $\mathbf{w}$ and $\mathbf{y_u}$ iteratively. **Algorithm 1** summarizes the solution of optimization problem (18) with the disparate impact.

Algorithm 1 The algorithm of optimizing problem (18) with disparate impact

Input: Labeled dataset $\mathcal{D}_l$, unlabeled dataset $\mathcal{D}_u$, fairness thresholds c
Parameter: ξ, σ
Initialize: Given initial values of $\mathbf{y_u}$ by label propagation
Output: $\mathbf{w}$ and $\mathbf{y_u}$
1: Calculate the adjacency matrix A according to Eq.(13)
2: **repeat**
3: Fix $\mathbf{y_u}$ and update $\mathbf{w}$ with KKT
4: Fix $\mathbf{w}$ and update $\mathbf{y_u}$ by Eq.(22)
5: Set $y_{u,i} \in \mathbf{y_u}$ to 0 or 1 by Eq. (23)
6: **until** The optimization problem (18) convergs

3.2.2 Disparate mistreatment

Disparate mistreatment metrics include overall misclassification rate, false positive rate and false negative rate. For simplicity, overall misclassification rate is used to analyze disparate mistreatment. However, false positive rate and false negative rate can also be analyzed easily, and the result of three disparate mistreatment metrics are presented in the experiment.

With the overall misclassification rate as the fairness metric, the objective function is denoted as,

$$\begin{aligned} &\min_{\mathbf{w}} -\ln(\mathbf{p})\mathbf{y} - \ln(\mathbf{1}-\mathbf{p})(\mathbf{1}-\mathbf{y}) + \alpha Tr(\mathbf{y}^T L \mathbf{y}) \\ &s.t. \mid \frac{1}{k}\mathbf{g_w}(\mathbf{x})\,(\mathbf{z}-\bar{z}) \mid \leq c \end{aligned} \tag{24}$$

Note that fairness constraints of disparate mistreatment are non-convex, and the solution of the optimization problem (24) is more challenging than the optimization problem in (18). Next, we convert these constraints into a Disciplined Convex-Concave Program (DCCP). Thus, the optimization problem (24) can be solved efficiently with the recent advances in convex-concave programming [27].

The fairness constraint of disparate mistreatment can be split into two terms,

$$\frac{1}{k} \mid \sum_{\mathcal{D}_0}(0-\bar{z})\mathbf{g_w} + \sum_{\mathcal{D}_1}(1-\bar{z})\mathbf{g_w} \mid \leq c \tag{25}$$

where $\mathcal{D}_0$ and $\mathcal{D}_1$ are the subsets of the labeled dataset $\mathcal{D}_l$ and unlabeled dataset $\mathcal{D}_u$ with values $z = 0$ and $z = 1$, respectively. k_0 and k_1 are defined as the number of data points in the $\mathcal{D}_0$ and $\mathcal{D}_1$, and thus $\bar{z}$ can be rewritten as $\bar{z} = \frac{0*k_0+1*k_1}{k} = \frac{k_1}{k}$. Then the fairness constraint of disparate mistreatment can be rewriten as,

$$\frac{k_1}{k} \mid \sum_{\mathcal{D}_0} \mathbf{g_w} + \sum_{\mathcal{D}_1} \mathbf{g_w} \mid \leq c \tag{26}$$

Solving w when $\mathbf{y_u}$ is fixed, the problem (24) becomes

$$\begin{aligned} &\min_{\mathbf{w}} -\ln(\mathbf{p})\mathbf{y} - \ln(\mathbf{1}-\mathbf{p})(\mathbf{1}-\mathbf{y}) \\ &s.t. \frac{k_1}{k} \mid \sum_{\mathcal{D}_0} \mathbf{g_w} + \sum_{\mathcal{D}_1} \mathbf{g_w} \mid \leq c \end{aligned} \tag{27}$$

The optimization problem (27) is a Disciplined Convex-Concave Program (DCCP) for any convex loss, and can be solved with some efficient heuristics [27].

Solving $\mathbf{y_u}$ when w is fixed, the problem (24) becomes

$$\min_{\mathbf{y_u}} -\ln(\mathbf{p})\mathbf{y} - \ln(\mathbf{1}-\mathbf{p})(\mathbf{1}-\mathbf{y}) + \alpha Tr(\mathbf{y}^T L\mathbf{y}) \tag{28}$$

The solution of Eq. (28) is the same as the solution of the Eq. (22). The closed form of $\mathbf{y_u}$ can be obtained via Eq. (23), and then the optimization problem (23) can be solved by updating $\mathbf{y_u}$ and $\mathbf{w}$ iteratively. **Algorithm 2** summarizes this process.

Algorithm 2 The algorithm of optimizing problem (24)

Input: Labeled dataset $\mathcal{D}_l$, unlabeled dataset $\mathcal{D}_u$, fairness thresholds c
Parameter: ξ, σ
Initialize:Given initial values of $\mathbf{y_u}$ by label propagation
Output: $\mathbf{w}$ and $\mathbf{y_u}$

```
Calculate the adjacency matrix A according to Eq.(13)
Choose a metric in disparate mistreatment
repeat
    Divide D into D_0 and D_1
    Calculate k_0 and k_1
    Fix y_u and update w with DCCP
    Fix w and update y_u by Eq.(22)
    Set y_{u,i} ∈ y_u to 0 or 1 by Eq. (23)
until The optimization problem (24) convergs
```

3.3 Discussion

Based on above analysis, some conclusions can be drawn:

1. Since unlabeled data do not contain any label information, they do not label biased information so that we can take advantage of the unlabeled data to improve the trade off between accuracy and fairness. In our framework, due to the fairness constraint, the weight $\mathbf{w}$ is updated towards a fair direction. Using the updated $\mathbf{w}$ to update $\mathbf{y_u}$ also ensures that $\mathbf{y_u}$ is directed towards fairness. In this way, fairness is enforced in labeled and unlabeled data by updating $\mathbf{w}$ and $\mathbf{y_u}$ iteratively. Therefore, labels of unlabeled data are calculated in a fair way, which is beneficial to the accuracy of the classifier as well as the fairness of the classifier.
2. Fairness constraints on labeled data and unlabeled data have different impact on the training result because labeled and unlabeled data may present different covariance between the sensitive attribute and the signed distance between feature vectors to the decision boundaries.

4 Fairness regularizers in SSL on graph neural networks

In this section, we present the proposed method of how to achieve fair SSL on GNNs. The main idea of the proposed method is to impose fairness regularizers on GNNs that is implemented in the SSL setting. In this way, GNN models can allocate gradient information from the classification loss and the fairness loss to ensure fairness. Firstly, we introduce a framework for fair SSL on GNNs, and then present a case of fair graph convolutional networks.

4.1 The proposed methods

Our goal is to learn a neural network function $f(W)$ that optimize two main objectives: the classification accuracy and fairness. The loss function of the model is defined as,

$$\mathcal{J}(\mathcal{D}; W) = \mathcal{J}_{\mathcal{C}}(\mathcal{D}; W) + \beta \mathcal{J}_{\mathcal{F}}(\mathcal{D}; W) \tag{29}$$

where $\mathcal{J}(\mathcal{D}; W)$ denotes the classification loss, and $\mathcal{J}_{\mathcal{F}}(\mathcal{D}; W)$ denotes the fairness loss that imposes fairness regularizers on the output of the model. β adjusts the trade-off between fairness and accuracy loss. Typically, the cross-entropy loss is used to calculate the classification loss.

4.1.1 Fairness constraints

The second item in the loss function exerts fairness on the learning function. Since fairness constraints Eqs. (7)–(10) are not differentiable, fairness regularizers are defined according to literal definitions of fairness metrics, these regularizers are able to handle and optimize different fairness definitions so as to adjust the appropriate fairness definition according to the application.

The fairness regularizer of disparate impact is defined as,

$$\mathcal{J}_{\mathcal{F}}^{DI} = \left| \frac{\sum_{i=1}^{k} p_i z_i}{\sum_{i=1}^{k} z_i} - \frac{\sum_{i=1}^{k} p_i (1 - z_i)}{\sum_{i=1}^{k} 1 - z_i} \right| \tag{30}$$

where p_i denotes the predicted probability of the i-th data point belonging to one class calculated by a softmax function in the last layer of the network.

Disparate mistreatment, including FPR, FNR, and OMR are defined in the following,

$$\mathcal{J}_{\mathcal{F}}^{FPR} = \left| \frac{\sum_{i=1}^{k} p_i (1 - y_i) z_i}{\sum_{i=1}^{k} z_i} - \frac{\sum_{i=1}^{k} p_i (1 - y_i) (1 - z_i)}{\sum_{i=1}^{k} 1 - z_i} \right| \tag{31}$$

$$\mathcal{J}_{\mathcal{F}}^{FNR} = \left| \frac{\sum_{i=1}^{k} (1 - p_i) y_i z_i}{\sum_{i=1}^{k} z_i} - \frac{\sum_{i=1}^{k} (1 - p_i) y_i (1 - z_i)}{\sum_{i=1}^{k} 1 - z_i} \right| \tag{32}$$

$$\mathcal{J}_{\mathcal{F}}^{OMR} = \mathcal{J}_{\mathcal{F}}^{FPR} + \mathcal{J}_{\mathcal{F}}^{FNR} \tag{33}$$

4.2 Fair SSL of convolutional GNN

In this section, we study a case of fair graph convolutional network (GCN), where a multi-layer graph convolutional networks is used to optimize the classification loss in the Eq. (30). We take GCN as an example since GCN achieves high performance in SSL tasks, and our method can also apply in other GNNs. The GCN model combines the graph structure and

vertex features in the convolution, in which the features of unlabeled vertices are mixed with those of neighboring labeled vertices, and then propagated to the graph through multiple layers.

The propagation rule of a multi-layer GCN is defined as [26],

$$H^{(l+1)} = \upsilon\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \tag{34}$$

where $\tilde{A} = A + I_N$ is the adjacency matrix of the undirected graph with added self-connections and $\tilde{D} = \sum_j \tilde{A}_{ij}$.

The model used in this paper is a two-layer GCN, and softmax classifier is applied to the output features,

$$S = \text{softmax}\left(\hat{A}\,\text{ReLU}\left(\hat{A}XW^{(0)}\right)W^{(1)}\right) \tag{35}$$

where $\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$.

The loss function is defined as the cross entropy error of all labeled data points,

$$\mathcal{J}_C = -\sum_{l\in\mathcal{Y}_L}\sum_{f=1}^{F} Y_{lf}\ln S_{lf} \tag{36}$$

where $\mathcal{Y}_L$ is the set of indices of labeled vertices and F is the number of classes.

Given the GCN loss and the fairness regularizer, the Eq. (29) adopts the form,

$$-\sum_{l\in\mathcal{Y}_L}\sum_{f=1}^{F} Y_{lf}\ln S_{lf} + \beta\mathcal{J}_{\mathcal{F}}(\mathcal{D}; W) \tag{37}$$

The model parameters W can be trained via gradient descent. In this paper, batch gradient descent is used to train datasets for each iteration.

4.3 Discussion

1. GCN naturally combines the structure and features of the graph in the convolution, and thus avoids graph Laplacian regularization. Our method allows the GCN model to allocate gradient information from the classification loss and the fairness loss. Therefore, fair representation of nodes with labeled data and unlabeled data can be learned to achieve fair SSL.
2. Parameter β adjusts the discrimination level. A higher β will impose a higher penalty on fairness loss, and thus decrease the discrimination level. However, a very large β may destroy the expression ability of the model.

5 Experiment

In this section, we first describe the experimental setup, including datasets, baselines, and parameters. Then, we evaluate our method on three real-world datasets under the fairness metric of disparate impact and disparate mistreatment (including OMR, FNR and FPR). The aim of our experiments is to assess: the effectiveness of our methods to achieve fair semi-supervised learning; the impact of different fairness constraints on fairness; and the extent to which unlabeled data can balance fairness with accuracy.

5.1 Experimental setup

5.1.1 Dataset

Our experiments involve three real-world datasets: Health dataset[1], Titanic dataset[2] and Bank dataset[3] When GNN models are used for training, structured datasets need processing into graphs. To construct graph-structured data based on structured data, we need to build an adjacency matrix to describe the topological relationship. In our experiment, we instinctively using Euclidean distance calculated by Eq. (13) as our adjacency matrix for simplicity.

- The task in the Health dataset is to predict whether people will spend time in the hospital. In order to convert the problem into the binary classification task, we only predict whether people will spend any day in the hospital. After data preprocessing, the dataset contains 27,000 data points with 132 features. We divide patients into two groups based on age ($\geq$65 years) and consider 'Age' to be the sensitive attribute.
- The Bank dataset contains a total of 41,188 records with 20 attributes and a binary label, which indicates whether the client has subscribed (positive class) or not (negative class) to a term deposit. We consider 'Age' as sensitive attribute.
- The Titanic dataset comes from a Kaggle competition where the goal is to analyze which sorts of people were likely to survive the sinking of the Titanic. We consider "Gender" as the sensitive attribute. After data preprocessing, we extract 891 data points with 9 features.

5.1.2 Parameters

The sensitive attributes are excluded from the training set to ensure fairness between groups and are only used to evaluate discrimination in the test phrases. In the Health, Bank and Titanic datasets, data are all labeled. In the Health dataset, we sample 4,000 data points as labeled dataset, 4,000 data points as test dataset, and left as unlabeled dataset. In the Bank dataset, we sample 4,000 data points as labeled dataset, 4,000 data points as test dataset, and left as unlabeled dataset. In the Titanic dataset, we sample 200 data points as labeled dataset, 200 data points as test dataset, and left as unlabeled dataset. Therefore, $\mathcal{D}_l$ and $\mathcal{D}_u$ are collected from the similar data distribution.

In the experiments, the results are an average of 10 results by randomly sampling labeled dataset, test dataset and unlabeled dataset.

We set $\alpha = 1$ and $\xi = 0.5$ in all datasets. σ is a length scale parameter. This parameter has an impact on the graph structure, and we set $\sigma = 0.5$ in the Health dataset and Bank dataset, and $\sigma = 0.1$ in the Titanic dataset by using binary search. τ and μ are parameters in DCCP. τ is a parameter that trades off satisfying the constraints and minimizing the objective in DCCP, and we set $\tau = 0.05$ and $\tau = 1$ in Bank and Titanic dataset by binary search. μ parameter sets the rate at which τ increases inside the algorithm, and we set $/mu$ as the default value 1.2 in Bank and Titanic datasets.

[1] https://foreverdata.org/1015/index.html.

[2] https://www.kaggle.com/c/titanic/data.

[3] https://archive.ics.uci.edu/ml/datasets/bank+marketing.

5.1.3 Baseline methods

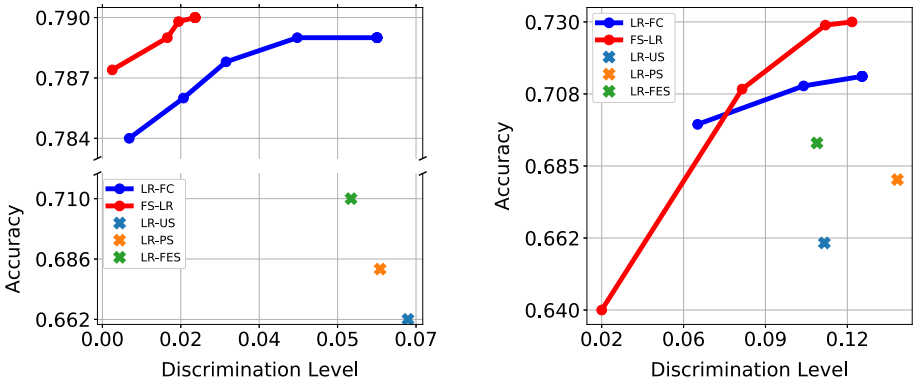
The methods chosen for comparison are listed as follows. PS, US and FES are only applied in the fairness metric of disparate impact, so they are compared with the performance with our methods using disparate impact. FC and FMLP are compared with the performance with our methods using disparate impact and disparate mistreatment. It is worth to note that [28] also used unlabeled data on fairness. However, they only applied the equal opportunity metric, which is different to ours. Hence, we did not compare the proposed method with them.

- Fairness Constraints (FC): Fairness constraints are used to ensure fairness for classifiers. [6]
- Uniform Sampling (US): The number of data points in each groups is equalized through oversampling and/ undersampling. [29]
- Preferential Sampling (PS): The number of data points in each groups is equalized by taking samples near the borderline data points. [29]
- Fair multilayer perceptron neural networks (FMLP): the proposed method is built on Multilayer Perceptron (MLP) neural network for SSL in the in-processing phase, where unlabeled data is marked labels with pseudo labeling. [30]
- Fairness-enhanced sampling (FES): A fair SSL framework includes pseudo labeling, re-sampling and ensemble learning. [31]

5.2 Experimental results of disparate impact

5.2.1 Trade-off between accuracy and discrimination

Figure 1 shows that as c varies, accuracy and discrimination level in the proposed method fair semi-LR (FS-LR) and other methods with LR on two datasets. From the results, we can observe that our framework provides the better trade-off between accuracy and discrimination. A better trade-off means that with the same accuracy, discrimination is low or with



(a) FS-LR-Health **(b)** FS-LR-Titanic

Fig. 1 The trade-off between accuracy and discrimination in the proposed method FS-LR (Red), FC (Blue), US (Blue cross), PS (Yellow cross) and FES (Green cross) under the fairness metric of disparate impact with LR in two datasets. As the threshold of covariance c increases, accuracy and discrimination increase. The results demonstrate that our method achieves a better trade-off between accuracy and discrimination than other methods

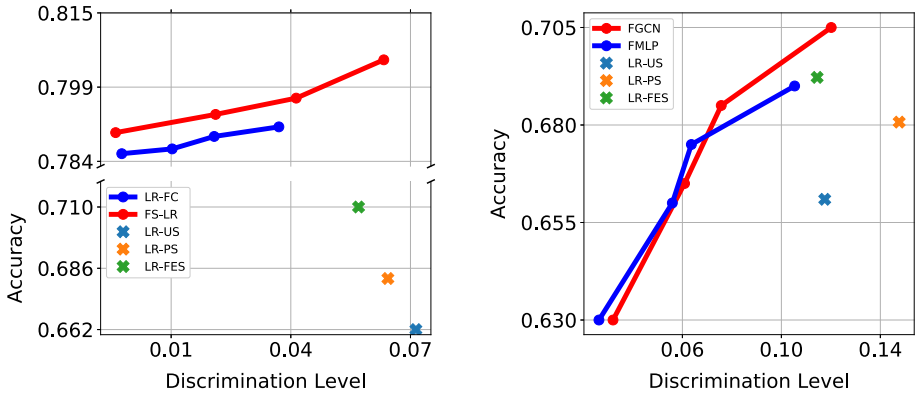


Fig. 2 The trade-off between accuracy and discrimination in the proposed method FGCN (Red), FMLP (Blue), US (Blue cross), PS (Yellow cross) and FES (Green cross) under the fairness metric of disparate impact in two datasets. As the parameter β increases, accuracy and discrimination decrease

the same discrimination, accuracy is higher. For example, at the same level of accuracy on the Titanic dataset, our method FS-LR has a discrimination level of around 0.08, while FC method has a discrimination level of 0.11. A similar observation can be made from the results with PS method (Yellow cross), US method (Blue cross) and FES method (Green cross). Note that the discrimination level (red line) with LR in the Health dataset does not extend because discrimination does not increase as c grows.

Figure 2 shows that accuracy and discrimination level in the proposed method fair GCN (FGCN) and the baseline method FMLP as β varies. The result shows that FGCN performs the better trade-off between accuracy and discrimination than FMLP. This contributes to GCN has effective utilization of structural and feature information of unlabeled data.

5.2.2 Different fairness constraints

Our next set of experiments is to determine the impact of different fairness constraints. For these tests, the size of unlabeled data is set to 12,000 data points in the Health dataset and 400 data points in the Titanic dataset. Due to space limitation, we have only reported the results for the LR, which appear in Tables 1 and 2. The result shows that, when varying the threshold of covariance c, different fairness constraints on labeled and unlabeled data have different impacts on the training results. As the threshold of covariance increases, both accuracy and discrimination level increase before steadying off for the duration. In terms of accuracy, this is because a larger c allows for a larger space to find better weights $\mathbf{w}$ to inform classification. In terms of discrimination, a larger c tends to introduce more discrimination in noise.

It is also observed that the fairness constraint on mixed data generally has the best performance in the trade-off between accuracy and discrimination. Other three constraints have very similar accuracy and discrimination levels. We attribute this to the assumption that labeled and unlabeled data have the similar data distribution, and therefore the mixed fairness constraint on labeled and unlabeled data gives the best description of the covariance between sensitive attributes and signed distance from feature vectors to the decision boundary.

Table 1 The impact of fairness constraints on different datasets in terms of accuracy (Acc) and discrimination level (Dis) under the fairness metric of disparate impact with FS-LR in the Health dataset

Dataset	Health dataset							
Constraint	Labeled		Unlabeled		Combined		Mixed	
	Acc	Dis	Acc	Dis	Acc	Dis	Acc	Dis
c = 0.0	0.7868	0.0042	N/A	N/A	0.7874	0.0022	0.7862	0.0003
c = 0.1	0.7890	0.0129	N/A	N/A	0.7890	0.0145	0.7892	0.0149
c = 0.2	0.7900	0.0170	0.7900	0.0207	0.7898	0.0170	0.7898	0.0170
c = 0.3	0.7898	0.0207	0.7898	0.0170	0.7900	0.0207	0.7900	0.0207
c = 0.4	0.7902	0.0178	0.7898	0.0170	0.7900	0.0207	0.7900	0.0207
c = 0.5	0.7900	0.0207	0.7900	0.0207	0.7900	0.0207	0.7900	0.0207
c = 0.6	0.7900	0.0207	0.7906	0.0186	0.7900	0.0207	0.7900	0.0207
c = 0.7	0.7900	0.0207	0.7900	0.0207	0.7900	0.0207	0.7900	0.0207
c = 0.8	0.7900	0.0207	0.7904	0.0191	0.7900	0.0207	0.7900	0.0207
c = 0.9	0.7900	0.0207	0.7908	0.0190	0.7900	0.0207	0.7900	0.0207
c = 1.0	0.7900	0.0207	0.7900	0.0207	0.7900	0.0207	0.7900	0.0207

Table 2 The impact of fairness constraints on different datasets in terms of accuracy (Acc) and discrimination level (Dis) under the fairness metric of disparate impact with FS-LR in the Titanic dataset

Dataset	Titanic dataset							
Constraint	Labeled		Unlabeled		Combined		Mixed	
	Acc	Dis	Acc	Dis	Acc	Dis	Acc	Dis
c = 0.0	0.6330	0.0128	0.6970	0.1244	0.6290	0.0139	0.6440	0.0402
c = 0.05	0.6690	0.0579	0.7070	0.1265	0.6690	0.0716	0.6810	0.0948
c = 0.1	0.7150	0.1272	0.7140	0.1332	0.7100	0.1239	0.7150	0.1256
c = 0.15	0.7200	0.1366	0.7190	0.1336	0.7190	0.1336	0.7200	0.1366
c = 0.2	0.7200	0.1366	0.7200	0.1366	0.7200	0.1366	0.7200	0.1366
c = 0.25	0.7200	0.1366	0.7200	0.1366	0.7200	0.1366	0.7200	0.1366

5.2.3 The impact of unlabeled data

For these experiments, we set the covariance threshold $c = 1$ for the Health and Titanic datasets, and parameter $\beta = 0.5$ in the Health dataset and $\beta = 0.8$ in the Titanic dataset. Figure 7 shows that accuracy and discrimination level varies with the amount of unlabeled data with FS-LR and FGCN methods on both datasets. As shown, accuracy increases as the amount of unlabeled data increases in both datasets before stabilizing at its peak. Discrimination level sharply decreases almost immediately, then stabilize or decrease. We can explain why unlabeled data help to reduce discrimination according to [19, 31]. In [19, 31], discrimination is decoupled into discrimination in bias, discrimination in variance and discrimination in noise. With an increasing size of unlabeled data, discrimination in variance decreases, leading to the whole discrimination decreases.

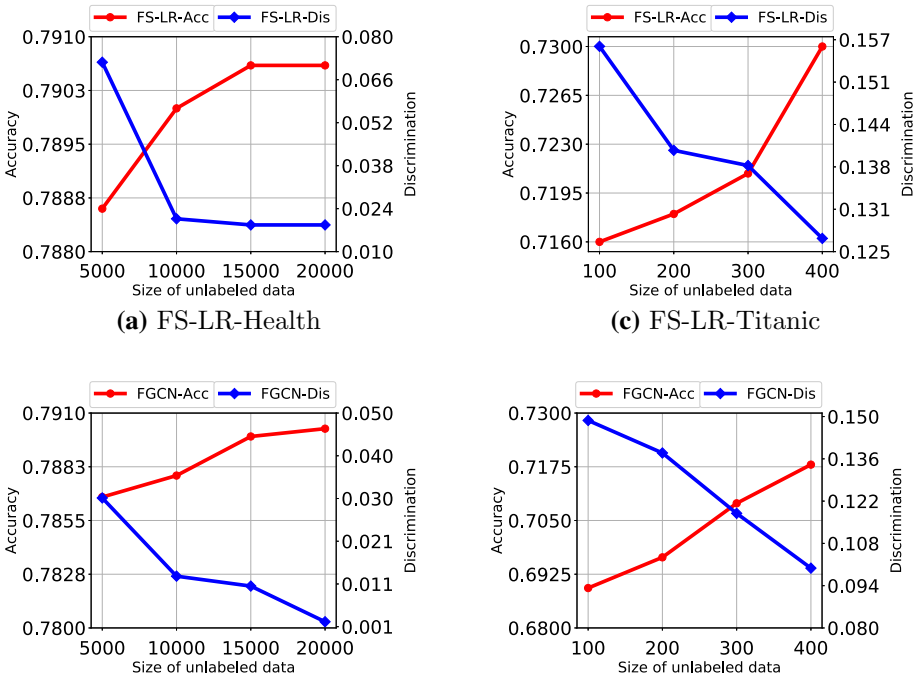
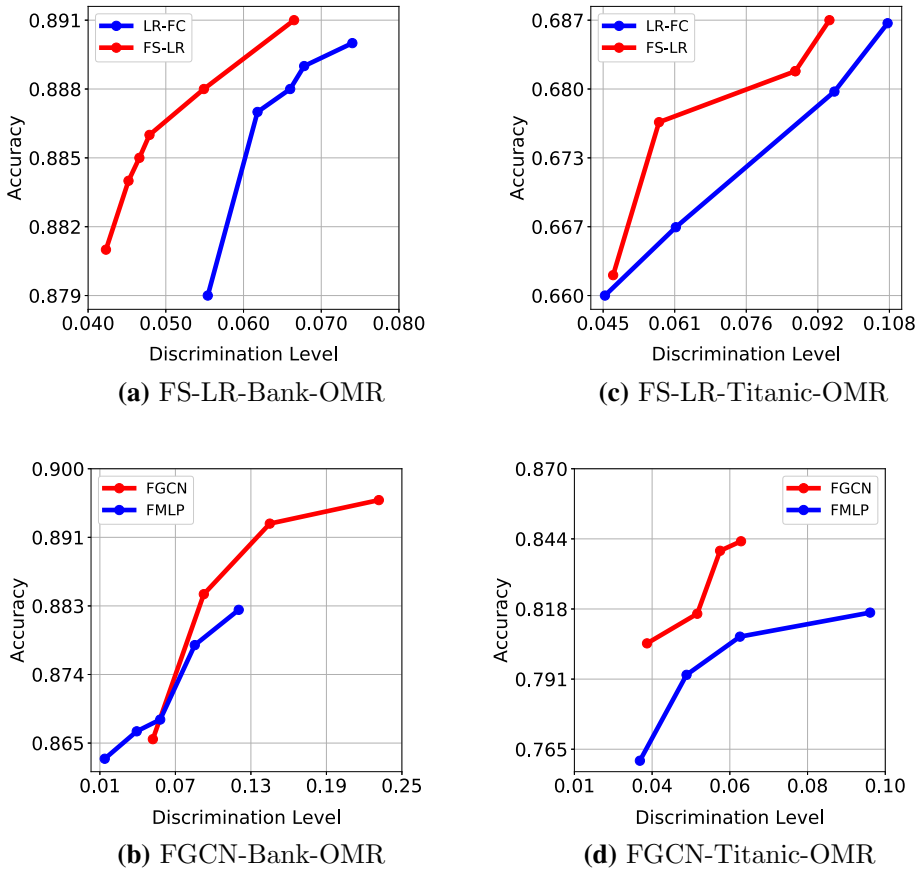


Fig. 3 The impact of the amount of unlabeled data in the training set on accuracy (Red) and discrimination level (Blue) under the fairness metric of disparate impact with FS-LR and FGCN in two datasets. The X-axis is the size of unlabeled dataset; left y-axis is accuracy; and right y-axis is discrimination level

5.3 Experimental results of disparate mistreatment

5.3.1 Trade-off between accuracy and discrimination

Figures 4, 5 and 6 show that as the threshold of covariance c increases in FS-LR, and parameter β increases in FGCN, accuracy and discrimination increase under the fairness metric of OMR, FPR and FNR. From the results, we can observe that our proposed methods FS-LR and FGCN (Red line) generally are in the left above the FC method and FMLP method (Blue line). This indicates that our framework provides the better trade-off between accuracy and discrimination in three metrics for the most time. For example, at the same level of accuracy (Acc = 0.885) on the Bank dataset under OMR, our method with FS-LR has a discrimination level of around 0.045, while FC method has a discrimination level of 0.06. We also observe that discrimination level is quite different under fairness metrics. For example, discrimination level can reach 0.17 at the end under FNR, while discrimination level only shows 0.01 under FPR. In addition, we note that accuracy and discrimination level have different performance on training models. In the Bank dataset, FGCN generally has a lower accuracy and discrimination than FS-LR.



(a) FS-LR-Bank-OMR **(c)** FS-LR-Titanic-OMR

(b) FGCN-Bank-OMR **(d)** FGCN-Titanic-OMR

Fig. 4 The trade-off between accuracy and discrimination in proposed method FS-LR and FGCN (Red), FC and FMLP (Blue) in two datasets under the metric of overall misclassification rate. The results demonstrate that our methods using unlabeled data achieves a better trade-off between accuracy and discrimination

5.3.2 Different fairness constraints under OMR

Tables 3 and 4 shows that different fairness constraints on labeled and unlabeled data have different impacts on the training results. Due to space limitation, we have only reported the results for the FS-LR under the metric of OMR on the Bank and Titanic datasets. For these tests, the size of unlabeled data is set to 4,000 data points in the Bank dataset and 400 data points in the Titanic dataset. As shown, when varying the threshold of covariance c, different fairness constraints on labeled and unlabeled data have huge difference on the training results. When the fairness constraint is enforced in labeled data, accuracy and discrimination increases with the increase in c in the Titanic dataset. This is because a smaller c enforces the lowest discrimination level, which results in a lower accuracy.

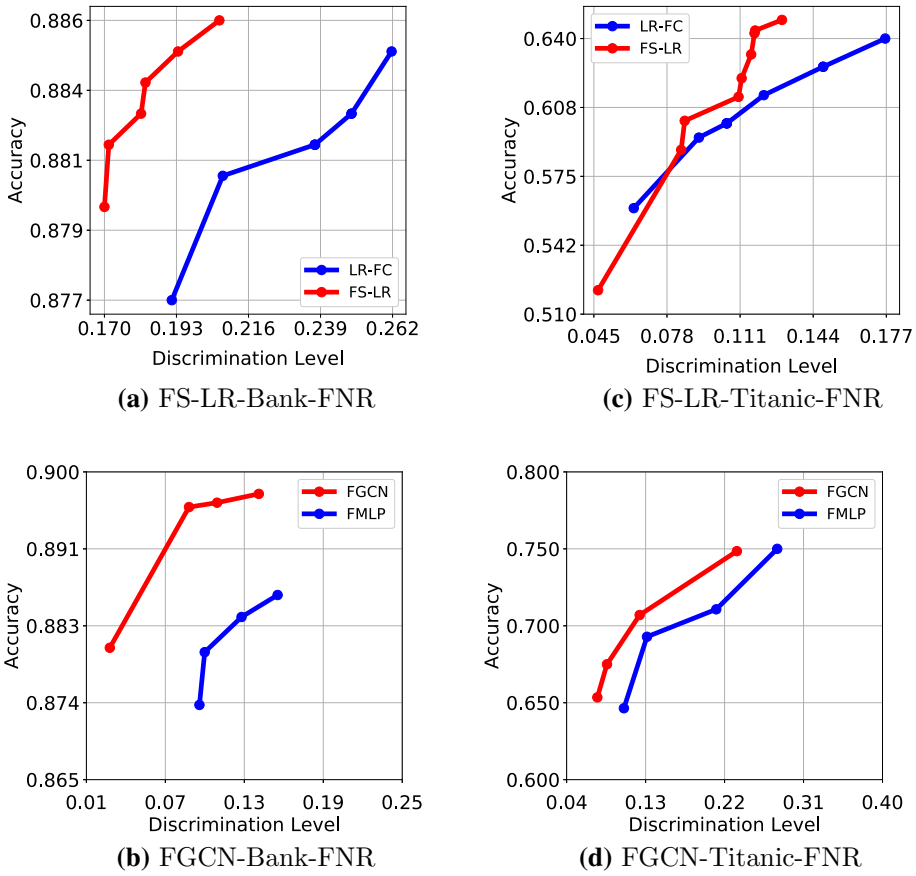
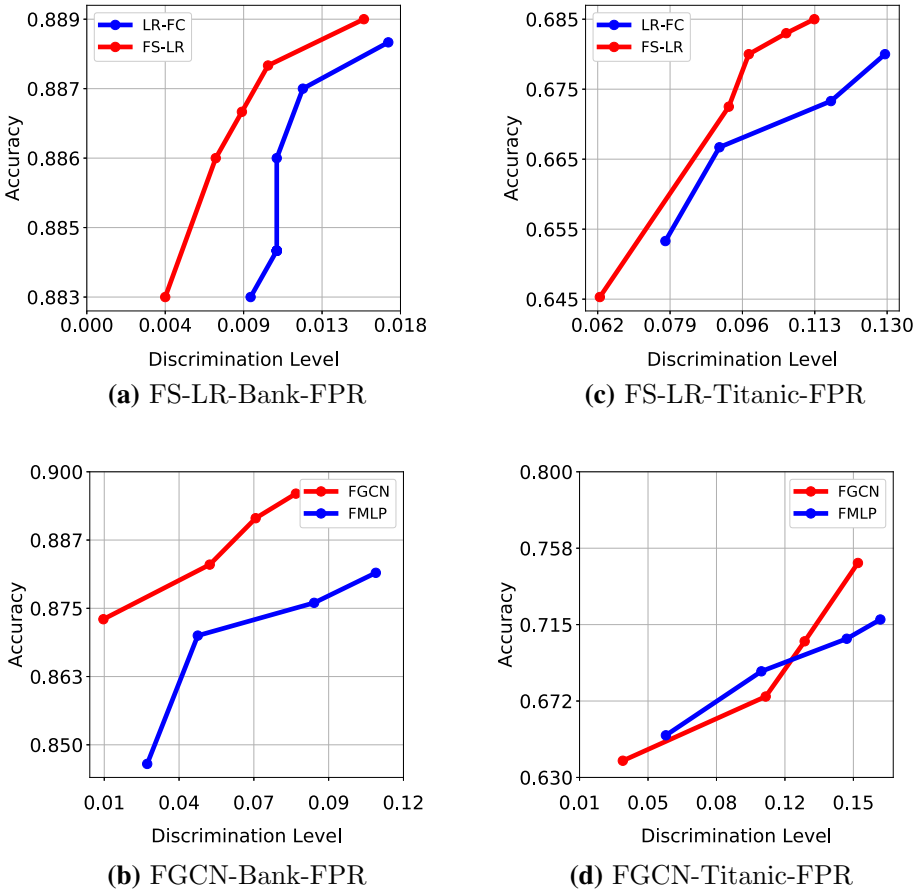


Fig. 5 The trade-off between accuracy and discrimination in the proposed method FS-LR and FGCN (Red), FC and FMLP (Blue) in two datasets under the metric of false negative rate

However, when the fairness constraint is enforced in unlabeled data, accuracy and discrimination could decrease with the increase in c. This is because the label of unlabeled data appears in the fairness constraint of disparate mistreatment, and it is updated during the training. This means that the distribution of unlabeled data is not described well during the training. As a result, the fairness constraint on unlabeled data is not that effective.

5.3.3 The impact of unlabeled data under OMR

For these experiments, we show the impact of unlabeled data on OMR. The covariance threshold is set as $c = 1$ for the Bank and Titanic datasets. Figure 7 shows accuracy and discrimination level varies given different size of unlabeled data with FS-LR and FGCN on two datasets. As shown, before the peak is reached, as the amount of unlabeled data increases in the two data sets, accuracy will also increase. Discrimination level decreases at the beginning, and then stabilize in the Titanic dataset. These results indicate that discrimination in variance decreases as the amount of unlabeled data in the training set increases.



(a) FS-LR-Bank-FPR **(c)** FS-LR-Titanic-FPR

(b) FGCN-Bank-FPR **(d)** FGCN-Titanic-FPR

Fig. 6 The trade-off between accuracy and discrimination in proposed method FS-LR and FGCN (Red), FS and FMLP (Blue) in two datasets under the metric of false positive rate

5.4 Discussion and summary

5.4.1 Discussion

We have some comparison of two methods. This provides some suggestions to choose which method to use in practice. 1) FGCN is suitable to train a large dataset, while FSMC may not work because the DCCP solver is difficult to process a large number of data points. 2) FGCN is suitable for multi-classification problems, while FSMC cannot directly be applied in multi-classification problems. 3) FSMC admits a closed form solution which makes it attractive in practice with a low computationally cost, while FGCN is generally more computational.

5.4.2 Summary

From these experiments, we can obtain some conclusions. 1) The proposed methods, FSMC and FGNN can make use of unlabeled data to achieve a better trade-off between accuracy and discrimination. 2) In FSMC, the fairness constraint on mixed labeled and unlabeled

Table 3 The impact of fairness constraints on different datasets in terms of accuracy (Acc) and discrimination level (Dis) under the fairness metric of overall misclassification rate with FS-LR in the Bank dataset

Dataset	Bank dataset							
Constraint	Labeled		Unlabeled		Combined		Mixed	
	Acc	Dis	Acc	Dis	Acc	Dis	Acc	Dis
c = 0.0	0.8635	0.0905	0.8407	0.1847	0.8342	0.147	0.8605	0.0987
c = 0.5	0.8625	0.092	0.8402	0.1854	0.8342	0.1442	0.8605	0.0987
c = 1.0	0.8638	0.0922	0.8402	0.1854	0.835	0.1452	0.8635	0.1071
c = 1.5	0.8645	0.0918	0.8407	0.1833	0.835	0.1452	0.8635	0.1071
c = 2.0	0.8648	0.0907	0.841	0.1822	0.8347	0.1462	0.8625	0.1071
c = 2.5	0.8652	0.0914	0.8413	0.1812	0.8353	0.1469	0.8635	0.1084
c = 3.0	0.866	0.0923	0.8413	0.1784	0.8342	0.147	0.8627	0.1084
c = 3.5	0.8662	0.0927	0.8407	0.1805	0.8342	0.147	0.8627	0.1097
c = 4.0	0.8665	0.093	0.841	0.1795	0.8342	0.147	0.8627	0.1097
c = 4.5	0.8668	0.0919	0.8407	0.1791	0.835	0.1452	0.8635	0.1113
c = 5.0	0.867	0.0909	0.8407	0.1791	0.8355	0.1444	0.8635	0.1113

Table 4 The impact of fairness constraints on different datasets in terms of accuracy (Acc) and discrimination level (Dis) under the fairness metric of overall misclassification rate with FS-LR in the Titanic dataset

Dataset	Titanic dataset							
Constraint	Labeled		Unlabeled		Combined		Mixed	
	Acc	Dis	Acc	Dis	Acc	Dis	Acc	Dis
c = 0.0	0.7448	0.0285	0.7138	0.3996	0.7655	0.1387	0.7483	0.0175
c = 0.5	0.7483	0.0335	0.6966	0.4386	0.7655	0.1547	0.7483	0.0175
c = 1.0	0.7517	0.0385	0.6931	0.4656	0.7552	0.1397	0.7517	0.0225
c = 1.5	0.7552	0.0436	0.7103	0.3946	0.7793	0.1748	0.7448	0.0445
c = 2.0	0.7552	0.0436	0.7069	0.4216	0.7724	0.1648	0.7483	0.0495
c = 2.5	0.7586	0.0326	0.7103	0.4106	0.7759	0.1378	0.7448	0.0605
c = 3.0	0.7552	0.0596	0.7552	0.2678	0.7552	0.0596	0.7483	0.0655
c = 3.5	0.7552	0.0596	0.6931	0.4656	0.7552	0.0596	0.7483	0.0816
c = 4.0	0.7586	0.0646	0.7103	0.4106	0.7586	0.0646	0.7517	0.0866
c = 4.5	0.7586	0.0646	0.7138	0.3996	0.7586	0.0646	0.7517	0.0866
c = 5.0	0.7552	0.0756	0.7103	0.4106	0.7552	0.0756	0.7483	0.0816

data generally has the best trade-off between accuracy and discrimination under disparate impact. The fairness constraint on labeled data achieves the trade-off between accuracy and discrimination under disparate mistreatment. 3) More unlabeled data generally helps to make a better compromise between accuracy and discrimination. 4) Model choice can affect the trade-off between accuracy and discrimination. Our experiments show that FGNN is more friendly to achieve a better trade-off than FSMC.

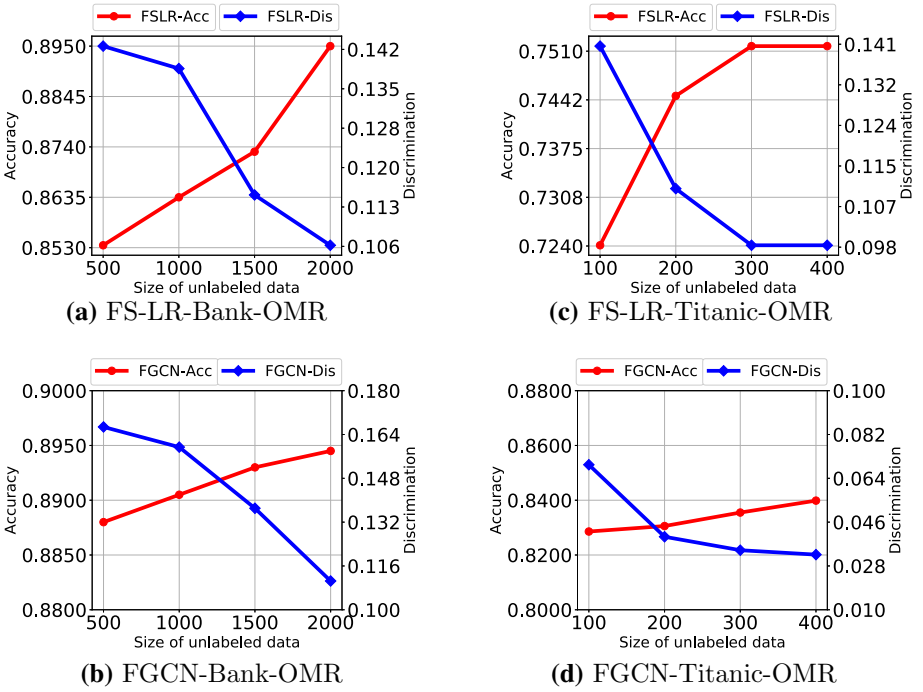


Fig. 7 The impact of the amount of unlabeled data in the training set on accuracy (Red) and discrimination level (Blue) under the fairness metric of overall mistreatment rate with FS-LR and FGCN in two datasets. The X-axis is the size of unlabeled dataset; left y-axis is accuracy; and right y-axis is discrimination level

6 Related work

6.1 Fair supervised learning

Methods for fair supervised learning include pre-processing, in-processing and post-processing methods. In pre-processing, discrimination is eliminated by guiding the distribution of training data towards a fairer direction [29] or by transforming the training data into a new space [14, 32–34]. Subsequent studies extended fair representations into more fairness metrics and more generalized tasks [35–38]. The main advantage of the pre-processing method is that it does not require changes to the machine learning algorithm, so it is very simple to use.

In in-processing, discrimination is constrained by fair constraints or regularizers during the training phase. For example, Kamishima et al. [39] used regularizer term to penalize discrimination in the learning objective. Konstantinov et al. designed fairness regularizers during training can greatly improve the fairness of rankings [40]. [6, 24, 41] designed the convex fairness constraint, called decision boundary covariance to achieve fair classification for classifiers. Some work presented the constrained optimization problem as a two-player game, and formalized the definition of fairness as a linear inequality [42–45]. This is more flexible for optimizing different fair constraints, and solutions using this method are considered to be the most robust. Recent work extended in-processing methods to more complex cases [46–48]. For example, Perrone et al. proposed a general constrained Bayesian optimiza-

tion framework to optimize the model performance [47]. Chikahara et al. studied individual fairness with path-specific causal-effect constraint [48].

A third approach to achieving fairness is post-processing, where a learned classifier is modified to adjust the decisions to be non-discriminatory for different groups [13, 49, 50]. Post-processing does not need changes in the classifier, but it cannot guarantee a optimal classifier. Awasthi et al. further studied equalized odds post-processing method with a perturbed attribute [51]. Putzel et al. worked on the predictions of a blackbox machine learning classifier in order to achieve fairness in a multiclass setting [52].

6.2 Fair unsupervised learning

Chierichetti et al. [15] was the first to study fairness in clustering problems. Their solution, under both k-center and k-median objectives, was required every group to be (approximately) equally represented in each cluster. Many subsequent works have since been undertaken on the subject of fair clustering. Among these, Rosner et al. [18] extended fair clustering to more than two groups. Schmidt et al. [53] consider the fair k-means problem in the streaming model, define fair coresets and show how to compute them in a streaming setting, resulting in significant reduction in the input size. Bera et al. [54] presented a more generalized approach to fair clustering, providing a tunable notion of fairness in clustering. Li et al. [55] defined a new fairness metric in clustering and incorporated group fairness into the algorithmic centroid clustering problem.

6.3 Comparing with other work

Existing fair learning methods mainly focus on supervised and unsupervised learning, and cannot be directly applied to SSL. As far as we know, only [28, 30, 31] has explored fairness in SSL. Chzhen et al. [28] studied Bayes classifier under the fairness metric of equal opportunity, where labeled data is used to learn the output conditional probability, and unlabeled data is used for to calibrate threshold in the post-processing phase. However, unlabeled data is not fully used to eliminate discrimination, and the proposed method only applies in equal opportunity. In [30], the proposed method is built on neural networks for SSL in the in-processing phase, where unlabeled data is marked labels with pseudo labeling. Zhang et al. [31] proposed a pre-processing framework which includes pseudo labeling, re-sampling and ensemble learning to remove discrimination. Our solution will focus on margin-based classifier in the in-processing stage, as in-processing methods have demonstrated good flexibility in both balancing fairness and supporting multiple classifiers and fairness metrics. A few studies have studied fair graph learning. For example, Rahman et al. studied how to learn fair node representations [56], while we focus on fair graph-based SSL. Kang et al. studied individual fairness on graph mining [57], while we focus on group fairness on graph-based SSL.

7 Conclusion

In this paper, we study how to improve the trade-off between fairness and accuracy with unlabeled data. We propose two methods of fair graph-based SSL that operates during in-processing phase. Our first method is formulated as an optimization problem with the goal of finding weights and labeling unlabeled data by minimizing the loss function subject to fairness

constraints. We analyze several different cases of fairness constraints for their effects on the optimization problem plus the accuracy and discrimination level in the results. The second method is built on GNN models with fairness regularizers that ensures fair representations of nodes with labeled and unlabeled data can be learned. Our experiments confirm this analysis, showing that the proposed framework provides accuracy and fairness at high levels in semi-supervised settings.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions

Declarations

Conflict of interest The authors declare that there are no conflicts of interest regarding the publication of this paper

References

1. Abid A, Farooqi M, Zou J (2021) Persistent anti-muslim bias in large language models. arXiv preprint arXiv:2101.05783
2. Vigdor N (2019) Apple card investigated after gender discrimination complaints. The New York Times
3. Suresh H, Guttag JV (2019) A framework for understanding unintended consequences of machine learning. arXiv preprint arXiv:1901.10002
4. Mehrabi N, Morstatter F, Saxena N, Lerman K, Galstyan A (2021) A survey on bias and fairness in machine learning. ACM Comput Surv (CSUR) 54(6):1–35
5. Chouldechova A (2017) Fair prediction with disparate impact: a study of bias in recidivism prediction instruments. Big Data 5(2):153–163
6. Zafar MB, Valera I, Rodriguez MG, Gummadi KP (2017) Fairness constraints: mechanisms for fair classification. In: Proceedings of the 20th international conference on artificial intelligence and statistics, vol 54, pp 962–970
7. Dwork C, Hardt M, Pitassi T, Reingold O, Zemel R (2012) Fairness through awareness. In: Proceedings of the 3rd innovations in theoretical computer science conference, pp 214–226. ACM
8. Jung C, Kearns M, Neel S, Roth A, Stapleton L, Wu ZS (2019) Eliciting and enforcing subjective individual fairness. arXiv preprint arXiv:1905.10660
9. Zhu T, Yu PS (2019) Applying differential privacy mechanism in artificial intelligence. In: 2019 IEEE 39th international conference on distributed computing systems (ICDCS), pp 1601–1609
10. Dwork C, Ilvento C, Jagadeesan M (2020) Individual fairness in pipelines. arXiv preprint arXiv:2004.05167
11. Kusner MJ, Loftus J, Russell C, Silva R (2017) Counterfactual fairness. Adv Neural Inf Process Syst 30:4066–4076
12. Wu Y, Zhang L, Wu X, Tong H (2019) Pc-fairness: a unified framework for measuring causality-based fairness. Adv Neural Inf Process Syst 32:3404–3414
13. Hardt M, Price E, Srebro N et al (2016) Equality of opportunity in supervised learning. Adv Neural Inf Process Syst 29:3315–3323
14. Song J, Kalluri P, Grover A, Zhao S, Ermon S (2019) Learning controllable fair representations. In: Proceedings of the 22nd international conference on artificial intelligence and statistics (AISTATS) 2019, vol 89, pp 2164–2173
15. Chierichetti F, Kumar R, Lattanzi S, Vassilvitskii S (2017) Fair clustering through fairlets. Adv Neural Inf Process Syst 30:5029–5037

16. Backurs A, Indyk P, Onak K, Schieber B, Vakilian A, Wagner T (2019) Scalable fair clustering. arXiv preprint arXiv:1902.03519
17. Chen X, Fain B, Lyu C, Munagala K (2019) Proportionally fair clustering. In: ICML
18. Rösner C, Schmidt M (2018) Privacy preserving clustering with constraints. In: 45th international colloquium on automata, languages, and programming (ICALP 2018), vol 107, pp 96–19614
19. Chen I, Johansson FD, Sontag D (2018) Why is my classifier discriminatory? Adv Neural Inf Process Syst 31:3539–3550
20. Zhu X, Ghahramani Z (2002) Learning from labeled and unlabeled data with label propagation
21. Wang F, Zhang C (2007) Label propagation through linear neighborhoods. IEEE Trans Knowl Data Eng 20(1):55–67
22. Von Luxburg U (2007) A tutorial on spectral clustering. Stat Comput 17(4):395–416
23. Wu Z, Pan S, Chen F, Long G, Zhang C, Philip SY (2020) A comprehensive survey on graph neural networks. IEEE Trans Neural Netw Learn Syst 32(1):4–24
24. Zafar MB, Valera I, Gomez Rodriguez M, Gummadi KP (2017) Fairness beyond disparate treatment & disparate impact: learning classification without disparate mistreatment. In: Proceedings of the 26th international conference on world wide web, pp 1171–1180
25. Agarwal A, Beygelzimer A, Dudík M, Langford J, Wallach H (2018) A reductions approach to fair classification. arXiv preprint arXiv:1803.02453
26. Kipf TN, Welling M (2017) Semi-supervised classification with graph convolutional networks. In: International conference on learning representations (ICLR)
27. Shen X, Diamond S, Gu Y, Boyd S (2016) Disciplined convex-concave programming. In: 2016 IEEE 55th conference on decision and control (CDC), pp 1009–1014. IEEE
28. Chzhen E, Denis C, Hebiri M, Oneto L, Pontil M (2019) Leveraging labeled and unlabeled data for consistent fair binary classification. Adv Neural Inf Process Syst 32:12739–12750
29. Kamiran F, Calders T (2012) Data preprocessing techniques for classification without discrimination. Knowl Inf Syst 33(1):1–33
30. Noroozi V, Bahaadini S, Sheikhi S, Mojab N, Yu PS (2019) Leveraging semi-supervised learning for fairness using neural networks. arXiv preprint arXiv:1912.13230
31. Zhang T, Zhu T, Li J, Han M, Zhou W, Yu P (2020) Fairness in semi-supervised learning: unlabeled data help to reduce discrimination. IEEE Trans Knowl Data Eng 34(4):1763–1774
32. Ruoss A, Balunovic M, Fischer M, Vechev M (2020) Learning certified individually fair representations. Adv Neural Inf Process Syst 33:7584–7596
33. Feng R, Yang Y, Lyu Y, Tan C, Sun Y, Wang C (2019) Learning fair representations via an adversarial framework. arXiv preprint arXiv:1904.13341
34. Zhao H, Gordon G (2019) Inherent tradeoffs in learning fair representations. Adv Neural Inf Process Syst 32:15675–15685
35. Ruoss A, Balunovic M, Fischer M, Vechev M (2020) Learning certified individually fair representations. Adv Neural Inf Process Syst 33:7584–7596
36. Gitiaux X, Rangwala H (2021) Learning smooth and fair representations. In: International conference on artificial intelligence and statistics, pp 253–261. PMLR
37. Shen X, Wong Y, Kankanhalli M (2022) Fair representation: guaranteeing approximate multiple group fairness for unknown tasks. IEEE Transa Pattern Anal Mach Intell. https://doi.org/10.1109/TPAMI.2022.3148905
38. Ma J, Guo R, Wan M, Yang L, Zhang A, Li J (2022) Learning fair node representations with graph counterfactual fairness. arXiv preprint arXiv:2201.03662
39. Kamishima T, Akaho S, Asoh H, Sakuma J (2012) Fairness-aware classifier with prejudice remover regularizer. In: Joint European conference on machine learning and knowledge discovery in databases, Springer, pp 35–50
40. Konstantinov N, Lampert CH (2021) Fairness through regularization for learning to rank. arXiv preprint arXiv:2102.05996
41. Goh G, Cotter A, Gupta M, Friedlander MP (2016) Satisfying real-world goals with dataset constraints. Adv Neural Inf Process Syst 29:2415–2423
42. Donini M, Oneto L, Ben-David S, Shawe-Taylor JS, Pontil M (2018) Empirical risk minimization under fairness constraints. Adv Neural Inf Process Syst 31:2791–2801
43. Agarwal A, Beygelzimer A, Dudík M, Langford J, Wallach H (2018) A reductions approach to fair classification. arXiv preprint arXiv:1803.02453
44. Cotter A, Jiang H, Wang S, Narayan T, You S, Sridharan K, Gupta MR (2019) Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. J Mach Learn Res 20(172):1–59

45. Mandal D, Deng S, Jana S, Wing J, Hsu DJ (2020) Ensuring fairness beyond the training data. Adv Neural Inf Process Syst 33:18445–18456
46. Liu H, Zhao N, Zhang X, Lin H, Yang L, Xu B, Lin Y, Fan W (2022) Dual constraints and adversarial learning for fair recommenders. Knowl-Based Syst 239:108058
47. Perrone V, Donini M, Zafar MB, Schmucker R, Kenthapadi K, Archambeau C (2021) Fair bayesian optimization. In: Proceedings of the 2021 AAAI/ACM conference on AI, ethics, and society, pp 854–863
48. Chikahara Y, Sakaue S, Fujino A, Kashima H (2021) Learning individually fair classifier with path-specific causal-effect constraint. In: International conference on artificial intelligence and statistics, pp 145–153 . PMLR
49. Kim MP, Ghorbani A, Zou J (2019) Multiaccuracy: Black-box post-processing for fairness in classification. In: Proceedings of the 2019 AAAI/ACM conference on AI, ethics, and society, pp 247–254
50. Lohia PK, Ramamurthy KN, Bhide M, Saha D, Varshney KR, Puri R (2019) Bias mitigation post-processing for individual and group fairness. In: Icassp 2019-2019 IEEE international conference on acoustics, speech and signal processing (icassp), pp 2847–2851 . IEEE
51. Awasthi P, Kleindessner M, Morgenstern J (2020) Equalized odds postprocessing under imperfect group information. In: International conference on artificial intelligence and statistics, pp 1770–1780 . PMLR
52. Putzel P, Lee S (2022) Blackbox post-processing for multiclass fairness. arXiv preprint arXiv:2201.04461
53. Schmidt M, Schwiegelshohn C, Sohler C (2018) Fair coresets and streaming algorithms for fair k-means clustering. CoRR **abs/1812.10854**
54. Bera S, Chakrabarty D, Flores N, Negahbani M (2019) Fair algorithms for clustering. Adv Neural Inf Process Syst 32:4955–4966
55. Li B, Li L, Sun A, Wang C, Wang Y (2021) Approximate group fairness for clustering. In: International conference on machine learning, pp 6381–6391 . PMLR
56. Rahman T, Surma B, Backes M, Zhang Y (2019) Fairwalk: Towards fair graph embedding
57. Kang J, He J, Maciejewski R, Tong H (2020) Inform: individual fairness on graph mining. In: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, pp 379–389

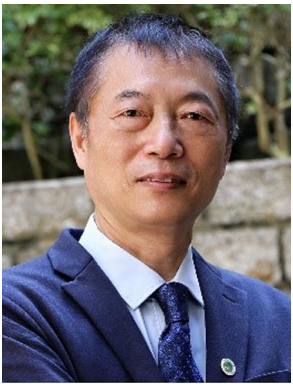
Tao Zhang works towards his Ph.D degree with the school of Computer Science in the University of Technology Sydney, Australia. His research interests include privacy preserving, algorithmic fairness, and machine learning.

Tianqing Zhu received the B.Eng. degree in chemistry and M.Eng. degree in automation from Wuhan University, Wuhan, China, in 2000 and 2004, respectively, and the Ph.D. degree in computer science from Deakin University, Geelong, Australia, in 2014. She is currently a Associate Professor in the Faulty of Engineering and Information Technology with the School of Computer Science, University of Technology Sydney, Sydney, Australia. Before that, she was a Lecturer in the School of Information Technology, Deakin University, from 2014 to 2018. Her research interests include privacy preserving, data mining, and network security.

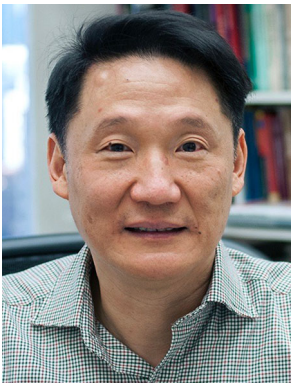
Mengde Han is a PhD student at University of Technology Sydney with a focus on Local Differential Privacy. He completed his Master's at the Johns Hopkins University.

Fengwen Chen received the B.S. degree in computer science (software engineering) from Arizona State University, Tempe, AZ, USA. He is currently pursuing the Ph.D. degree in computer science with the University of Technology Sydney (UTS), Ultimo, NSW, Australia. His research concentrates on data mining and deep learning on graphs.

Jing Li received the B.Eng and M.Eng degrees in computer science and technology from Northwestern Polytechnical University, Xi'an, China, in 2015 and 2018, respectively. Currently, he is pursuing the Ph.D degree with the Centre for Artificial Intelligence in the University of Technology Sydney, Australia. His research interests include machine learning and privacy preserving.



Wanlei Zhou is currently the Vice Rector (Academic Affairs) and Dean of Institute of Data Science, City University of Macau, Macao SAR, China. He received the B.Eng and M.Eng degrees from Harbin Institute of Technology, Harbin, China in 1982 and 1984, respectively, and the PhD degree from The Australian National University, Canberra, Australia, in 1991, all in Computer Science and Engineering. He also received a DSc degree (a higher Doctorate degree) from Deakin University in 2002. Before joining City University of Macau, Professor Zhou held various positions including the Head of School of Computer Science in University of Technology Sydney, Australia, the Alfred Deakin Professor, Chair of Information Technology, Associate Dean, and Head of School of Information Technology in Deakin University, Australia. Professor Zhou also served as a lecturer in University of Electronic Science and Technology of China, a system programmer in HP at Massachusetts, USA; a lecturer in Monash University, Melbourne, Australia; and a lecturer in National University of Singapore, Singapore. His main research interests include security, privacy, and distributed computing. Professor Zhou has published more than 400 papers in refereed international journals and refereed international conferences proceedings, including many articles in IEEE transactions and journals.



Philip S Yu received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA, and the M.B.A. degree from New York University, New York, NY, USA. He was with IBM, Armonk, NY, USA, where he was a Manager of the Software Tools and Techniques Department with the Thomas J. Watson Research Center. He is a Distinguished Professor of computer science with the University of Illinois at Chicago, Chicago, IL, USA, where he also holds the Wexler Chair in information technology. He has published over 1200 papers in peer-reviewed journals, such as the IEEE Transactions on Knowledge and Data Engineering, ACM Transactions on Knowledge Discovery from Data, VLDBJ, and the ACM Transactions on Intelligent Systems and Technology and conferences, such as KDD, ICDE, WWW, AAAI, SIGIR, ICML, and CIKM. He holds or has applied for over 300 U.S. patents. His current research interests include data mining, data streams, databases, and privacy. Dr. Yu was a recipient of the ACM SIGKDD 2016 Innovation Award for his influential research and scientific contributions on mining, fusion, and anonymization of Big Data and the IEEE Computer Society 2013 Technical Achievement Award. He was the Editor-in-Chief of ACM Transactions on Knowledge Discovery from Data (2011–2017) and IEEE Transactions on Knowledge and Data Engineering (2001–2004). He is a fellow of ACM.