UNIVERSITY OF SOUTHAMPTON
Faculty of Engineering and Physical Sciences
Electronics and Computer Science

Group vs. Individual Algorithmic Fairness

Wanying Zhou

Supervised by Doctor Enrico Gerding, Professor Frank McGroarty and Doctor Ramin Okhrati

25th April 2022

Abstract

Machine learning algorithms are increasingly used in making people's life decisions across a range of different areas such as loan applications, university admissions, insurance pricing and criminal justice sentencing. If the historical data used to train the algorithm is biased against certain demographic groups (e.g. black people or women), the predictive results of the algorithms will too. From both regulation and ethical perspectives, we need to reduce discrimination and improve group fairness which concentrates on equalizing the outcomes across distinct groups. However, there are cases where the outcomes are unfair from an individual's point of view when group fairness is satisfied. Individual fairness states that similar individuals should be treated similarly. It is also an important concept of fairness and needs to be considered carefully while we improve group fairness, but it has not yet received much attention in the literature. Most of the existing fairness algorithms concentrates on achieving group fairness disregarding individual fairness.

It is important to explore the relationship between group fairness and individual fairness, specifically, in what cases the individual fairness would be affected when we improve group fairness. We show by practical results from real data sets that, after removing the sensitive attributes, there generally exists a trade off between group fairness and individual fairness. Moreover, we use experimental results from simulated data sets to show that satisfying group fairness decreases the level of individual fairness when the Wasserstein distance (which is a measure of the distance between two distributions) between the attribute distributions of two groups is large. By adjusting the parameters of the simulated distributions, we show that, if a large Wasserstein distance is caused by a large mean difference rather than a large variance difference, individual fairness is more likely to be affected when group fairness is satisfied.

Furthermore, we not only tweak the existing reweighing algorithm to obtain more flexible performance on individual fairness and group fairness, but also construct a new algorithm to achieve fairness. This approach reduces the mean difference in attribute values between different groups so that the association between the sensitive attribute and non-sensitive attributes is decreased. This method can be used to achieve fairness among more than two demographic groups and solve fairness problems in multi-classification or regression scenarios. We assess the performance of this method in terms of both group fairness and individual fairness and the results show that our method outperforms two existing fairness algorithms: reweighing and reject option based classification.

Acknowledgements

I would like to thank Dr. Enrico Gerding, Prof. Frank McGroarty and Dr. Ramin Okhrati for their guidance and constructive feedback during the completion of this work.

Table of Contents

1. Introduction

Nowadays, increasingly machine learning techniques are replacing humans and they are widely used in decision-making systems in many areas such as credit scoring, insurance rates, job applications, criminal justice and so on [1, 2, 3]. It is commonly believed that automated algorithms improve objectivity since decisions made by humans are highly subjective [4]. For instance, bank officers may prefer to issue loans to white applicants than non-white applicants and make decisions depending on their mood. Moreover, two officers may make different decisions on the same candidate which means that there is a lack of consistency. However, machine learning can also lead to unfairness and there are some potential causes [5]. The most prevalent cause is that the algorithm inherits past biases from the historical data on which it is built. If we train an algorithm on data where the outcomes are biased against a demographic group, the algorithm is highly likely to predict outcomes that are biased against that demographic group. Missing data and selection bias can also cause discrimination [6]. The problem of missing values leads to less informative data since people are reluctant to provide information that are disadvantageous to them. Selection biases occurs when the number of individuals selected from each demographic group is very different during the data collection process and would cause the data being less representative of the population. Training algorithms on less informative or representative data can lead to biases. Thus, automated algorithms are not always as objective as we expect and the growing use of automated algorithms has raised social and ethical concerns [7, 8].

Now, there are two types of discrimination in the legal domain: *direct discrimination* and *indirect discrimination* [9]. Direct discrimination, also known as disparate treatment, applies when you treat someone less favourably because the person belongs to one of the protected groups. In machine learning, it occurs if the decisions made by the algorithm are based on the sensitive attributes including gender, race and sexual orientation, also known as protected characteristics [10]. There are laws in many countries that prohibit unfair treatment such as the Civil Rights Act of 1964 in the US which ended racial discrimination in public places and the Equality Act 2010 in the UK which legally protects people from discrimination in the workplace [11]. Indirect discrimination, also referred to as disparate impact, is present if the system produces outcomes that disproportionately hurt people with certain sensitive attribute values compared to other people. The 80% rule advocated by the US Equal Employment Opportunity Commission (EEOC) states that disparate impact is admitted when a selection rate for any group is less than 80% of that for the group with the highest rate [12].

If we simply eliminate the sensitive attribute, the algorithm does not make use of the sensitive attribute so that direct discrimination is avoided. However, this is insufficient for removing indirect discrimination as the other attributes may be highly correlated to the sensitive attribute so that they carry the sensitive information [13, 14]. For instance, in credit scoring, although the attribute race is not explicitly used, race still has an indirect influence on the outcome via a strongly linked attribute, namely postcode. One might suggest to remove both race and postcode so that most influence of race can be removed but, in this case, the information that postcode contains which is independent of race is also removed. This method is not recommended since losing more information may reduce the accuracy of the predictive model. The literature identifies three different types of approaches: pre-processing, in-processing and post-processing [15]. Pre-processing techniques modify the biased data to remove discrimination before we feed it into a machine learning algorithm. In-processing involves tweaking a specific algorithm during the training time such as adding a regularizer to restrict its behavior. Post-processing methods adjust some of the decisions which are obtained from the predictive model. Each of them has its own advantages and disadvantages, and the best approach varies depending on the application. For instance, pre-processing may be selected because of its flexibility since it can be used with any algorithm unlike in-processing [16].

If both direct discrimination and indirect discrimination are removed, different demographic groups are being treated more equally. The fairness that we achieve here is group fairness, which concentrates on equalizing the outcomes across distinct groups. There are many definitions of group fairness in machine learning and we introduce them in Chapter 2. However, there are situations where group fairness is satisfied but the outcomes may be unfair from an individual's point of view. Consider the following example. Suppose there are 50 male and 50 female applicants for a university course and each individual has a score. The university admission team makes decisions on whether to accept or reject an individual based on the score only and they accept 10 male and 10 female applicants. Also, we suppose that the 10$^{\text{th}}$ highest score of male applicants is 90 and the 10$^{\text{th}}$ highest score of female applicants is 85. In this situation, as the proportion of female accepted is equal to the proportion of male accepted, statistical parity — a typical notion of group fairness — is satisfied. However, a female applicant with score 85 will be accepted but a male applicant with the same score will be rejected. Although group fairness is satisfied in this example, the problem of individual unfairness occurs since two individuals with the same score obtain different outcomes. In fact, individual fairness is also an important concept of fairness and needs to be monitored carefully while we improve group fairness.

Individual fairness requires similar individuals to be treated similarly [17]. Similarity is evaluated by the distance between individuals which is measured by a distance metric based on their attributes values. The shorter the distance is, the more similar the two individuals are. However, two individuals can be similar in one domain but different in another. For example, if we consider two individuals with exactly the same features but different A-level grades, they would be considered as similar individuals when a bank makes decisions on whether to issue a credit card, but are considered to be different in the university admission domain. Therefore, defining the distance metric is a challenge since we require different distance metric in different situations.

Some prominent definitions and measures of group and individual fairness are introduced in Chapter 2. One of our interests is to study the relationship between group fairness and individual fairness, specifically, in what cases there would exist a trade-off between them. Most of the existing fairness algorithms on fairness has concentrated on group fairness only without considering individual fairness. One goal is to build an algorithm so that we improve group fairness and make fairer decisions without sacrificing too much accuracy or individual fairness. There are some other shortcomings of existing algorithms. One is that they have been focusing on achieving group fairness across only two groups such as *male* and *female*, or *white* and *non-white.* There are not only two distinct groups in many real-world problems, so we need to consider group fairness across more than two groups. Another one is that they solve only binary classification problems where the outcome is either positive or negative such as whether or not to issue a loan. However, fairness is also essential in many multi-class classification or regression problems. For instance, we want to achieve fairness in the cases where an individual is predicted with a continuous score or is classified into one of the three risk levels: low, medium or high.

1.1. Research Challenges

There is extensive literature on algorithmic fairness over the past decade, yet some challenges still need to be addressed. Specifically:

1. One of the challenges is to explore the relationship between group fairness and individual fairness, specifically, in what cases the individual fairness would be affected when we improve group fairness. Satisfying group fairness has been the goal of most existing algorithms whereas individual fairness has not received much attention. It is challenging to build algorithms that consider both group fairness and individual fairness.

2. Most fairness definitions and algorithms deal with only one binary sensitive attribute and consider only two demographic groups. For example, when race is the sensitive attribute, we often divide the individuals into white people and non-white people since most algorithms can only deal with two groups. When we consider race and sex at the same time, there are four groups in total: black females, black males, white females and white males. The discrimination problem becomes more complicated and difficult when we handle multiple attributes or a sensitive attribute with multiple values [18].

3. To date, most existing algorithms to achieve fairness have focused on only binary classification problems. A challenge is to develop algorithms that can be used to solve multi-class classification problems and even regression problems. Also, we need to carefully define fairness in multi-class classification and regression problems as they are slightly different from binary classification. In regression, one might measure fairness by the magnitude of the mean difference in the continuous outcome between distinct groups.

4. Similarity is measured by a distance metric, but defining distance metric is a challenge. We can generate a principled approach that allows us to set the appropriate distance measures for any domain. For example, we can explore the relevance of each feature in a specific situation and assign weights to features based on their relevance. The features with higher relevance can be assigned with higher weights and have a larger influence on the distance between two individuals.

1.2. Research Contributions

We have addressed several challenges from Section 1.1 in this report.

- For the first time, we show by some practical results that when we want to improve group fairness further after removing the sensitive attributes, there generally exists a trade off between group fairness and individual fairness. Although simply eliminating the sensitive attribute has been criticised for incompletely removing sensitive information, we point out that in general this method maximises individual fairness. This partially addresses Challenge 1.

- Wasserstein distance, also known as Earthmover distance is a measure of the distance between two probability distributions. It is shown that individual fairness implies group fairness when the Wasserstein distance between two demographic groups' attribute distributions is small (i.e. the attribute follows similar distributions for the two groups) [19]. In this report, we show from practical results that individual fairness can be worsened when we try to achieve group fairness if the Wasserstein distance is large. There are different causes which lead to large Wasserstein distance such as large mean difference between the two distributions or large variance difference. A new result is that the influence on individual fairness can be affected by the causes of large Wasserstein distance. This partially addresses Challenge 1 as well.

- We discover ways of slightly altering the reweighing algorithm to change the weights assigned to each individual so that we can change group and individual fairness performance according to specific requirements. Also, we extend reweighing algorithm and show that it can be applied to fairness problems with more than two demographic groups. This partially addresses Challenge 1 and 2.

- We propose a new pre-processing approach: we adjust the feature values to reduce the gap between the mean of the groups so that we lower the correlation coefficient between the sensitive attribute and other attributes, and we monitor individual fairness while improving group fairness. The performance results show that it outperforms existing approaches. This method allows a sensitive attribute with multiple levels or multiple sensitive attributes and can be applied to not only binary classification but also multi-class classification and regression problems. This contribution partially addresses Challenges 2 and 3.

1.3. Report Outline

The outline of the report is as follows. Chapter 2 introduces the background of algorithm fairness and gives a literature review. We present some prominent definitions of both group and individual fairness measures, theory of existing algorithms and the definition of Wasserstein distance. In Chapter 3, we discover the relationship between group fairness and individual fairness, then analyse the influence of Wasserstein distance on individual fairness when satisfying group fairness. In Chapter 4, we extend the reweighing algorithm, describe a new pre-processing approach in detail and present its performance based on real data. Finally, we conclude the report in Chapter 5.

2. Background

Since algorithmic fairness started to attract our attention, many notions of fairness have been proposed in the literature, yet there is no clear agreement on the most appropriate way to define it. In the first section, we provide the most prominent definitions and measures of group fairness. We also detail the most common group fairness algorithms that have been used in existing literature and their mathematical descriptions. The advantages and limitations of each algorithm are analysed. Next, we introduce the definition of individual fairness and algorithms the aim to achieve individual fairness. Then, we introduce Wasserstein distance, an prominent statistical distance measure, since we need to discover the influence that applying group fairness algorithms has on individual fairness when the distance between attributes in two demographic groups are different. Finally, we summarise the state of art and main gaps from the literature.

2.1. Group Fairness

In Section 2.1.1, we start with introducing the most common definitions of group fairness in the basic setting of a binary classification problem with a binary sensitive attribute. Then, we extend them to more complicated settings. Next, in Section 2.1.2, we describe the most prominent group fairness algorithms.

2.1.1. Group fairness definitions

The first concept of fairness we introduce is demographic parity, also known as statistical parity [20]. It is satisfied if a decision is independent of the protected attribute. That is, the proportion of individuals in any group receiving a positive outcome is equal to the proportion of the population as a whole [21]. For a binary sensitive attribute $S \in \{0, 1\}$, the privileged group is the set of all individuals for which $S = 1$ and the unprivileged group is the set of all individuals for which $S = 0$. In binary classification $\hat{Y} \in \{0, 1\}$ with a single binary sensitive attribute $S \in \{0, 1\}$, statistical parity can be formulated mathematically as $P(\hat{Y} = 1) = P(\hat{Y} = 1|S = 0) = P(\hat{Y} = 1|S = 1)$, where $\hat{Y} = 1$ if the individual is predicted to have a positive outcome and $\hat{Y} = 0$ if the individual is predicted to have a negative outcome. To measure it, we can use statistical parity difference, i.e.:

$$P(\hat{Y} = 1|S = 0) - P(\hat{Y} = 1|S = 1). \tag{2.1}$$

Here, a value close to 0 means that statistical parity is satisfied. Another measure is the fraction $\frac{P(\hat{Y}=1|S=0)}{P(\hat{Y}=1|S=1)}$ where its value is close to 1 if statistical parity is satisfied. However, this definition ignores any possible correlation between the outcome Y and the sensitive attribute S. In particular, a perfect classifier (i.e. $\hat{Y} = Y$) does not even ensure statistical parity when base rates are different (i.e. $P(Y = 1|S = 1) \neq P(Y = 1|S = 0)$).

The second definition we introduce is equalized odds. Unlike statistical parity, it allows the decisions to depend on the sensitive attribute but only through the actual outcome [22]. It states that the decision $\hat{Y}$ and the sensitive attributes are independent conditional on the real outcome Y. In binary classification with a single binary attribute, it requires $P(\hat{Y} = 1|Y = y, S = 0) = P(\hat{Y} = 1|Y = y, S = 1)$ for $y = 0, 1$. This can be interpreted as equalizing false positive rates across the groups and true positive rates across the groups. We can use average odds difference to measure it. Formally, $\frac{1}{2}\Big(\big[P(\hat{Y} = 1|Y = 1, S = 0) - P(\hat{Y} = 1|Y = 1, S = 1)\big] + \big[P(\hat{Y} = 1|Y = 0, S = 0) - P(\hat{Y} = 1|Y = 0, S = 1)\big]\Big)$ where a value close to 0 means that equalized odds is satisfied.

The third one is called equal opportunity which is a relaxation of equal odds as it only requires true positive rates of each group to be equal. Formally: $P(\hat{Y} = 1|Y = 1, S = 0) = P(\hat{Y} = 1|Y = 1, S = 1)$. We can evaluate it by equal opportunity difference $P(\hat{Y} = 1|Y = 1, S = 0) - P(\hat{Y} = 1|Y = 1, S = 1)$, a value close to 0 indicates that we satisfy equal opportunity.

There are other common notions of fairness such as fairness through unawareness, calibration and predictive parity that have been proposed, which we introduce now. First of all, fairness through unawareness is a naive definition which states that an algorithm is fair if no sensitive attributes are explicitly used in the decision-making process [23]. It was proposed as a baseline and simply requires us to ignore all the sensitive attributes when training the algorithms. Despite its simplicity, this definition has a clear weakness as other existing attributes can be highly correlated to the sensitive attribute and still contain the discriminatory information. In such a situation, fairness through unawareness is ineffective as the algorithm that uses high-correlated attributes will indirectly discriminate [24]. The second one is calibration which is defined such that, for a set of individuals whose predicted probability of being positive is p, we expect a p fraction of them to have a positive outcome [25]. *Calibration within groups* is satisfied if calibration holds simultaneously within each demographic group. Thirdly, predictive parity is satisfied if positive predictive values for the groups are equal. In binary classification with a binary sensitive attribute, it is equivalent to $P(Y = 1|\hat{Y} = 1, S = 0) = P(Y = 1|\hat{Y} = 1, S = 1)$ [26].

We have introduced the most common notion of group fairness in the basic setting of binary classification with a binary sensitive attribute. When there are multiple sensitive attributes or a single sensitive attribute is multi-dimensional instead of binary, the population is divided into more than two groups. Then statistical parity means that all the groups have equal probabilities of being predicted positive. For instance, when there are two binary sensitive attributes, it means $P(\hat{Y} = 1|S_1 = s_1, S_2 = s_2)$ is equal for any $s_1, s_2 \in \{0, 1\}$. When there is a single sensitive attribute with three categories, statistical parity is satisfied if $P(\hat{Y} = 1|S = 0) = P(\hat{Y} = 1|S = 1) = P(\hat{Y} = 1|S = 2)$ for $S \in \{0, 1, 2\}$. When there are more than two distinct groups, we often compare the most privileged group with the other groups [24]. For example, when the two binary sensitive attributes are S_1 sex and S_2 race (consider white and black only), there are four distinct groups: white male, black male, white female and black female. Then we compare the most privileged group white male with the other three groups. The idea is similar for equalized odds and equal opportunity, but the only difference is that the condition of the actual outcome is given.

Now we review the fairness concepts used for solving multi-class classification or regression problem instead of binary classification. Regression problems have a continuous outcome and multi-class classification problems have a categorical outcome with more than two categories. Therefore, it would not be appropriate to use equalized odds or equal opportunity since false positive rates and true positive rates are restricted to a binary outcome. Instead, statistical parity can be easily extended to the multi-class classification setting. Specifically, statistical parity is satisfied if the proportion of individuals in any group receiving any type of outcome is equal to the proportion of the population as a whole. In regression, if we suppose that the predicted outcome $\hat{Y}$ is continuous and ranges from 0 to 1, then statistical parity is satisfied if $P(\hat{Y} \geq z|S = s) = P(\hat{Y} \geq z)$ for all $s \in \mathcal{S}$ and $z \in [0, 1]$ [27].

Since statistical parity is the most common group fairness definition and can be adjusted flexibly in more complicated settings, we use it to measure group fairness in this report. In a real-world scenario, we can select the most appropriate measure based on the specific interest or requirement. Another possible approach is to investigate public views on definitions of fairness by survey and opinion polling so that we can understand how people perceive the meaning of fairness [28].

2.1.2. Group Fairness Algorithms

After understanding common definitions of fairness, we now review existing fairness algorithms in this section. The three main approaches to achieve fairness in the literature are pre-processing, in-processing and post-processing. In what follows, we describe the ideas behind each approach with some mathematical rationale and analyse the limitations critically.

Pre-processing indicates that, before learning any classifier on the data, we modify the data first to remove discrimination. An advantage of pre-processing is its flexibility since we can apply any machine learning algorithm after obtaining a fairer data set. The idea is that, if we learn a classifier on fairer training data, it is more likely that the predicted outcomes will be fairer. In the literature, pre-processing principles include changing the outcome labels, assigning different weights to different individuals and finding fairer representations of $\mathbf{X}$ [21, 24, 29, 30, 31]. We now present three typical methods that corresponds to the principles: Massaging, reweighing and disparate impact remover.

Massaging focuses on modifying outcome labels Y in the training data [30]. It changes the labels of some individuals in the privileged group (i.e $S = 1$) from positive to negative and changes the same number of individuals in the unprivileged group from negative to positive so that the modified data satisfies statistical parity, i.e. $P(\tilde{Y} = 1|S = 0) = P(\tilde{Y} = 1|S = 1)$ where $\tilde{Y}$ represents the modified label. The individuals to relabel are selected based on their scores (predicted probabilities of being positive) produced by a ranker. The explicit explanation on how to determine the number of individual to relabel and how to select those individuals is presented in Appendix A.1. This method is straightforward and simple, but can be criticised for being intrusive. In addition, massaging is restricted to only two demographic groups (one privileged and one unprivileged group) and a binary outcome label. Thus, it cannot deal with multiple sensitive attributes or an attribute with multiple values, and cannot be extended to solve multi-class classification or regression problems.

A less intrusive approach is reweighing which assigns different weights to different individuals without modifying labels or attribute values [30]. For instance, individuals with $S = 0$ and $Y = 1$ have higher weights than those with $S = 0$ and $Y = 0$ and individuals with $S = 1$ and $Y = 1$ have lower weights than those with $S = 1$ and $Y = 0$. When we assign weights

$$W_{s,y} = \frac{|S = s| \times |Y = y|}{|D| \times |S = s \cap Y = y|}$$

to each individual (e.g. each positive instance from the unprivileged group is assigned with a weight $W_{0,1}$), the weighted data can achieve statistical parity. The mathematical proof is presented in Appendix A.2. It is easier to be implemented than Massaging since it does not require the process of training a ranker first to obtain scores and select which individuals to relabel. However, the weights $W_{s,y}$ still consider binary classification problems only and cannot be used in multi-class classification or regression.

Disparate impact remover removes the sensitive information about S from the numerical attributes [24]. The main idea is that the data is fair if $\mathbf{X}$ does not contain the information content about S and cannot be used to predict the sensitive S. It modifies the attribute values to make the modified distributions of $\mathbf{X}$ for the two demographic groups closer to each other, then the modified data becomes fairer. The modification level $\lambda \in [0, 1]$ indicates the extent in which we modify the non-sensitive attribute values $\mathbf{X}$, where $\lambda = 1$ indicates that $\mathbf{X}$ is fully modified and $\lambda = 0$ indicates that $\mathbf{X}$ is not modified. The modifying procedures are detailed in Appendix A.3. Disparate impact remover has several advantages compared with the previous two methods. Firstly, it can be used to solve multi-class classification and regression problem since the procedures do not involve the outcome labels. Secondly, it allows multiple sensitive attributes and a sensitive attribute with multiple values. Finally, the modification level can be adjusted depending on how close we make the distributions of $\mathbf{X}$ be. However, a shortcoming is that only numerical attributes can be modified using this method.

We focus on pre-processing approaches in this report, and so we introduce in-processing and post-processing only briefly. In-processing approaches tweak the machine learning algorithm to achieve fairness during the training time, which make them inflexible since they are tightly coupled with the specific algorithm itself [20, 32, 33]. Take prejudice remover as an example [32]. It restricts the learner's behavior by adding a regularization term that penalizes the mutual information between the sensitive attribute S and the outcome label Y to the objective function. Nevertheless, it can only be applied to probabilistic models such as logistic regression. It cannot be applied directly if the sensitive attribute is multivariate or there are multiple sensitive attributes.

Post-processing adjusts some of the outputs which are obtained from the classifier to make the decisions satisfy fairness definitions such as statistical parity, equalized odds or equal opportunity [22, 34, 35]. A drawback is that they are likely to give inferior results due to the fact that they are applied at a late stage [36]. We introduce two methods that focus on statistical parity [35], the first one is called discrimination-aware ensemble (DAE). In this method, there is an ensemble of different classifiers. If all the members predict the same outcome label, we assign the individual with the agreed label. If any of the members produces the opposite labels, we assign individuals belonging to the privileged group with negative labels and assign individuals belonging to the unprivileged group with positive labels. The idea is that instances which are close to the decision boundary are more likely to be misclassified and cause disagreement among the classifiers. The other method is reject option based classification (ROC) which can be applied to probabilistic classifiers only. Each individual has a predicted probability of being positive and we choose a threshold $\theta \in [0.5, 1]$. A positive label is assigned if the predicted probability is greater than θ and a negative label is assigned if the predicted probability is lower than $(1 - \theta)$. If the predicted probability is between $(1 - \theta)$ and θ, we predict the individuals who belong to the unprivileged group as positive and predict those who belong to the privileged group as negative. DAE and ROC are both restricted to a single binary sensitive attribute and binary classification. Also, they damage the individual fairness deliberately since two similar individuals who are close to the decision boundary are treated differently because they belong to different groups.

2.2. Individual Fairness

In the previous section we discussed the most common definitions of group fairness, which require parity of some statistical measure across groups. Now we introduce the most common definitions of individual fairness in the literature. One way to define individual fairness is that if two individuals from different groups have exactly the same attributes (except the sensitive attribute), the classifier should predict the same outcome [26]. To measure it, we first obtain test individuals' predicted outcomes using the classifier, then switch the label sensitive attribute and obtain new predicted outcomes, finally we compute the fraction of test individuals whose predicted outcome remains the same after switching the label of the sensitive attribute. A value of 1 means that this individual fairness definition is satisfied. However, a drawback is that simply removing the sensitive attribute when training an algorithm will give 100% fairness in terms of this definition.

The most common concept of individual fairness is fairness through awareness [19, 37]: similar individuals should have similar classification where similarity is defined by a distance metric. We can formulate it in mathematical expression as follows. Suppose that we have N individuals in a data set with a sensitive attribute S and non-sensitive attributes $\mathbf{X}$, and suppose that the output of an algorithm is $\hat{Y} = (\hat{y}_1, \dots, \hat{y}_N)$. Fix two thresholds ϵ, ϵ', the algorithm is (ϵ, ϵ')-fair if $d(\mathbf{x}_i, \mathbf{x}_j) \leq \epsilon \implies d_O(\hat{y}_i, \hat{y}_j) \leq \epsilon'$ for any $\mathbf{x}_i, \mathbf{x}_j$ where $i, j \in \{1, \dots, N\}$, a distance metric for the input space d and a distance metric for the outcome space d_O. It is assumed that the distance function which measures similarity between individuals is somehow pre-defined, but in fact, it is difficult to be determined [38]. One simple approach is to apply nearest neighbours algorithm to each individual so that we identify individuals that are similar to it. Consistency is a measure used to evaluate individual fairness and it compares the classifier's predicted outcome of each individual to its k-nearest neighbours where the k-nearest neighbours is found based on attributes excluding the sensitive attribute [21]. It is formulated as

$$1 - \frac{1}{N} \sum_{i=1}^{N} |\hat{y}_i - \frac{1}{k} \sum_{j \in \mathrm{kNN}(\mathbf{x}_i)} \hat{y}_j|, \tag{2.2}$$

where $\hat{y}_i, \hat{y}_j \in \{0, 1\}$, N is the number of test individuals and $\mathrm{kNN}(\mathbf{x}_i)$ represents the k-nearest neighbours of the individual $\mathbf{x}_i$. A value close to 1 indicates that the classifier treats similar individuals similarly. Inspired by fairness through awareness, [39] proposed another notion of individual fairness for pre-processing approaches. Individual fairness is what the existing fairness algorithms tend to ignore, but in fact, it should be monitored while we try to achieve group fairness.

In the literature, most fairness algorithms have exclusively concentrated on achieving group fairness such as statistical parity or equalized odds disregarding individual fairness. The most notable work on individual fairness is learning fair representations which formulates fairness as an optimization problem of finding a good representation of the data to compromise classifier accuracy, group fairness and individual fairness [21]. However, it burdens the learning since its objective function tries to find a compromise over all three components. A detailed procedure is in Appendix A.4. A recent approach iFair inspired from it has considered individual fairness but disregarded group fairness[39].

2.3. Wasserstein Distance Metric

Researchers became motivated to discover the relationship between individual fairness and group fairness after it was initially formulated in [19], where it has been shown that individual fairness implies group fairness when the Wasserstein distance between two demographic groups are small. Despite Wasserstein distance (also known as Earthmover distance), there are other statistical distance measures which can be used to measure the distance between two groups' attribute distributions KL-Divergence and Total Variation Distance. In this section, we introduce the most prominent statistical distance measure, Wasserstein distance [40].

Wasserstein distance is a distance function defined between probability distributions on a given metric space. One way to understand Wasserstein distance is to consider the optimal mass transport problem which seeks the most efficient way of transforming one distribution of mass to another relative to a given cost function. Consider two probability density functions $I_0, I_1 : \mathbb{R}^d \to \mathbb{R}$ defined over respective domains Ω_0 and Ω_1 such that $\int_{\Omega_0} I_0(x)dx = \int_{\Omega_1} I_1(y)dy = 1$. Let $\mathcal{J}(I_0, I_1)$ denote all joint distributions J for (X, Y) that have marginals I_0 and I_1. We define the p-Wasserstein distance using the optimal transportation problem with the cost function $||x - y||^p$,

$$W_p(I_0, I_1) = \Big(\inf_{J \in \mathcal{J}(I_0,I_1)} \int ||x-y||^p dJ(x,y)\Big)^{1/p}$$

where $p \geq 1$. When $p = 1$, this is also called the Earthmover distance.

Consider two data matrices

$$\mathbf{X} = \begin{bmatrix} - & X_1 & - \\ - & X_2 & - \\ & \vdots & \\ - & X_n & - \end{bmatrix} = \begin{bmatrix} x_{11} & \dots & x_{1d} \\ x_{21} & \dots & x_{2d} \\ \vdots & & \vdots \\ x_{n1} & \dots & x_{nd} \end{bmatrix} \text{ and } \mathbf{Y} = \begin{bmatrix} - & Y_1 & - \\ - & Y_2 & - \\ & \vdots & \\ - & Y_n & - \end{bmatrix} = \begin{bmatrix} y_{11} & \dots & y_{1d} \\ y_{21} & \dots & y_{2d} \\ \vdots & & \vdots \\ y_{n1} & \dots & y_{nd} \end{bmatrix}.$$

For any d, let I_0 and I_1 be empirical distributions of $\mathbf{X}$ and $\mathbf{Y}$ respectively, then the p- Wasserstein distance is

$$W_p(I_0, I_1) = \inf_\pi \Big(\sum_{i=1}^{n} ||X_i - Y_{\pi(i)}||^p\Big)^{1/p}$$

where infimum is over all permutations π.

When $d = 1$, the one dimensional p-Wasserstein distance is a simple function of ordered statistics:

$$W_p(I_0, I_1) = \Big(\sum_{i=1}^{n} |X_{(i)} - Y_{(i)}|^p\Big)^{1/p}$$

where ordered statistics $X_{(1)} \leq X_{(2)} \leq \cdots \leq X_{(n)}$ and $Y_{(1)} \leq Y_{(2)} \leq \cdots \leq Y_{(n)}$.

When n and d become large, the computation of the Wasserstein distance is computationally too demanding. Therefore, we introduce a method where we obtain a set of one-dimensional representations ($d = 1$) for a higher-dimensional probability distribution through projections, then calculate the distance as a functional on the Wasserstein distance of their one-dimensional representations so that we only solve several one-dimensional optimal transport problems. We use Radon transform $\mathcal{R}$ to transform a function $I : \mathbb{R}^d \to \mathbb{R}$ to $\mathcal{R}I : \mathbb{R} \to \mathbb{R}$ which is defined as

$$\mathcal{R}I(t;\theta) = \int_{\mathbb{R}} I(t\theta + s\theta^{\perp})ds \quad \forall t \in \mathbb{R}, \forall \theta \in \mathbb{S}^{d-1},$$

where $\theta^{\perp}$ is the subspace orthogonal to θ and $\mathbb{S}^{d-1} = \{\theta \in \mathbb{R}^d : ||\theta|| = 1\}$ is the unit sphere in $\mathbb{R}^d$. Radon transform projects a pdf into an infinite set of one-dimensional pdf $\mathcal{R}I(.;\theta)$. The sliced Wasserstein distance for pdf I_0 and I_1 is defined as

$$\mathrm{SW}_p(I_0, I_1) = \Big(\int_{\mathbb{S}^{d-1}} \mathrm{W}_p(\mathcal{R}I_0(.;\theta), \mathcal{R}I_1(.;\theta))^p d\theta\Big)^{1/p},$$

which we use to compute multi-dimensional Wasserstein distance.

2.4. State of the Art and Limitations

In this section, we summarise the state of the art and the main gaps from the literature. There are many prominent fairness definitions that have been defined in Section 2.1.1 and 2.2, but recent studies have shown that some of them are incompatible [41, 42]. For example, it has been shown that, except in the most trivial cases, equalized odds and calibration within groups cannot be satisfied simultaneously [43]. In order to maintain calibration, we have to relax equalized odds conditions e.g. by only requiring equal opportunity. Some real-world cases also show a lack of agreement among the definitions of fairness. For instance, COMPAS is a risk assessment tool which is used to assign a score that predicts a defendant's risk of recidivism. In terms of predictive parity, COMPAS is fair since positive predictive values for white and black people are similar [44, 45]. However, ProPublica analysed COMPAS predictions for 10,000 criminals and claimed that the tool is biased against blacks [46]. The team pointed out that blacks are twice as likely as whites to be predicted as high risk but not actually reoffend, whereas whites are much more likely than black to be predicted as low risk but actually reoffend, so the algorithm is discriminatory by equalized odds. This example is an evidence for incompatibility between predictive parity and equalized odds. Therefore, a main gap from literature is that although researchers have proposed different notions of algorithmic fairness, there is no clear agreement on the most appropriate fairness definition yet since different domains would require different definitions.

Although individual fairness was initially formulated in 2012 in [19], most of the work in the literature has only focused on group fairness where different group fairness measures and algorithms have been introduced [16, 17, 26]. We have introduced prominent group and individual fairness measures and mentioned that it is difficult to determine a similarity metric for individual fairness. One of the challenges is to provide an approach that allows us to set appropriate distance functions for any domain, which corresponds to Challenge 4 in Section 1.1.

Moreover, there has been discussion on the relationship between group fairness and individual fairness in the literature. On one hand, it has been shown theoretically that fairness for individuals implies group fairness (statistical parity) if and only if the Earthmover distance between two groups is small [19]. Thus, when the distribution of features are similar across different groups, group fairness and individual fairness can be satisfied simultaneously. On the other hand, group fairness measures and individual fairness measures appear to conflict if we have to assign similar individuals from different groups with opposite outcomes in order to satisfy group fairness. However, the literature needs more work on theoretical discussion on the conflict or when there would be a trade off between group fairness and individual fairness, which is part of Challenge 1 in Section 1.1. There are many areas which need to be explored such as *(i)* finding out whether there is a trade off when Wasserstein distance is large, *(ii)* the influence on individual fairness when a large Wasserstein distance is caused by different factors, *(iii)* analysing how different algorithms respond to different Wasserstein distance.

Furthermore, in the literature, most fairness algorithms have concentrated on improving group fairness measures such as statistical parity and equalized odds [13, 20, 22, 24, 35]. The most notable work on individual fairness is [21] which formulates fairness as an optimization problem of finding a good representation of the data to compromise classifier accuracy, group fairness and individual fairness. However, a big limitation is that its objective function tries to find a compromise over all three components. We also point out two major limitations of the existing algorithms which are *i)* they only allow one single binary sensitive attribute, which means that they can only be used when there are only two demographic groups. *ii)* they are restricted to a binary classification problem but multi-class classification and regression scenarios also require fairness. These correspond to part of Challenge 1, Challenge 2 and 3 in Section 1.1.

3. Group Fairness vs. Individual Fairness

In this chapter, we explore the relationship between group fairness and individual fairness especially when there exists a trade-off between them. As we mentioned in Section 2.1.2, one of the pre-processing approaches to achieve group fairness is to first eliminate the sensitive attribute, then modify the non-sensitive attributes to obtain a fairer representation. Suppose that simply removing the sensitive attribute will give us a group fairness level G_1 and an individual fairness level I_1, in Section 3.1, we experiment on a real data set *Adult* and observe the influence on individual fairness level I_1 if we modify the non-sensitive attributes to improve the group fairness level G_1 further.

Furthermore, it has been shown that, when the Wasserstein distance between attribute distributions of the two groups is small, individual fairness implies group fairness [19]. In Section 3.2, on a simulated data set, we discover how satisfying group fairness affects individual fairness when the Wasserstein distance between two groups' attribute distributions is large. There are different factors that affect the Wasserstein distance such as mean difference and variance difference between two groups. We also discover how individual fairness is affected when a large Wasserstein distance is only caused by a large mean difference or a large variance difference.

3.1. Trade-off between Group Fairness and Individual Fairness

Suppose that we have N instances which represent N individuals in a data set with a sensitive attribute S and other attributes $\mathbf{X}$: $(S, \mathbf{X}) = \{s_i, \mathbf{x}_i\}_{i=1}^{N}$ where $s_i, \mathbf{x}_i$ are the i^{th} individual's sensitive attribute value and the non-sensitive attribute values respectively. Also, we suppose that the predicted outcome of a binary classification algorithm is $\hat{Y} = (\hat{y}_1, \ldots, \hat{y}_N)$ which takes value 1 or 0. In the case where we train an algorithm f_1 without group fairness consideration and both the sensitive attribute and the other attributes are used, the predicted outcome of the i^{th} individual is $\hat{y}_i = f_1(s_i, \mathbf{x}_i)$. To improve group fairness, we can train an algorithm f_2 after removing the sensitive attribute. If only non-sensitive attributes are used in the algorithm, then $\hat{y}_i = f_2(\mathbf{x}_i)$.

Now that we have introduced the necessary notation, we can proceed with defining individual fairness more precisely. It can be defined in a mathematical form as follows [37]: Fix two thresholds ϵ, ϵ', the algorithm is (ϵ, ϵ')-fair if, for any $\mathbf{x}_i, \mathbf{x}_j$ where $i, j \in \{1, \ldots, N\}$, a distance metric for the input space d and a distance metric for the outcome space d_O,

$$d(\mathbf{x}_i, \mathbf{x}_j) \leq \epsilon \implies d_O(\hat{y}_i, \hat{y}_j) \leq \epsilon'. \tag{3.1}$$

This equation can be interpreted as follows. If individuals are similar in terms of their non-sensitive attributes, their outputs should be close as well. It is natural to think that when the outputs come from an algorithm which was trained using only the original non-sensitive attributes, this definition is the most likely to be satisfied.

Inspired by Definition 3.1, another notion of individual fairness was proposed by [39]. Suppose we apply a pre-processing approach where a mapping ϕ maps the original attributes to modified attributes, then the mapping is individually fair if, for any two individuals and for small ϵ'', we have

$$|d(\phi(s_i, \mathbf{x}_i), \phi(s_j, \mathbf{x}_j)) - d(\mathbf{x}_i, \mathbf{x}_j)| \leq \epsilon''. \tag{3.2}$$

The mapping is individually fair if individuals who are close on their non-sensitive attributes are also close in their modified representations. If Definition 3.2 is satisfied, individuals who are close on their non-sensitive attributes are also close in their modified representations, then outputs of an algorithm which was trained using modified data would be close and Definition 3.1 is satisfied. From Definition 3.2, we notice that, when $d(\phi(s_i, \mathbf{x}_i), \phi(s_j, \mathbf{x}_j))$ is equal to $d(\mathbf{x}_i, \mathbf{x}_j)$, the left hand side equals 0 and individual fairness can always be satisfied even when ϵ'' is set to 0. The simplest way to achieve $d(\phi(s_i, \mathbf{x}_i), \phi(s_j, \mathbf{x}_j)) = d(\mathbf{x}_i, \mathbf{x}_j)$ is to let $\phi(s_i, \mathbf{x}_i) = \mathbf{x}_i$ for all i (i.e. simply remove the sensitive attribute).

We have mentioned earlier that non-sensitive attributes may be highly correlated to the sensitive one and still contain sensitive information. Thus, to improve group fairness further, we can modify the non-sensitive attributes and train an algorithm f_3 on modified non-sensitive attributes only. Then, the output is $\hat{y}_i = f_3(\phi(s_i, \mathbf{x}_i)) = f_3(\tilde{\mathbf{x}}_i)$ where ϕ is the modification function and $\tilde{\mathbf{x}}_i$ represents the modified $\mathbf{x}_i$. The idea behind improving group fairness by modifying the non-sensitive attributes is to make the attribute distributions for the privileged group and the unprivileged group become closer to each other. If we feed an algorithm with data where it is more difficult to recognize which demographic group an individual belongs to from its non-sensitive attributes, group fairness of the algorithm would improve.

Now, we consider the influence on individual fairness if we remove the sensitive attribute and modify the non-sensitive attributes to improve group fairness. From Definition 3.2, we can see that the left hand side becomes $|d(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j) - d(\mathbf{x}_i, \mathbf{x}_j)|$. The two individuals i and j can either belong to different groups or belong to the same group. When they belong to different groups, $d(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j)$ will generally decrease as the distributions for two groups are moved closer to each other. Since $d(\mathbf{x}_i, \mathbf{x}_j)$ is fixed and $d(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j) \leq d(\mathbf{x}_i, \mathbf{x}_j)$, decreasing $d(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j)$ will increase $|d(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j) - d(\mathbf{x}_i, \mathbf{x}_j)|$, which means that individual fairness gets worse. For any two individuals from the same group, the change in the distance between them would be negligible compared to the change in distance between two individuals belonging to different groups. Therefore, after the sensitive attribute has been removed, if the distributions of non-sensitive attributes for the two groups become closer to each other, although group fairness will improve, individual fairness would fall. In fact, if the non-sensitive attributes are modified by too much, not only the individual fairness will get worse, the accuracy of the algorithm will fall as well.

We can illustrate the trade-off between group fairness and individual fairness using a real data set *Adult* [47]. The data set $\mathcal{D} = (S, \mathbf{X}, Y)$ consists of one binary sensitive attribute S: *sex (1: Male; 0: Female)* or *race (1: White; 0: Non-white)*, five non-sensitive attributes $\mathbf{X}$: *age*, *education-num*, *hours-per-week*, *capital-loss* and *capital-gain* and a binary outcome label Y (1: income exceeds 50K/yr; 0: income does not exceed 50K/yr). Then, we apply disparate impact remover with repair levels $\lambda = (0, 0.05, \ldots, 0.95, 1)$ which we have introduced in Section 2.1.2 and obtain a modified data set $\tilde{\mathcal{D}} = (\tilde{\mathbf{X}}, Y)$. Notice that, at repair level $\lambda = 0$, we remove the sensitive attribute only and do not modify the non-sensitive attributes. For each repair level, we compute statistical parity difference defined by Equation 2.1 to measure group fairness and consistency defined by Equation 2.2 to measures individual fairness. The procedure is as follows:

1. Split the modified data $\tilde{\mathcal{D}}$ randomly into a training, validation and test data set in the ratio 7:1.5:1.5, then standardise the data.

2. Train a logistic regression model on the training data.

3. Fit the model on the validation data and obtain each validation individual's score, also known as the predicted probability of having a positive outcome. For a threshold $t \in (0, 1)$, we predict labels $\hat{Y} = 1$ if score $\geq t$ and $\hat{Y} = 0$ if score $\leq t$. For a set of thresholds, we compute the balanced accuracy $\frac{1}{2}[P(\hat{Y} = 1|Y = 1) + P(\hat{Y} = 0|Y = 0)]$ of the model on these validation individuals for each threshold. Then we find the optimal threshold that maximises the balanced accuracy.

4. Fit the model on test data and predict the outcome labels of test individuals using the optimal threshold.

5. Compute statistical parity difference and consistency.

After obtaining the statistical parity difference and consistency for each repair level, we can plot them against the repair level on the same graph for a better visualisation. Consistency is shown as the blue thick line with the scale showing on the y-axis on the left hand side. Statistical parity difference is represented by the green thin line with scale showing on the y-axis on the right hand side. The two plots are shown in Figure 3.1 and Figure 3.2.

In the first case where *sex* is the sensitive attribute, the green thin line shows an upward trend which indicates that the statistical parity difference between female and male is getting closer to zero as we increase the repair level. When the repair level is around 0.6, it reaches 0 and statistical parity is satisfied. The blue thick line represents the consistency and shows a downward trend. We can see that a trade-off exists since individual fairness level gradually decreases as we improve group fairness by increasing the repair level from 0 to 0.6. In the second case where *race* is the sensitive attribute, group fairness and individual fairness also show opposite trends although there are some fluctuations. These practical results show that generally, after the sensitive attribute has been removed, if we modify the non-sensitive attributes for a further improvement in group fairness, the level of individual fairness would fall.

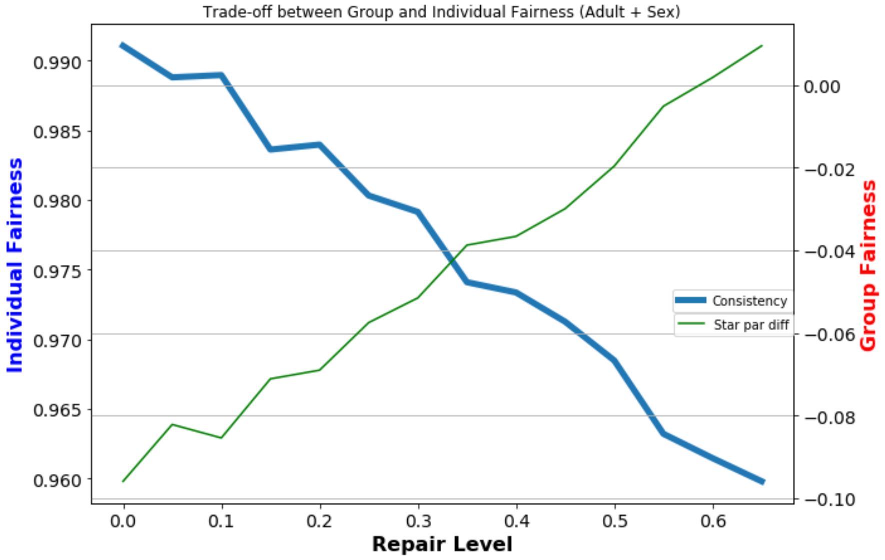

Figure 3.1.: Illustration of a trade-off between group fairness (thin line) and individual fairness (thick line) after the sensitive attribute *sex* is removed.

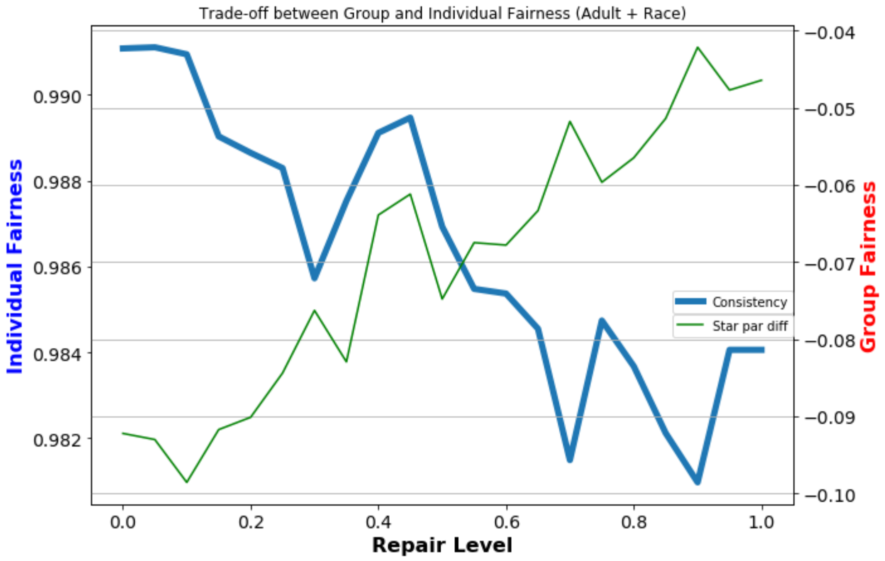

Figure 3.2.: Illustration of trade-off between group fairness (thin line) and individual fairness (thick line) after the sensitive attribute *race* is removed.

3.2. Influence of Wasserstein Distance on Individual Fairness

It has been proven that individual fairness implies group fairness if and only if the Wassertein distance between the two demographic groups is small [19]. However, the relationship between group fairness and individual fairness when the Wassertein distance between two groups is large has not been studied yet in the literature. In this section, we investigate how individual fairness performs at different Wasserstein distance when we force group fairness to be satisfied. There are different factors that affect Wassertein distance such as mean difference and variance difference between two groups. The Wasserstein distance between two distributions with the same mean but different variance can be the same as between two distributions with the same variance but different mean. Another interesting problem is whether individual fairness will show different performance when the Wasserstein distance between two groups is the same but is caused by different factors. Therefore, we require data sets with different magnitude and causing factors of Wasserstein distance to address the two problems. By simulating data sets, we can adjust parameters and control Wasserstein distance more easily. In Section 3.2.1, we introduce the main idea and process of our experiments. Then we provide the method on how to adjust Wasserstein distance in details in Section 3.2.3. The results of our initial experiments are shown in Section 3.2.4.

3.2.1. Main Process of the Experiment

In this section, we give an overall process. To start with, we need to simulate the data sets for our experiments. Each individual in the simulated data set belongs to either the privileged group or the unprivileged group and has five non-sensitive features and a positive/negative label. We use five features because it is sufficient enough to train an algorithm and is computationally efficient. Also, we include not only discrete attributes but also continuous attributes from different distributions. The procedure is as follows:

- Randomly generate a sensitive attribute S of N individuals where the probability of each individual being in the privileged group is p (N needs to be large so that the results are more reliable, so we choose $N = 20000$; We chose a range of p values from 0.1 to 0.9 and found out the experimental results are not sensitive to the value p, so we choose an appropriate value $p = 0.4$, i.e. there are 8000 individuals in the privileged group and 12000 in the unprivileged group).

- Randomly generate the label Y where the probability of each privileged individual being positive is p_1 and the probability of each unprivileged individual being positive is p_2 (Since we assume that privileged individuals are more likely to be predicted positive than unprivileged individuals due to discrimination, we require that $p_1 > p_2$. To make it more realistic, the difference between them is not set too large. We set $p_1 = 0.55$ and $p_2 = 0.4$, i.e. $P(Y = 1|S = 1) = 0.55$, $P(Y = 1|S = 0) = 0.4$).

- Randomly generate three features V_1, V_2, V_3, V_4 and V_5. [Details in Section 3.2.3]

- Randomly generate noise and add to each feature to make it more realistic.

- Form a data frame which consists of $S, V_1, V_2, V_3, V_4, V_5$ and Y.

Then we split the simulated data set into a training, validation and test data set, standardise each data set, build a logistic regression model on the training data set, then use the validation data set to find the optimal threshold and finally predict the outcome of each test individual. Based on the predicted outcomes of the test data set, we compute consistency c_b and statistical parity difference s_b which demonstrate individual fairness and group fairness respectively. The subscript b means ‘before’ since these are the fairness performance before we apply fairness algorithms (i.e. no fairness algorithms are applied).

Since we are interested in how individual fairness changes if we force group fairness to be satisfied, we need to compare the fairness performance when we do and do not apply fairness algorithms. We have obtained the fairness measures c_b and s_b when we do not apply fairness algorithms, therefore, the next step is to record the fairness performance when a fairness algorithm is applied. We take a well-performed pre-processing approach (DI remover or reweighing) and build a new logistic regression model, then record consistency c_a and statistical parity difference s_a. If $c_a > c_b$, it indicates that individual fairness improves after we apply group fairness algorithm. Otherwise, it indicates that individual fairness gets worse after applying the fairness algorithm. By appropriately adjusting the parameters of distribution from which $V_1/V_2/V_3/V_4/V_5$ is generated, we can obtain different data sets with different Wasserstein distance between the multivariate distributions of the privileged group and the unprivileged group. For each data set, we compute the sliced Wasserstein distance between the multivariate distributions as well as consistency and statistical parity difference before and after applying the fairness algorithm. After testing with DI remover and reweighing, we found out that the results are not sensitive to which fairness algorithm we use, so we only show the results of applying DI remover for this experiment. Finally, we plot the fairness performance measures against the sliced Wasserstein distance to observe the relationship between them.

3.2.2. Exploratory Data Analysis of Real Datasets

In order to generate features more realistically, we carry out exploratory data analysis (EDA) of real datasets *Adult*, *German* and *COMPAS*. The data set *Adult* has been described in Section 3.1. The data set *German* consists of one sensitive attribute *sex (1: Female; 0: Male)*, six non-sensitive attributes: *month*, *credit amount*,*investment as percentage*, *residence since*, *number of credits* and *people liable for* and an outcome label (1: good credit risk; 0: bad credit risk). The data set *COMPAS* consists of two sensitive attributes: *sex (1: Female; 0: Male)* and *race (1: Caucasian; 0: Non-Caucasian)*, three non-sensitive attributes: *age*, *decile_score* and *priors_count* and an outcome label (1: the defendant has reoffended in 2 years' time; 0: the defendant has not).

For each dataset, we first transform the non-sensitive attributes using standard scalar since transformed data is the data we feed into the model. Then for each transformed attribute in each dataset, we summarise the mean difference, standard deviation difference and one-dimensional Wasserstein distance between two groups. Also, for each dataset, we compute multi-dimensional sliced Wasserstein distance between two groups' multi-variate distributions of all non-sensitive attributes. The summary is shown in Table 3.1. From these values, we obtain a rough range of 1-D Wasserstein distance of a transformed attribute between 0 to 0.6, and a rough range of sliced Wasserstein distance between 0.1 to 0.3. Since 1-D Wasserstein distance range is very small in magnitude after all the attributes are transformed into a standard scale, there is a big difference for an attribute having 1-D Wasserstein distance 0.1 compared to having distance 0.2. Sliced Wasserstein distance has a even smaller range, thus, a dataset with sliced Wasserstein distance 0.1 will be very different if its attributes have been adjusted so that the sliced Wasserstein distance becomes 0.2. Also, we notice that in most cases, 1-D Wasserstein distance is affected by mean difference rather than standard deviation difference.

Furthermore, we plot histograms of two groups and obtain a visualisation of their distribution for some attributes from these three datasets. The datasets are popular and have been widely used in the study of fairness as they include a wide range of attributes, from discrete distributions where they only take a limited number of values to continuous distributions including both heavy-tailed and light-tailed. We present two discrete examples in Figure 3.3 and four typical continuous examples in Figure 3.4. It is important that we include different types of attributes when simulating datasets to make them more general and realistic, so we include discrete attributes and choose continuous attributes from appropriate distributions: Gamma, Log-normal, Exponential, Normal, Cauchy.

Table 3.1.: Mean difference, standard deviation difference and 1-D Wasserstein distance of each attribute and the sliced Wasserstein distance of each dataset.

Dataset	Attributes (Transformed)	Mean diff.	Standard deviation diff.	1-D Wasserstein distance	Multi-dimensional sliced Wasserstein distance
Adult + **Sex**	Age	0.1872	0.0528	0.1878	0.2243
	Education-num	0.0198	0.1094	0.1007	
	Capital-gain	0.1000	0.4393	0.1000	
	Capital-loss	0.0966	0.2236	0.1004	
	Hours-per-week	0.4855	0.0138	0.4855	
Adult + **Race**	Age	0.0910	0.0760	0.1038	0.1057
	Education-num	0.1400	0.0051	0.1400	
	Capital-loss	0.0584	0.1459	0.0587	
	Capital-gain	0.0420	0.1315	0.0420	
	Hours-per-week	0.1324	0.1301	0.1575	
German + **Sex**	Month	0.1761	0.1160	0.1762	0.2273
	Credit Amount	0.2021	0.1061	0.2080	
	Investment as %	0.1866	0.1127	0.1866	
	Residence Since	0.0299	0.0803	0.1121	
	Number of Credits	0.2038	0.1031	0.2061	
	People liable for	0.4399	0.5404	0.4399	
COMPAS + **Sex**	Age	0.0214	0.0291	0.0415	0.1899
	Decile Score	0.1543	0.0791	0.1543	
	Priors Count	0.3024	0.2960	0.3024	
COMPAS + **Race**	Age	0.3823	0.1527	0.3831	0.2934
	Decile Score	0.4184	0.1156	0.4184	
	Priors Count	0.3061	0.3384	0.3064	

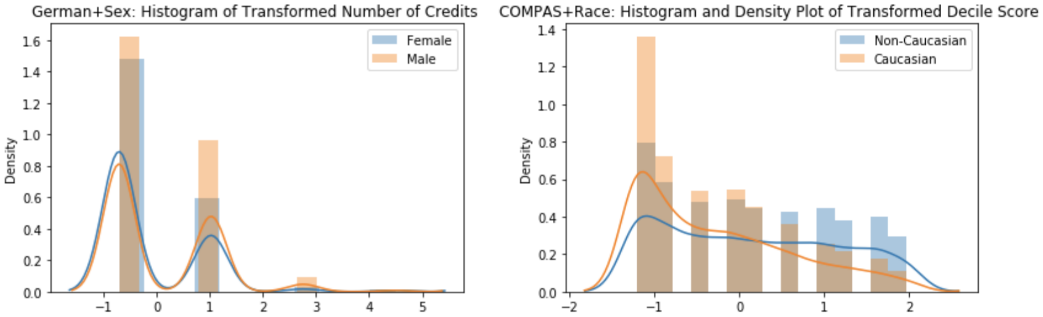

Figure 3.3.: Two examples of transformed discrete attribute histograms (Left: Number of Credits in German dataset with sex as the sensitive attribute. Right: Decile Score in COMPAS dataset with race as the sensitive attribute.)

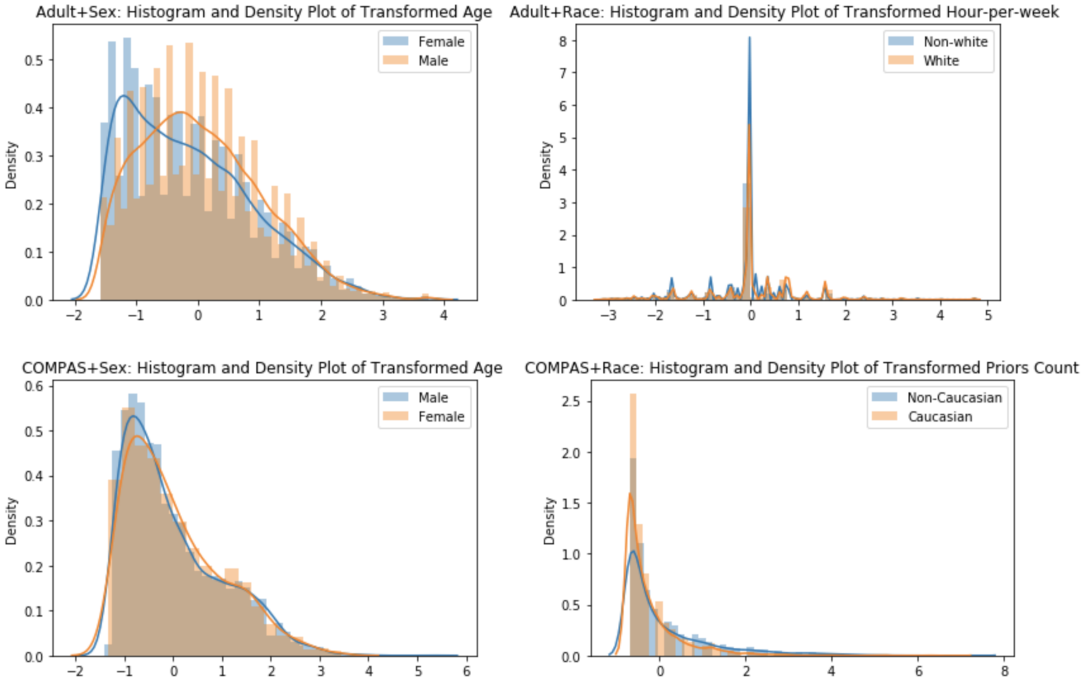

Figure 3.4.: Four examples of transformed continuous attribute histograms and distribution plots (Top left: Age in Adult dataset with sex as the sensitive attribute. Top right: Hour-per-week in Adult dataset with race as the sensitive attribute. Bottom left: Age in COMPAS dataset with sex as the sensitive attribute. Bottom right: Priors Count in COMPAS dataset with race as the sensitive attribute.)

3.2.3. Generating the Features

After observing real datasets, we now simulate our own datasets which we have better control on. This section describes the details of how to generate the features V_1, V_2, V_3, V_4 and V_5. By adjusting the parameters of an attribute distribution, we can adjust the mean difference or variance difference between two groups so that one-dimensional Wasserstein distance varies. A change in one-dimensional Wasserstein distance of any attribute will lead to a change in multi-dimensional Wasserstein distance over multiple attributes. According to EDA of attributes in real datasets, we decide to generate the five attributes from distributions with similar shape. We generate 2 discrete attributes V_1 and V_2 and 3 continuous attributes V_3, V_4 and V_5 from normal, gamma and Cauchy distribution respectively. For Normal and Gamma, we can easily control mean and variance by adjusting their parameters. A normal distribution $\mathcal{N}(\mu, \sigma^2)$ has μ as the mean and σ^2 as the variance. A gamma distribution with shape parameter k and scale parameter θ has mean $k\theta$ and variance $k\theta^2$.

Our first task is to study the influence on individual fairness when large Wassertein distance is caused by different factors (either mean difference or variance difference), we adjust the difference between mean or variance of the two groups while keeping other factors constant. In our experiments, we choose to change Wasserstein distance in V_3 or V_4 only since it is convenient to adjust mean and variance of normal and gamma distributions. There are four cases: 1) change the difference between mean of V_3 of the two groups only; 2) change the difference between variance of V_3 of the two groups only; 3) change the difference between mean of V_4 of the two groups only and 4) change the difference between variance of V_4 of the two groups only.

To start with, we introduce the way of generating attributes V_1, V_2 and V_5. We only generate them once and use the same attributes in all four cases so that the 1-D Wasserstein distance in $V_1/V_2/V_5$ distribution of two groups is constant. For each attribute, we generate from two slightly different distributions based on their actual outcome to incorporate some correlation between the features and the outcome and, in doing so, to ensure some level of accuracy of the model. The method is as follows:

1. V_1 is a discrete variable which randomly takes values 0, 1, 2, 3, 4, 5, 6 with probability 0.1, 0.04, 0.2, 0.18, 0.3, 0.06, 0.12 respectively if $Y = 0$ and takes the same values with probability 0.1, 0.04, 0.3, 0.155, 0.275, 0.035, 0.095 respectively if $Y = 1$.

2. V_2 is another discrete variable which randomly takes values 0, 1, 2 with probability 0.25, 0.25, 0.5 respectively if $Y = 0$ and takes the same values with probability 0.2, 0.2, 0.6 respectively if $Y = 1$.

3. V_5 is a continuous variable generated from $Cauchy(x_0, \gamma)$ where x_0 is location parameter and γ is scale parameter. We randomly generate V_5 $Cauchy(0,4)$ if $Y = 0$

and *Cauchy*(0,2.4) if $Y = 1$.

After having attributes V_1, V_2 and V_5, we now concentrate on how to generate V_3 and V_4 in four cases. In the first case where we only change the difference in mean of V_3 of the two groups, we generate V_3 and V_4 as follows. To start with, we randomly generate V_4 from *Gamma*(4,1) if $Y = 0$ and *Gamma*(4.8,1) if $Y = 1$, then the Wasserstein distance in $V_1/V_2/V_4/V_5$ distributions of two groups is constant. Then, we generate the first feature V_3 from a set of normal distributions: *i)* $\mathcal{N}(100m, 20)$ if $Y = 0$ and $S = 0$, *ii)* $\mathcal{N}(100, 20)$ if $Y = 0$ and $S = 1$, *iii)* $\mathcal{N}(100, 20)$ if $Y = 1$ and $S = 0$, *iv)* $\mathcal{N}(100(2 - m), 20)$ if $Y = 1$ and $S = 1$ where $m = 0.99, \ldots, 0.8$. There are 30 m values where each m corresponds to a new V_3, so we obtain 30 datasets with different values for V_3 but unchanged values for the other four attributes. When m decreases, the mean for the group $S = 0$ decreases while the mean for the group $S = 1$ increases so that the mean difference increases and the one-dimensional Wasserstein distance of V_3 increases, which leads to an increase in multi-dimensional sliced Wasserstein distance over multivariate distribution over five attributes. The variance difference of V_1 between the two groups hardly changes with m, which implies that the increase in Wasserstein distance is led by the increase in mean difference in this case. For the selection of parameter values and m values, we can select any as long as they provide a good range of Wasserstein distance between two groups since we need to observe fairness performance at different Wasserstein distance.

In the second case where we only change the difference in variance of V_3 of the two groups, we use the same V_1, V_2, V_4 and V_5 as in the first case, then generate V_3 from *i)* $\mathcal{N}(100, 20s)$ if $Y = 0$ and $S = 0$, *ii)* $\mathcal{N}(100, 20)$ if $Y = 0$ and $S = 1$, *iii)* $\mathcal{N}(100, 20)$ if $Y = 1$ and $S = 0$, *iv)* $\mathcal{N}(100, 20/s)$ if $Y = 1$ and $S = 1$ where $s = 0.96, \ldots, 0.15$. As s decreases, the variance for the group $S = 0$ decreases but the variance for the group $S = 1$ increases so that the variance difference and the Wasserstein distance both increase. In this case, the increase in multi-dimensional Wasserstein distance is caused by the increase in variance difference since the mean difference between the two groups barely changes.

In the third case where we only concentrate on changing the mean of V_4, we generate V_3 from $\mathcal{N}(90, 20)$ if $Y = 0$ and $\mathcal{N}(110, 20)$ if $Y = 1$, then the Wasserstein distance in $V_1/V_2/V_3/V_5$ distributions of two groups is constant. Then we generate V_4 from *i)* *Gamma*$(\frac{100}{k}, 30\sqrt{k})$ if $Y = 0$ and $S = 0$, *ii)* *Gamma*$(100, 30)$ if $Y = 0$ and $S = 1$, *iii)* *Gamma*$(100, 30)$ if $Y = 1$ and $S = 0$, *iv)* *Gamma*$(100k, \frac{30}{\sqrt{k}})$ if $Y = 1$ and $S = 1$ where $k = 0.9975, \ldots, 0.85$. The mean for *Gamma*$(100/k, 30\sqrt{k})$ is $\frac{100}{k} \times 30\sqrt{k} = 3000/\sqrt{k}$ and the mean for *Gamma*$(100k, 30/\sqrt{k})$ is $100k \times 30/\sqrt{k} = 3000\sqrt{k}$. The variance for these two gamma distributions is theoretically the same since $\frac{100}{k} \times (30\sqrt{k})^2 = 100 \times 30^2 = 100k \times (30/\sqrt{k})^2$. Therefore, the mean of group $S = 0$ increases whereas the mean of group $S = 1$ decreases as k decreases. The mean difference between two groups increases but the variance difference does not change, thus, the multi-dimensional Wasserstein distance increases due to the increase in the mean difference.

In the fourth case where we consider changing the variance of V_4 only, we use the same V_1, V_2, V_3 and V_5 as in the third case, then generate V_4 from *i)* $Gamma(100/l, 30l)$ if $Y = 0$ and $S = 0$, *ii)* $Gamma(100, 30)$ if $Y = 0$ and $S = 1$, *iii)* $Gamma(100, 30)$ if $Y = 1$ and $S = 0$, *iv)* $Gamma(100l, 30/l)$ if $Y = 1$ and $S = 1$. As l decreases, the mean difference remains unchanged, but the variance difference increases since the variance for group $S = 0$ increases but the variance for group $S = 1$ decreases. Therefore, the cause of an increase in multi-dimensional Wasserstein distance in this case is the increase in variance difference.

3.2.4. Experimental Results

In each of the four cases mentioned above, we generate 30 data sets where the multi-dimensional sliced Wasserstein distance between two groups is different in every data set, then compute statistical parity difference and consistency before and after we apply the fairness algorithm (DI remover) for every data set. Finally, we plot these fairness measures on one graph with x-axis representing the sliced Wasserstein distance. The results of these four cases are shown in Figures 3.5, 3.6, 3.7, 3.8. The thick blue dashed line and thick red solid line represent the consistency before and after applying DI remover respectively, with the scale showing on the y-axis on the left hand side. If the red thick line is below the blue thick line, it means that consistency decreases after applying the fairness algorithm. The thin blue dashed line and think red solid line represent the statistical parity difference before and after DI remover respectively, with the scale showing on the y-axis on the right hand side. If the red thin line is closer to zero compared to the blue thin line, it shows that group fairness has been improved after applying the fairness algorithm.

We notice that in all four cases, 'statistical parity difference after' gets closer to zero after DI remover is applied which means that group fairness is achieved regardless of multi-dimensional Wasserstein distance. Also, we notice that when Wasserstein distance is large, 'consistency after' is generally lower than 'consistency before' which indicates that individual fairness is harmed after satisfying group fairness. However, it is difficult to determine 'large' quantitatively since we observe that individual fairness shows different performance in different cases even at the same multi-dimensional Wasserstein distance. For instance, when it equals 0.2, *i)* applying DI remover decreases consistency from 0.98 to 0.94 approximately in the first case, *ii)* consistency increases from 0.953 to 0.963 approximately after applying DI remover in the second case, *iii)* consistency decreases from 0.95 to 0.93 approximately in the third case, *iv)* consistency increases 0.92 to 0.95 in the fourth case.

Moreover, by comparing Figure 3.5 and 3.6, we see that individual fairness gets worse after applying the DI remover algorithm when multi-dimensional Wasserstein distance reaches 0.125 if it is only affected by mean difference of V_3. However, if the variance difference of V_3 is the causing factor, up to the point when Wasserstein distance reaches

0.225, individual fairness does not worsen after applying DI remover. Similarly, we can compare Figure 3.7 and 3.8: Individual fairness gets worse after applying the DI remover algorithm when multi-dimensional Wasserstein distance reaches 0.16 if it is only affected by mean difference of V_4, but does not get worse until Wasserstein distance reaches 0.24 if it is only affected by variance difference of V_4. Therefore, a large Wasserstein distance caused by mean difference of a feature is more likely to have a bad influence on individual fairness than the same amount of Wasserstein distance caused by variance difference of that feature. This means that if we want to achieve group fairness in a dataset and we find out the attribute distributions of two groups are very different due to a large difference between their mean, we need to be careful as individual fairness as it is likely to be sacrificed.

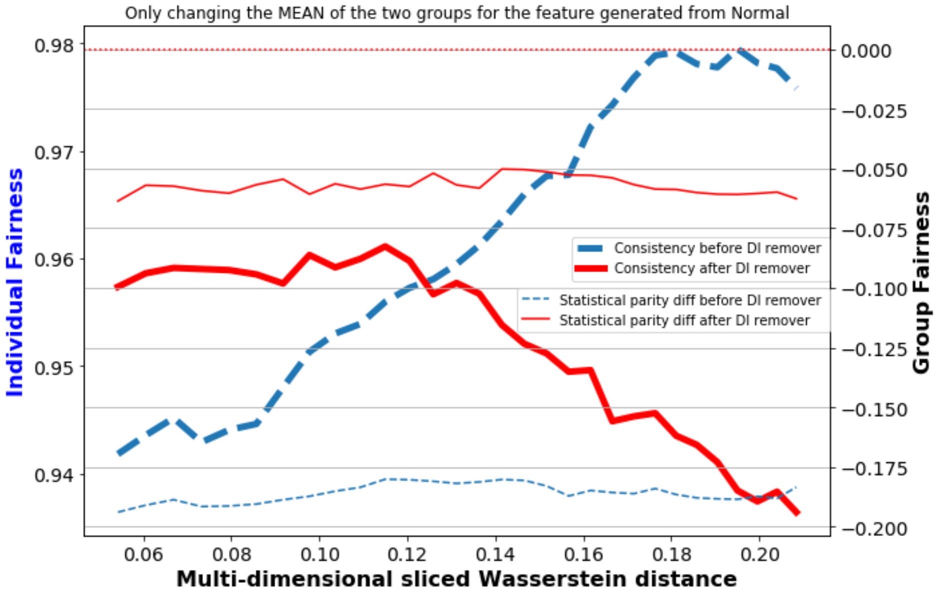

Figure 3.5.: When only changing the difference in mean of V_3 of the two groups, consistency decreases after applying DI remover when multi-dimensional Wasserstein distance reaches 0.125.

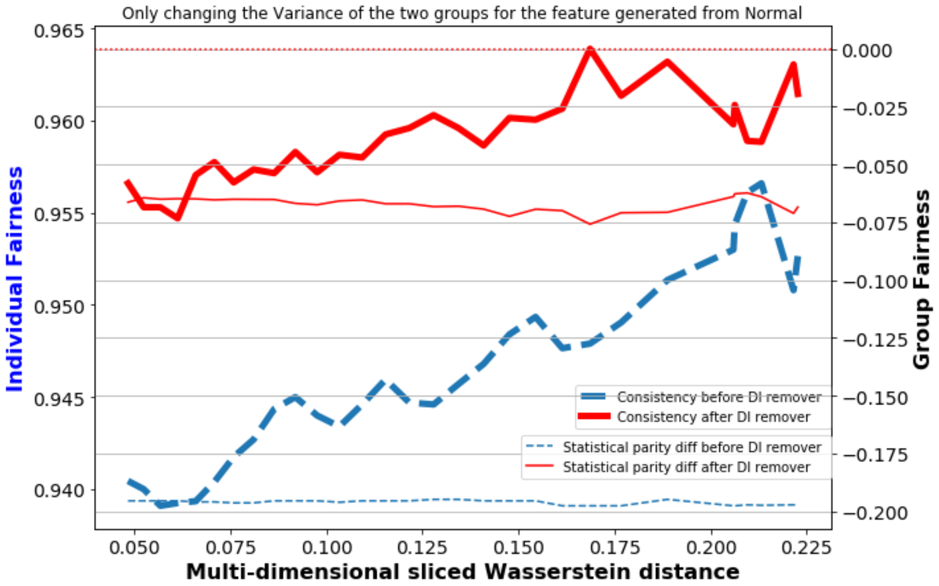

Figure 3.6.: When only changing the difference in variance of V_3 of the two groups, consistency does not fall after applying DI remover up to the point when Wasserstein distance reaches 0.225.

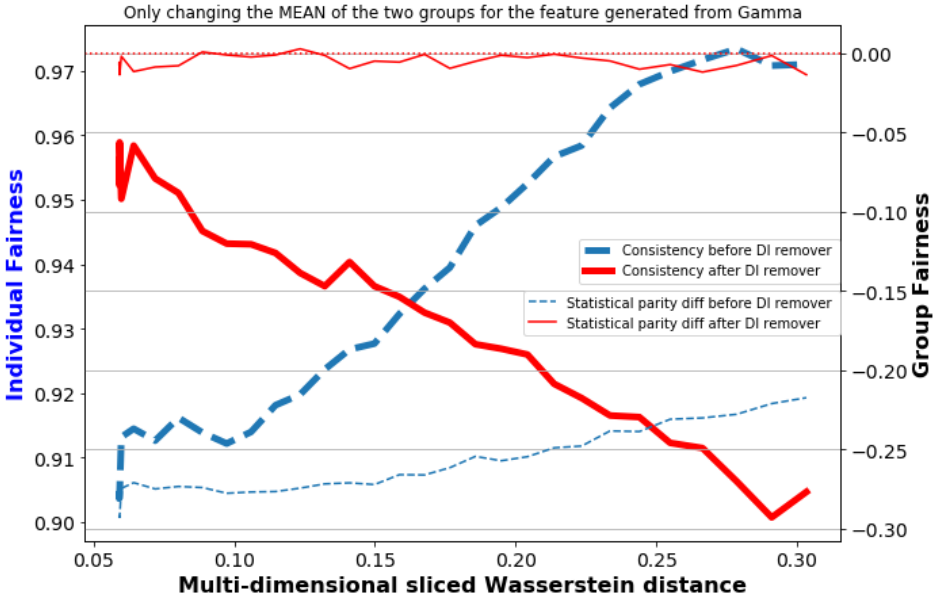

Figure 3.7.: When only changing the difference in mean of V_4 of the two groups, consistency falls after applying DI remover when Wasserstein distance reaches 0.16.

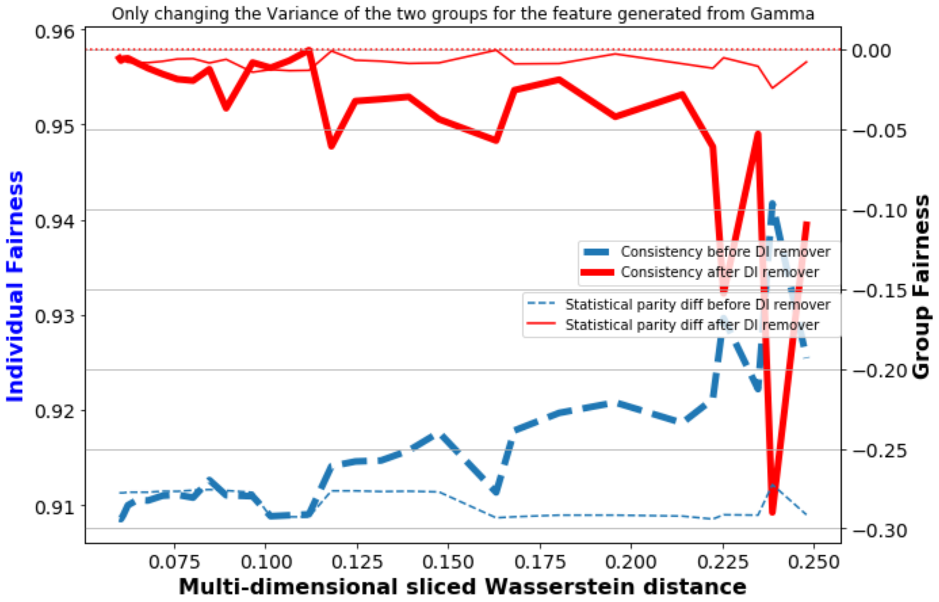

Figure 3.8.: When only changing the difference in variance of V_4 of the two groups, consistency falls after applying DI remover when Wasserstein distance reaches 0.24.

3.2.5. More Experimental results

In Section 3.2.3 and 3.2.4, we have concentrated on changing the mean/variance difference between two groups for only one of the attributes and generated 30 different datasets each time. In this section, we expand the experiments further by adjusting mean for multiple attributes at one time or adjusting their variance at one time. In both cases, we generate 1000 datasets. The generation procedure is only slightly different from the one we introduced in Section 3.2.3.

To start with, we simulate exactly the same attributes V_1, V_2 and V_5. Then in the first case where we focus on adjusting mean for multiple attributes, we randomly generate V_3 from a set of normal distributions: *i)* $\mathcal{N}(100m, 20)$ if $Y = 0$ and $S = 0$, *ii)* $\mathcal{N}(100, 20)$ if $Y = 0$ and $S = 1$, *iii)* $\mathcal{N}(100, 20)$ if $Y = 1$ and $S = 0$, *iv)* $\mathcal{N}(100(2-m), 20)$ if $Y = 1$ and $S = 1$ where $m = 0.99, \dots, 0.8$, and randomly generate V_4 from *i)* $Gamma(\frac{100}{k}, 30\sqrt{k})$ if $Y = 0$ and $S = 0$, *ii)* $Gamma(100, 30)$ if $Y = 0$ and $S = 1$, *iii)* $Gamma(100, 30)$ if $Y = 1$ and $S = 0$, *iv)* $Gamma(100k, \frac{30}{\sqrt{k}})$ if $Y = 1$ and $S = 1$ where $k = 0.9975, \dots, 0.85$. There are 32 m values and 32 k values where each m corresponds to a new V_3 and each k corresponds to a new V_4, so we obtain $32 \times 32 = 1024$ datasets with different combinations for V_3 and V_4. For simplicity, we then randomly select 1000 datasets from these 1024 datasets without loss of generality. In second case when we concentrate on adjusting variance for multiple attributes, we randomly generate V_3 from *i)* $\mathcal{N}(100, 20s)$

if $Y = 0$ and $S = 0$, *ii)* $\mathcal{N}(100, 20)$ if $Y = 0$ and $S = 1$, *iii)* $\mathcal{N}(100, 20)$ if $Y = 1$ and $S = 0$, *iv)* $\mathcal{N}(100, 20/s)$ if $Y = 1$ and $S = 1$ where $s = 0.96, \ldots, 0.15$ and randomly generate V_4 from *i)* $Gamma(100/l, 30l)$ if $Y = 0$ and $S = 0$, *ii)* $Gamma(100, 30)$ if $Y = 0$ and $S = 1$, *iii)* $Gamma(100, 30)$ if $Y = 1$ and $S = 0$, *iv)* $Gamma(100l, 30/l)$ if $Y = 1$ and $S = 1$. We also set 32 s values and l values, then select 1000 datasets from the 1024 generated.

Finally, we compute multi-dimensional sliced Wasserstein distance and the change in consistency after applying a fairness algorithm, then present the results in scatter plots shown in Figure 3.9 and 3.10. We plot a horizontal line $y = 0$ as a reference line, a point above the 0 line indicates that there is an increase in consistency (i.e. improvement in individual fairness) after satisfying group fairness whereas a point below 0 means that individual fairness falls after satisfying group fairness. The two figures also show that when a large multi-dimensional Wasserstein distance is caused by large mean differences in attributes rather than large variance differences, individual fairness is more likely to be sacrificed when group fairness is satisfied.

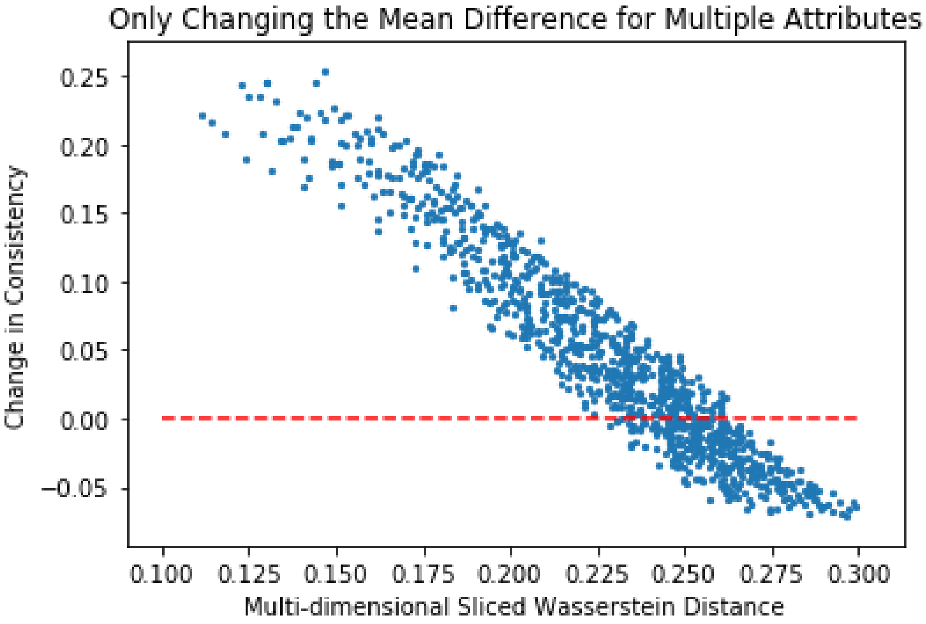

Figure 3.9.: When changing the difference in mean of multiple attributes only, consistency starts to fall after applying DI remover when multi-dimensional Wasserstein distance is in the range (0.225, 0.265).

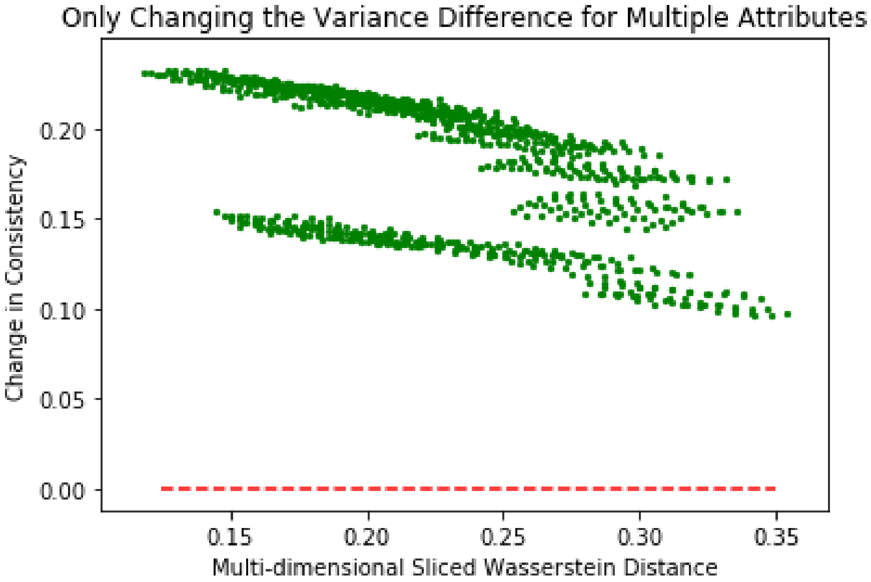

Figure 3.10.: When changing the difference in variance of multiple attributes only, consistency increases after applying DI remover (change in consistency is always positive).

3.3. Summary

In this chapter, we show on a real data set *Adult* that if we achieve a group fairness level and an individual fairness level by simply removing the sensitive attribute and we then modify the non-sensitive attributes to improve the group fairness level further, the individual fairness level decreases. Therefore, simply eliminating the sensitive attributes has an advantage if we concentrate on achieving individual fairness. On simulated data sets, we show that satisfying group fairness worsens individual fairness when the Wasserstein distance between two groups' attribute distributions is large. Moreover, experimental results indicate that if a large Wasserstein distance is caused by a large mean difference rather than a large variance difference, individual fairness is more likely to be affected when group fairness is satisfied. This means that if the attribute distributions of two groups are very different, especially when there is a large difference between their mean, individual fairness tends to be sacrificed when we force group fairness to be satisfied.

4. Pre-processing Approaches

In Chapter 3, we investigated the relationship between group fairness and individual fairness. In this chapter, we concentrate on studying pre-processing fairness algorithms and assessing their performance in terms of both group and individual fairness. We start with tweaking the existing reweighing algorithm in Section 4.1 by changing the assigned weights to individuals. For example, if original reweighing algorithm achieves a statistical parity difference of -0.01 and we only require -0.05, then we could adjust the weights to achieve a higher level of individual fairness while meeting the requirements for group fairness. Then, in Section 4.2, we propose a new pre-processing algorithm. To achieve group fairness, one way is to modify the data so that distributions of the attributes of different groups become more similar. In other words, we want to make the non-sensitive attributes to be less associated with the sensitive attribute. Our new algorithm reduces the mean difference in attribute values between different groups so that the association between the sensitive attribute and non-sensitive attributes is decreased.

4.1. Tweaking Reweighing Algorithm

As we introduced in Section 2.1.2, disparate impact remover has an outstanding advantage which is that modification level can be adjusted from 0 to 1, where 0 gives original data and 1 gives fully modified data. Adjusting modification level allows us to choose a desirable performance which meets specific requirements of individual fairness and group fairness. Therefore, similarly, we can slightly alter reweighing algorithm by changing the weights assigned to each individual. In Section 4.1.1, we introduce the idea of changing the reweighing level for all individuals and analyse the performance on 1000 simulated datasets. Then we evaluate the performance of reweighing algorithms with a mixture of two reweighing levels in Section 4.1.2. Finally, we introduce other forms of reweighing such as conditional reweighing and reweighing with multiple attributes in Section 4.1.3.

4.1.1. Reweighing Level

To start with, we remind ourselves of the theory of reweighing, which is described in Appendix A.2. For each individual in training data, reweighing algorithm assigns weight

$$W(s, y) = \frac{P_{exp}(S = s \cap Y = y)}{P_{obs}(S = s \cap Y = y)} = \frac{|S = s| \times |Y = y|}{|D| \times |S = s \cap Y = y|},$$

then trains a classifier with the weights.

Since training individuals with weights $W(s, y)$ means that reweighing algorithm is fully applied whereas training individuals with a weight equal to 1 means that no fairness algorithm is applied, it is natural to consider adjusting reweighing level by assigning weights with values between $W(s, y)$ and 1. This method is feasible since $W(0, 1), W(1, 0) > 1$ and $W(0, 0), W(1, 1) < 1$, which are proven as follows:

	$S = 0$	$S = 1$	Total
$Y = 0$	a	c	$\lvert Y = 0\rvert = a + c$
$Y = 1$	b	d	$\lvert Y = 1 = b + d$
Total	$\lvert S = 0\rvert = a + b$	$\lvert S = 1\rvert = c + d$	$\lvert D\rvert = a + b + c + d$

Table 4.1.: Number of individuals in each group in an unfair dataset

Suppose that we have an unfair data set shown in Table 4.1 with $P(Y = 1|S = 0) < P(Y = 1|S = 1)$, then

$$\begin{gathered}
P(Y = 1|S = 0) < P(Y = 1|S = 1) \\
\frac{|Y = 1 \cap S = 0|}{|S = 0|} < \frac{|Y = 1 \cap S = 1|}{|S = 1|} \\
\frac{b}{a + b} < \frac{d}{c + d} \\
bc + bd < ad + bd \\
bc < ad
\end{gathered}$$

Therefore, we have

$$\begin{aligned}
W(0, 1) &= \frac{|S = 0| \times |Y = 1|}{|D| \times |S = 0 \cap Y = 1|} = \frac{(a + b)(b + d)}{(a + b + c + d)b} \\
&= \frac{ab + ad + b^2 + bd}{ab + b^2 + bc + bd} = \frac{(ab + b^2 + bd) + ad}{(ab + b^2 + bd) + bc} > 1
\end{aligned}$$

$$\begin{aligned}
W(1, 0) &= \frac{|S = 1| \times |Y = 0|}{|D| \times |S = 1 \cap Y = 0|} = \frac{(c + d)(a + c)}{(a + b + c + d)c} \\
&= \frac{ac + c^2 + ad + cd}{ac + bc + c^2 + cd} = \frac{(ac + c^2 + cd) + ad}{(ac + c^2 + cd) + bc} > 1
\end{aligned}$$

$$
\begin{aligned}
W(0,0) &= \frac{|S=0| \times |Y=0|}{|D| \times |S=0 \cap Y=0|} = \frac{(a+b)(a+c)}{(a+b+c+d)a} \\
&= \frac{a^2+ac+ab+bc}{a^2+ab+ac+ad} = \frac{(a^2+ac+ab)+bc}{(a^2+ac+ab)+ad} < 1
\end{aligned}
$$

$$
\begin{aligned}
W(1,1) &= \frac{|S=1| \times |Y=1|}{|D| \times |S=1 \cap Y=1|} = \frac{(c+d)(b+d)}{(a+b+c+d)d} \\
&= \frac{bc+cd+bd+d^2}{ad+bd+cd+d^2} = \frac{(cd+bd+d^2)+bc}{(cd+bd+d^2)+ad} < 1
\end{aligned}
$$

Suppose that reweighing level 1 indicates that individuals are assigned with original weights $W(0,1)$, $W(1,0)$, $W(0,0)$, $W(1,1)$, we can lower the reweighing level by moving the weights towards 1. We introduce a method to adjust the reweighing level L:

$$
\begin{aligned}
W_L(0,1) &= 1 + (W(0,1) - 1)L \\
W_L(1,0) &= 1 + (W(1,0) - 1)L \\
W_L(0,0) &= 1 - (1 - W(0,0))L \\
W_L(1,1) &= 1 - (1 - W(1,1))L
\end{aligned}
$$

To evaluate the performance of reweighing algorithms with different levels, we apply reweighing algorithms with levels $L = 0, 0.1, \ldots, 1$ to 1000 data sets generated in Section 3.2.5. When we lower the reweighing level by 0.1 each time, we record the number of datasets in which the absolute value of statistical parity difference increases (i.e. group fairness level decreases) and the number of datasets in which consistency increases (i.e. individual fairness level increases). For a better visualisation, we plot mean of 1000 datasets' absolute values of statistical parity difference and consistency against reweighing level. The results are shown in Table 4.2 and Figure 4.1.

From the results, we can see that when reweighing level decreases by 0.1, absolute value of statistical parity difference increases in most data sets. This indicates that group fairness level tends to decrease when reweighing level decreases, which is also demonstrated in the first plot in Figure 4.1. We also notice that in the most of the time when reweighing level decreases by 0.1, consistency increases in most data sets although there are cases where consistency only increases in around half of the datasets at relatively low reweighing levels. The general decreasing trend shown in the second plot in Figure 4.1 reflects that individual fairness level tends to decrease when reweighing level increases.

Change in Reweighing level	Number of Data Sets where Absolute Value of Statistical Parity Difference Increases	Number of Data Sets where Consistency Increases
From 1 to 0.9	972	942
From 0.9 to 0.8	983	900
From 0.8 to 0.7	974	857
From 0.7 to 0.6	984	836
From 0.6 to 0.5	982	776
From 0.5 to 0.4	985	697
From 0.4 to 0.3	982	663
From 0.3 to 0.2	979	594
From 0.2 to 0.1	989	519
From 0.1 to 0	971	486

Table 4.2.: A table which records the number of datasets out of 1000 in which the absolute value of statistical parity difference increases and the number of datasets in which consistency increases when decreasing the reweighing level by 0.1

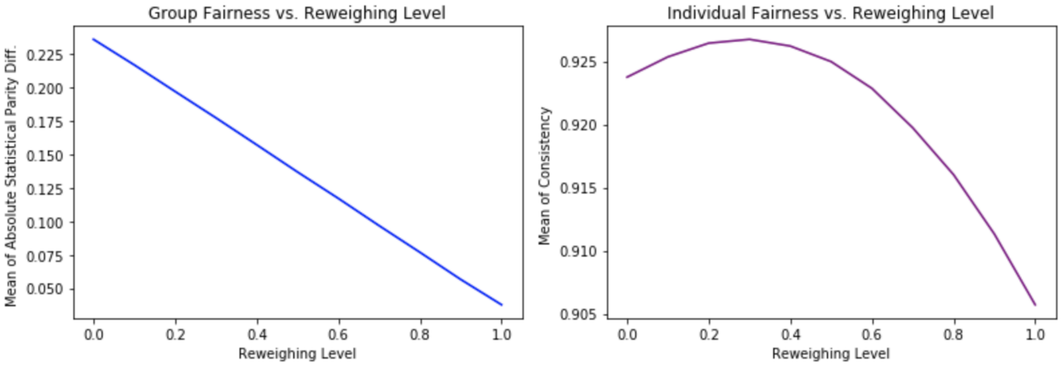

Figure 4.1.: Reweighing algorithm's average performance over 1000 datasets at different reweighing level (Left: Mean of absolute values of statistical parity difference against reweighing level. Right: Mean of consistency against reweighing level.)

4.1.2. Reweighing Individuals

In Section 4.1.1, we assign the weights to every individual according to one's actual outcome and protected attribute. We now discover the situation where only a fraction of individuals from each group are assigned with the weights $W_{L_1}(0,1)$, $W_{L_1}(1,0)$, $W_{L_1}(0,0)$, $W_{L_1}(1,1)$ while the rest is assigned with weights from a lower level L_2 ($L_1 > L_2$). Take the data set in Table 4.1 as an example. Suppose the fraction of the first reweighing level is m and the fraction of the second reweighing level is $(1-m)$. Then ma individuals with $S=0$ and $Y=0$ are assigned with $W_{L_1}(0,0)$, $(1-m)a$ individuals with $S=0$ and $Y=0$ are assigned with $W_{L_2}(0,0)$, mb individuals with $S=0$ and $Y=1$ are assigned with $W_{L_1}(0,1)$, $(1-m)b$ individuals with $S=0$ and $Y=1$ are assigned with $W_{L_2}(0,1)$ etc. Then the required number of individuals are selected randomly and assigned with proper weights.

To evaluate the performance of reweighing algorithms with two levels taking different proportion, we apply reweighing algorithms with levels $L_1=1$ and $L_2=0.9$ with fraction $m=0,0.1,\ldots,1$ to 1000 data sets generated in Section 3.2.5. When we decrease the proportion of reweighing level 1 by 10% and increase the proportion of reweighing level 0.9 by 10% each time, we record the number of datasets in which the absolute value of statistical parity difference increases (i.e. group fairness level decreases) out of 1000, a large value indicates that group fairness level decreases in most datasets whereas a small number indicates that group fairness level increases in most datasets. Also, we record the number of datasets in which consistency increases (i.e. individual fairness level increases) out of 1000 each time, a large value means that individual fairness level increases in most datasets whereas a small value means that individual fairness level decreases in most datasets. For a better visualisation, we plot mean of 1000 datasets' absolute values of statistical parity difference and consistency against reweighing level. The results are shown in Table 4.3 and Figure 4.2.

From Table 4.3, we can see that when the percentage of individuals who are assigned with level 1 weights decreases, absolute value of statistical parity difference increases in most data sets, in other words, group fairness level decreases in most data sets. Take the first row as an example, when we assign level 1 to 90% individuals and level 0.9 to 10% individuals instead of assigning level 1 to all individuals, group fairness level decreases in 993 data sets out of 1000. Generally, group fairness level decreases as we increase the proportion of individuals being assigned a lower reweighing level. However, when we assign level 0.9 to all individuals instead of assigning level 0.9 to 90% individuals and level 1 to 10% individuals, only 14 datasets show a decrease in group fairness level, in other words, group fairness improves in most datasets although the proportion of lower reweighing level increases. In fact, group fairness level tends to be higher when applying an algorithm where all the weights are assigned at level 0.9 rather than one where there is a mixture of two reweighing levels. Similarly, from the results of consistency, we see that individual fairness level generally increases as we increase the proportion of individuals being assigned a lower reweighing level, but the level tends to be lower when applying an algorithm where all the weights are assigned at level 0.9 rather than one where there is a mixture of two reweighing levels.

Therefore, the performance results suggest that we should avoid mixture of reweighing levels and set the same reweighing level for all individuals for more efficient results. For instance, if we are not satisfied with the group fairness level when reweighing every individual at level 0.9 but the group fairness level is beyond what we require when we reweigh every individual at level 1, then reweighing every individual at a level between 0.9 and 1 (e.g. 0.95) is a more effective solution than reweighing some individuals at level 1 and some at level 0.9.

Change in reweighing level proportion	Number of Data Sets where Absolute Value of Statistical Parity Difference Increases	Number of Data Sets where Consistency Increases
From (1: 100%, 0.9: 0%) to (1: 90%, 0.9: 10%)	993	893
From (1: 90%, 0.9: 10%) to (1: 80%, 0.9: 20%)	794	587
From (1: 80%, 0.9: 20%) to (1: 70%, 0.9: 30%)	788	613
From (1: 70%, 0.9: 30%) to (1: 60%, 0.9: 40%)	808	619
From (1: 60%, 0.9: 40%) to (1: 50%, 0.9: 50%)	786	575
From (1: 50%, 0.9: 50%) to (1: 40%, 0.9: 60%)	786	619
From (1: 40%, 0.9: 60%) to (1: 30%, 0.9: 70%)	789	567
From (1: 30%, 0.9: 70%) to (1: 20%, 0.9: 80%)	810	594
From (1: 20%, 0.9: 80%) to (1: 10%, 0.9: 90%)	767	555
From (1: 10%, 0.9: 90%) to (1: 0%, 0.9: 100%)	14	141
From (1: 100%, 0.9: 0%) to (1: 0%, 0.9: 100%)	980	889
From (1: 90%, 0.9: 10%) to (1: 0%, 0.9: 100%)	213	370

Table 4.3.: A table which records the number of datasets out of 1000 in which the absolute value of statistical parity difference increases and the number of datasets in which consistency increases when the proportion of reweighing level 1 decreases by 10% and the proportion of reweighing level 0.9 increases by 10%

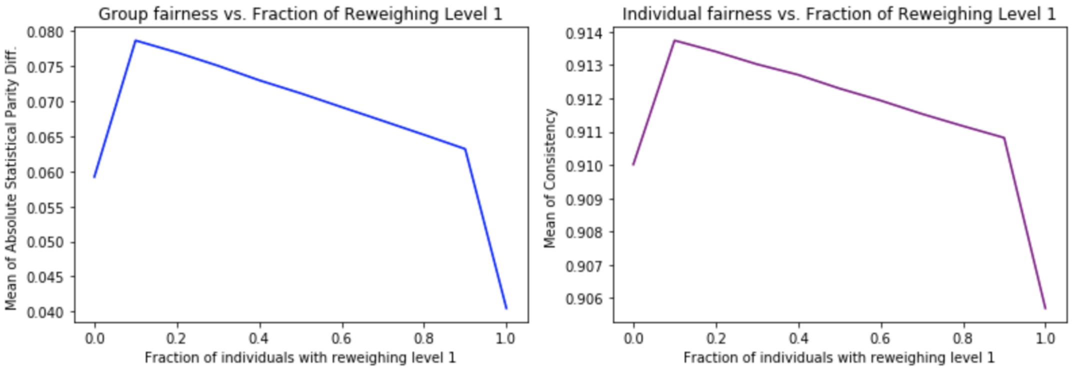

Figure 4.2.: Reweighing algorithm's average performance over 1000 datasets at different proportion of reweighing levels (Left: Mean of absolute values of statistical parity difference against fraction of reweighing level 1. Right: Mean of consistency against fraction of reweighing level 1.)

4.1.3. Other Forms of Reweighing

In reality, there are cases where part of the differences in the probability of positive outcome for the two groups may be objectively explainable by other attributes. For instance, suppose that average annual income for females is lower than males but females work less hours than males on average, then part of difference in annual income can be explained by working hours. Researchers argue that only the discrimination conditioned on an explanatory attribute should be removed. Therefore, we introduce conditional reweighing. If we want to achieve statistical independence between protected attribute S and outcome Y conditional on another attribute A, i.e. $S \perp\!\!\!\perp Y|A$ instead of $S \perp\!\!\!\perp Y$, then the expected probability is

$$\begin{aligned} P_{exp}(S = s \cap Y = y|A = a) &= P(S = s|A = a)P(Y = y|A = a) \\ &= \frac{|S = s \cap A = a|}{|A = a|} \times \frac{|Y = y \cap A = a|}{|A = a|} \end{aligned}$$

The observed probability is

$$P_{obs}(S = s \cap Y = y|A = a) = \frac{|S = s \cap Y = y \cap A = a|}{|A = a|}$$

For each observation in training data, assign weight

$$W(s, y, a) = \frac{P_{exp}(S = s \cap Y = y|A = a)}{P_{obs}(S = s \cap Y = y|A = a)} = \frac{|S = s, A = a| \times |Y = y, A = a|}{|A = a| \times |S = s \cap Y = y \cap A = a|}$$

When A is binary, there will be 8 weights.

Furthermore, reweighing can also be applied when there are multiple sensitive attributes in the data set. Suppose that there are two independent sensitive attributes S_1 and S_2 and we want to achieve statistical independence between S_1, S_2 and outcome Y, i.e. We expect $S_1 \perp\!\!\!\perp Y, S_2 \perp\!\!\!\perp Y$, the expected probability is

$$\begin{aligned} P_{exp}(S_1 = s_1 \cap S_2 = s_2 \cap Y = y) &= P(S_1 = s_1 \cap S_2 = s_2)P(Y = y) \\ &= \frac{|S_1 = s_1 \cap S_2 = s_2|}{|D|} \cdot \frac{|Y = y|}{|D|} \end{aligned}$$

The observed probability is

$$P_{obs}(S_1 = s_1 \cap S_2 = s_2 \cap Y = y) = \frac{|S_1 = s_1, S_2 = s_2, Y = y|}{|D|}$$

For each observation in training data, assign weight

$$W(s_1, s_2, y) = \frac{P_{exp}}{P_{obs}} = \frac{|S_1 = s_1 \cap S_2 = s_2| \times |Y = y|}{|D| \times |S_1 = s_1 \cap S_2 = s_2 \cap Y = y|}$$

4.2. New Pre-processing Approach

After exploring the existing reweighing algorithm, we construct a new pre-processing approach in this section. We start with introducing a statistic called point-biserial correlation coefficient which measures the association between a categorical variable and a continuous variable, and another statistic called Cramér's V which measures the association between two categorical variables in Section 4.2.1. Inspired from their formula, we then construct a new pre-processing approach in Section 4.2.2: reduce the mean difference in attribute values between different groups so that the association between the sensitive attribute and non-sensitive attributes is decreased. Finally, we apply this new method to the real data sets *Adult* and *COMPAS* and compare its group fairness and individual fairness performance with existing fairness algorithms: reweighing and reject option based classification (ROC) in Section 4.2.3.

4.2.1. Point-biserial Correlation Coefficient and Cramér's V

Before we apply a pre-processing approach to the original data, a useful step is to check the correlation between the sensitive attribute and the other attributes and the significance of the correlation. Since the sensitive attribute is often categorical, we will introduce correlation ratio, a measure of the correlation between a categorical variable and a continuous variable [48]. Then we introduce a point-biserial correlation coefficient which measures the correlation between a binary variable and a continuous variable [49].

In more details, we consider a categorical variable S and a continuous variable X, the continuous variable of each individual is represented as x_{si}, where s indicates the category that the individual belongs to and i is the index. Let n be the total number of individuals and n_s be the number of individuals in category s, then the mean of group s is $\bar{x}_s = \frac{1}{n_s}\sum_i^{n_s} x_{si}$ and the mean of the whole population is $\bar{x} = \frac{1}{n}\sum_s n_s\bar{x}_s$. Given this, the correlation ratio η is defined as:

$$\eta = \sqrt{\frac{\sum_s n_s(\bar{x}_s - \bar{x})^2}{\sum_{s,i}(x_{si} - \bar{x})^2}}. \tag{4.1}$$

The fraction inside the square root is the equivalent to the proportion of the weighted variance of the group means to the variance of all individuals. Its range is between 0 and 1, where a value of 1 indicates that all the variance is due to the variance between different groups and no variance is due to the variance within groups, whereas a value of 0 indicates the opposite. Correlation ratio reflects the relative importance of the "between group variance" and "within group variance".

Specifically, when S is binary, there are two categories: $s = 0$ or 1. The number of individuals in each group is represented by n_1 and n_0 and the mean of each group is represented $\bar{x}_1$ and $\bar{x}_0$. Then we define point-biserial correlation coefficient as [49]:

$$\frac{\bar{x}_1 - \bar{x}_0}{\sqrt{\frac{1}{n}\sum_{s,i}(x_{si} - \bar{x})^2}}\sqrt{\frac{n_0 n_1}{n^2}}. \tag{4.2}$$

A value of 0 indicates no correlation and a value of 1 or -1 indicates perfect correlation. Moreover, a p-value is yielded from the point-biserial correlation coefficient. The interpretation of the p-value is as follows. Our null hypothesis H_0 is that the point-biserial correlation coefficient is 0. The alternative hypothesis, H_1, is that the coefficient is not equal to 0. If the p-value is less than the significance level that has been set, there is sufficient evidence to show that the point-biserial correlation coefficient does not equal 0, so we reject H_0, equivalently it means that there is evidence showing the existence of association between the two variables. If the p-value is greater than the significance level, there is insufficient evidence to reject H_0 so that there is insufficient evidence showing the existence of association between the two variables. We show that the magnitude of point-biserial correlation coefficient is equivalent to correlation ratio when S is binary.

Formula 4.1 can be written as:

$$
\begin{aligned}
&\frac{1}{\sqrt{\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2}}\sqrt{\frac{1}{n}\Big[n_0(\bar{x}_0-\bar{x})^2+n_1(\bar{x}_1-\bar{x})^2\Big]}\\
&=\frac{1}{\sqrt{\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2}}\sqrt{\frac{1}{n}\Big[n_0(\bar{x}_0-\frac{n_0\bar{x}_0+n_1\bar{x}_1}{n})^2+n_1(\bar{x}_1-\frac{n_0\bar{x}_0+n_1\bar{x}_1}{n})^2\Big]}\\
&=\frac{1}{\sqrt{\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2}}\sqrt{\frac{1}{n}\Big[n_0(\frac{n\bar{x}_0-n_0\bar{x}_0-n_1\bar{x}_1}{n})^2+n_1(\frac{n\bar{x}_1-n_0\bar{x}_0-n_1\bar{x}_1}{n})^2\Big]}\\
&=\frac{1}{\sqrt{\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2}}\sqrt{\frac{1}{n}\Big[\frac{n_0n_1^2}{n^2}(\bar{x}_0-\bar{x}_1)^2+\frac{n_1n_0^2}{n^2}(\bar{x}_1-\bar{x}_0)^2\Big]}\\
&=\frac{|\bar{x}_1-\bar{x}_0|}{\sqrt{\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2}}\sqrt{\frac{1}{n}\Big[\frac{n_0n_1^2+n_1n_0^2}{n^2}\Big]}\\
&=\frac{|\bar{x}_1-\bar{x}_0|}{\sqrt{\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2}}\sqrt{\frac{n_0n_1}{n^2}},
\end{aligned}
$$

which is equal to the absolute value of Formula 4.2.

So far, we have described how to measure the association between a categorical variable (the sensitive attribute) and a continuous variable. However, the non-sensitive attributes may include categorical attributes as well. Therefore, we now move onto a statistic that measures the association between a categorical variable and another categorical variable: Cramér's V [50]. Consider a categorical variable, S, and another categorical variable, C, of n individuals. In addition, suppose that there are r categories in variable S and k categories in variable C. Let $n_{i\cdot}$ be the number of individuals with $S=i$, $n_{\cdot j}$ be the number of individuals with $C=j$, and n_{ij} be the number of individuals with $S=i$ and $C=j$ for $i=1,\dots,r$ and $j=1,\dots,k$. Then, Cramér's V is defined as:

$$
V=\sqrt{\frac{\chi^2/n}{\min(k-1,r-1)}}, \tag{4.3}
$$

where $\chi^2=\sum_{i,j}\frac{(n_{ij}-\frac{n_{i\cdot}n_{\cdot j}}{n})^2}{\frac{n_{i\cdot}n_{\cdot j}}{n}}$. When the variable S is binary, Cramér's V is simply $\sqrt{\chi^2/n}$ since $r=2$. It varies from 0 to 1, where a value of 0 corresponds to no association between the variables and a value of 1 corresponds to perfect/complete association. Also, as with the point-biserial correlation coefficient, we can compute the p-value yielded from Cramér's V to evaluate the significance.

4.2.2. Framework of a New Algorithm

Inspired from point-biserial correlation coefficient and Cramér's V, we start to build a new pre-processing fairness algorithm. One of the approaches to improve group fairness is to obtain a fairer representation of the non-sensitive attributes by making the attribute distributions for different groups similar to each other. Equivalently, we decrease the correlation between the sensitive attribute and non-sensitive attributes and make point-biserial correlation coefficient or Cramér's V closer to 0.

To decrease Cramér's V, we decrease χ^2 by adjusting the number of individuals in each group so that n_{ij} increases or decreases towards $\frac{n_{i\cdot}n_{\cdot j}}{n}$. Taking the sensitive attribute S and a categorical variable C in Table 4.4 as an example, we can see that $n_{0\cdot} = 98$, $n_{1\cdot} = 102$, $n_{\cdot 0} = 104$ and $n_{\cdot 1} = 96$. Thus, the expected frequencies are $\frac{n_{0\cdot}n_{\cdot 0}}{n} = 51$, $\frac{n_{1\cdot}n_{\cdot 0}}{n} = 53$, $\frac{n_{0\cdot}n_{\cdot 1}}{n} = 47$ and $\frac{n_{1\cdot}n_{\cdot 1}}{n} = 49$, which are shown in Table 4.5. We modify some individuals' C categories so that the modified frequencies are closer to the expected frequencies. The closer we move the frequencies to the expected ones, the higher the attribute's modification level is.

Table 4.4.: Observed Frequencies

Observed	S=0	S=1	Total
C=0	56	48	104
C=1	42	54	96
Total	98	102	200

Table 4.5.: Expected Frequencies

Expected	S=0	S=1	Total
C=0	51	53	104
C=1	47	49	96
Total	98	102	200

To lower the magnitude of point-biserial correlation coefficient, the first intuition is to reduce the gap between the mean of the two groups so that the numerator term decreases. Nevertheless, the denominator also decreases since the variance of all individuals tends to decrease as the mean difference between two groups gets smaller. Now we show mathematically how point-biserial correlation coefficient responds to a decrease in the mean difference. Suppose that, originally, $\bar{x}_1 - \bar{x}_0 = a$ so that the numerator is $a\sqrt{\frac{n_0 n_1}{n^2}}$. If we do not change the attribute values of individuals in group $s = 1$ but add λa to every individual's value in group $s = 0$ where $0 \leq \lambda \leq 1$, then $\bar{x}_{0\text{new}} = \bar{x}_0 + \lambda a$ and the new numerator becomes $(1-\lambda)a\sqrt{\frac{n_0 n_1}{n^2}}$. The new population mean is $\bar{x}_{\text{new}} = \frac{n_1\bar{x}_1 + n_0\bar{x}_{0\text{new}}}{n} = \frac{n_1\bar{x}_1 + n_0(\bar{x}_0 + \lambda a)}{n} = \frac{n_1\bar{x}_1 + n_0\bar{x}_0}{n} + \frac{n_0}{n}\lambda a = \bar{x} + \frac{n_0}{n}\lambda a$. Then, the part inside the square root of the new denominator becomes:

$$\frac{1}{n}\Big[\sum_{i:s=0}(x_{0i}+\lambda a-\bar{x}-\frac{n_0}{n}\lambda a)^2+\sum_{i:s=1}(x_{1i}-\bar{x}-\frac{n_0}{n}\lambda a)^2\Big]$$
$$=\frac{1}{n}\Big[\sum_{i:s=0}(x_{0i}-\bar{x}+\frac{n_1}{n}\lambda a)^2+\sum_{i:s=1}(x_{1i}-\bar{x}-\frac{n_0}{n}\lambda a)^2\Big]$$
$$=\frac{1}{n}\Big[\sum_{i:s=0}[(x_{0i}-\bar{x})^2+\frac{2n_1}{n}\lambda a(x_{0i}-\bar{x})+(\frac{n_1}{n}\lambda a)^2]$$
$$+\sum_{i:s=1}[(x_{1i}-\bar{x})^2-\frac{2n_0}{n}\lambda a(x_{1i}-\bar{x})+(\frac{n_0}{n}\lambda a)^2]\Big]$$
$$=\frac{1}{n}\Big[\sum_{s,i}(x_{si}-\bar{x})^2+\frac{2n_1\lambda a}{n}\sum_{i:s=0}(x_{0i}-\bar{x})-\frac{2n_0\lambda a}{n}\sum_{i:s=1}(x_{1i}-\bar{x})$$
$$+\frac{n_0n_1^2\lambda^2a^2}{n^2}+\frac{n_1n_0^2\lambda^2a^2}{n^2}\Big]$$
$$=\frac{1}{n}\Big[\sum_{s,i}(x_{si}-\bar{x})^2+2n_1\lambda a(\frac{n_0\bar{x}_0}{n}-\frac{n_0\bar{x}}{n})-2n_0\lambda a(\frac{n_1\bar{x}_1}{n}-\frac{n_1\bar{x}}{n})+\frac{n_1n_0\lambda^2a^2(n_1+n_0)}{n^2}\Big]$$
$$=\frac{1}{n}\Big[\sum_{s,i}(x_{si}-\bar{x})^2+\frac{2n_1n_0\lambda a}{n}(\bar{x}_0-\bar{x})-\frac{2n_1n_0\lambda a}{n}(\bar{x}_1-\bar{x})+\frac{n_1n_0\lambda^2a^2}{n}\Big]$$
$$=\frac{1}{n}\Big[\sum_{s,i}(x_{si}-\bar{x})^2+\frac{2n_1n_0\lambda a}{n}(\bar{x}_0-\bar{x}_1)+\frac{n_1n_0\lambda^2a^2}{n}\Big]$$
$$=\frac{1}{n}\Big[\sum_{s,i}(x_{si}-\bar{x})^2-\frac{2n_1n_0\lambda a^2}{n}+\frac{n_1n_0\lambda^2a^2}{n}\Big]$$
$$=-\frac{n_1n_0a^2}{n^2}\lambda(2-\lambda)+\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2.$$

Therefore, the new point-biserial correlation coefficient becomes

$$\frac{(1-\lambda)(\bar{x}_1-\bar{x}_0)\sqrt{\frac{n_0n_1}{n^2}}}{\sqrt{-\frac{n_1n_0a^2}{n^2}\lambda(2-\lambda)+\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2}}.$$

We can see that the numerator has been multiplied by a factor of $(1-\lambda)$, the new coefficient will be smaller than the old one as long as the denominator is multiplied by a factor larger than $(1-\lambda)$. That is,

$$\sqrt{-\frac{n_1n_0a^2}{n^2}\lambda(2-\lambda)+\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2}>(1-\lambda)\sqrt{\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2},$$

which is equivalent to,

$$-\frac{n_1n_0a^2}{n^2}\lambda(2-\lambda)+\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2>(1-\lambda)^2\Big[\frac{1}{n}\sum_{s,i}(x_{si}-\bar{x})^2\Big].$$

By representing the variance of the whole population as V, the condition required for the point-biserial correlation coefficient of the attribute to decrease is simply

$$V - \frac{n_1 n_0 a^2}{n^2}\lambda(2-\lambda) > (1-\lambda)^2 V,$$

where a is difference between the mean of the two groups and $0 \leq \lambda \leq 1$ is repair level of the attribute.

After describing the methods on how to decrease Cramér's V and point-biserial correlation coefficient, we now construct a simple pre-processing approach. When there is one single binary sensitive attribute S, the procedure is as follows:

- Transform the data matrix which consists of non-sensitive attributes by standard scalar.

- If a categorical non-sensitive attribute C is to be modified, we switch some individuals' categories so that Cramér's V is lowered. If a continuous non-sensitive attribute X is to be modified, we calculate the mean $\bar{x}_1$ and $\bar{x}_0$ for the group $S = 0$ and $S = 1$ respectively. The mean difference is $a = \bar{x}_1 - \bar{x}_0$. Then we add λa to the attribute value of every individual in group $S = 0$ but keep the value of every individual in group $S = 1$ the same, where $0 \leq \lambda \leq 1$.

- Monitor balanced accuracy, statistical fairness difference and consistency as we modify a single non-sensitive attribute at its different repair levels.

If the sensitive attribute has multiple levels, then the population is divided into multiple groups and we decrease correlation ratio or Cramér's V of non-sensitive attributes. For continuous attributes, we compute the mean for each group and reduce the gap between every two groups. When there are multiple sensitive attributes, we create a categorical variable where each category represents one of the groups. Then the categorical variable becomes a sensitive attribute with multiple levels.

One way to select which non-sensitive attributes to modify is based on their correlation coefficient and p-value. Firstly, we compute the correlation coefficient and corresponding p-value of each non-sensitive attribute and select those attributes whose p-value is lower than significance level. Then we put those attributes in the descending order of the correlation coefficient's magnitude and modify them in this order. If group fairness still needs to be improved after the first attribute is fully modified, we will continue modifying the second one. For better visualisation, we summarise the procedure in a diagram, shown in Figure 4.3. For example, if the rule is that consistency should not be lower than 98%, we select the repair level of the first attribute which provides the highest group fairness level given at least 98% consistency. If the consistency does not fall to 98% after fully modifying the first attribute, we then modify a second attribute, a third attribute and so on until we obtain an optimal result.

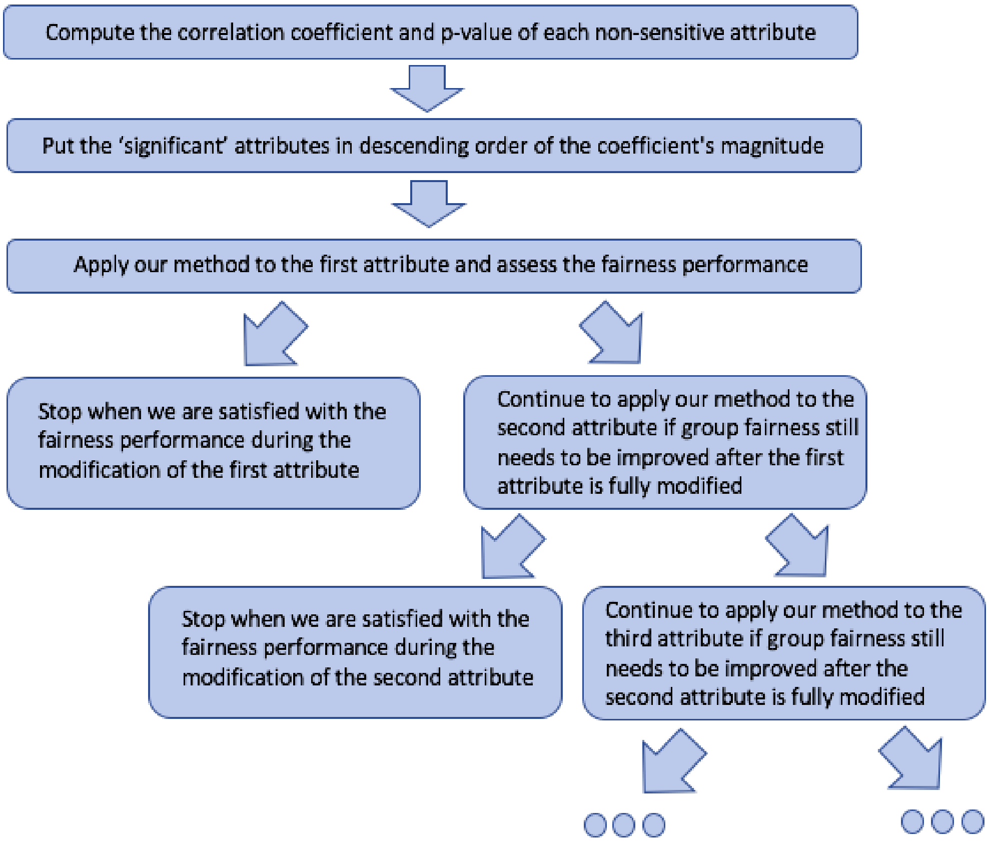
Compute the correlation coefficient and p-value of each non-sensitive attribute
Put the 'significant' attributes in descending order of the coefficient's magnitude
Apply our method to the first attribute and assess the fairness performance
Stop when we are satisfied with the fairness performance during the modification of the first attribute
Continue to apply our method to the second attribute if group fairness still needs to be improved after the first attribute is fully modified
Stop when we are satisfied with the fairness performance during the modification of the second attribute
Continue to apply our method to the third attribute if group fairness still needs to be improved after the second attribute is fully modified

Figure 4.3.: Demonstration of the order in which we apply our method

4.2.3. Performance on Real Data Sets

In this section, we test this new algorithm on real data sets: *Adult* and *COMPAS*, which have been described in Section 3.1 and Section 3.2.2. On these data sets, we first train an algorithm without any fairness consideration, and then apply two existing algorithms which are detailed in Section 2.1.2, namely the original reweighing algorithm and reject option based classification (ROC), and record their performance results. The results are summarised in Table 4.6.

Then, we test the new method on the same data sets. Specifically, we first compute the point-biserial correlation coefficient and its p-value of each non-sensitive attribute for each data set. These are shown in Table 4.7. After obtaining these statistics, we apply our new method on the attributes in descending order of their correlation coefficient (i.e. modify the attribute which has the highest correlation coefficient first). At different levels of modification, we obtain a set of corresponding statistical parity difference, consistency

Data set	Algorithms	Statistical Parity Difference	Consistency	Balanced Accuracy
Adult (sensitive: sex)	*i)* Reweighing *ii)* ROC	-0.0399 -0.0451	0.9611 0.9615	0.7083 0.7131
Adult (sensitive: race)	*i)* Reweighing *ii)* ROC	-0.0467 -0.0528	0.9684 0.9725	0.7100 0.7115
COMPAS (sensitive: sex)	*i)* Reweighing *ii)* ROC	0.0694 0.0961	0.9594 0.9549	0.6703 0.6672
COMPAS (sensitive: race)	*i)* Reweighing *ii)* ROC	-0.0090 0.0362	0.9184 0.9037	0.6677 0.6652

Table 4.6.: Performance on the four data sets when we *i)* apply reweighing algorithm, *ii)* apply ROC algorithm.

and balanced accuracy. We observe the performance results by plotting consistency and balanced accuracy against statistical parity difference. Finally, we compare its performance with the performance of reweighing and ROC which is shown in Table 4.6.

For the data set *Adult* with sex as the sensitive attribute, we apply the new method on the attribute 'Hours-per-week' since it has the highest correlation coefficient with the lowest p-value. The results are shown in Figure 4.4. From Figure 4.4 and Table 4.6, we can see that when the statistical parity difference of the new algorithm is the same as reweighing (-0.0399), the consistency is approximately 0.975 and balanced accuracy is approximately 0.711, both higher than reweighing which has consistency 0.9611 and balanced accuracy 0.7083. Similarly, when the statistical parity difference of the new algorithm is the same as ROC (-0.0451), the consistency (approximately 0.978) is higher than ROC (0.9615) and balanced accuracy (approximately 0.713) is similar to ROC (0.7131). Thus, our method outperforms reweighing and ROC since it obtains an approximately 1.5% higher level of individual fairness and similar balanced accuracy given the same level of group fairness.

Then, for the data set *Adult* with race as the sensitive attribute, we modify the attributes in descending order of correlation: 'Education-num', 'Hours-per-week' and 'Age'. From Figure 4.5, if we select the modification level where the statistical parity difference is -0.02, then consistency is approximately 0.978 and balanced accuracy is approximately 0.711. This result outperforms both reweighing (statistical parity difference: -0.0467; consistency: 0.9684; balanced accuracy: 0.7100) and ROC (statistical parity difference: -0.0528; consistency: 0.9725; balanced accuracy: 0.7115) since it has similar balanced accuracy but higher group fairness level and individual fairness level.

Next, for the data set *COMPAS* with sex as the sensitive attribute, we modify the attribute 'Priors count' which has the highest correlation and significance (lowest p-value). From Figure 4.6, if we select the modification level where the statistical parity difference

Data set	Non-sensitive attributes	Correlation coefficient	P-value
Adult (sensitive: sex)	Age	0.0881	8.65×10^{-85}
	Education-num	0.0093	0.0393
	Hours-per-week	0.2286	0.0
Adult (sensitive: race)	Age	0.0320	1.45×10^{-12}
	Education-num	0.0493	1.17×10^{-27}
	Hours-per-week	0.0466	6.66×10^{-25}
COMPAS (sensitive: sex)	Age	0.0084	0.5090
	Decile score	-0.0606	1.91×10^{-6}
	Priors count	-0.1187	8.11×10^{-21}
COMPAS (sensitive: race)	Age	0.1812	1.06×10^{-46}
	Decile score	-0.1983	8.91×10^{-56}
	Priors count	-0.1451	2.16×10^{-30}

Table 4.7.: Point-biserial correlation coefficient and its p-value of each non-sensitive attribute in each data set.

is -0.0095, then consistency is 0.9844 and balanced accuracy is 0.6779. Our method outperforms both reweighing (statistical parity difference: 0.0694; consistency: 0.9594; balanced accuracy: 0.6703) and ROC (statistical parity difference: 0.0961; consistency: 0.9549; balanced accuracy: 0.6672) since it has higher balanced accuracy, group fairness level and individual fairness level.

Finally, for data set *COMPAS* with race as the sensitive attribute, we modify the attributes in the order: 'Decile score', 'Age' and 'Priors count' and obtain the performance results in Figure 4.7. One of the performance results we obtain is that statistical parity difference, consistency and balanced accuracy are 0.0006, 0.9235 and 0.6672 respectively. This result outperforms both reweighing (statistical parity difference: -0.0090; consistency: 0.9184; balanced accuracy: 0.6677) and ROC (statistical parity difference: 0.0362; consistency: 0.9037; balanced accuracy: 0.6652) since it has similar balanced accuracy but higher group fairness level individual fairness level.

To summarise, our algorithm performs better than both reweighing and ROC on all the data sets Adult+sex, Adult+race Compas+sex and Compas+race. We obtain 0.5∼3% higher consistency and 1∼9% lower statistical parity difference. On the data set Adult+sex, our algorithm achieves a higher individual fairness level and higher balanced accuracy given the same level of group fairness. On the other data sets, our algorithm obtains higher group and individual fairness levels with similar or higher balanced accuracy.

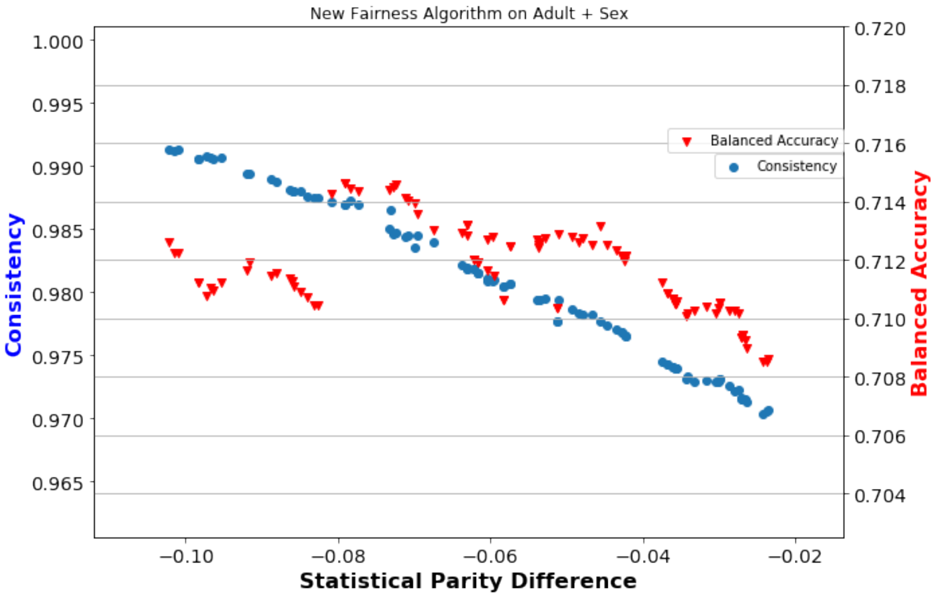

Figure 4.4.: Performance of our new algorithm on Adult+Sex

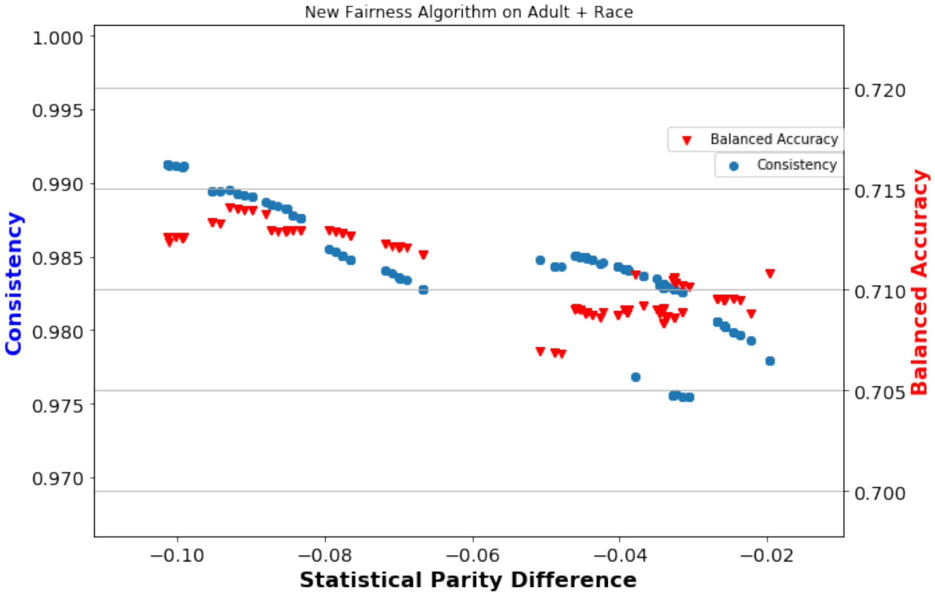

Figure 4.5.: Performance of our new algorithm on Adult+Race

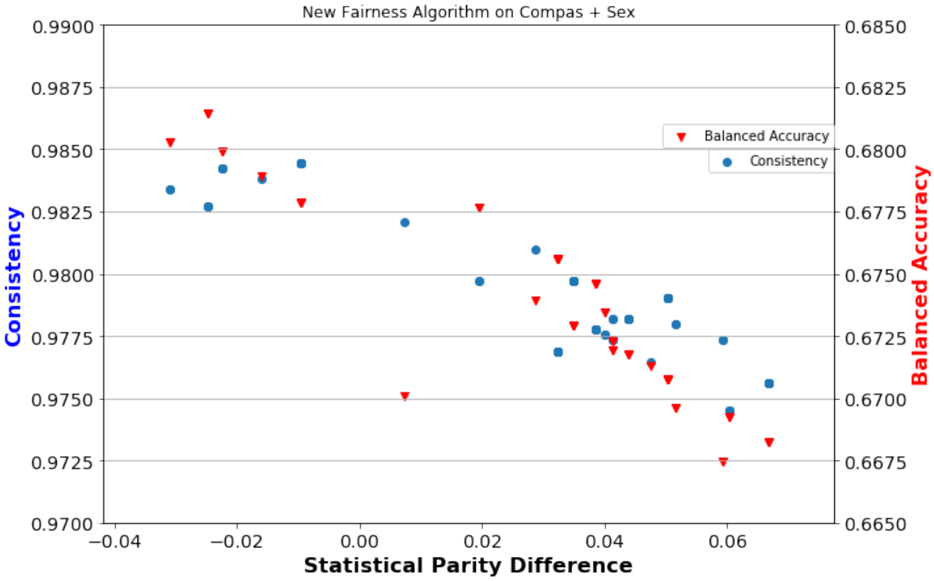

Figure 4.6.: Performance of our new algorithm on Compas+Sex

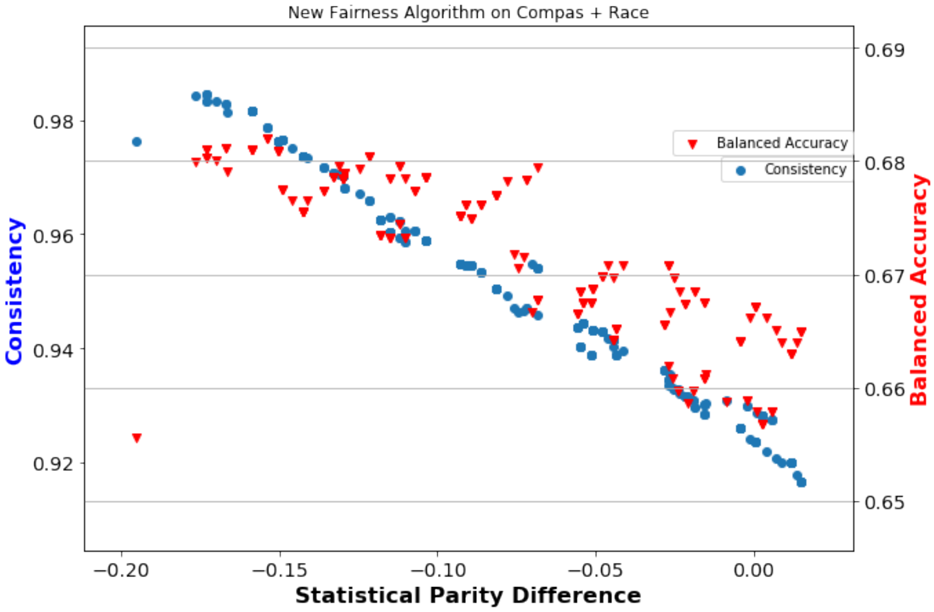

Figure 4.7.: Performance of our new algorithm on Compas+Race

4.3. Summary

In this chapter, we introduce the idea of reweighing level and find out that when reweighing level increases, group fairness tends to increase and individual fairness tends to decrease. Also, it is suggested that we should avoid mixture of reweighing and set the same reweighing level for all individuals in order to obtain group fairness more effectively.

Moreover, we introduce two statistics, namely the point-biserial correlation coefficient and Cramér's V, which measure the association between a categorical variable and another variable. Since a sensitive attribute is categorical, we use these statistics to measure the association between the sensitive attribute and non-sensitive attributes. Then we build a new pre-processing fairness algorithm. This algorithm reduces the mean difference in attribute values between different groups so that the magnitude of these statistic and the association between the sensitive attribute and non-sensitive attributes is decreased. Finally, we apply this new method to the real data sets *Adult* and *COMPAS* and compare our performance with reweighing and ROC. We obtain 0.5∼3% higher consistency, 1∼9% lower statistical parity difference and similar balanced accuracy, the performance results show that our method outperforms both original reweighing and ROC algorithm.

5. Conclusions and Future Work

This chapter summarises the work presented in this report in Section 5.1 and introduces a plan for the future work in Section 5.2.

5.1. Conclusions

In this report, we present three research contributions. Firstly, we explore the relationship between group fairness and individual fairness after the sensitive attribute is removed. On the real data set *Adult*, we apply DI remover and record the fairness performance at difference repair levels. The results show that if we improve group fairness further by modifying the non-sensitive attribute values after the sensitive attribute is removed, individual fairness falls. In other words, there is a trade-off between individual fairness and group fairness after the sensitive attribute is removed. Thus, although simply eliminating the sensitive attribute has been criticised as it may not sufficiently remove discrimination since non-sensitive attributes may be highly correlated to the sensitive attribute and carry sensitive information, it can be advantageous if we concentrate on achieving individual fairness.

Secondly, we simulate data sets with different Wasserstein distance and record the fairness performance when applying DI remover to these data sets. We show that when Wasserstein distance between the attribute distributions of two groups is large, the level of individual fairness decreases when we apply fairness algorithms to satisfy group fairness. It is useful if we can find a universal threshold such that a distance above it is classified as being large. However, the threshold varies in different cases which makes it difficult to define 'large' quantitatively. Moreover, if a large Wasserstein distance is caused by a large mean difference rather than a large variance difference, individual fairness is more likely to be affected when group fairness is satisfied. This result indicates that if we want to achieve group fairness when the attribute distributions of two groups are very different, especially when there is a large difference between their mean, individual fairness is likely to be sacrificed.

Thirdly, we slightly alter the existing reweighing algorithm by inserting a setting called reweighing level. By decreasing the reweighing level, group fairness level tends to decrease whereas individual fairness level tends to increase. Also, the results show that in order to achieve group fairness more effectively, we should use the same reweighing level for all individuals rather than use a mixture of reweighing levels. We not only expand an existing algorithm, but also build a simple pre-processing algorithm to achieve fairness. The idea is to reduce the association between the sensitive attribute and the other attributes. This method can be applied to both continuous and categorical attributes. Also, this algorithm allows a single sensitive attribute with multiple levels or multiple sensitive attributes so that it is not restricted to achieve fairness across only two demographic groups. Since it is a pre-processing method without involving outcome labels, we can use it to solve multi-class classification and regression problems. We test this new algorithm on real data sets *Adult* and *COMPAS* and compare its performance with two existing algorithms, namely reweighing and ROC. The results show that our algorithm outperforms the existing ones since it obtains 0.5∼3% higher consistency, 1∼9% lower statistical parity difference and similar balanced accuracy.

5.2. Future Work

Future work will concentrate on improving our algorithm which was introduced in Chapter 4 so that the algorithm can deal with high-dimensional datasets, considers both group and individual fairness and are not restricted to one binary attribute or binary classification only. We applied the method to the non-sensitive attributes in the descending order of their magnitude of correlation coefficients in this report, but we have not tested the algorithm using other ways of ordering the attributes. In the future, we are going to explore the influence on the performance if we change the order in which we apply the algorithm. Also, to improve our algorithm, we are going to learn a fairer presentation of the data using optimisation. To optimize the performance of the algorithm and find a good compromise between individual fairness and group fairness, we are going to define a penalty term for group unfairness and another penalty term for individual unfairness. One possible penalty term for individual fairness is data distortion after the non-sensitive attributes are modified and the penalty term for group fairness will be defined based on statistical parity difference, but these will be discussed in more detail in the future. Then we will solve the optimisation problem by finding the minimum of an objective function which involves both penalty terms. Moreover, we need to consider how the algorithm deals with the case where there is a large number of features. Dimensionality reduction will be a good approach. There are some popular techniques such as principal component analysis (PCA) and autoencoders which are used to learn a lower-dimensional representation for a set of data which ignores nuisance factors but minimises information loss at the same time. We are going to combine these methods with our algorithm and test on high-dimensional data sets.

In addition, we can address Challenge 4 mentioned in Section 1.1 which states the difficulty on defining distance metric when measuring individual fairness. In more detail, we have only used consistency to measure individual fairness so far. In future work, we plan to study how to define new metrics which may depend on the specifics of each domain. In particular, given a specific domain, we will produce a survey to investigate public views on the relative importance of each feature and define a new distance metric according to public opinions. We will also use a survey to determine the most appropriate fairness definition given the domain. Insurance will be the first domain that we look at.

Furthermore, we will look at non-binary classification or regression problems where the outcomes are multi-class or continuous. For instance, we will look at an insurance premium data set where the outcome is premiums and discuss whether they are fair. The definition of individual fairness and group fairness in regression or multi-class classification is different from the fairness definition in binary classification and needs to be carefully defined. We will also look at the trade-off between individual fairness and group fairness when applying a group fairness algorithm, then interpret the results to the insurance company.

Bibliography

[1] W. Lin, Y. Hu, and C. Tsai, "Machine learning in financial crisis prediction: A survey," *In IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, pp. 421–436, 2012.

[2] P. Dua and S. Bais, "Supervised learning methods for fraud detection in healthcare insurance," *Machine Learning in Healthcare Informatics*, vol. 56, pp. 261–285, 2014.

[3] C. Clancy, J. Hecker, E. Stuntebeck, and T. O'Shea, "Applications of machine learning to cognitive radio networks," *In IEEE Wireless Communications*, vol. 14, pp. 47–52, 2007.

[4] J. M. Logg, J. A. Minson, and D. A. Moore, "Algorithm appreciation: People prefer algorithmic to human judgment," *Organizational Behavior and Human Decision Processes*, vol. 151, pp. 90–103, 2019.

[5] A. Chouldechova and A. Roth, "A snapshot of the frontiers of fairness in machine learning," *Communications of the ACM*, vol. 63 No.5, pp. 82–89, 2020.

[6] F. Martínez-Plumed, C. Ferri, D. Nieves, and J. Hernández-Orallo, "Missing the missing values: The ugly duckling of fairness in machine learning," *International Journal of Intelligent Systems*, vol. 36, 2021.

[7] K. Crawford and R. Calo, "There is a blind spot in ai research," *Nature*, vol. 538, 2016.

[8] B. D. Mittelstadt, P. Allo, M. Taddeo, S. Wachter, and L. Floridi, "The ethics of algorithms: Mapping the debate," *Big Data & Society*, vol. 538, 2016.

[9] A. Romei and S. Ruggieri, "A multidisciplinary survey on discrimination analysis," *Knowledge Engineering Review*, vol. 29, 2013.

[10] "Protected characteristics," 2019. [Available from `https://www.equalityhumanrights.com/en/equality-act/protected-characteristics` ; accessed 18-June-2020].

[11] "Civil rights act of 1964," 2010. [Available from `https://www.history.com/topics/black-history/civil-rights-act` ; accessed 10-May-2020].

[12] "Adverse impact analysis / four-fifths rule," 2009. [Available from `https://www.prevuehr.com/resources/insights/adverse-impact-analysis-four-fifths-rule/` ; accessed 10-May-2020].

[13] F. Kamiran and T. Calders, "Classifying without discriminating," *2009 2nd International Conference on Computer, Control and Communication*, pp. 1–6, 2009.

[14] P. Adler, C. Falk, S. A. Friedler, G. Rybeck, C. Scheidegger, B. Smith, and S. Venkatasubramanian, "Auditing black-box models for indirect influence," *Knowl Inf Syst*, vol. 54, pp. 95–122, 2018.

[15] S. Hajian and J. Domingo-Ferrer, "A methodology for direct and indirect discrimination prevention in data mining," *IEEE Transactions on Knowledge and Data Engineering*, 2013.

[16] D. Pessach and E. Shmueli, "Algorithmic fairness," *ArXiv*, vol. abs/2001.09784, 2020.

[17] I. Zliobaite, "A survey on measuring indirect discrimination in machine learning," *arXiv: Computers and Society*, vol. abs/1511.00148v1, 2015.

[18] M. Kearns, S. Neel, A. Roth, and Z. S. Wu, "Preventing fairness gerrymandering: Auditing and learning for subgroup fairness," *In Proceedings of the 35th International Conference on Machine Learning*, 2018.

[19] 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*, p. 214–226, 2012.

[20] T. Calders and S. Verwer, "Three naive bayes approaches for discrimination-free classification," *Data Mining and Knowledge Discovery*, vol. 21, p. 277–292, 2010.

[21] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, "Learning fair representations," *In International Conference on Machine Learning*, p. 325–333, 2013.

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

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

[24] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian, "Certifying and removing disparate impact," *In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 259–268, 2015.

[25] G. Pleiss, M. Raghavan, F. Wu, J. Kleinberg, and K. Q. Weinberger, “On fairness and calibration,” *In Proceedings of the 31st International Conference on Neural Information Processing Systems*, p. 5684–5693, 2017.

[26] S. Verma and J. Rubin, “Fairness definitions explained,” *In Proceedings of the International Workshop on Software Fairness*, pp. 1–7, 2018.

[27] A. Agarwal, M. Dudik, and Z. S. Wu, “Fair regression: Quantitative definitions and reduction-based algorithms,” *In Proceedings of the 36th International Conference on Machine Learning*, 2019.

[28] N. AniSaxenaa, K. Huang, E. DeFilippis, G. Radanovic, D. C. Parkes, and Y. Liu, “How do fairness definitions fare? testing public attitudes towards three algorithmic definitions of fairness in loan allocations,” *Artificial Intelligence*, vol. 283, 2020.

[29] F. Calmon, D. Wei, B. Vinzamuri, K. N. Ramamurthy, and K. R. Varshney, “Optimized pre-processing for discrimination prevention,” *In Advances in Neural Information Processing Systems*, p. 3992–4001, 2017.

[30] F. Kamiran and T. Calders, “Data preprocessing techniques for classification without discrimination,” *Knowledge and Information Systems*, vol. 33, p. 1–33, 2012.

[31] F. Kamiran, I. Zliobaite, and T. Calders, “Quantifying explainable discrimination and removing illegal discrimination in automated decision making,” *Knowledge and Information Systems*, vol. 35, p. 613–644, 2013.

[32] 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*, p. 35–50, 2012.

[33] M. B. Zafar, I. Valera, M. G. Rodriguez, and K. P. Gummadi, “Fairness constraints: Mechanisms for fair classification,” *In Artificial Intelligence and Statistics*, p. 962–970, 2017.

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

[35] F. Kamiran, A. Karim, and X. Zhang, “Decision theory for discrimination-aware classification,” *In Proceedings of the 2012 IEEE 12th International Conference on Data Mining*, p. 924–929, 2012.

[36] B. Woodworth, S. Gunasekar, M. I. Ohannessian, and N. Srebro, “Learning non-discriminatory predictors,” *In Conference on Learning Theory*, p. 1920–1953, 2017.

[37] S. A. Friedler, C. Scheidegger, and S. Venkatasubramanian, “On the (im)possibility

of fairness," 2016.

[38] R. Binns, "On the apparent conflict between individual and group fairness," *In Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 2020.

[39] P. Lahoti, K. P. Gummadi, and G. Weikum, "ifair: Learning individually fair data representations for algorithmic decision making," *In Proceedings of the IEEE 35th International Conference on Data Engineering*, 2019.

[40] S. Kolouri, S. Park, M. Thorpe, D. Slepčev, and G. K. Rohde, "Optimal mass transport: Signal processing and machine-learning applications," *IEEE Signal Process Mag*, vol. 34, pp. 43–59, 2017.

[41] R. Berk, H. Heidari, S. Jabbari, M. Kearns, and A. Roth, "Fairness in criminal justice risk assessments: The state of the art," *Sociological Methods & Research*, 2018.

[42] S. Corbett-Davies and S. Goel, "The measure and mismeasure of fairness: A critical review of fair machine learning," *In Proceedings of the 35th International Conference on Machine Learning*, 2018.

[43] J. K. S. Mullainathan and M. Raghavan, "Inherent trade-offs in the fair determination of risk scores," *In 8th Innovations in Theoretical Computer Science Conference*, 2017.

[44] William Dieterich and Christina Mendoza and Tim Brennan, "Compas risk scales: demonstrating accuracy equity and predictive parity," 2016. [Available from `http://go.volarisgroup.com/rs/430-MBX-989/images/ProPublica_Commentary_Final_070616.pdf` ; accessed 30-April-2020].

[45] A. Chouldechova, "Fair prediction with disparate impact: A study of bias in recidivism prediction instruments," *Big Data*, vol. 5, 2017.

[46] Julia Angwin and Jeff Larson and Surya Mattu and Lauren Kirchner, "Machine bias risk assessments in criminal sentencing," 2016. [Available from `https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing` ; accessed 30-April-2020].

[47] Dheeru Dua and Casey Graff, "Uci machine learning repository," 2017. [Available from `http://archive.ics.uci.edu/ml` ; accessed 20-May-2020].

[48] "Correlation ratio," 2020. [Available from `https://www.statisticssolutions.com/directory-of-statistical-analyses-correlation-ratio/` ; accessed 1-July-2020].

[49] “Conduct and interpret a point-biserial correlation,” 2020. [Available from https://www.statisticssolutions.com/point-biserial-correlation/ ; accessed 1-July-2020].

[50] “Nominal association: Phi and cramer’s v,” 2020. [Available from http://www.people.vcu.edu/~pdattalo/702SuppRead/MeasAssoc/NominalAssoc.html ; accessed 1-July-2020].

A. Appendix

A.1. Massaging

We modify M individuals in each demographic group, the statistical parity difference is:

$$\begin{aligned} P(\tilde{Y}=1|S=0)-P(\tilde{Y}=1|S=1) &= \frac{|\tilde{Y}=1\cap S=0|}{|S=0|}-\frac{|\tilde{Y}=1\cap S=1|}{|S=1|} \\ &= \frac{|Y=1\cap S=0|+M}{|S=0|}-\frac{|Y=1\cap S=1|-M}{|S=1|} \\ &= \frac{|Y=1\cap S=0|}{|S=0|}-\frac{|Y=1\cap S=1|}{|S=1|}-M\Big(\frac{1}{|S=0|}+\frac{1}{|S=1|}\Big) \\ &= \frac{|Y=1\cap S=0|}{|S=0|}-\frac{|Y=1\cap S=1|}{|S=1|}-M\frac{N}{|S=0|\times|S=1|} \end{aligned}$$

We aim to reduce the left hand side to zero, thus

$$M=\frac{1}{N}\Big[\Big(\frac{|Y=1\cap S=0|}{|S=0|}-\frac{|Y=1\cap S=1|}{|S=1|}\Big)\times|S=0|\times|S=1|\Big]$$

How to select individuals to relabel using a ranker which is a model that can produce scores (predicted probabilities of being positive):

1. Learn a ranker on training data and produce scores for each individual

2. Order the group of individuals with $S=0$ and $Y=0$ in descending scores and call it G_0, and order the group of individuals with $S=1$ and $Y=1$ in ascending scores and call it G_1

3. Change the labels of the top M individuals in G_0 from negative to positive and change the labels of the top M individuals in G_1 from positive to negative

4. Keep the labels of the other individuals the same, now we have a new outcome label $\tilde{Y}$ for the training data

5. Train any classifier on the training data with modified outcome labels.

A.2. Reweighing

We expect S and Y to be statistically independent.
The expected probability is

$$P_{\text{exp}}(S=s \cap Y=y) = P(S=s)P(Y=y) = \frac{|S=s|}{|D|} \times \frac{|Y=y|}{|D|}$$

where $|D|$ indicates the number of individuals in the training data set.
The observed probability is

$$P_{\text{obs}}(S=s \cap Y=y) = \frac{|S=s \cap Y=y|}{|D|}$$

For each observation in training data, assign weight

$$W_{s,y} = \frac{P_{\text{exp}}(S=s \cap Y=y)}{P_{\text{obs}}(S=s \cap Y=y)} = \frac{|S=s| \times |Y=y|}{|D| \times |S=s \cap Y=y|}$$

We can show that after multiplying each individual by its weight, the weighted data satisfies statistical parity.

$$P_w(Y=1|S=0) - P_w(Y=1|S=1) = \frac{|Y=1 \cap S=0| \times W_{0,1}}{|S=0|} - \frac{|Y=1 \cap S=1| \times W_{1,1}}{|S=1|}$$

$$= \frac{|Y=1 \cap S=0|}{|S=0|} \times \frac{|S=0| \times |Y=1|}{|D| \times |S=0 \cap Y=1|} - \frac{|Y=1 \cap S=1|}{|S=1|} \times \frac{|S=1| \times |Y=1|}{|D| \times |S=1 \cap Y=1|} = 0$$

A.3. Disparate Impact Remover

The modifying procedures when focusing on a single numerical X:

- Let $F_s : X_s \to [0,1]$ be the cumulative distribution function where X_s is the domain of X condition on $S=s$ and $F_s^{-1} : [0,1] \to X_s$ be the quantile function.
- Define median distribution $A : F_A^{-1}(u) = \text{median}_{s \in S} F_s^{-1}(u)$ for $u \in [0,1]$
- $\forall x \in X_s$, the corresponding $\tilde{x} = F_A^{-1}(F_s(x))$
- Delete the sensitive attribute from the data, the resulting data is $(\tilde{X}, Y)$

The above shows the procedures of obtaining a fully repaired data (distributions are the same), we can obtain a partially repaired data by:

$$\tilde{x} = (1-\lambda)x + \lambda F_A^{-1}(F_s(x))$$

where $\lambda \in [0,1]$ indicates the repair level. $\lambda = 0$ stands for unmodified data and $\lambda = 1$ stands for fully modified data.

A.4. Learning Fair Representations

Let $X^+ = \{\mathbf{x} \in X : S = 1\}$, $X^- = \{\mathbf{x} \in X : S = 0\}$, X_0 be the training dataset (N observations) with $X_0 = X_0^+ \cup X_0^-$, $\mathbf{x}_n = (x_{n1}, x_{n2}, \dots, x_{np}) \in X$ and $(\alpha_1, \alpha_2, \dots, \alpha_p)$ is the weight for each feature. Let Z be a multinomial random variable with K possible outcomes-“prototypes" which are associated with K vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_K$ where $\mathbf{v}_k = (v_{k1}, v_{k2}, \dots, v_{kp})$

A probabilistic mapping $X \to Z$ via softmax:

$$M_{n,k} = P(Z = k|\mathbf{x}_n) = \exp(-d(\mathbf{x}_n, \mathbf{v}_k, \alpha))/\sum_{j=1}^{K} \exp(-d(\mathbf{x}_n, \mathbf{v}_j, \alpha))$$

where $d(\mathbf{x}_n, \mathbf{v}_k, \alpha) = \sum_{i=1}^{p} \alpha_i (x_{ni} - v_{ki})^2$ is the Euclidean distance measure.

The reconstructions of $\mathbf{x}_n$ from Z:

$$\hat{\mathbf{x}}_n = \sum_{k=1}^{K} M_{n,k} \mathbf{v}_k$$

The prediction $\hat{y}_n$ from $M_{n,k}$ and parameters $\{w_k\}$ between 0 and 1:

$$\hat{y}_n = \sum_{k=1}^{K} M_{n,k} w_k$$

Then we learn two sets of parameters $\{\mathbf{v}_k\}$ and $\{w_k\}$ by L-BFGS to minimize $L = A_z \cdot L_z + A_x \cdot L_x + A_y \cdot L_y$ where A_x, A_y, A_z are hyperparameters and

$$L_z = \sum_{k=1}^{K} |M_k^+ - M_k^-| = \sum_{k=1}^{K} |\frac{1}{|X_0^+|} \sum_{n \in X_0^+} M_{n,k} - \frac{1}{|X_0^-|} \sum_{n \in X_0^-} M_{n,k}|$$

aims at group fairness/statistical parity,

$$L_x = \sum_{n=1}^{N} (\mathbf{x}_n - \hat{\mathbf{x}}_n)^2 = \sum_{n=1}^{N} \sum_{i=1}^{p} (x_{ni} - \hat{x}_{ni})^2$$

quantifies the information lost in the new representation, and

$$L_y = \sum_{n=1}^{N} -y_n \log \hat{y}_n - (1 - y_n) \log(1 - \hat{y}_n)$$

requires the prediction of y is as accurate as possible.