

Faculté des sciences

Obtaining Fairness and Removal of Algorithmic Bias in Machine Learning using Optimal Transport Theory

Auteur : Bryan LEURQUIN

Promoteur : Prof. Johan SEGERS

Lecteurs : Prof. Eugen PIRCALABELU, Prof. Donatien HAINAUT

Année académique 2020-2021

Mémoire de fin d'études présenté en vue de l'obtention
du master en statistiques : **orientation générale, à finalité spécialisée**

Contents

1	Introduction	1
2	Definition, Detection and Quantification of Fairness	4
2.1	Motivation and Framework for the Fairness Problem	4
2.2	Exploratory Analysis of the <i>Adult Income Dataset</i>	6
2.3	Quantifying Fairness	11
3	Futility of Removing Bias with the Use of Naive Procedures	16
3.1	Removing the Sensitive Variable	16
3.2	Positive Discrimination	18
3.3	Discussion	20
4	Mathematical Framework for Optimal Transport Theory	21
4.1	Probability vectors and measures	21
4.2	Assignment and Monge Problem	22
4.3	Kantorovich Relaxation	24
4.4	Dual problem	26
4.5	Metric Properties of Optimal Transport	28
4.6	The Kantorovich linear program	30
4.7	Complementary	31
5	Obtaining Fairness using Optimal Transport Theory	32
5.1	Repairing the data	32
5.1.1	Total repair	33
5.1.2	Partial repair	37
5.2	Application with Simulated Data	41
5.3	Application to a Real Example	44
6	Conclusion	51
6.1	Overview and Results	51
6.2	Limitations and Future Work Avenues	52

Chapter 1

Introduction

In today's day and age machine learning is omnipresent. It directly or indirectly affects or interferes in almost every aspect of our lives and has without a doubt made human life easier on a day to day basis. Machine learning can be encountered everywhere, in our private life, whether it be when we are watching Netflix and are given suggestions on new TV shows to watch next or browsing on Amazon and receive suggestions on what book could interest us, as well as in our professional lives, where machine learning algorithms are able to make decision processes easier by showing us which applicant would make for the best hire for a firm or whether an individual should qualify for a loan at the bank. Not only are machine learning algorithms already widespread, they could be ever more present in the near future with the great strides made towards the construction of fully self-driving cars and the development of algorithms capable of recognising cancer cell on radiographs. The democratisation of machine learning in most fields can be attributed to the simple fact that it is able to make more decisions than humans and do so faster than any one ever could. The speeding up of decision making processes is an enormous gain for firms which potentially save a lot of money but it can also be the difference between life and death in medical fields. While this all seems perfect, let us remember that if something seems too good to be true, it most likely is. In the case of machine learning algorithms, the potentially dangerous pitfall is that since they are trained using databases regrouping information on potentially thousands of individuals, some of this data may be biased and thus introduce discriminatory behaviour to our decision making algorithms which can lead to unfortunate consequences. The emergence and implementation of machine learning in almost every facet of our lives begs for a thorough multidisciplinary investigation on how to detect and remove bias in machine learning algorithms.

Since machine learning is not naturally occurring and we have to create it, we could ask ourselves how could an algorithm possibly be prejudiced. Couldn't we simply built an algorithm without introducing discriminatory behaviour ? Lets us recall that a decision making algorithm is built by giving it information on many thousands of individuals which the algorithm then analyses to find patterns linking the response variable to all the others. This data can be anything from the age of the individual to their job, their height, weight, etc. Datasets can be as thorough as we want and large datasets can, unbeknownst to the user, harbour significant bias towards certain group of individuals in a population. Bias can stem from multiple avenues such as the historical or socio-economic background of the person being considered. This problem has created the need of a new field of research in machine learning, that of fair learning. Fair learning is a field that specialises

on the study of the impact of bias in databases and the repercussions that can arise from the use of such databases in machine learning. Fair learning also looks at how to limit the reach of bias and avoid algorithms making decisions on the basis of characteristics such as sex, race, political or sexual orientation, religion, etc.

This paper is soundly grounded in this rather recent field that is fair learning and seeks to shed some new light on a mathematical approach used to eliminate or at the very least dampen the effects of bias in machine learning leading to a fairer treatment of individuals.

This work is divided into four chapters. Chapter 2 will specialize in defining fairness in the field of machine learning as well as how to quantify fairness. Chapter 3 will introduce and put to the test two methods that follow General Data Protection Regulations concerning mitigating bias in machine learning. These methods will serve to illustrate the need for procedures such as optimal transport to eliminate bias in decision making algorithms. We will then move onto introducing and laying down the framework for optimal transport theory in Chapter 4. We will look at what is the assignment problem and how to solve it. Finally, Chapter 5 will be an application of optimal transport on simulated data as well as on the *Adult Income Dataset* [1]. We will look at the different data repair procedures and apply them.

The motivation behind the rising need to be able to detect difference in treatments between different groups of individuals lies in the ever expanding use of machine learning as a tool to make decisions. Machine learning algorithms are entirely dependent on the data that is fed to them when building them, in effect, if discriminatory patterns are given to the algorithm, it will learn from them and use them as a basis for future decisions. This can have disastrous consequences. For example, in [2], it was found that an algorithm in charge of rating the likelihood of recidivism of inmates in the United States, would rate black individuals more harshly than whites. It is for these reasons and more that a multidisciplinary approach to detect and remove bias in machine learning is paramount. In order to remove bias, one firstly needs to be able to reliably quantify it. The two main bias quantifiers used throughout the literature are disparate impact (DI) and balanced error rate (BER). Given a classifier g such as a decision tree, disparate impact was defined in [3] as the ratio of the probability of success given that the individual is part of the minority class over the probability of success given that the individual is part of the default class.

An individual is part of a minority class if they are part of a group of people that supposedly suffer from discrimination and part of the default class if they are not part of that group. For example, regarding sensitive variables such as sex or race, the minority classes would be women and non-white respectively due to the fact that men and white individuals allegedly benefit from being favoured in everyday life. An algorithm presents discriminatory bias when the predictive outcome is intimately related to the protected attribute. This means that if we are given only the outcome of a decision making algorithm and we can accurately and confidently guess what class each individual was a member of, then we may be in a situation when the decision was heavily influenced by the sex or the race of the individual for example. This predictability of the protected attribute can be computed with the help of the BER. This quantify can loosely be understood as the probability of the decision being the opposite of the expected outcome. The lower the value of the BER, the higher the probability of correctly guessing the value of the protected variable based only on the value of the output. The best bound achievable is $BER = \frac{1}{2}$ as this would translate to being correct half of the time if we were to guess the class of the protected attribute based on the decision of the algorithm.

The next chapter, Chapter 3, will focus on fixing unfairness in machine learning by applying two methods: the first is removing the sensitive variable from the data and the second is the process of positive discrimination. Removing the sensitive variables from the dataset is the recommendation made by the General Data Protection Regulation in order to combat bias in machine learning. While this should most definitely always be done to preserve anonymity, simply removing sensitive variables does little to reduce bias. In effect, due to socio-economic and/or historical reasons, sensitive variables may very well be reconstructed from the set of other variables in the dataset. This measure on its own is quasi useless. Another procedure is the use of positive discrimination. This method essentially boils down to lowering the acceptance threshold in favour of the minority class while keeping it at the original value for the default class. This threshold can be lowered as much as possible, to the point of artificially increasing the disparate impact to a level of 0.8 which legally would indicate no bias in the data. However, this method is fundamentally unfair as different groups are treated differently. This opens up new legal problems far beyond the scope of this work. These two methods serve as examples to illustrate the dire need of a more thorough and sound method of removing bias in datasets.

Chapter 4 will set the foundations and mathematical framework of the optimal transport theory. In the words of Gaspard Monge ([4]), optimal transport can be described with the following problem: a person is told to move a large pile of sand from one point to another and erect this pile of sand with a specific shape, while wanting to minimise the amount of energy or time spent carrying the sand. Optimal transport is thus a mathematical field which specializes in comparing two distributions and looks at all the possible ways to reshape the first distribution into the second distribution under the constraints that dictates the cost of displacing one unit of mass of the first distribution to a point in the second distribution.

In this work, optimal transport theory will be used to fracture the link between the variables in a dataset and the protected attribute S , this is referred to as repairing the data. The last chapter, Chapter 5, is an overview of how optimal transport procedures are implemented in order to repair data. These procedures can be classified into two groups: total and partial repair procedures. Total repair is when a dataset is divided according to the value of the sensitive variable and the resulting conditional distributions are all moved to a single and common distribution. Partial repair on the other hand are procedures in which the conditional distributions are move towards a common distribution, without reaching it. Partial repair procedures offer more control on the amount of repair desired in the data to reach the wanted disparate impact level. Users also have more control over the loss in classification accuracy and the predictability of the sensitive variables as data is repaired. This chapter also includes an application of these repair methods using the *Adult Income Dataset*.

Chapter 2

Definition, Detection and Quantification of Fairness

2.1 Motivation and Framework for the Fairness Problem

Over the last decade, the automatisisation of decision making processes has increased exponentially. Originally designed to improve recommendation systems on the internet for services such as streaming or shopping services, machine learning algorithms and artificial intelligence have now become the baseline for any and all decision making operations such as in the automobile industry with the rise of self-driving vehicles, in banking and insurance industries with algorithms that are used to quantify a client's credit risk [5] [6] or even in the justice system where algorithms such as the COMPAS algorithm are used to assess potential recidivism risk in the United States [2].

It is quite easy to understand why machine controlled algorithms are so widely used nowadays, in effect, machines do not get tired and thus can work endlessly and can work faster than any human could ever. Machines can thus lower costs incurred by a company and can also free up human workforce for other tasks. They are a massive quality of life improvement. Moreover, we know that humans are inherently biased beings. Our decision making stems from our beliefs whether they be conscious or otherwise. Therefore one may think that using automatized algorithms and bots to help us in our decision making would render the process unbiased and fair in regards to all aspects as machines do not prejudice against certain beliefs or against certain socio-economic backgrounds for instance. Over the last decade, the use of machine learning in decision making situations has been expanding rapidly.

The tricky part when using machine learning algorithms reside in how they are built. In effect, data used to train an algorithm could, unbeknownst to the user, harbor bias with regards to a certain group of individuals and thus teach discriminatory behaviour to the algorithm. This discrimination could be linked to several factors such as the historical and socio-economic background of the person being considered. Let us consider an individual A who resides and goes to school in a part of a city which is known to provide a lesser standard of education than other parts of the city. This could be for historical reasons, for example mass immigration could have seen certain poor populations setting residence in that part of the city. These groups being poor could have resorted to petty crimes to make a living, making that part of the city less desirable, thus

less funds and infrastructure were allocated. All this could contribute to having a lesser standard of education in schools. Let us say that individual A applies for a job and all applications are run through an algorithm which helps weed out unwanted applicants. If the algorithm had been trained on data including individuals living in and with high school diplomas from the same part of the city as A and was told that those individuals should be rejected, it could quickly draw a parallel between what it had been taught and A. The algorithm could automatically reject A’s application even if he would have been more than qualified for the position. In our case, the algorithm could very well have learnt to discriminate against certain ethnicities as A and his peers stemmed from immigration. A real life example showing how an algorithm could discriminate on the basis of race or ethnicity is the COMPAS algorithm which is used to predict the risk of recidivism of inmates in the United States ([2]). In effect, it was found that the COMPAS algorithm used to assess potential recidivism risk in the United States predicted significantly higher proportion of false positives for black individuals than for white people. Such discriminatory behaviours open up companies to serious and costly legal battles. The notion of fairness, defined as the impartial and just treatment or behaviour without favouritism or discrimination, thus becomes paramount when developing decision making algorithms.

The first imperative when trying to study the impact of bias in databases and the repercussions that can arise from the use of such databases in machine learning is to quantify this bias. An empirical definition of bias permits users to compare the “amount” of bias from one algorithm to another and enables users to make changes accordingly. In this study, we will firstly try to quantify fairness so as to use several methods whose aim is to modify, or repair, the data to nullify the bias or at the very least bring this bias down to an acceptable level and keep the algorithm from learning discriminatory and potentially false patterns. Machine learning algorithms are widely used in a number of sensitive areas such as human resources, hiring or lending policies. Thus, having an algorithm making decisions with inherent bias can prove damaging to a firm’s reputation and/or costly if a discriminatory pattern is indeed proven, making it a priority to minimize the bias and bring it under a legal and acceptable limit.

For example, in 2003, the New Haven Department conducted oral and written exams for promotion to Lieutenant and Captain. Among the 118 firefighters that took the test, 77 took the test to become Lieutenant and 41 took the Captain exam. The test scores would remain valid for two years, during which a total of 16 Lieutenant and 8 Captain positions became available. The City mandated that if x promotions were made, then the $x + 2$ top scores should be considered. For the Lieutenant exam, 15 white and 3 black people were considered whereas for the Captain exam, the top 10 top scorers included 8 whites and 2 Hispanics. Because of a perceived lack of minorities able to be promoted to an existing position, the City of New Haven invalidated the exam results and did not promote any of the test takers. In turn, those that would have been considered for promotion decided to sue the city on grounds of reverse discrimination. In the hearing, the court accepted the city’s argument stating that the decision to cancel the exam results was based on the effect of disparate impact affecting minorities. In order to rule in favor or against this stance, the United States District Court for the District of Connecticut used the government’s “four-fifths rule” which is often used as a legal threshold in decision making. The four-fifths rule, or 80% rule, is defined as:

“A selection rate for any race, sex, or ethnic group which is less than four-fifths ($4/5$) (or 80%) of the rate for the group with the highest rate will generally be regarded by the

federal enforcement agencies as evidence of adverse impact, while a greater than four-fifths rate will generally not be regarded by federal enforcement agencies as evidence of adverse impact. Smaller differences in selection rate may nevertheless constitute adverse impact, where they are significant in both statistical and practical terms or where a user's actions have discouraged applicants disproportionately on grounds of race, sex, or ethnic group. Greater differences in selection rate may not constitute adverse impact where the differences are based on small numbers and are not statistically significant."

[7]

The four-fifths rule was first introduced by the State of California Fair Employment Practice Commission (FEPC) in 1971 and has since served as the benchmark to determine whether an algorithm presented discriminatory behaviour or not. This rule serves as the basis for an index called Adverse Impact. Adverse impact is the measure of unfairness defined as: "a substantially different rate of selection in hiring, promotion, or other employment decision which works to the disadvantage of members of a race, sex, or ethnic group" [7]. The adverse impact ratios are the ratios of the pass rates where the numerator is that pass rate of a minority and the denominator is that of the favoured class. In this case, the adverse impact ratios for blacks and Hispanics were 54% and 34.4% respectively. It was found that since both ratios were well below the 80% threshold allowed, the district court upheld the City's decision to not certify the test results.

This example perfectly illustrates how being able to quantify bias can shape the way we proceed to make more informed and fair decisions. This is a great advantage for all as every individual knows that if he or she is refused a job, it is not because of petty discrimination and the form knows it is shielded from frivolous lawsuits.

One of the most well known and important applications of machine learning algorithms is in credit scoring. Decision making algorithms are widely used to determine whether a client can qualify for a credit or for a loan. For example, the Fair Isaac Corporation (FICO) score in the USA and the Schutzorganisation für Allgemeine Kreditsicherung (SCHUFA) score in Germany are two credit rating scores determined entirely by algorithmic processing of data. However, such algorithms have often been outed as biased namely in [5] and [6]. Being able to quantify bias permits us to make changes to both these scoring procedures in order to make them fairer for all individuals.

2.2 Exploratory Analysis of the *Adult Income Dataset*

In this paper, we will work with the *Adult Income Dataset* [1] for which the aim is to predict whether an individual has an income of more or less than 50 000\$ per annum. This decision is based on multiple factors. First of all, this dataset was built upon an existing census made in the United States in 1994, it contains 15 variables which provide information on 48 842 individuals. This table is used to predict whether the annual income of each individual is more or less than 50 000\$. While predicting whether and individual earns more than a certain amount per year may not be the most interesting analysis, it serves as an illustration of what can be done in the banking or insurance industries since the machine learning procedures are similar to those used to assess the credit risks of a client. Two major advantages that this dataset possesses in contrast to most other credit score or loan databases, are that we are given both the true value of the target variable as well as the values of potential sensitive variable such as the gender and ethnicity of each individual, making

it possible to test the robustness and accuracy of any algorithmic procedure. This dataset has also been shown, and this will be shown again throughout this paper, to harbour discriminatory behaviour with respect to both `sex` and `race` variables. These reasons make the *Adult Income Dataset* one of the most interesting datasets in the context of fair learning. However this dataset is very extensive. The variables in the datasets are as follow:

- `age` : age of the individual,
- `workclass` : whether the individual works in private practice, is self or government employed,
- `fnlwgt` : no clear indication on what this variable stands for,
- `education` : highest degree achieved by the individual,
- `education_years` : number of years spend in the educational system,
- `marital_status` : whether the individual is single, married, divorced or widowed,
- `occupation` : job of the individual,
- `relationship` : no clear indication on what this variable stands for,
- `race` : race of the individual,
- `gender` : sex of the individual - henceforth referred to as `sex`,
- `capital.gain` : profit earned on the sale of the individual's assets,
- `capital.loss` : loss made on the sale of the individual's assets,
- `hpw` : hours per week the individual spends at work,
- `income` : binary variable indicating whether the individual has an annual income of more or less than 50 000\$.

As good as this dataset is, it is not perfect and needs to be pre-processed before using it for analysis. For example, some variables are redundant and others were not well defined, it is thus not particularly interesting to keep them. Such variables are as follow:

- `fnlwgt` was removed since it had no clear meaning,
- `relationship` was removed since it is redundant with `gender` and `marital_status`,
- `native.country` and `race` were removed since this paper will focus on sex-based discrimination,
- `education` was removed since it is redundant with `educational.num`.

Variable `sex` were changed to binary variables where 1 is the default class and 0 the minority. In this context, a value of 1 would translate to the individual being male whereas a value of 0 would mean the individual is female. We also made the conscious choice to remove any and all rows presenting one or more missing value as we would like to train the algorithm on complete data. After removal of those rows, we are left with a database containing ten columns and 46 033

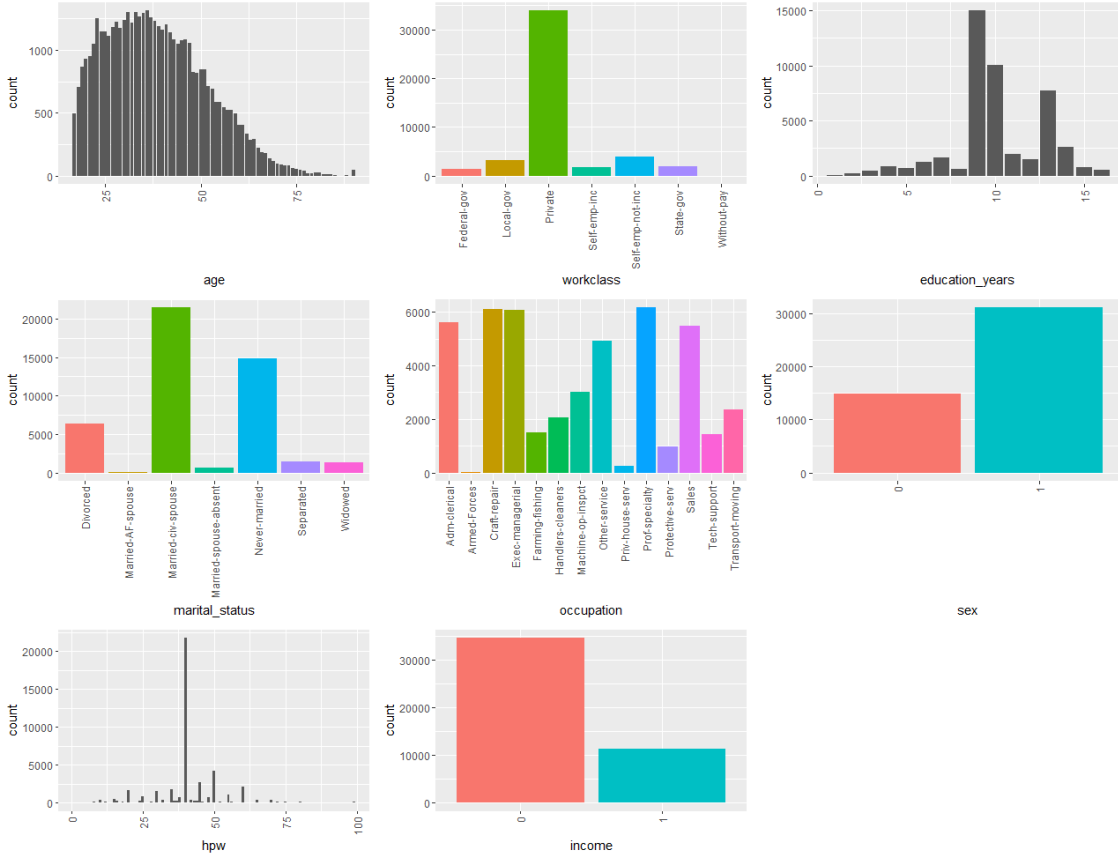


Figure 2.1: Distributions of every variable used in the *Adult Income Dataset*. From left to right, top to bottom, we have the distributions of `age`, `workclass`, `education_years`, `marital_status`, `occupation`, `sex`, `hpw` and `income`.

observations. Below we have a graphical representation of the distribution of each variable except `capital.gain` and `capital.loss` as most entries are null for these two columns.

We can clearly see that we have an uneven split of the data with respect to the variable `sex`. This variable is one that we would expect to be the sensitive variable. A preliminary analysis of the dataset shows us that there is an underrepresentation of the “woman” class in the data, making it the minority class with respect to `sex`. The underrepresentation of certain classes could be potentially problematic when building a decision making algorithm. We will be focusing on the variable `sex` as the procedures undertaken throughout this paper are similar when using the sensitive variable `race`.

We can also observe on the mosaic plot below an unbalanced representation of the default and minority classes of the variable `sex` with respect to the `income` variable. In effect, we can observe on the right a larger proportion of males making more than 50 000\$ a year than females whereas on the left we can see a larger proportion of white making more than 50 000\$ a year.

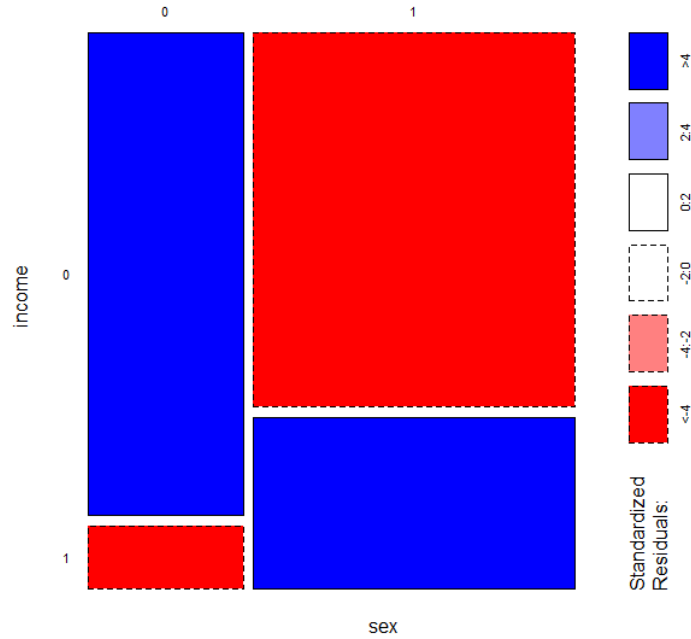


Figure 2.2: Unbalanced representation of the default and minority classes with respect to the target variable. For **income**, a value of 0 means that the individual makes less than 50 000\$ a year while 1 means the individuals makes more than 50 000\$ a year. For **sex**, the value 0 means the individual considered is a woman, while a value of 1 means we have a male subject.

In effect, blue shades mean that there are more observations in this cell than what would be expected under the null hypothesis whereas red shades mean there are fewer observations than expected. The null hypothesis in this case is the independence of the two variables. If **income** and **sex** were truly independent from one another then we would expect roughly equal sized cells. This mosaic plot shows us that there are significantly more men earning more than 50 000\$ per year than there are men earning less while also pointing out the exact opposite relationship for women. We thus have a strong dependence between our sensitive variable and our response variable, this could very well indicate that any machine learning algorithm trained on this data is susceptible to learning sexist behaviour.

A point which needs to be addressed is whether the apparent discrimination we observe is really due to the sex or can it be explained by another variable. For example, it could be the case that in our dataset, women generally have a lower level of education which could explain the trend of women earning less than men. Figure 2.3 shows us the distribution of men and women with respect

to the number of years spent in the academic system. The two distributions seem to mimic one another in shape. This would indicate that there are virtually no differences between men and women when it comes down to the level of education and thus the disparity in the treatment between the two is due to sex-based discrimination.

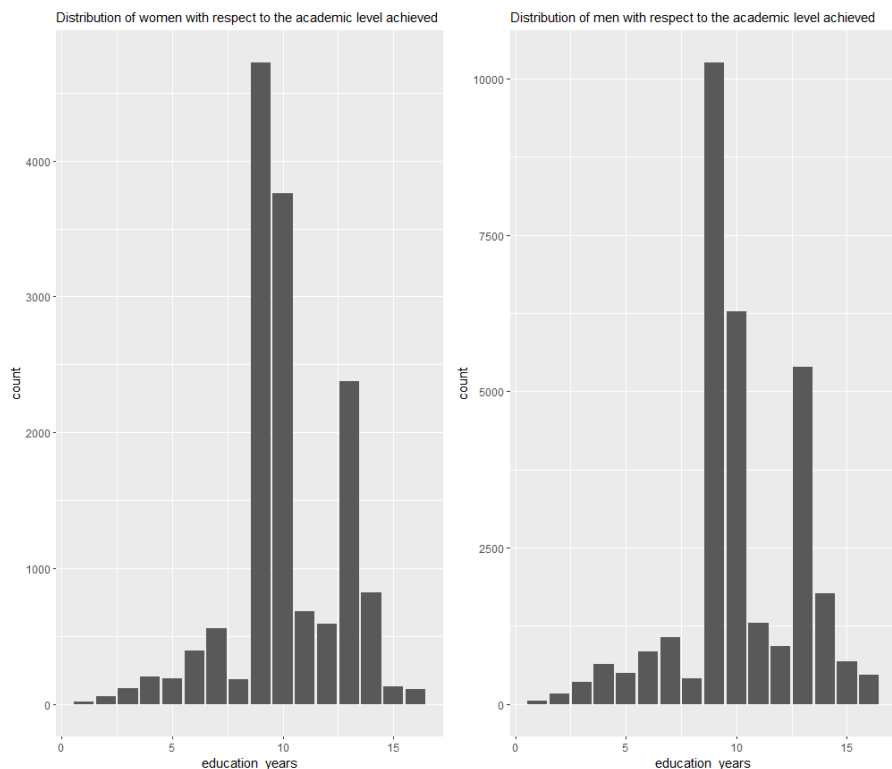


Figure 2.3: Distribution of the variable `education_years` for women (*left*) and men (*men*)

We can test whether we truly have a discriminatory behaviour or whether we could explain this with the level of education of women being on average lower than that of men by performing a conditional independence test. Using the library `bnlearn` and the function `ci.test()` in R, we test if the two variables `sex` and `income` are independent conditionally on `education_years`. The null hypothesis being tested is that both variables are conditionally independent with respect to `education_years`. In effect, we are verifying whether the probability of success, given the sex and level of education of an individual is equal to the probability of success given the level of education of the individual. If we have this equality, then the sex of the individual has no bearing on the outcome of the algorithm and thus we would have no discrimination. Once the test has been conducted, the R console outputs a p-value of less than $2.2e^{-16}$ meaning that the null hypothesis is to be rejected in favour of the alternative stating the `sex` and `income` are conditionally dependent. This test was performed for every variable in our dataset and in each case, the null hypothesis was rejected in favour of the alternative, stating that both `sex` and `income` were conditionally dependent with respect to the third variable. We can therefore conclude that the *Adult Income Dataset* does house

sex-based discrimination, therefore any decision making algorithm trained on this data will end up learning sexist behaviour.

2.3 Quantifying Fairness

In order to reliably quantify bias in a dataset, we will make use of two types of fairness quantifiers: disparate impact, previously also referred to as adverse impact in section 2.1 and balanced error rate. These fairness indicators are based upon the ability to infer on the value of certain variables characterizing individuals, which are prone to bias. Such variables include gender, race, age or any other type of potential discriminatory attribute. These type of variables are called protected or sensitive attributes. An algorithm is considered fair with respect to these attribute if the value of the protected attribute of an individual cannot be inferred from the outcome of the algorithm. A naive approach to solving this problem is to ignore these variables when building the algorithm. This method will be discussed in Chapter 3 where we will see why it unfortunately does not work, as a protected variable can be potentially closely correlated to other variables and thus could be inferred from these other variables.

Let us introduce the following: a binary variable $Y \in \{0, 1\}$ which codifies whether the outcome is a success or not, a set of covariates $X \in \mathbb{R}^d, d \geq 1$ which encompasses the data upon which the decision will be made, and a second binary variable $S \in \{0, 1\}$. This last variable, called protected variable, dissects the population into two categories which classifies whether an observation is part of the minority class, supposedly unfavoured and discriminated against by the algorithm and takes the value $S = 0$, or part of the default class, which is supposedly favored and takes the value $S = 1$. For example, in a gender based discrimination scenario, the value 0 would be attributed to “female” whereas 1 would be attributed to “male”.

The aim is to be able to predict the value of $Y \in \{0, 1\}$, where $Y = 0$ is a failure and $Y = 1$ a success, using the set of covariates $X \in \mathbb{R}^d, d \geq 1$ and a family of binary classifiers $g \in \mathcal{G}: \mathbb{R}^d \rightarrow \{0, 1\}$, such that for every classifier g , the outcome of the prediction will be $\hat{Y} = g(X)$. We want to determine whether there is an algorithmic bias and if so, quantify it. Let us note that most algorithms do not output a strictly binary value but rather a continuous probability spectrum and the decision to mark an outcome as successful or not is based on a predefined threshold.

We say that an algorithm presents discriminatory bias when the predictive outcome is intimately related to the protected attribute as statistical inference on Y may most likely lead to figuring out the distribution of the protected class. The issue at hand is to first and foremost quantify the amount of bias present in the output of a specific algorithm as without this quantity known, it would be arduous to erase this discrimination. Two methods of bias quantification were introduced in *Certifying and Removing Disparate Impact* ([3]): disparate impact and the balanced error rate.

When the real distribution of the variable Y is known, we say that a classifier g achieves overall accuracy equality if:

$$P(g(X) = Y \mid S = 0) = P(g(X) = Y \mid S = 1). \quad (2.1)$$

This indicates that the probability of a correct classification is independent of the protected variable group. However, in most cases the true values of the response variable Y are unavailable

as they may be sensitive data that the owner might not be willing to share. In this case, we must take into account the predictive variable $\hat{Y}$ instead. We say that a classifier g achieves statistical parity if:

$$P(g(X) = 1 \mid S = 0) = P(g(X) = 1 \mid S = 1). \quad (2.2)$$

This indicates that the probability of predicted success does not depend on the group. For example, if we were to have two people, one Caucasian and one Hispanic, then the algorithm would achieve statistical parity if they both have the same probability of being credited a loan at a bank.

Through these definitions, the concept of disparate impact of the classifier g in order to quantify fairness of the algorithm can be written as

$$DI(g, X, S) = \frac{P(g(X) = 1 \mid S = 0)}{P(g(X) = 1 \mid S = 1)}, \quad (2.3)$$

where we suppose that $0 < P(g(X) = 1 \mid S = 0) \leq P(g(X) = 1 \mid S = 1)$.

Disparate impact is a measure of fairness in the sense that smaller values of this quantity indicate a greater bias in the algorithm. In effect, let us suppose that we have a classifier g^* for which the disparate impact is $DI(g^*, X, S) < 1$. We would conclude that the probability of success given that the individual is part of the unfavoured class $S = 0$ is less than the probability of success if an individual were to be part of the default class $S = 1$. A classifier g is said to have disparate impact at level τ with respect to the joint distribution (S, X) if:

$$DI(g, X, S) \leq \tau, \quad (2.4)$$

where $\tau \in]0, 1]$. If $\tau = 1$, the probability of a successful outcome is the same whether the individual is from the minority or majority group and thus the classifier g for which this outcome is verified is said to achieve statistical parity.

Another way to quantify fairness is the Balanced Error Rate (BER). Let us set a practical example. Let us suppose that a person A designs a decision making algorithm based on a dataset that is composed of a protected attribute S and unprotected variables X which outputs a binary decision Y . The protected attribute cannot be used in the decision making whether it would be through the use of the attribute directly or through the use of variables that are too closely related to the protected class. A person B who would like to verify that this is the case without having access to the algorithm will have need of the BER. The BER is a quantity that measures the predictability of S from X . If the protected class cannot be predicted from the unprotected classes, then the algorithm is considered fair.

Given a classifier $g \in \mathcal{G}$, its BER with respect to the joint distribution of (X, S) is defined as:

$$\text{BER}(g, X, S) = \frac{P(g(X) = 0 \mid S = 1) + P(g(X) = 1 \mid S = 0)}{2}. \quad (2.5)$$

This quantity defines the ϵ -predictability of the protected attribute. The attribute S is said to be ϵ -predictable from the set of covariates X if there exists a classifier g such that:

$$\text{BER}(g, X, S) \leq \epsilon. \quad (2.6)$$

This can be summed as any decision with disparate impact can be converted into one when the protected attribute leaks, i.e. where the protected variable can be deduced from a low BER. On the other hand, S is said to be ϵ -fair if for any classifier $g \in \mathcal{G}$:

$$\text{BER}(g, X, S) > \epsilon. \quad (2.7)$$

Therefore, let g^* a classifier which minimizes the BER such that $\text{BER}(g^*, X, S) = \epsilon^*$, then we have that, for all $\epsilon \leq \epsilon^*$, S is not ϵ -predictable from X .

The following two theorems underline the relationship between disparate impact and BER as well as between total variation distance and the BER (Theorems 2.1 and 2.2 in *Obtaining Fairness using Optimal Transport Theory* [8])

Theorem 2.1. *Let $X \in \mathbb{R}^d$ a vector of random variables and $S \in \{0, 1\}$ the protected variable. A classifier $g \in \mathcal{G}$ has disparate impact at level $\tau \in]0, 1]$ with respect to X and S if and only if:*

$$\text{BER}(g, X, S) \leq \frac{1}{2} - \frac{P(g(X) = 1 \mid S = 0)}{2} \left(\frac{1}{\tau} - 1 \right). \quad (2.8)$$

Proof. Let $g \in \mathcal{G}$, we have

$$\begin{aligned} \text{BER}(g, X, S) &\leq \frac{1}{2} - \frac{P(g(X) = 1 \mid S = 0)}{2} \left(\frac{1}{\tau} - 1 \right) = \frac{1}{2} - \frac{(\frac{1}{\tau} - 1)}{2} P(g(X) = 1 \mid S = 0) \\ &\iff P(g(X) = 0 \mid S = 1) + P(g(X) = 1 \mid S = 0) \leq 1 - \left(\frac{1}{\tau} - 1 \right) P(g(X) = 1 \mid S = 0) \\ &\iff \left(1 + \left(\frac{1}{\tau} - 1 \right) \right) P(g(X) = 1 \mid S = 0) + P(g(X) = 0 \mid S = 1) \leq 1 \\ &\iff \frac{1}{\tau} P(g(X) = 1 \mid S = 0) \leq 1 - P(g(X) = 0 \mid S = 1) = P(g(X) = 1 \mid S = 1) \\ &\iff \tau = \frac{P(g(X) = 1 \mid S = 0)}{P(g(X) = 1 \mid S = 1)}, \end{aligned}$$

where we have that a classifier g has disparate impact at level τ if $\text{DI}(g, X, S) \leq \tau$. $\square$

Definition 2.2. *The Total Variation distance between two probability measures P and Q on a σ -algebra $\mathcal{F}$ of subsets of the sample space Ω is:*

$$d_{TV}(P, Q) = \sup_{A \in \mathcal{F}} |P(A) - Q(A)|. \quad (2.9)$$

Theorem 2.3. *Given $X \in \mathbb{R}^d$ $d \geq 1$ and $S \in \{0, 1\}$,*

$$\min_{g \in \mathcal{G}} \text{BER}(g, X, S) = \frac{1}{2} (1 - d_{TV}(\mathcal{L}(X \mid S = 0), \mathcal{L}(X \mid S = 1))). \quad (2.10)$$

Proof. Let $s = 0, 1$ and let f_s the conditional distribution with respect to the protected variable S . whose probability measures are considered absolute continuous with respect to a measure μ . In general, the misclassification error is given by

$$\begin{aligned}
P(g(X) \neq S) &= P(S=0)P(g(X)=1 \mid S=0) + P(S=1)P(g(X)=0 \mid S=1) \\
&= P(S=0) \int_{g(X)=1} f_0(x) d\mu(x) + P(S=1) \int_{g(X)=0} f_1(x) d\mu(x).
\end{aligned} \tag{2.11}$$

Let us fix the value of $\pi_s = P(S=s)$. From Bayes's formula, we have

$$P(S=s \mid X) = \frac{\pi_s f_s(X)}{\pi_0 f_0(X) + \pi_1 f_1(X)}.$$

Hence

$$\{P(S=0 \mid X) > P(S=1 \mid X)\} = \{\pi_0 f_0(X) > \pi_1 f_1(X)\},$$

Thus the classifier that minimizes the misclassification error rate is

$$g^*(x) = \begin{cases} 1 & \text{if } \pi_0 f_0(x) \leq \pi_1 f_1(x) \\ 0 & \text{if } \pi_0 f_0(x) > \pi_1 f_1(x) \end{cases},$$

and from equation [2.11](#), we have

$$\min_{g \in \mathcal{G}} P(g(X) \neq S) = \int_{\pi_0 f_0(x) \leq \pi_1 f_1(x)} \pi_0 f_0(x) d\mu(x) + \int_{\pi_0 f_0(x) > \pi_1 f_1(x)} \pi_1 f_1(x) d\mu(x).$$

In our particular case, $\text{BER}(g, X, S) = P(g(X) \neq S)$ when considering $\pi_0 - \pi_1 = 1/2$. Thus we have that

$$g^*(x) = \begin{cases} 1 & \text{if } f_0(x) \leq f_1(x) \\ 0 & \text{if } f_0(x) > f_1(x) \end{cases},$$

and

$$\begin{aligned}
\min_{g \in \mathcal{G}} \text{BER}(g, X, S) &= \text{BER}(g^*, X, S) \\
&= \frac{1}{2} \left[\int_{f_0(x) \leq f_1(x)} f_0(x) d\mu(x) + \int_{f_0(x) > f_1(x)} f_1(x) d\mu(x) \right] \\
&= \frac{1}{2} \int (f_0 \wedge f_1)(x) d\mu(x).
\end{aligned}$$

By definition we have

$$d_{TV}(\mu_0, \mu_1) = \frac{1}{2} \int |f_0 - f_1| d\mu = 1 - \int (f_0 \wedge f_1)(x) d\mu(x).$$

□

This result shows that the disparate impact expression of fairness depends greatly on the conditional distributions of X given the value of the protected attribute. From Theorem 2.0.3, we can conclude that the protected attribute S is not ϵ -predictable from X if, and only if,

$$d_{TV}(\mathcal{L}(X | S = 0), \mathcal{L}(X | S = 1)) \leq 1 - 2\epsilon. \quad (2.12)$$

The smaller the Total Variation distance, the greater the value of ϵ that can satisfy the equation above and, thus, the less predictable S is from X . In the case where $d_{TV}(\mathcal{L}(X | S = 0), \mathcal{L}(X | S = 1)) = 0$, we have that both conditional distributions are equal and therefore the protected attribute is not ϵ -predictable from X . Let us note that for any binary classifier, the best bound achievable is $\epsilon = \frac{1}{2}$ non predictability. In effect, in the case of a binary classifier, if the sensitive variable can be correctly inferred from the value of the outcome at most half the time, then we have achieved total non-predictability of the attribute.

Chapter 3

Futility of Removing Bias with the Use of Naive Procedures

In this chapter, we will talk about two methods to deal with bias in machine learning algorithms. These two methods are the removal of the sensitive variable and positive discrimination. We will talk about them in order to justify why more advanced methods such as optimal transport theory are needed to treat machine learning bias. The first method, removal of sensitive variable, is based on analysing a dataset once any and all variables that could lead to discrimination are removed whereas the second method, positive discrimination, is based on treating each class of the protected variable differently from the others. The rationale between these two methods are as follows: removing the sensitive variable will in theory stop the classifier from using this variable as a basis in its decision making whereas positive discrimination is used to provide a more easily attainable decision threshold for the minority class.

This analysis will be made on the *Adult Income Dataset* as described in section [2.2](#). Using Rstudio, we will train two decision making algorithm, a random forest using the `randomForest` library and a logit regression classifier. We will also implement a 10-fold cross-validation of the data. A 10-fold cross-validation is a technique used to evaluate predictive models by randomly dividing the initial dataset in 10 subsets, each containing the same amount of observations. Nine subsets are used as training data while the 10th subset is used as a test set for the predictive model. The cross-validation process is repeated 10 times with each subset being used as a test set once. This allows us to compute more robust results when computing disparate impact and accuracy of classification.

3.1 Removing the Sensitive Variable

General Data Protection Regulation (GDPR) state that sensitive variables such as sex, ethnic group or other sensitive attributes should not be used in machine learning algorithms. This chapter thus deals with the removal of any sensitive variable from the dataset to study whether this is an effective strategy in order to minimize bias in automated decision making. The rationale behind this manoeuvre is that if this variable is absent from the dataset, then it cannot be used in the

decision making and thus it should remove the inherent bias. We will put this method to the test by using the *Adult Income Dataset*.

However, it can be seen in figure 3.1 below that removing the sensitive variable from the dataset does not in fact provide any significant improvement in the value of the disparate impact. In fact, in some case such as when using a random forest classifier, removing the variable **sex** actually worsens the disparate impact while the accuracy remains rather unchanged. Therefore, removing the sensitive variable has little impact on the bias of a machine learning algorithm. This may be explained by the fact that machine learning algorithms use every single piece of information to draw links between variables in a dataset. If we have a protected variable S which is closely correlated to other variables in the data, then simply removing it will not stop the algorithm from predicting the value of the sensitive variable using the set of covariates X . In effect, one possible explanation of this effect may be that many women after having a child end up having to stay at home to take care of the newborn. Since this takes time and effort, most of these women cannot possibly hold a full-time job and thus will end up with part-time offers. Part-time contracts mean less hours worked per weeks translating in a lower income. Therefore the algorithm could very well have found a strong link between the variable **hpw** and **income**. Once the sensitive variable was removed, this link may have had an even bigger impact in the classification of each individuals, hence why we may observe a small decrease in disparate impact when using a random forest classifier.

We may ask ourselves, if a random forest classifier increases the bias when we remove the sensitive variable, why is it that a logit regression classifier removes some bias when the sensitive variable is removed ? The same argument presented in the paragraph above can be repeated here. In effect, the logit regression may have focused on some other link that, once the sensitive variable is removed, slightly puts women back on equal footing with men. This should of course be taken with a grain of salt as the removal of the sensitive attributes only slightly increases disparate impact and does so in a non significant manner. Going from a mean disparate impact of 0.19 to 0.21 is not a significant improvement, we can hardly say that women were concretely given an advantage when removing the sensitive variable using a logit classifier.

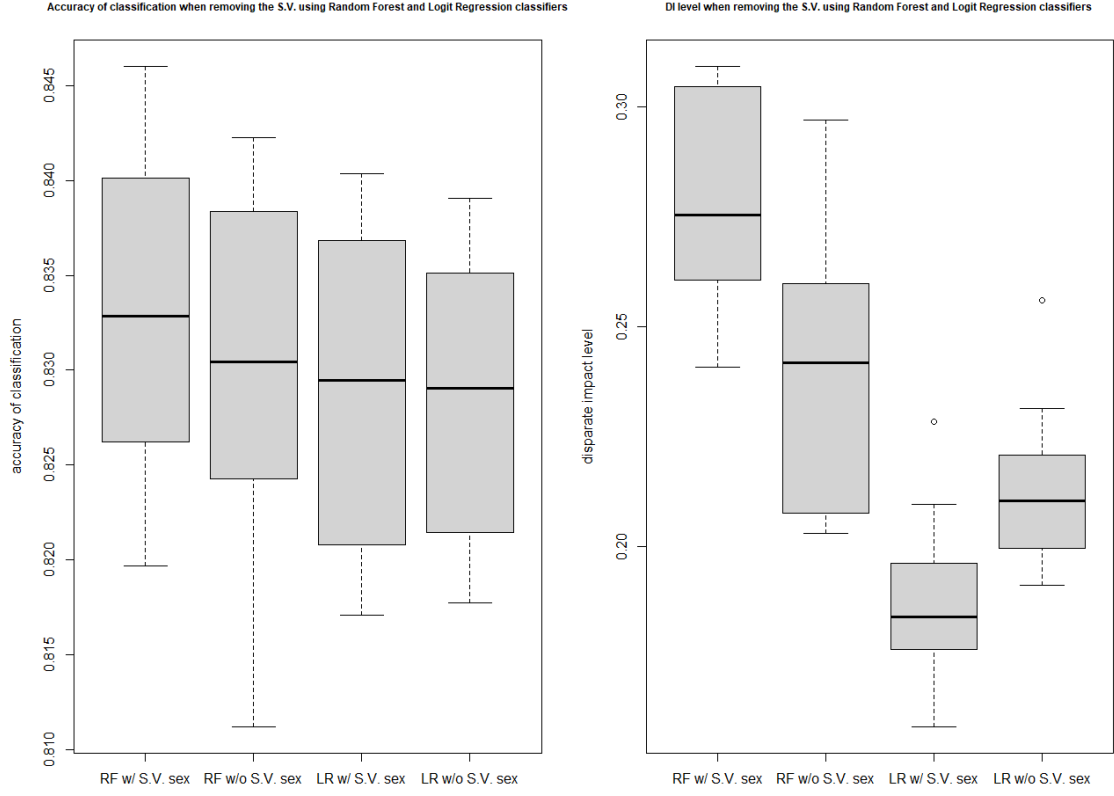


Figure 3.1: Performance of the random forest and logit regression classifiers when removing the variable `sex`. We can observe the change in classification accuracy for both classifiers when the sensitive attribute is removed from the data (*left*) as well as the change in disparate impact for our random forest and logit regression when the sensitive variable is removed (*right*). The boxplots represent the spread of the values computed for the accuracy of classification and disparate impact after all iterations of the cross-validation have been computed.

3.2 Positive Discrimination

In this case, contrary to separate treatment, we use a single classifier that is trained on the data which will in turn return a score value to us. This value will not be binary, therefore it will not immediately inform us whether the classifier predicts a success or a failure. In this case the prediction is based on whether the score value obtained is greater or less than a threshold value that has been set beforehand. If the score is greater than the threshold, the outcome will be a success and if the score is smaller than the threshold, we will have a failure. In the case of positive discrimination the threshold value is made *S*-dependent, meaning that the cutoff value will differ if the observation is part of the majority or minority class. Usually, a threshold value of 0.5 is used

when the observation is part of the group $S = 1$. The cutoff value for a success for observations in the minority group $S = 0$ is set so that the value of the disparate impact approaches 0.8.

A second method that can help with mitigating disparate impact is that of positive discrimination.

Positive discrimination is the act of lowering the requirement for success for the class that is supposedly being discriminated against. When a classifier decides if we have a success or failure for a specific observation, it compares the score this observation is given by the decision making algorithm to a set threshold value. If the score is greater than the threshold, the outcome will be a success and if the score is smaller than the threshold, we will have a failure. In the case of positive discrimination the threshold value is made S -dependent, meaning that the cutoff value will differ if the observation is part of the majority or minority class. In our case, the sensitive variable being the sex of the individual, the threshold for success will be kept at its original value for men but lowered for women. When an observation is passed through a classifier, a probability of success is outputted. In general, if this probability is greater than 0.5, then we classify the output as being a success but if it is lower then we have a failure. In our analysis the threshold for men will be kept at 0.5 while that of women will be arbitrarily lowered so as to achieve a disparate impact at level 0.8 minimum.

In figure [3.2](#) below we can see the disparate impact level before and after applying positive discrimination for both random forests and logit regressions. In effect, we had a mean disparate impact at level 0.27 and 0.19 for our random forest and logit classifiers respectively. After applying a positive discrimination modifier to the classifiers, these values increased abruptly to 0.9 and 0.84 respectively.

Looking back at the legal limit for discrimination, a disparate impact at level 0.8 would indicate that there is legally no discernible discriminatory behaviour in the classification of men and women. However, if we look at our boxplots showing classification accuracy, we can observe a sharp decrease once positive discrimination is introduced. In effect, positive discrimination lowers the barrier to entry for an entire group of individuals whilst keeping it at its original level for other groups. This barrier sometime has to be drastically lowered to reach the legal limit of 0.8 for disparate impact and doing so sharply increases the number of false positive in the minority class. False positives can be extremely harmful for a firm depending on the type of work needed to be done by the individual. On a legal point of view, this procedure, while it increases the level of disparate impact is fundamentally discriminatory. This unfairness is easily quantifiable through the value of the decision threshold. Such behaviour can have serious legal ramifications which go far outside the scope of this paper.

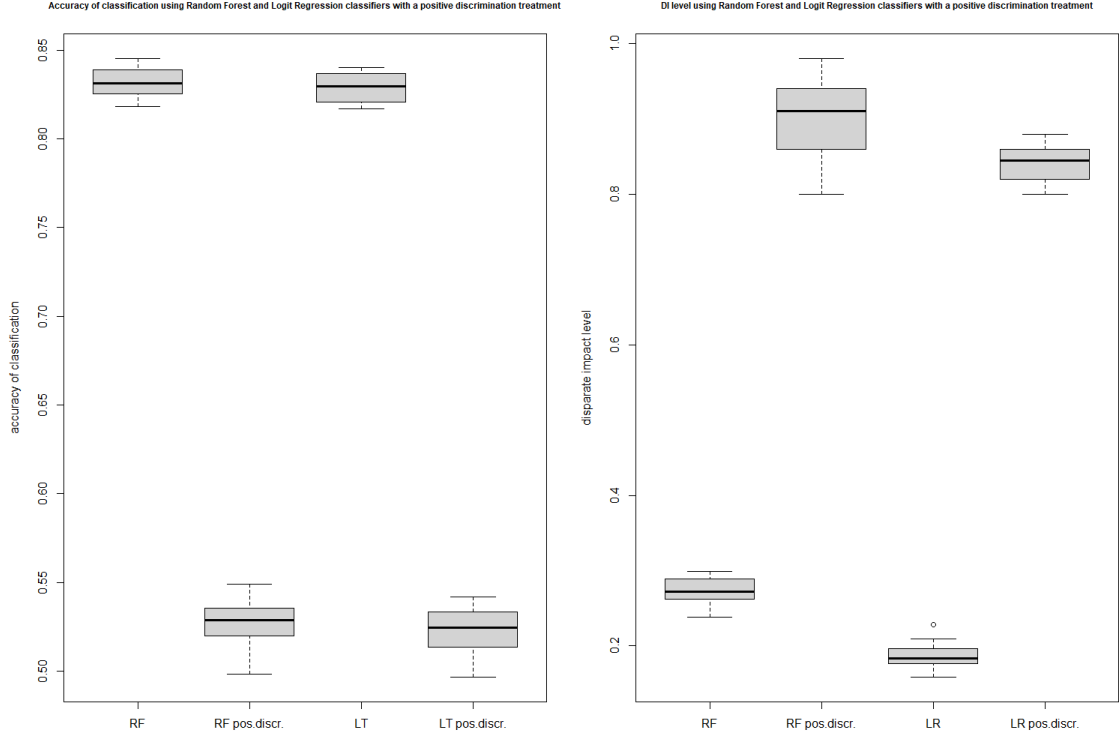


Figure 3.2: Performance of the random forest and logit regression classifiers. We can observe the change in classification accuracy for both classifiers (*left*) as well as the change in disparate impact for our random forest and logit regression when positive discrimination is introduced (*right*). The boxplots represent the spread of the values computed for the accuracy of classification and disparate impact after all iterations of the cross-validation have been computed.

3.3 Discussion

While other methods which try to reduce discriminatory behaviour in algorithms exist, the two procedures presented above are flagship procedures. General Data Protection Regulation require that sensitive variables should not be used in decision making algorithms and while it is an appropriate measure to remove such variables, we have seen that this should not be the only strategy that should be undertaken to reduce bias in algorithms as it can be rather inefficient. On the other hand, positive discrimination, while it does increase disparate impact to acceptable levels, is still a form of discrimination and thus is a procedure that should be avoided at all cost. The failure of these “naive” methods in the quest of getting rid of bias in machine learning illustrates the need for more complete and thorough procedures. The rest of this work will concentrate on such a method, that of optimal transport.

Chapter 4

Mathematical Framework for Optimal Transport Theory

We have seen in the previous chapter that GDPR recommendations such as removing sensitive variables do very little to provide any significant improvement in disparate impact, in effect the level of disparate impact remains rather unchanged as the sensitive variable can most likely be reconstructed from the other variables in the dataset. Strategies such as positive discrimination, while effective on significantly changing the level of disparate impact, come with big problems in classification accuracy and brings along its fair share in discriminatory behaviour.

One main strategy to get rid of possible discrimination associated with a classifier is to modify the initial data such as to fracture the link between the data and the protected attribute S . This method is referred to as repairing the data. The aim is to map the conditional distributions to a new common distribution so as to achieve statistical parity, meaning the probability of success is independent of the value of the protected attribute. Two methods can be considered: total repair and partial repair. These repair methods are based on moving one or both distributions using the Wasserstein metric. This metric is the minimized cost required to change one distribution into another. For example, let us say that we have a pile of sand that represents a distribution. The metric represents the cost of lifting the pile and moving it a certain distance away. The cost would be the amount of sand that is left on the ground or falls out of our hand during the transfer and is akin to the information lost when moving a distribution from one point to another. This metric thus enables the user to compare distinct distributions and quantify the distance between them.

4.1 Probability vectors and measures

In this chapter, the terms probability vectors will refer to any element $\mathbf{a} \in \Sigma_n$ that belongs to the probability simplex

$$\Sigma_n = \left\{ \mathbf{a} \in \mathbb{R}_+^n : \sum_{i=1}^n a_i = 1 \right\} \quad (4.1)$$

Definition 4.1. A *discrete measure* with weights $\mathbf{a}$ and locations $x_1, \dots, x_n \in \mathcal{X}$ reads

$$\alpha = \sum_{i=1}^n a_i \delta_{x_i} \quad (4.2)$$

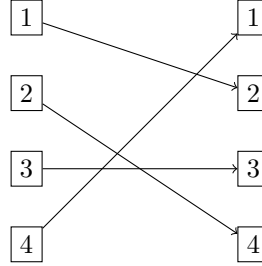
where δ_{x_i} is the Dirac distribution at position x_i and where $\mathcal{X} = \mathbb{R}^n$.

4.2 Assignment and Monge Problem

The assignment problem is a fundamental optimization problem which reads as follows: a certain number of tasks need to be done and the number of agents can be used to perform those tasks. Any agent can perform any tasks incurring a cost which will vary depending on the pair agent-task. Any agent may complete at most one task and each task can be attributed to at most one agent. The aim is to minimize the total cost all the while completing as many tasks as possible. Let $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y} = \mathbb{R}^m$. We suppose that $n = m$. Let $C: \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ be a cost function such that $\mathbf{C}_{i,j}, i = 1, \dots, n, j = 1, \dots, m$ is the cost of assigning x_i to y_j . We introduce the notation $\text{Perm}(n) = \{\sigma: \sigma \text{ is a bijection of } \{1, \dots, n\} \text{ to itself}\}$ to denote the set of permutations of n elements. For example, let $n = m = 4$. A possible permutation in Cauchy's two line notation is

$$\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 3 & 1 \end{pmatrix},$$

where the first line lists the elements of the set and the second row lists the image of each element. A more direct visualization of this permutation is as such:



The permutation in the figure above shows that an element in position 1 is sent to position 2, while the element in position 2 is sent to position 4, that in position 3 stays in position 3 and the last element in position 4 is sent to position 1.

The optimal assignment problem seeks to find the bijection $\sigma \in \text{Perm}(n)$ which minimises the cost

$$\frac{1}{n} \sum_{i=1}^n \mathbf{C}_{i, \sigma(i)} \quad (4.3)$$

where $\sigma(i)$ is to be understood as the image of i .

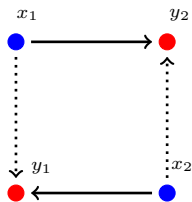
To further illustrate the use of the cost matrix, let us say that a manager needs to send supplies from a warehouse x_i to a factory y_j at a cost of $\mathbf{C}_{i,j}$ for each shipment from i to j . Assuming,

three warehouses and three factories, let us assume that the manager has the following cost matrix, where each entry is the cost incurred when moving supplies from a point x_i to a factory y_j :

5	3	1
3	2	2
4	2	4

A quick glance would indicated that the most cost efficient way of transporting supplies would be sending supplies from x_1 to y_3 , x_2 to y_1 and x_3 to y_2 .

Let us note that the optimal assignment does not necessarily have a unique solution. In the case where $n = m = 2$ and the cost matrix entries are the distance taken pairwise between the corners of the 2D square of length 1, there are two possible assignments which are both optimal.



The blue dots from measure α and the red dots from measure β are pairwise equidistant. Therefore, matching x_i to y_j along the solid or dotted line is equally optimal.

Let $\alpha = \sum_{i=1}^n \mathbf{a}_i \delta_{x_i}$ and $\beta = \sum_{j=1}^m \mathbf{b}_j \delta_{y_j}$ be two discrete measures on $\{x_i\}_{i=1}^n \subset \mathcal{X}$ and $\{y_j\}_{j=1}^m \subset \mathcal{Y}$, where $(\mathbf{a}, \mathbf{b})$ are probability vectors of the simplices $\Sigma_n \times \Sigma_m$. The Monge problem looks to displace the mass of α towards that of β by associating each x_i to a single y_j . We introduce the map

$$T: \mathcal{X} \rightarrow \mathcal{Y}$$

$$\{x_1, \dots, x_n\} \mapsto \{y_1, \dots, y_m\},$$

which sends each point in $\mathcal{X}$ to a single point in $\mathcal{Y}$ and its push-forward operator

$$T_{\#}: \mathcal{M}(\mathcal{X}) \rightarrow \mathcal{M}(\mathcal{Y})$$

$$\alpha \mapsto T_{\#}\alpha = \sum_i \mathbf{a}_i \delta_{T(x_i)}.$$

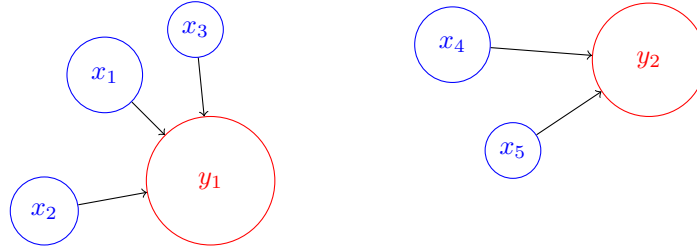
The operator T pushes forward each elementary mass of the measure α on $\mathcal{X}$ by applying the map T to obtain an elementary mass in $\mathcal{Y}$. The linear operator T is a surjective function as every point mass being displaced is discrete in nature. The map also verifies that, $\forall j = 1, \dots, m$:

$$\mathbf{b}_j = \sum_{i: T(x_i)=y_j}^n \mathbf{a}_i. \quad (4.4)$$

The aim of such a map is to minimize a certain number value which symbolizes the "energy" needed to transport a point mass from x_i to y_j . The aim is to be as efficient as possible therefore minimizing the spent "energy" on the transportation is paramount. This spent "energy" is parametrized by a function c defined for all points $(x, y) \in \mathcal{X} \times \mathcal{Y}$, such that the map T minimizes it:

$$\min_T \left\{ \sum_i c(x_i, T(x_i)) : T_{\#}\alpha = \beta \right\} \quad (4.5)$$

In the special case where $n = m$ and all weights are uniform, meaning that $a_i = b_j = \frac{1}{n}$, then the conservation of mass suggests that the operator T is not only surjective but is actually a bijection such that $T(x_i) = y_{\sigma(i)}$, where $\sigma(i)$ is the image of i . In this case the optimization problem is that of equation 4.3. On the other hand, when $n \neq m$, transportation maps may not exist at all. This is always the case when $m > n$, i.e. the target measure has more points than the source. For example, we can see below that there exists an application mapping the measure α to β but there is no map for the reverse transformation.



4.3 Kantorovich Relaxation

The assignment and Monge problems as presented earlier are not always relevant in practical study of discrete measures. In effect, the way the assignment problem is defined, it can only be used to compare uniform probability vectors of the same size.

Thus, Kantorovich proposes that mass transportation be treated as probabilistic in nature as opposed to the deterministic way it was considered to be. This, in effect, means that instead of having a point mass at an initial location i which can only be transported to another location j , mass can now be split and transported to multiple targets. This flexibility is encoded by a coupling matrix $\mathbf{P} \in \mathbb{R}_+^{n \times m}$. The entries $\mathbf{P}_{i,j}$ describe the amount of mass being transported from initial location i to final location j , or the amount of mass flowing from x_i to y_j .

$$\mathbf{U}(\mathbf{a}, \mathbf{b}) = \left\{ \mathbf{P} \in \mathbb{R}_+^{n \times m} : \mathbf{P}\mathbf{1}_m = \mathbf{a} \quad \text{and} \quad \mathbf{P}^T\mathbf{1}_n = \mathbf{b} \right\},$$

where we have the following with $\mathbf{1}_n$ and $\mathbf{1}_m$ two vectors of size n and m respectively for which every entry is 1:

$$\mathbf{P}\mathbf{1}_m = \left(\sum_j \mathbf{P}_{i,j} \right)_i \in \mathbb{R}^n \quad \text{and} \quad \mathbf{P}^T \mathbf{1}_n = \left(\sum_i \mathbf{P}_{i,j} \right)_j \in \mathbb{R}^m.$$

Kantorovich's optimal transport therefore reads:

$$L_C(\mathbf{a}, \mathbf{b}) = \min_{\mathbf{P} \in \mathbf{U}(\mathbf{a}, \mathbf{b})} \langle \mathbf{C}, \mathbf{P} \rangle = \sum_{i,j} \mathbf{C}_{i,j} \mathbf{P}_{i,j}, \quad (4.6)$$

where $\mathbf{C}$ is the cost matrix in which each entry details the cost of moving a unit mass from a point x_i to a point y_j .

Let us illustrate the Kantorovich problem using a fictitious example. Let us suppose that a manager runs n warehouses containing raw materials used in the making of goods and services in his m factories. Each warehouse is indexed by an integer i while the factories are indexed by j . Each warehouse contains an amount $\mathbf{a}_i$ of raw material while each factory needs an amount $\mathbf{b}_j$ in order to operate. In order to transfer a units of raw material from a warehouse i to a factory j , the manager uses a transportation company that will charge him $a \times \mathbf{C}_{i,j}$, where $\mathbf{C}_{i,j}$ is the cost of shipping a single unit from i to j . The manager therefore needs to solve the linear program described by equation (4.6) in order to obtain a logistical plan $\mathbf{P}^*$ which describes what amount of raw material needs to be transported from each warehouse i to each factory j . The manager will be paying the transportation company a total of $\langle \mathbf{C}, \mathbf{P}^* \rangle$, where $\mathbf{P}^*$ is such that it minimizes the cost of transportation for the manager.

For a permutation $\sigma \in \text{Perm}(n)$, we write $\mathbf{P}_\sigma$ for the corresponding permutation matrix, $\forall i = 1, \dots, n, j = 1, \dots, n$,

$$(\mathbf{P}_\sigma)_{i,j} = \begin{cases} \frac{1}{n} & \text{if } j = \sigma_i \\ 0 & \text{otherwise} \end{cases} \quad (4.7)$$

We can check that

$$\langle \mathbf{C}, \mathbf{P}_\sigma \rangle = \frac{1}{n} \sum_{i=1}^n \mathbf{C}_{i, \sigma(i)}, \quad (4.8)$$

which equates to the fact that the assignment problem discussed earlier can be remodeled as a Kantorovich problem where the couplings $\mathbf{P}$ are restricted to be exactly permutation matrices, meaning that we have deterministic couplings.

$$\min_{\sigma \in \text{Perm}(n)} \frac{1}{n} \sum_{i=1}^n \mathbf{C}_{i, \sigma(i)} = \min_{\sigma \in \text{Perm}(n)} \langle \mathbf{C}, \mathbf{P}_\sigma \rangle \quad (4.9)$$

Let us illustrate what the coupling matrix entails by going back to the example 4.2. We assume that we are still in the deterministic case for simplicity. The matrix $\mathbf{P}$ describes the amount of produce moving from a warehouse x_i to a factory y_j . Since it was established that the most cost efficient way of transporting supplies was to transport supplies from x_1 to y_3 , x_2 to y_1 and x_3 to y_2 , the coupling matrix associated with the given cost matrix is

0	0	1/3
1/3	0	0
0	1/3	0

For discrete measures of the form (4.2), we store in the matrix $\mathbf{C}$ all pairwise costs between points in the supports of α, β , namely $\mathbf{C}_{i,j} = c(x_i, y_j)$ to define

$$\mathcal{L}_c(\alpha, \beta) = L_{\mathbf{C}}(\mathbf{a}, \mathbf{b}). \quad (4.10)$$

Thus, the Kantorovich formulation of the optimal transport problem between measures is the same as that between their associated probability weight vectors $\mathbf{a}, \mathbf{b}$ except that the cost matrix $\mathbf{C}$ makes use of the support of α and β .

Equation 4.10 is written to work in the case where discrete measures are used and can be generalized to deal with arbitrary measures instead. Let us consider couplings $\pi \in \mathcal{M}_+^1(\mathcal{X}, \mathcal{Y})$ which are joint distributions over the product space. Admissible couplings are reformulated as a marginal constraint on joint probability distributions:

$$\mathcal{U}(\alpha, \beta) = \{\pi \in \mathcal{M}_+^1(\mathcal{X}, \mathcal{Y}) : P_{\mathcal{X}\#}\pi = \alpha \text{ and } P_{\mathcal{Y}\#}\pi = \beta\} \quad (4.11)$$

such that $P_{\mathcal{X}\#}$ and $P_{\mathcal{Y}\#}$ are the push-forward operators of the projections $P_{\mathcal{X}}(x, y) = x$ and $P_{\mathcal{Y}}(x, y) = y$. The Kantorovich problem thus reads

$$\mathcal{L}_c(\alpha, \beta) = \min_{\pi \in \mathcal{U}(\alpha, \beta)} \int_{\mathcal{X} \times \mathcal{Y}} c(x, y) d\pi(x, y). \quad (4.12)$$

4.4 Dual problem

Because the Kantorovich problem is a constrained convex minimization problem, there naturally exists a constrained concave maximization problem, called dual problem.

Let us preface the explanation by illustrating the dual problem using the analogy made in the previous section with the Kantorovich problem. We had a manager looking to transfer a certain amount of raw materials from his warehouses to his factories while minimizing the cost of transportation given a set cost for each shipping route. We suppose that the manager does not have the computational means to obtain the logistical plan $\mathbf{P}^*$. The manager therefore decides to outsource that task to a vendor. The vendor decides to split the task in two: collecting and delivering. The vendor sets a price $\mathbf{f}_i$ to collect a single unit from a warehouse i regardless of where the unit is sent to, and a delivery price $\mathbf{g}_j$ which depends on which factory j the unit is delivered to. Since there are $\mathbf{a}_i$ units in warehouse i and each factory j requires $\mathbf{b}_j$ to function properly, the pricing scheme chosen by the vendor will result in a total cost of $\langle \mathbf{f}, \mathbf{a} \rangle + \langle \mathbf{g}, \mathbf{b} \rangle$.

We assume that there is no other vendor available for the manager to compare prices to, therefore the manager will need to determine whether the price scheme proposed by the vendor is reasonable or not. The manager could compute the price $L_{\mathbf{C}}(\mathbf{a}, \mathbf{b})$ by solving the linear program and verify whether the vendor's asking price is at least not larger than that amount. However,

let us recall that the manager does not have the capabilities of computing the most efficient plan possible, thus another method will be needed. Recall that $\mathbf{f}_i$ is the price to pick up a single unit at warehouse i and $\mathbf{g}_j$ is the price to deliver a single unit at factory j . Thus the cost incurred by the manager to have a single unit shipped from i to j is $\mathbf{f}_i + \mathbf{g}_j$. The manager also knows that the price of transferring a unit of raw material from one warehouse to a factory is $\mathbf{C}_{i,j}$. Therefore, if for any shipping route linking i to j , the aggregate price $\mathbf{f}_i + \mathbf{g}_j$ is greater than $\mathbf{C}_{i,j}$, the manager can conclude that the offer from the vendor is not reasonable and should naturally refuse. Should the manager verify that, for all pairs (i, j) , the inequality $\mathbf{f}_i + \mathbf{g}_j \leq \mathbf{C}_{i,j}$ does indeed hold true, then any attempt to transport the goods himself would prove more costly than having the vendor do the job. The vendor must thus find a set of prices $\mathbf{f}, \mathbf{g}$ which maximizes $\langle \mathbf{f}, \mathbf{a} \rangle + \langle \mathbf{g}, \mathbf{b} \rangle$ all the while verifying $\mathbf{f}_i + \mathbf{g}_j \leq \mathbf{C}_{i,j}$ for all (i, j) . It is shown that the best possible outcome for the vendor is in fact the price which would be obtained by the manager should he have computed $L_{\mathbf{C}}(\mathbf{a}, \mathbf{b})$.

We can extend this primal-dual construction to arbitrary measures. Their relationship is explained in the following proposition.

Definition 4.2. Let $(\mathbf{f}, \mathbf{g}) \in \mathbb{R}^n \times \mathbb{R}^m$ be two vectors. Let $\mathbf{1}_n$ and $\mathbf{1}_m$ be two vectors of size n and m respectively for which every entry is 1. The direct sum of $\mathbf{f}$ and $\mathbf{g}$ is defined as:

$$\mathbf{f} \oplus \mathbf{g} = \mathbf{f} \mathbf{1}_m^T + \mathbf{1}_n \mathbf{g}^T$$

Proposition 4.3. The Kantorovich problem as described in equation [4.6](#) admits the dual

$$L_{\mathbf{C}}(\mathbf{a}, \mathbf{b}) = \max_{(\mathbf{f}, \mathbf{g}) \in \mathcal{R}(\mathbf{C})} \langle \mathbf{f}, \mathbf{a} \rangle + \langle \mathbf{g}, \mathbf{b} \rangle \quad (4.13)$$

such that the set of admissible Kantorovich potentials is

$$\mathcal{R}(\mathbf{C}) = \{(\mathbf{f}, \mathbf{g}) \in \mathbb{R}^n \times \mathbb{R}^m : \forall i = 1, \dots, n, j = 1, \dots, m, \mathbf{f} \oplus \mathbf{g} \leq \mathbf{C}\}.$$

such that $\oplus$ is the direct sum operator. The notation $\mathbf{f} \oplus \mathbf{g} \leq \mathbf{C}$ means that $\forall i = 1, \dots, n, j = 1, \dots, m$, we have $f_i + g_j \leq C_{i,j}$ where $\mathbf{f}, \mathbf{g}$ are price vectors and $\mathbf{C}$ is a cost matrix.

Proof. The proof can be consulted in [\[9\]](#), p.23, Proposition 2.4. □

The following proposition extends the dual problem to arbitrary measures.

Proposition 4.4. Let $\alpha \in \mathcal{M}(\mathcal{X})$ and $\beta \in \mathcal{M}(\mathcal{Y})$ be two arbitrary measures on $\mathcal{X}$ and $\mathcal{Y}$. We have

$$\mathcal{L}_c(\alpha, \beta) = \sup_{(f, g) \in \mathcal{R}(c)} \int_{\mathcal{X}} f(x) d\alpha(x) + \int_{\mathcal{Y}} g(y) d\beta(y), \quad (4.14)$$

where the set of admissible dual potentials is

$$\mathcal{R}(c) = \{(f, g) \in \mathcal{C}(\mathcal{X}) \times \mathcal{C}(\mathcal{Y}) : \forall (x, y), f(x) + g(y) \leq c(x, y)\},$$

where $\mathcal{C}(\mathcal{X})$ and $\mathcal{C}(\mathcal{Y})$ are the sets of continuous functions on $\mathcal{X}$ and $\mathcal{Y}$.

4.5 Metric Properties of Optimal Transport

Optimal transport defines a distance between probability vectors and probability measures as soon as the cost matrix satisfies certain suitable properties. Optimal transport can be understood as follows: it is a way to lift a ground distance between points to a distance between probability vectors or measures. We first consider the case where the “ground metric” matrix $\mathbf{C}$ is fixed, representing substitution costs between bins, and shared across several probability vectors we would like to compare. The following proposition states that OT provides a valid distance between probability vectors supported on these bins.

Proposition 4.5. *Suppose $n = m$ and that for some $p \geq 1$, $C = D^p = (\mathbf{D})_{i,j}^p \in \mathbb{R}^{n \times n}$, where $\mathbf{D} \in \mathbb{R}_+^{n \times n}$ is ground distance matrix meaning that each entry contains the distance, taken pairwise, between two elements of different sets such that:*

- $\mathbf{D} \in \mathbb{R}_+^{n \times n}$ is symmetric;
- $D_{i,j} = 0 \iff i = j$;
- $\forall (i, j, k), \quad D_{i,k} \leq D_{i,j} + D_{j,k}$.

Then,

$$W_p(\mathbf{a}, \mathbf{b}) = L_{D^p}(\mathbf{a}, \mathbf{b})^{1/p} \quad (4.15)$$

is the p -Wasserstein distance on Σ_n . We have that W_p is symmetric, positive, $W_p(\mathbf{a}, \mathbf{b}) = 0 \iff \mathbf{a} = \mathbf{b}$ and satisfies the triangle inequality,

$$\forall \mathbf{a}, \mathbf{b}, \mathbf{c} \in \Sigma_n, \quad W_p(\mathbf{a}, \mathbf{c}) \leq W_p(\mathbf{a}, \mathbf{b}) + W_p(\mathbf{b}, \mathbf{c})$$

Proof. Since $\mathbf{C} = \mathbf{D}^p$ has a null diagonal, $W_p(\mathbf{a}, \mathbf{a}) = 0$, with corresponding optimal transport matrix $\mathbf{P}^* = \text{diag}(\mathbf{a})$; by the positivity of all off-diagonal elements of $\mathbf{D}^p$, $W_p(\mathbf{a}, \mathbf{b}) > 0$ whenever $\mathbf{a} \neq \mathbf{b}$. By symmetry of $\mathbf{D}^p$, $W_p(\mathbf{a}, \mathbf{b})$ is also a symmetric function.

To prove the triangle inequality we will refer back to Theorem 7.3 in [10]. Let $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \Sigma_n$ and $\mathbf{P}, \mathbf{Q}$ be two optimal solutions to the transport problems between $\mathbf{a}$ and $\mathbf{b}$ and $\mathbf{b}$ and $\mathbf{c}$ respectively. Let us define a vector $\mathbf{d} = \mathbf{b}$ if $\mathbf{b}_j > 0$ and $\mathbf{d}_j = 1$ otherwise, to write

$$\mathbf{S} = \mathbf{P} \text{diag}(1/\mathbf{d}) \mathbf{Q} \in \mathbb{R}_+^{n \times n},$$

and notice that $\mathbf{S} \in \mathbf{U}(\mathbf{a}, \mathbf{b})$ because

$$\mathbf{S} \mathbf{1}_n = \mathbf{P} \text{diag}(1/\mathbf{d}) \mathbf{Q} \mathbf{1}_n = \mathbf{P}(\mathbf{b}/\mathbf{d}) = \mathbf{P} \mathbf{1}_{\text{Supp}(\mathbf{b})} = \mathbf{a},$$

where $\mathbf{1}_{\text{Supp}(\mathbf{b})}$ is the vector of size n with entries equal to ones at indices j where $\mathbf{b}_j > 0$ and zero otherwise. We also have that $\mathbf{P} \mathbf{1}_{\text{Supp}(\mathbf{b})} = \mathbf{P} \mathbf{1} = \mathbf{a}$ because $\mathbf{P}_{i,j} = 0$ for j where $\mathbf{b}_j = 0$. Similarly, we verify that $\mathbf{S}^T \mathbf{1}_n = \mathbf{c}$. The triangle inequality is thus such that:

$$\begin{aligned}
W_p(\mathbf{a}, \mathbf{b}) &= \left(\min_{\mathbf{P} \in \mathbf{U}(\mathbf{a}), \mathbf{c}} \langle \mathbf{P}, \mathbf{D}^p \rangle \right)^{1/p} \leq \langle \mathbf{S}, \mathbf{D}^p \rangle^{1/p} \\
&= \left(\sum_{ik} \mathbf{D}_{ik}^p \sum_j \frac{\mathbf{P}_{ij} \mathbf{Q}_{jk}}{\mathbf{d}_j} \right)^{1/p} \\
&= \left(\sum_{ijk} (\mathbf{D}_{ij} + \mathbf{D}_{jk})^p \frac{\mathbf{P}_{ij} \mathbf{Q}_{jk}}{\mathbf{d}_j} \right)^{1/p} \\
&= \left(\sum_{ijk} \mathbf{D}_{ij}^p \frac{\mathbf{P}_{ij} \mathbf{Q}_{jk}}{\mathbf{d}_j} \right)^{1/p} + \left(\sum_{ijk} \mathbf{D}_{jk}^p \frac{\mathbf{P}_{ij} \mathbf{Q}_{jk}}{\mathbf{d}_j} \right)^{1/p}.
\end{aligned}$$

Therefore we have

$$\begin{aligned}
W_p(\mathbf{a}, \mathbf{b}) &\leq \left(\sum_{ij} \mathbf{D}_{ij}^p \mathbf{P}_{ij} \sum_k \frac{\mathbf{Q}_{jk}}{\mathbf{d}_j} \right)^{1/p} + \left(\sum_{jk} \mathbf{D}_{jk}^p \mathbf{Q}_{jk} \sum_i \frac{\mathbf{P}_{ij}}{\mathbf{d}_j} \right)^{1/p} \\
&= \left(\sum_{ij} \mathbf{D}_{ij}^p \mathbf{P}_{ij} \right)^{1/p} + \left(\sum_{jk} \mathbf{D}_{jk}^p \mathbf{Q}_{jk} \right)^{1/p} \\
&= W_p(\mathbf{a}, \mathbf{b}) + W_p(\mathbf{b}, \mathbf{c}).
\end{aligned}$$

□

This last proposition can be rewritten to use arbitrary measures that need not be discrete.

Proposition 4.6. *Let us assume that $\mathcal{X} = \mathcal{Y}$ and that for some $p \geq 1$, $c(x, y) = d(x, y)^p$, where d is a distance on $\mathcal{X}$ such that:*

- $d(x, y) = d(y, x) \geq 0$;
- $d(x, y) = 0 \iff x = y$;
- $\forall (x, y, z) \in \mathcal{X}^3, \quad d(x, z) \leq d(x, y) + d(y, z)$.

Then,

$$\mathcal{W}_p(\alpha, \beta) = \mathcal{L}_{d^p}(\alpha, \beta)^{1/p} \quad (4.16)$$

is the p -Wasserstein distance on $\mathcal{X}$, where $\alpha \in \mathcal{M}(\mathcal{X})$ and $\beta \in \mathcal{M}(\mathcal{X})$ are arbitrary measures on the space $\mathcal{X}$. We have that $\mathcal{W}_p$ is symmetric, non-negative, $\mathcal{W}_p(\alpha, \beta) = 0 \iff \alpha = \beta$ and satisfies the triangle inequality,

$$\forall (\alpha, \beta, \gamma) \in \mathcal{M}_+^1(\mathcal{X})^3, \quad \mathcal{W}_p(\alpha, \gamma) \leq \mathcal{W}_p(\alpha, \beta) + \mathcal{W}_p(\beta, \gamma)$$

Proof. The proof follows the same approach as that of Proposition 4.5 but uses arbitrary measures instead of purely discrete ones. This proof relies on the existence of optimal couplings (α, β) and (β, γ) where $(\alpha, \beta, \gamma) \in \mathcal{M}_+^1(\mathcal{X})^3$. □

Since the Wasserstein distance is able to compare measures on different probabilities vectors, this makes it a great candidate to use it as a loss function.

4.6 The Kantorovich linear program

Let us recall Kantorovich's optimal transport formulation introduced in the last chapter:

$$L_C(\mathbf{a}, \mathbf{b}) = \min_{\mathbf{P} \in \mathbf{U}(\mathbf{a}, \mathbf{b})} \langle \mathbf{C}, \mathbf{P} \rangle = \sum_{i,j} \mathbf{C}_{i,j} \mathbf{P}_{i,j}, \quad (4.17)$$

where $(\mathbf{a}, \mathbf{b})$ are probability vectors of the simplices $\Sigma_n \times \Sigma_m$.

To make use of linear programming to solve this equation, we may rewrite the coupling matrix $\mathbf{P} \in \mathbb{R}^{n \times m}$ as a vector $\mathbf{p} \in \mathbb{R}^{nm}$ such that $\mathbf{p}$ is equal to $\mathbf{P}$ enumerated column-wise, i.e. the $i + n(j-1)^{\text{th}}$ element of $\mathbf{p}$ is $\mathbf{P}_{i,j}$. We can also rewrite the cost matrix $\mathbf{C}$ as a cost vector $\mathbf{c}$ using the same method. To be solvable, the linear program also needs equality constraints and non-negative constraints on variables. The following $(n+m) \times nm$ matrix can be used to store both the row and column-sum constraints that need to be satisfied for any coupling matrix in $\mathbf{U}(\mathbf{a}, \mathbf{b})$:

$$\mathbf{A} = \begin{bmatrix} \mathbf{1}_n^T \otimes \mathbf{1}_m \\ \mathbf{1}_n \otimes \mathbf{1}_m^T \end{bmatrix} \in \mathbb{R}^{(n+m) \times nm}$$

where $\otimes$ is Kronecker's product.

Definition 4.7. Let $\mathbf{A} \in \mathbb{R}^{n \times m}$ and $\mathbf{B} \in \mathbb{R}^{p \times q}$ two matrices. The **Kronecker product** of these two matrices is defined as:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1m}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{n1}\mathbf{B} & \cdots & a_{nm}\mathbf{B} \end{bmatrix} \in \mathbb{R}^{np \times qm}.$$

Then the linear reformulation of the original problem is thus:

$$L_C(\mathbf{a}, \mathbf{b}) = \min_{\substack{\mathbf{p} \in \mathbb{R}^{nm} \\ \mathbf{A}\mathbf{p} = \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}}} \mathbf{c}^T \mathbf{p}. \quad (4.18)$$

The corresponding dual problem is defined as

$$L_C(\mathbf{a}, \mathbf{b}) = \max_{\substack{\mathbf{h} \in \mathbb{R}^{n+m} \\ \mathbf{A}^T \mathbf{h} \leq \mathbf{c}}} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}^T \mathbf{h}, \quad (4.19)$$

where the cost vector $\mathbf{h} \in \mathbb{R}^{n+m}$ is equal to the two price vectors $\mathbf{f} \in \mathbb{R}^n$ and $\mathbf{g} \in \mathbb{R}^m$ from section 4.4 stacked together.

4.7 Complementary

Let us recall the Kantorovich transport problem and its dual. The primal problem, described in equation 4.6 is the linear program in which one needs to minimise the cost of displacing any unit from a point i to a point j in space given an associated cost matrix. The dual problem as described in Proposition 4.3 is a linear program where the costs of picking up one unit at point i and the cost of delivering it at point j are split. The aim is to maximize both costs while keeping the total at most equal to the total cost in the primal problem. While they are closely related, the original and dual problems can be solved independently of one another. These two problems will yield the optimal $\mathbf{P}^*$ and $(\mathbf{f}^*, \mathbf{g}^*)$ solutions to the primal and dual problems respectively. The link between the two solutions is detailed in the following proposition.

Proposition 4.8. *Let $\mathbf{P}^* \in \mathbb{R}^{n \times m}$ and $(\mathbf{f}^*, \mathbf{g}^*) \in \mathbb{R}^n \times \mathbb{R}^m$ the optimal solutions to the primal and dual problems respectively. For all $i = 1, \dots, n$ and $j = 1, \dots, m$, we have*

$$\mathbf{P}_{i,j}^* (\mathbf{C}_{i,j} - \mathbf{f}_i^* + \mathbf{g}_j^*) = 0.$$

Proof. The dual aspect of the Kantorovich problem as described in Proposition 4.3 indicates that $\langle \mathbf{P}^*, \mathbf{C} \rangle = \langle \mathbf{f}^*, \mathbf{a} \rangle + \langle \mathbf{g}^*, \mathbf{b} \rangle$. Let us recall that $\mathbf{P} \mathbf{1}_m = \mathbf{a}$ and $\mathbf{P}^T \mathbf{1}_n = \mathbf{b}$. Thus,

$$\begin{aligned} \langle \mathbf{f}^*, \mathbf{a} \rangle + \langle \mathbf{g}^*, \mathbf{b} \rangle &= \langle \mathbf{f}^*, \mathbf{P}^* \mathbf{1}_m \rangle + \langle \mathbf{g}^*, \mathbf{P}^{*T} \mathbf{1}_n \rangle \\ &= \langle \mathbf{f}^* \mathbf{1}_m^T, \mathbf{P}^* \rangle + \langle \mathbf{1}_n \mathbf{g}^{*T}, \mathbf{P}^* \rangle \end{aligned}$$

where $\mathbf{f}^* \oplus \mathbf{g}^* = \mathbf{f}^* \mathbf{1}_m^T + \mathbf{1}_n \mathbf{g}^{*T}$. Therefore, we have

$$\langle \mathbf{f}^*, \mathbf{a} \rangle + \langle \mathbf{g}^*, \mathbf{b} \rangle = \langle \mathbf{P}^*, \mathbf{f}^* \oplus \mathbf{g}^* \rangle.$$

Thus,

$$\begin{aligned} \langle \mathbf{P}^*, \mathbf{C} \rangle - [\langle \mathbf{f}^*, \mathbf{a} \rangle + \langle \mathbf{g}^*, \mathbf{b} \rangle] &= \langle \mathbf{P}^*, \mathbf{C} \rangle - \langle \mathbf{P}^*, \mathbf{f}^* \oplus \mathbf{g}^* \rangle \\ &= \langle \mathbf{P}^*, \mathbf{C} - \mathbf{f}^* \oplus \mathbf{g}^* \rangle \\ &= 0. \end{aligned}$$

□

Because each entry of the $\mathbf{C} - \mathbf{f}^* \oplus \mathbf{g}^*$ and $\mathbf{P}^*$ matrices are non negative, the dot product being necessarily equal to 0 forces the fact that, for any and all pair of (i, j) , if $\mathbf{P}_{i,j}^* > 0$, then $\mathbf{C}_{i,j} - (\mathbf{f}_i^* + \mathbf{g}_j^*) = 0$ and if $\mathbf{C}_{i,j} > \mathbf{f}_i^* + \mathbf{g}_j^*$, then we have that $\mathbf{P}_{i,j}^* = 0$.

Definition 4.9. *Let $\mathbf{P} \in \mathbb{R}^{n \times m}$ and $(\mathbf{f}, \mathbf{g}) \in (\mathbb{R}^n \times \mathbb{R}^m)$ the optimal solutions to the primal and dual problems respectively. The matrix $\mathbf{P}$ and vector pair $(\mathbf{f}, \mathbf{g})$ are complementary with respect to the cost matrix $\mathbf{C}$ if $\forall (i, j)$ such that $\mathbf{P}_{i,j} > 0$ we have $\mathbf{C}_{i,j} = \mathbf{f}_i + \mathbf{g}_j$.*

Proposition 4.10. *If a pair of primal and dual solutions $\mathbf{P}$ and $(\mathbf{f}, \mathbf{g})$ is complementary, we can infer that the solutions are optimal.*

Proof. By duality, we have that

$$\mathbf{L}_C(\mathbf{a}, \mathbf{b}) \leq \langle \mathbf{P}, \mathbf{C} \rangle = \langle \mathbf{P}, \mathbf{f} \oplus \mathbf{g} \rangle = \langle \mathbf{a}, \mathbf{f} \rangle + \langle \mathbf{b}, \mathbf{g} \rangle \leq \mathbf{L}_C(\mathbf{a}, \mathbf{b}).$$

Thus, $\mathbf{P}$ and $(\mathbf{f}, \mathbf{g})$ are primal and dual optimal respectively.

□

Chapter 5

Obtaining Fairness using Optimal Transport Theory

This chapter will dwell into the different repair procedures that will be explored and used in this work. We will focus on four procedures, two of them will be total repair procedures whereas the other two are partial repair procedures.

Total repair procedures are procedures where the data set is divided according to the value of the sensitive variable. We then take conditional distributions and remap them to a singular, common distribution. Both distributions will be transformed through the use of a continuous map which will depend on the value of the attribution S .

On the other hand, partial repair procedures are procedures in which the conditional distributions are remapped to a singular, common distribution, however not fully. There will be a parameter λ which will dictate the amount of repair desired. Partial repair allows the user to have more control over the classification error of the algorithm and the predictability of the repaired data. We will see that partial repair procedures on the whole provide the user with a better classification error rate.

This chapter will also detail a real life application of the total and partial repair procedures. Using the *Adult Income Database*, we will apply each repair procedure and compare the level of repair and the classification accuracy of each procedure. This will be done twice over, once using a logistic regression model to model the data and once using a random forest classifier.

5.1 Repairing the data

We have an initial dataset X which can be dissected into two groups depending on whether each observation is part of the minority group ($S = 0$) or the default group ($S = 1$). We thus have two conditional distributions $\mu_s = \mathcal{L}(X \mid S = s)$, where $s = 0, 1$ which need to be transformed into a new distribution μ in order to suppress the dependence on the value of S . This is called "repairing the data". This process modifies the initial data X into a new dataset $\tilde{X}$ for which the prediction $\hat{Y} = g(\tilde{X})$ of Y using any classifier $g \in \mathcal{G}$ yields a disparate impact value $DI(g, \tilde{X}, Y)$ as close to 1 as possible depending on the level of repair desired. There are two types of data repair: total and partial repair. Total repair, as the name implies, results in total repair of the data, meaning that we aim to have both conditional distributions equal to one another leading to having disparate

impact at level 1, whereas partial repair solely aims to make both conditional distributions closer to one another. Both methods offer pros and cons. For example, total repair ensures full fairness at the expense of some accuracy of the classification. This trade-off will lead one to choose one method rather than the other.

5.1.1 Total repair

As stated in the paragraph above, total repair is the process of modifying a dataset X into a new dataset $\tilde{X}$. It is this new dataset which will be then used to predict the outcome value $\hat{Y}$ using a classifier $g \in \mathcal{G}$ such that the new modified data set achieves statistical parity, i.e. the probability of success is independent of the value of the protected attribute S . Of course, modifying the distribution of data means that information is in effect lost or modified as well. Hence, the distribution to which the initial data is mapped to should mirror the original distribution as much as possible so as to minimize information loss and maximize the accuracy of the prediction of the new classification. In order to achieve statistical parity, we wish the new modified conditional distributions relative to each value of the protected attribute be the same. Let X and S be a pair of random variables. The conditional probability distribution of X given S is the probability of observing X given that S is known to satisfy a particular condition. In our case, we wish that the conditional probability of X given S is the same whether $S = 0$ or $S = 1$ for all classifiers $g \in \mathcal{G}$. Thus, we have:

$$\mathcal{L}(\tilde{X} \mid S = 0) = \mathcal{L}(\tilde{X} \mid S = 1) \implies \mathcal{L}(g(\tilde{X}) \mid S = 0) = \mathcal{L}(g(\tilde{X}) \mid S = 1)$$

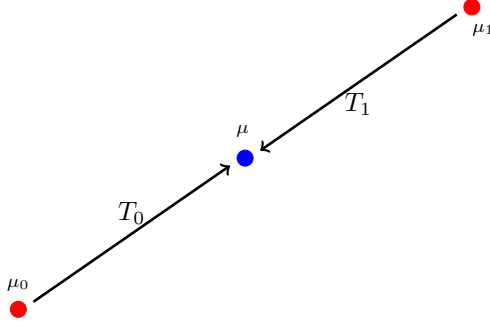
Both conditional distributions, $\mu_0 = \mathcal{L}(X \mid S = 0)$ and $\mu_1 = \mathcal{L}(X \mid S = 1)$, need to be remapped to a singular, common third distribution μ . This is done through transforming X through the use of a continuous map $T: X \rightarrow \tilde{X}$ which depends on the value of the attribute S . For example, for $s = 0, 1$, we have:

$$\begin{aligned} T_s: \mathbb{R}^d &\rightarrow \mathbb{R}^d \\ X &\mapsto T_s(X) = \tilde{X} \end{aligned}$$

Where T_0 maps the values of $X \mid S = 0$ to $\tilde{X} \mid S = 0$ and T_1 maps the values of $X \mid S = 1$ to $\tilde{X} \mid S = 1$. It is expected that the correct choice of maps would lead to:

$$\mathcal{L}(T_0(X) \mid S = 0) = \mathcal{L}(T_1(X) \mid S = 1) \tag{5.1}$$

The common distribution μ needs to be as close to both μ_0 and μ_1 so that the amount of information lost during the transformation of the data is minimized and the use of the new dataset $\tilde{X}$ for predictions is still warranted. We thus need to find the optimal way to transport both conditional distributions to the common distribution. Thus, the aim of total repair is to find a mapping function T_s for which [5.1](#) holds true. The aim is to map both conditional distributions $\mu_s \sim X \mid S$ to the new common distribution $\mu = T_s \circ \mu_s$.



We need to quantify the amount of information lost under the transportation plan $T_s: \mathbb{R}^d \rightarrow \mathbb{R}^d$ which maps the original data X to $\tilde{X}$. In the event that the whole original data is available, the classification risk of a classifier $g \in \mathcal{G}$ is

$$R(g, X, S) = P(g(X, S) \neq Y).$$

On the other hand, when using the repair data, the risk of misclassification of a classifier h is

$$R(h, \tilde{X}) = P(h(\tilde{X}) \neq Y)$$

We can study the efficiency of the transportation plan by providing a bound for the difference between the minimal risks obtained for the best classifier with input data $\tilde{X}$, denoted by $R_B(\tilde{X})$ and for the best classifier g_B with input data (X, S) , defined by $R_B(X, S) = \inf_{g \in \mathcal{G}} R(g, X, S) = R(g_B, X, S)$. This bound is defined by:

$$\mathcal{E}(\tilde{X}) = R_B(\tilde{X}) - R_B(X, S).$$

However, since the total repair process will more than likely impair our ability to classify accurately, we will prefer to turn to another process which will instead only partially repair the data. Such a process will be described in a following section.

Below we detail two total repair methods what will be used in this work. Both methods focus on repairing the data completely but do so differently. The first method, widely used in existing literature, treats the data as deterministic and therefore changes each point by a unique point. The second method, introduced in [8] treats the data as probabilistic and therefore repairs the data by sending each point to multiple points.

Method A

Let $\{(X_i, S_i, Y_i), i = 1, \dots, N\}$ be an observed sample of (X, S, Y) where n_0 and n_1 are the number of observations in each protected class such that $n_0 + n_1 = N$. We suppose that the observations are grouped by the value of the protected variable S such that

$$\begin{aligned} \{x_{0,i}\} &= \mathcal{X}_0, & \text{if } s_i = 0, i = 1, \dots, n_0, \\ \{x_{0,j-n_0}\} &= \mathcal{X}_1, & \text{if } s_i = 1, j = n_0 + 1, \dots, N. \end{aligned}$$

In practice, the implementation of the repair procedure is based on the solution of the Kantorovich optimal transport problem as seen in chapter 5. Let us recall that the solution $\hat{\mathbf{P}} \in \mathbb{R}^{n_0 \times n_1}$ to this transport problem is the matrix that minimizes the following transportation cost between two distributions:

$$\min_{\mathbf{P}} \langle \mathbf{C}, \mathbf{P} \rangle, \quad (5.2)$$

such that the cost matrix entries are given by $c_{ij} = \|x_{0,i} - x_{1,j}\|$.

This first method is a one-for-one repair method, meaning that each point in $(\mathcal{X}_0, \mathcal{X}_1)$ will be sent to a singular other point as dictated by the following equations:

$$\begin{aligned} \tilde{x}_{0,i} &= \frac{\sum_{j=1}^{n_1} p_{ij}(\pi_0 x_{0,i} + \pi_1 x_{1,j})}{\sum_{j=1}^{n_1} p_{ij}} = \pi_0 x_{0,i} + n_0 \pi_1 \sum_{j=1}^{n_1} p_{ij} x_{1,j}, & 1 \leq i \leq n_0, \\ \tilde{x}_{1,j} &= \frac{\sum_{i=1}^{n_0} p_{ij}(\pi_0 x_{0,i} + \pi_1 x_{1,j})}{\sum_{i=1}^{n_0} p_{ij}} = \pi_1 x_{1,j} + n_1 \pi_0 \sum_{i=1}^{n_0} p_{ij} x_{0,i}, & 1 \leq j \leq n_1. \end{aligned}$$

The repaired set $\tilde{\mathcal{X}} = (\tilde{\mathcal{X}}_0, \tilde{\mathcal{X}}_1)$ will contain exactly N point due to the one-for-one nature of the repair such that $\tilde{\mathcal{X}}_0 = \{\tilde{x}_{0,i}\}$ and $\tilde{\mathcal{X}}_1 = \{\tilde{x}_{1,j}\}$. While this method repairs of the data and ensures the decision making will ultimately be fairer, since the two repaired sets are different, we cannot ensure total fairness. In order to do just that, we will require another method.

Method B

This second method is a method that ensures total fairness. In order to do so, this repair procedure treats each point as probabilistic, its mass can therefore be split into multiple repaired points, generating an extended repair set $\tilde{\mathcal{X}} = \tilde{\mathcal{X}}_0 \cup \tilde{\mathcal{X}}_1$. For every $i = 1, \dots, n_0, j = 1, \dots, n_1$, if $\hat{p}_{ij} > 0$, we define two points

$$\tilde{x}_{0,i,j} = \pi_0 x_{0,i} + \pi_1 x_{1,j} = \tilde{x}_{1,j,i} \quad (5.3)$$

such that we have the following repaired sets:

$$\begin{aligned} \tilde{\mathcal{X}}_0 &= \bigcup_{1 \leq i \leq n_0} \{\tilde{x}_{0,i,j}\} \\ \tilde{\mathcal{X}}_1 &= \bigcup_{1 \leq j \leq n_1} \{\tilde{x}_{1,j,i}\}. \end{aligned}$$

The new sets will allow us to obtain full fairness and total unpredictability as both repaired sets will follow equally distributed conditional distributions.

Theoretical application of the total repair procedure

The following example will serve as a visual aid in order to better understand the intricacies of the repair mechanism. This example is taken from [8]. Two samples $\mathcal{X}_0$ and $\mathcal{X}_1$ of sizes $n_0 = 4$ and $n_1 = 7$ respectively were simulated. The solution matrix that minimizes the transportation cost of the problem in 5.2 is

$$\hat{P} = \begin{bmatrix} \frac{1}{7} & \frac{1}{4} - \frac{1}{7} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{2}{7} - \frac{1}{4} & \frac{1}{7} & \frac{1}{14} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{14} & \frac{1}{7} & \frac{2}{7} - \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{4} - \frac{1}{7} & \frac{1}{7} \end{bmatrix}.$$

In figure 5.1 below, we have a visual representation of the total repair processes. The left part of the figure describes method A of total repair, where we have a one-for-one repair process where the green points are the repaired set of points for minority class ($S = 0$) whereas the green squares are the repaired set of points for the default class ($S = 1$). The right part of the figure shows us the second total repair method, in which each point is transported to several new points. In method B, we can see that each observation from the minority class is repaired by sending its probabilistic mass to as many points as there are non-zero entries in the corresponding row in the solution matrix above. For example, for the first observation in the minority class, we see that there are two non-zero entries, therefore this observation will be sent to two different points in the repaired set. This remark is also true for any observation in the default class, for example, for the second observation, we have only two non-zero entries in its corresponding column meaning that this observation will be transported to two distinct points in the repaired set.

We once again observe that the repaired set when using method B will be the same for both classes of the protected variable which will lead to statistical parity meaning that the probability of success is independent of the value of the protected attribute.

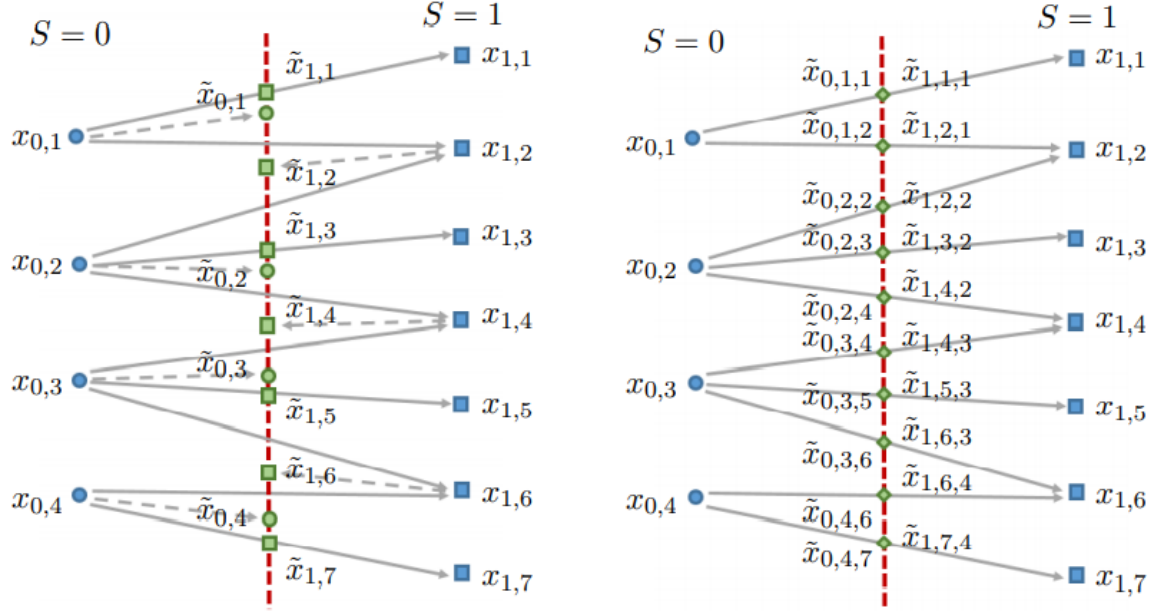


Figure 5.1: Visual representation of both method A (*left*) and method B (*right*) total repair procedures as taken from [8].

It is worthwhile to note that while method A usually does not achieve statistical parity, it is possible to make it so. In effect, in the event where both classes of the protected variable have

equal size n and all weights are uniform meaning that all non-zero entries of the solution to the Kantorovich optimal transport problem, the mass conservation constraint implies that each point is sent to a unique point as depicted in the following figure. In this special case, statistical parity is always achieved and the repaired set observations are given by:

$$\tilde{x}_{0,i} = \frac{x_{0,i} + x_{1,i}}{2} = \tilde{x}_{1,i}, \quad 1 \leq i \leq n.$$

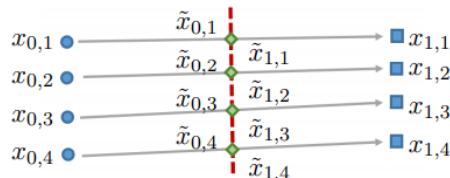


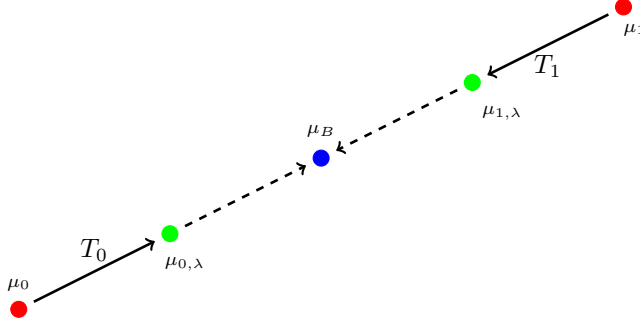
Figure 5.2: Visual representation of the total repair processes if the number of observations in both classes of the protected attribute are the same as taken from [8].

5.1.2 Partial repair

While total repair procedures can guarantee the repaired sets will approach statistical parity, it will be at the cost of classification accuracy. The aim being to ensure fair treatment to every individual, a high error rate in classification is not acceptable, as misclassification is by very essence unfair. Thus, [3] introduced a new repair procedure called partial repair. As the name implies, this procedure takes the data presented and only partially repairs it so as to avoid degrading the classification accuracy of the decision process. This method is a trade-off between classification accuracy and the fairness of the repaired data. This trade-off is executed by displacing each distribution towards the barycenter along the Wasserstein geodesic path between the two conditional distributions μ_0 and μ_1 , however not fully.

Let $\lambda \in [0, 1]$ be the parameter which encodes the level of repair desired for the dataset X such that $\lambda = 0$ translates to no repair and thus we are left with the original data and $\lambda = 1$ translates to total repair. Let Z be a random variable with distribution μ_B and T_s the optimal transport map transposing each conditional distribution towards that of the Wasserstein barycenter, μ_B . With $s = 0, 1$, the partially repaired conditional distributions are given by:

$$\begin{aligned} \mu_{s,\lambda} &= \mathcal{L}(\lambda Z + (1 - \lambda)T_s^{-1}(Z)) \\ &= \mathcal{L}(\lambda T_s(X) + (1 - \lambda)X \mid S = s) \end{aligned} \tag{5.4}$$



The choice of λ depends on two major factors:

- the classification error needs to be as small as possible,
- the protected variable needs to be non predictable from the modified data.

These two factors cannot be both fully satisfied, the choice of λ is a tradeoff between one and the other as having a small classification error using the modified data means that the original data was largely left unchanged while assuring non-predictability of the protected attribute means that both conditional distributions were brought closer to one another.

The distance between two distributions is measured using Total Variation distance. If P and Q are two probability measures, then this distance can be computed as:

$$d_{TV}(P, Q) = \min_{\pi \in \Pi(P, Q)} \pi(x \neq y).$$

If $\lambda \in (0, 1)$, following the definition of partial repair in (5.4), we have

$$d_{TV}(\mu_{0,\lambda}, \mu_{1,\lambda}) \leq P(T_0^{-1}(Z) \neq T_1^{-1}(Z)).$$

This means that the level of repair does not influence the distance in Total Variation between the modified distributions. Moreover, in some distinct situations, the above equation may even be an equality. Let us take the following example. Let us consider the following distributions of X in each class of S :

$$\begin{aligned} \mu_{0,0} &= U(K, K+1) \\ \mu_{1,0} &= U(-K-1, -K). \end{aligned}$$

The barycenter is thus $\mu_{0,1} = \mu_{1,1} = U(-1/2, 1/2)$. The partially repaired distribution are given by

$$\begin{aligned} \mu_{0,\lambda} &= U(-\lambda/2 + (1-\lambda)K, -\lambda/2 + (1-\lambda)K+1) \\ \mu_{1,\lambda} &= U(-\lambda/2 - (1-\lambda)(K+1), -\lambda/2 - (1-\lambda)(K+1)+1). \end{aligned}$$

The Total Variation distance is thus $d_{TV}(\mu_{0,\lambda}, \mu_{1,\lambda}) = \min(1, (1-\lambda)(2K+1))$. If $\lambda \leq \frac{2K}{2K+1}$, we have that $d_{TV}(\mu_{0,\lambda}, \mu_{1,\lambda}) = 1$. Thus, according to Theorem 2.0.3, the attribute S is ϵ -predictable

from the partially repaired dataset for values of λ arbitrarily close to 1. This provides incentive to only use small repair distances. To fix this, a process called random repair is proposed in [8].

Let Z and B be the targeted random variable with distribution μ_B and a Bernoulli variable with parameter λ respectively. Let $B \sim \mathcal{B}(\lambda)$ be independent of (X, S, Y) . The random repair process consists of randomly modifying the original distribution of X by choosing the target distribution μ_B or the original conditional distributions. The choice between both is regulated by the Bernoulli parameter λ . For $s \in 0, 1$, the repaired distributions are:

$$\begin{aligned}\tilde{\mu}_{s,\lambda} &= \mathcal{L}(BZ + (1 - B)T_s^{-1}(Z)) \\ &= \mathcal{L}(BT_s(X) + (1 - B)X \mid S = s).\end{aligned}\tag{5.5}$$

Let us note that for $\lambda = 0$, we do obtain the original distribution of the data while we obtain the desired distribution $\mathcal{L}(Z)$ when $\lambda = 1$. We have,

$$\begin{aligned}d_{TV}(\tilde{\mu}_{0,\lambda}, \tilde{\mu}_{1,\lambda}) &\leq P(BZ + (1 - B)T_0^{-1}(Z) \neq BZ + (1 - B)T_1^{-1}(Z)) \\ &= 1 - P(BZ + (1 - B)T_0^{-1}(Z) = BZ + (1 - B)T_1^{-1}(Z)) \\ &= 1 - P(B = 1) \\ &= 1 - \lambda.\end{aligned}$$

Let us recall that the smaller the value of the Total variation distance between two distribution, the more alike the two distributions are to one another. Therefore, this result suggests that in order to have the two final conditional distributions be as similar to each other as possible, one needs to set the Bernoulli parameter λ as close to 1 as possible.

Moreover, we observe that the misclassification error using the random repair process is given by the following, where $\tilde{X}_\lambda$ is the repaired data using Bernoulli parameter λ .

$$\begin{aligned}R(g, \tilde{X}_\lambda) &= P(g(\tilde{X}_\lambda) \neq Y) \\ &= (1 - \lambda)P(g(BT_s(X) + (1 - B)X) \neq Y \mid B = 0) + \lambda P(g(BT_s(X) + (1 - B)X) \neq Y \mid B = 1) \\ &= (1 - \lambda)P(g(X) \neq Y \mid B = 0) + \lambda P(g(T_s(X)) \neq Y \mid B = 1) \\ &= (1 - \lambda)P(g(X) \neq Y) + (g(T_s(X)) \neq Y).\end{aligned}$$

The misclassification error is therefore a mix of the errors of the total repair variable $T_s(X) = \tilde{X}$ and the original X variable. Thus, the use of the Wasserstein barycenter $Z \sim \mu_B$ remains warranted. We will henceforth advocate for the use of random repair in order to increase disparate impact while minimizing the effect of data repair on the accuracy of the classification.

Below we detail two partial repair methods what will be used in this work. Both methods focus on repairing the data so as to avoid compromising the classification accuracy but do so in different ways. The first method, called combinatorial repair works similarly to total repair as each conditional distribution is moved towards the barycenter along the Wasserstein geodesic path between the two conditional distributions. However, for partial repair, the conditional distribution are not moved all the way from their starting positions to the Wasserstein barycenter but rather to a point in-between. The second partial repair method that will be explored is that of random repair. In the random repair procedure, each observation will either not be repaired or be repaired using the total repair method B procedure as detailed in figure 5.1

Classic partial repair

The classic partial repair procedure is nothing more than a total repair procedure which was interrupted before reaching the targeted distribution μ_B . This method allows the user to decide whether to interrupt the repair of the data when a certain level of repair has been reached with minimal loss of classification accuracy. The level of repair can be easily controlled using a repair parameter $\lambda \in [0, 1]$. Since this procedure is based on method A, every point in the dataset $\mathcal{X}$ is replaced by a single point in $\tilde{\mathcal{X}}$. For all $1 \leq i \leq n_0$, $1 \leq j \leq n_1$, each point in $(\mathcal{X}_0, \mathcal{X}_1)$ is sent to a unique point defined as

$$\begin{aligned}\tilde{x}_{0,i} &= (1 - \lambda\pi_1)x_{0,i} + n_0\lambda\pi_1 \sum_{j=1}^{n_1} p_{ij}x_{1,j}, & 1 \leq i \leq n_0, \\ \tilde{x}_{1,j} &= (1 - \lambda\pi_0)x_{1,j} + n_1\lambda\pi_0 \sum_{i=1}^{n_0} p_{ij}x_{0,i}, & 1 \leq j \leq n_1.\end{aligned}$$

where $p_{i,j}$ are the entries of the solution to the Kantorovich optimal transport problem.

The repaired set $\tilde{\mathcal{X}} = (\tilde{\mathcal{X}}_0, \tilde{\mathcal{X}}_1)$ will contain exactly N observations due to the deterministic nature of the repair such that $\tilde{\mathcal{X}}_0 = \{x_{0,i,\lambda}\}$ and $\tilde{\mathcal{X}}_1 = \{x_{0,j,\lambda}\}$.

Random repair

The random repair procedure aims to be the ultimate compromise between building a repaired set of data which achieves statistical parity and building a repaired set of data with minimal loss of classification accuracy. In order to achieve this goal, the procedure will either fully repair an observation with repair procedure B or not at all and use the original value from the dataset in the new randomly repaired dataset $\tilde{\mathcal{X}}_\lambda$. The level of repair desired is regulated with parameter $\lambda \in [0, 1]$. For every observation in the original dataset, a binary output will be simulated using a Bernoulli distribution of parameter λ . If the output is 0, then the observation will not be repaired and will be used as is, however if the output is 1, then the observation will be repaired using method B detailed in the previous section on total repair. For every $i = 1, \dots, n_0, j = 1, \dots, n_1$, such that $\hat{p}_{ij} > 0$, where $\mathbf{P}$ is the solution to the Kantorovich optimal transport problem, we define the randomly repaired sets

$$\begin{aligned}\tilde{\mathcal{X}}_{0,\lambda} &= \bigcup_{1 \leq i \leq n_0} R_{0,i,\lambda} \\ \tilde{\mathcal{X}}_{1,\lambda} &= \bigcup_{1 \leq j \leq n_1} R_{1,j,\lambda},\end{aligned}$$

where $R_{0,i,\lambda}$ and $R_{1,j,\lambda}$ are the repaired set of points $x_{0,i}$ and $x_{1,j}$ respectively such that

$$\begin{aligned}R_{0,i,\lambda} &= \begin{cases} \{x_{0,i}\} & \text{if } b_i = 0 \\ \{\tilde{x}_{0,i,j}\} & \text{if } b_i = 1 \end{cases} \\ R_{1,j,\lambda} &= \begin{cases} \{x_{1,j}\} & \text{if } b_{n_0+j} = 0 \\ \{\tilde{x}_{1,j,i}\} & \text{if } b_{n_0+j} = 1 \end{cases},\end{aligned}$$

with $\tilde{x}_{0,i,j}$ and $\tilde{x}_{1,j,i}$ defined in equation [5.3](#)

Theoretical application of the random repair procedure

Let us reconsider the mock example from the previous section on total repair procedures. Figure [5.3](#) represents the random repair process with repair parameter $\lambda = \frac{1}{2}$. The randomly repaired sets are given by

$$\begin{aligned}\tilde{\mathcal{X}}_{0,\lambda} &= \{x_{0,1}, \tilde{x}_{0,2,2}, \tilde{x}_{0,2,3}, \tilde{x}_{0,2,4}, x_{0,3}, \tilde{x}_{0,4,6}, \tilde{x}_{0,4,7}\} \\ \tilde{\mathcal{X}}_{1,\lambda} &= \{\tilde{x}_{1,1,1}, x_{1,2}, \tilde{x}_{1,3,2}, \tilde{x}_{1,4,2}, \tilde{x}_{1,4,3}, \tilde{x}_{1,5,3}, x_{1,6}, \tilde{x}_{1,7,4}\}.\end{aligned}$$

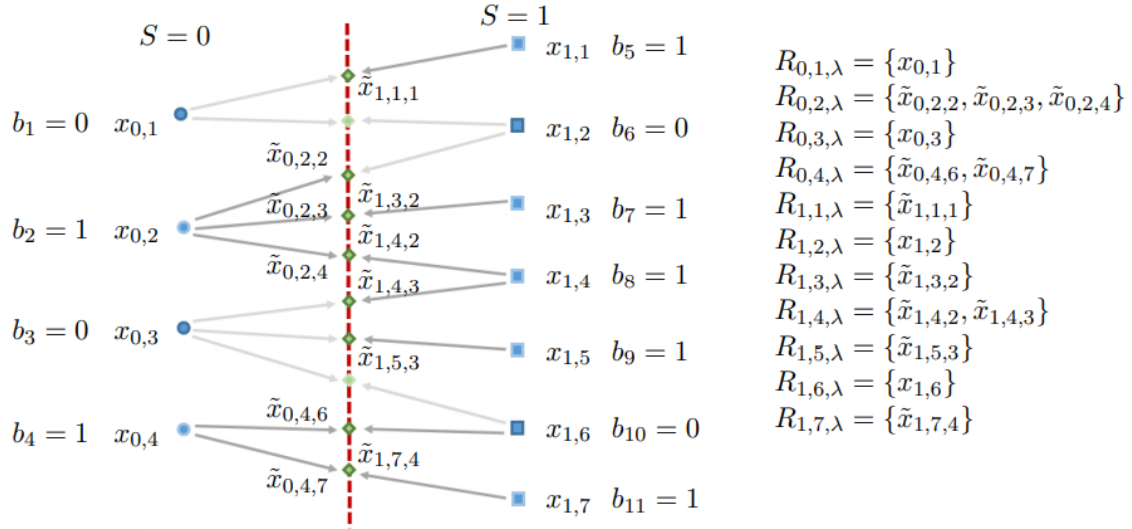


Figure 5.3: Example of a random repair process with repair parameter $\lambda = \frac{1}{2}$ as taken from [8](#)

5.2 Application with Simulated Data

In this section, we will illustrate the working of the four repair procedures we have introduced. We will create a dataset of five numerical variables, one sensitive variable S and one response variable Y . The sensitive attribute will be a binary variable such that $S=0$ is the minority class and $S=1$ is the default. To introduce bias in our dataset, we will simulate n_0 and n_1 observations from two multivariate normal distribution on $\mathbb{R}^5$ with means $\mu_0 = (4, 3, 3.5, 4, 4.5)$ and $\mu_1 = (5, 4, 4.5, 5, 5.5)$ for the minority and default class respectively. The covariance matrices for both classes are kept the same such that $\Sigma = \text{diag}(1, 1, 0.5, 1.5, 1)$. The response variable Y was simulated using a logit model for each group with different probabilities for success with the following parameters: $\beta_0 = (1, -1, -0.5, 1, -1, 1)$ and $\beta_1 = (1, -0.4, 1, -1, 1, -0.5)$. For $s_{0,1}$ the probability of success is given as $\pi_s(x) = \frac{e^{x\beta_s}}{1+e^{x\beta_s}}$. This is a crude way to ensure that there will be a certain level of bias

in our dataset. Let us remember that the purpose of this section is to investigate whether the implementation of the repair algorithms that were discussed earlier work correctly by looking at whether the results obtained make sense.

Once the simulated dataset has been completed, the data will be split using a 75-25% division of the data such that 75% is used as training data for the classifier and 25% of the data is used as a test set. The new logit classifier is then trained on our training set and then used to predict the binary response variable Y where a value of 0 indicates a failure and a value of 1 indicated a success. Using the predicted values, we can calculate the disparate impact as defined in equation 2.3:

$$DI(g, X, S) = \frac{P(g(X) = 1|S = 0)}{P(g(X) = 1|S = 1)}.$$

As per the following table, we can see that with a confidence of 95%, the logit classifier has a disparate impact at level 0.554 with respect to our sensitive variable and a BER of 0.267. A BER value as low as 0.267 indicates that there is a strong link between the value of the protected class of the individual and the decision of the classifier. We can thus safely declare that our data does indeed present discrimination.

DI	CI _{95%}	BER
0.434	(0.315, 0.554)	0.267

Table 5.1: Disparate impact and BER of the logit classifier on simulated data.

Since we can out discriminatory patterns in the data, we can proceed with repairing the data. Table 5.2 concisely shows the effect of both total repair procedures on the outcome of the classification model. As predicted, repair method A provides us with a disparate impact that is not quite 1 but that is very close such that 1 is included in the computed confidence interval. The value of the BER is also close to 0.5, meaning that we cannot accurately predict the which class an individual belongs to solely based on the value of the predicted response variable. We can thus say that our repaired dataset is, if not completely fair, fairer. On the other hand, when using total repair method B, we have disparate impact at level 1 and BER equal to exactly 0.5. This makes sense considering that the repaired data for both classes are made up of the same values.

Repair Procedure	DI	CI _{95%}	BER
A	0.998	(0.818, 1.179)	0.499
B	1	(0.882, 1.118)	0.5

Table 5.2: Disparate impact and BER of the logit classifier on simulated data.

Let us look at the implementation of partial repair procedures. The first procedure, is a method which resembles method A in the sense that the barycenters of both conditional distributions are shifted towards that of a third, common distribution. In the case of partial repair, however, the process is interrupted some way on the Wasserstein geodesic linking each conditional distribution to the third distribution. The second partial repair method is that of random repair. For each individual, a Bernoulli variable will be generated and, depending on the value of this variables, the observation will either not be repaired and used as such in the repaired dataset or it will be

repaired following the repair algorithm of method B. The level of repair applied to the dataset will be controlled with the use of repair parameter λ . Whether $\lambda = 0$ or $\lambda = 1$, then there will be either no repair or full repair and we expect the results of the classic and random repair procedures to mimic those of method A and B respectively. Figures 5.4 and 5.5 below show the variation of disparate impact and BER at each level of repair for both classic partial and random repair procedures. We can see that to obtain disparate impact at level 0.8 with a confidence of 95%, we only need a repair level of 0.525 for classic partial repair and 0.275 using random repair. Let us recall that 0.8 is the level at which one can legally suppose that a classifier does not discriminate with respect to a sensitive variable therefore attaining this bound is crucial when repair data. Being able to reach this bound with minimal repair is an advantage as it minimizes the changes in the original data, therefore less information will be lost. This is a great advantage of partial repair over total repair. We can also observe that for the classic partial repair procedure, at a level of repair of 0.525, we have a BER of 0.38 whereas a level of repair of 0.275 with random repair procedures yields a BER of 0.379. While these are considerable improvements on the original value of 0.267, these BER values do translate into a rather great predictability of the sensitive variable based on the value of the predicted response variable. Let us keep in mind that these results are based on simulated data and thus isn't representative of a real world example. Thus only the general trends can really be analysed and not the actual computed values of the different indexes. In the next section, we will be using a real dataset, with which we expect to see an increase in disparate impact and BER as the level of repair increases. Whether random repair remains the most efficient partial repair method to repair data remains to be seen.

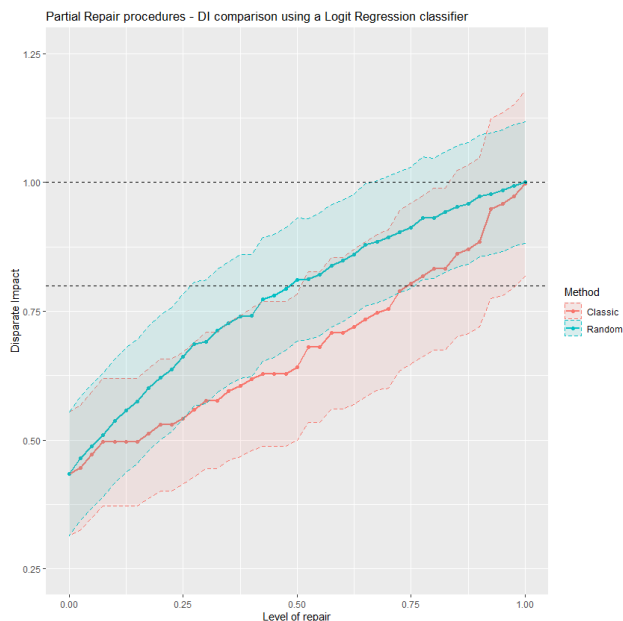


Figure 5.4: Disparate Impact and its 95% confidence interval of partially repaired data using a logit regression model with respect to the sensitive variable S and repair parameter λ . The values obtained with the classic partial repair are in red, the values obtained with random repair are in blue.

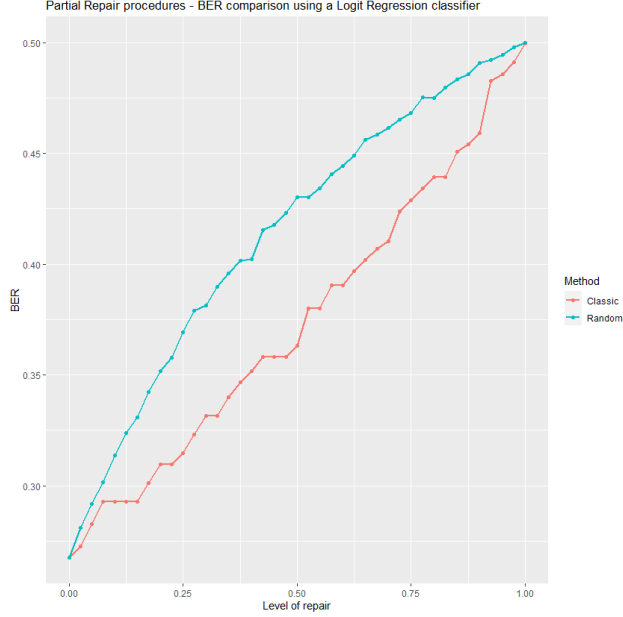


Figure 5.5: Balanced Error Rate of partially repaired data using a logit regression model with respect to the sensitive variable S and repair parameter λ . The values obtained with the classic partial repair are in red, the values obtained with random repair are in blue.

5.3 Application to a Real Example

In this section, we will put all four previously talked about repair procedure to the test using the *Adult Income Dataset*. This dataset contains 48 842 observations, each describing an individual according to 14 variables, 6 numeric and 8 categorical as well as a categorization of whether the individual has an income of more or less than 50 000\$ per annum. This last variable will be the focal point of our analysis. In effect, we will first look at whether we can distinguish a difference in salary with respect to a specific sensitive variable. In our case, the sensitive variable will be the sex of the individual. In order to flush out a potential difference in treatment, we will make use of Disparate Impact and provide a confidence interval as established in [11].

As good and complete as this dataset is, we will focus on only some numerical variables and keep only the sex of the participants. The variable `fnlwgt` was removed since it was only described as "finale weights" and had no clear meaning. We were thus left with five numerical variables: `age`, `education.num`, `capital.gain`, `capital.loss` and `hours.per.week`. Both classifiers, logistic regression and random forest were trained on those variables with the variable `income` used as the response attribute. The dataset contains 3 620 missing values. Due to the impressive size of the dataset, each row in which a missing value was present will be removed from the analysis.

We first need to make sure that our dataset does indeed present discriminatory behaviour. In order to corroborate this, we follow the same steps as we did in the previous section with our

simulated dataset. We separated the data into a training and a testing set using the pro rata of 75/25. The logit regression model and random forest are trained using the training set and then are used to predict the binary variable `pred.income`. A value of 0 translates to the individual being predicted to have an income of less than 50 000\$ per year whereas a value of 1 means the individuals is predicted to earn more than 50 000\$ per year. Using the definition for disparate impact as detailed in equation 2.3, we can quantify the amount of bias in our data.

Table 5.3 presents a summary of the performance of both classification algorithms. We can assert that with a confidence of 95%, the logit and random forest classifiers have disparate impact at level 0.490 and 0.532 respectively with respect to the sensitive variable `sex`. In plain words, this indicated that for every 100 successes in the protected variable default class, there are only 49 successes in the minority class using the logit regression model and 53 with random forests. Since our protected variable is the sex of the individuals, we can assert that the dataset clearly presents discriminatory behaviour favouring males. With such clear unfairness in our dataset, we can thus continue our analysis and repair our dataset to minimize the difference in treatment between males and females.

Statistical Model	DI	CI _{95%}	Error	Accuracy
Logit	0.383	(0.276, 0.490)	0.178	0.822
Random Forest	0.424	(0.315, 0.532)	0.156	0.844

Table 5.3: Disparate impact and classification accuracy with respect to the protected variable `sex` with logit regression and random forests models.

The total repair procedure is applied on the test set for practical reasons. Most firms using decision making algorithms have paid a lot of money to have their algorithm created and having to retrain a prediction model with slightly better results can be too costly and time consuming to be warranted. Moreover, training a new algorithm means handing over potentially sensitive data which can be risky. Therefore, the logit regression and random forests models trained on our data with not be retrained and only the test set will be subjected to the repair procedures.

Table 5.4 concisely shows the effect of both total repair procedures on the outcome of our two classification models. As predicted we can see that the repair method A does not translate into a disparate value of 1 meaning that we do not reach statistical parity. Moreover, when using the logit regression, we even compute a disparate impact of more than 1. Can we thus say that the minority class is now favoured ? Looking at the classification error, we can see that we have an almost 30% error rate, meaning that nearly a third of all predictions are misclassified. Furthermore, at a confidence level of 95%, since 1 is part of the confidence interval for the value of disparate impact, we can reject the hypothesis that our logit classifier offers an advantage to women in favour of the hypothesis which states that there is no perceptible discriminatory behaviour. The same argument can be made when using method A of repair with a random forest algorithm. A high error rate and a 95% confidence interval which encompasses the value 1 means that no discernible discrimination is present when applying our classification algorithm to the repaired set. On the other had, when using repair procedure B, we can see that we achieve statistical parity. In effect, this was we had predicted as both repaired sets for each value of the protected attribute were the same.

Finally lets us look at the results when using partial repair methods. The first method, which we will call the "classic" partial repair, is a method akin to method A for total repair as data is repaired using the same procedure however instead of shifting the barycenters, μ_0 and μ_1 , of both

Statistical Model	Repair Procedure	DI	CI _{95%}	Error	Accuracy
Logit	A	1.016	(0.799, 1.232)	0.292	0.708
Logit	B	1	(0.854, 1.146)	0.302	0.698
Random Forest	A	0.936	(0.718, 1.154)	0.283	0.717
Random Forest	B	1	(0.843, 1.157)	0.312	0.688

Table 5.4: Disparate impact and classification accuracy with respect to the protected variable **sex** with total repair procedures applied on the logit regression and random forests models.

conditional distributions to that of the desired totally repaired distributions, μ_B , we interrupt the process some way on the Wasserstein geodesic linking μ_0 and μ_1 to μ_B . We will be calculating the disparate impact and classification accuracy for every value of λ between 0 and 1, using an incrementation of 0.025 at every step. When $\lambda = 0$, we fully expect the values of disparate impact and accuracy to be those of table 5.3 whereas with $\lambda = 1$, we should expect disparate impact and error rate to mimic the values of the rows describing method A of total repair in table 5.4. The second method use is random repair. This method is based on method B, in effect depending on the value of a randomly generated Bernoulli variable, each observation in the dataset will either be taken as is and placed in the repaired set or fully repaired following method B. If $\lambda = 0$ then no repair will be made and we expect to find the same results as table 5.3 once again, whereas if $\lambda = 1$, we expect the results to follow that of method B described in table 5.4 as a repair level of 1 means that every single observation is being repaired. Since random repair is based on a random Bernoulli process, for every value of λ we will run the repair algorithm 100 times. We will thus compute the disparate impact, lower and upper intervals as well as the classification accuracy 100 times. For each of these four categories, we will average out all 100 values in order to get an accurate measurement.

Below, we can observe the outputs of the simulation undertaken with our dataset. We can visualise the improvement in disparate impact as the level of repair increases for both procedures for both logit regression and random forests classifiers. Figure 5.6 shows the evolution of disparate impact when using a logit model. While both repair procedures seem to converge towards a disparate impact of 1 as the level of repair tends to 1, we can see that the amount of repair needed to achieve a confidence interval for the disparate impact at level 0.8 is much smaller for random repair than for the classic partial repair. Lets us recall that a disparate impact of 0.8 or greater is the legal limit for which a selection rate is regarded as not discriminatory ([7]). It is therefore paramount for a firm to be able to clear this threshold. The amount of repair necessary to reach a confidence interval for disparate impact at level 0.8 using a logit classifier is 0.325 with the random repair procedure and 0.575 with the classic partial repair method. The accuracy of classification with such repairs are 0.69 and 0.71 respectively. When using a random forest classifier, the amount of repair need to reach a disparate impact at level 0.8 with a confidence interval at level 0.95 is 0.375 for random repair and 0.4 for classic partial repair. The accuracy of classification is thus 0.68 and 0.71 respectively. With both classifiers, the sensitive variable is also less predictable when using a random repair. This is noticeable with the logit classifier where not only is the amount of repair needed to achieved disparate impact at level 0.8 significantly lower than with classic repair, the value of the BER at this level of repair is also significantly higher, meaning that random repair makes it harder to accurately guess the value of the sensitive variable based on the value of the

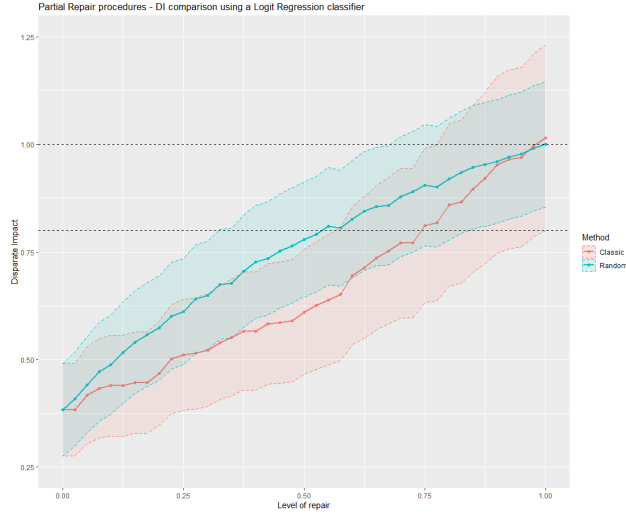


Figure 5.6: Disparate Impact and its 95% confidence interval of partially repaired data using a logit regression model with respect to the variable `sex`. The values obtained with the classic partial repair are in red, the values obtained with random repair are in blue.

outcome. The same can be observed using a random forest, though the effects are slightly less noticeable than with the logit regression.

When trying to repair data in order to reduce or eliminate bias and discrimination in a machine learning algorithm, a partial repair procedure will most likely be preferred to a total repair procedure. In effect, we have just seen that in order to achieve disparate impact at level 0.8, the level of repair is quite low, in our case it was at least 0.325 and at most 0.575. We can see that very little repair on the data can have enormous consequences on the fairness of the algorithm. We can also note the classification accuracy of partial repair remains rather unchanged with respect to the amount of repair done. Therefore, we can observe that total repair is most likely overkill when trying to reduce or eliminate bias in machine learning. We can now ask ourselves, is classic partial repair better than random repair or not? We can see that in order to achieved disparate impact at level 0.8, random repair requires less repair regardless of the classifier used. The fact that we need less repair makes it that less information is lost during the transfer of each conditional distribution towards the new distribution barycenter. This is a big advantage of random repair. Random repair also boasts better BER values than classic partial repair, in effect, at every level of repair, random repair provides a dataset where the sensitive variable is less predictable than with classic partial repair. On the other hand, in order to get robust results using the random repair procedure, we had to run the simulation 100 times for every level of repair using both logit and random forest classifiers and average out all results for each value of α . This can be an enormous computational demand. In effect, while the results for classic partial repair could be outputted in a few minutes, random repair computations took 4 to 5 times longer. While random repair procedures seems to output better results regardless of the classifier used, this is at a slight cost in accuracy of classification and at the cost of needing more computational power and time.

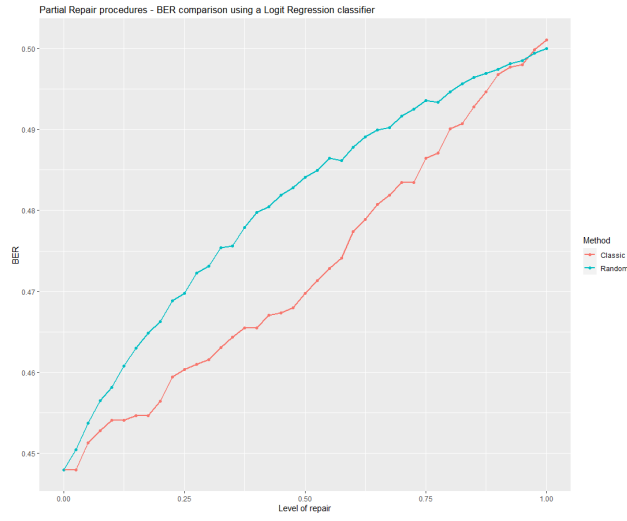


Figure 5.7: Balanced Error Rate of partially repaired data using a logit regression model with respect to the variable `sex`. The values obtained with the classic partial repair are in red, the values obtained with random repair are in blue.

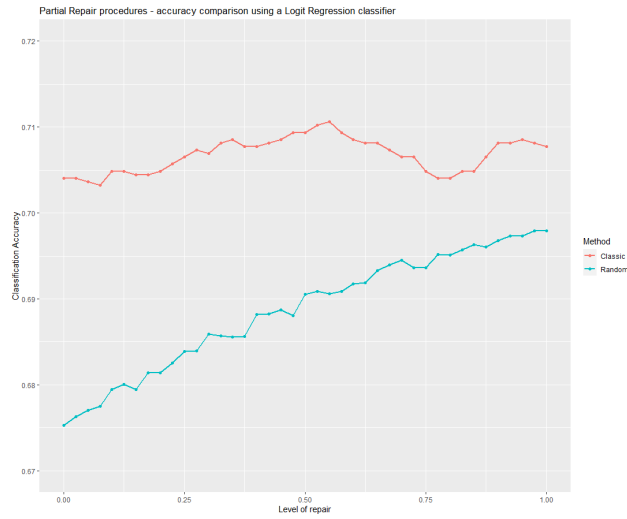


Figure 5.8: Classification accuracy of the logit classifier of both partial repair methods, the classic method in red and random repair method in blue.

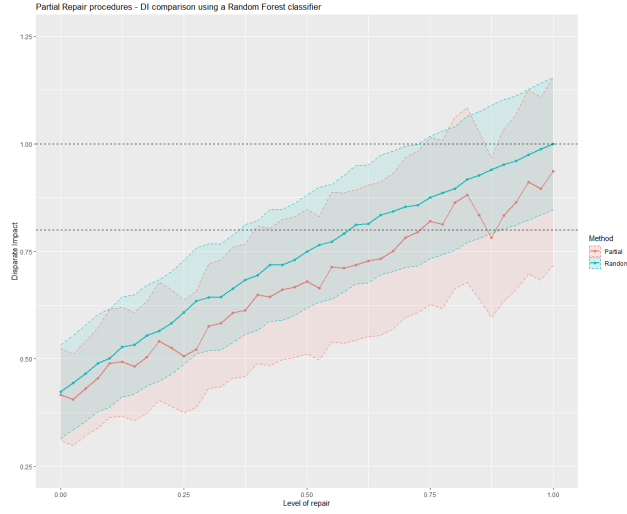


Figure 5.9: Disparate Impact and its 95% confidence interval of partially repaired data using a random forest model with respect to the variable `sex`. The values obtained with the classic partial repair are in red, the values obtained with random repair are in blue.

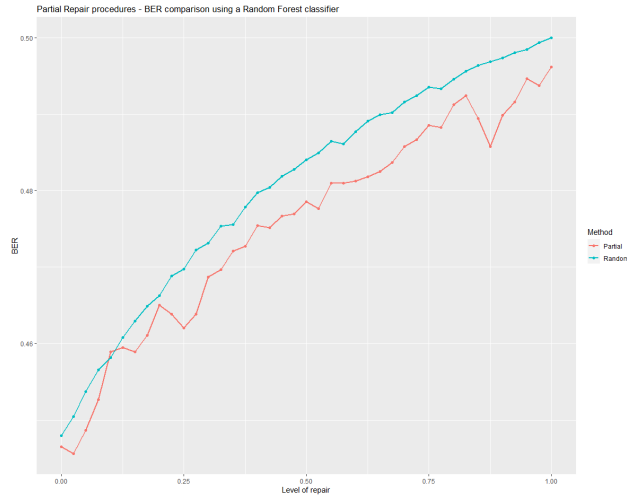


Figure 5.10: Balanced Error Rate of partially repaired data using a random forest model with respect to the variable `sex`. The values obtained with the classic partial repair are in red, the values obtained with random repair are in blue.

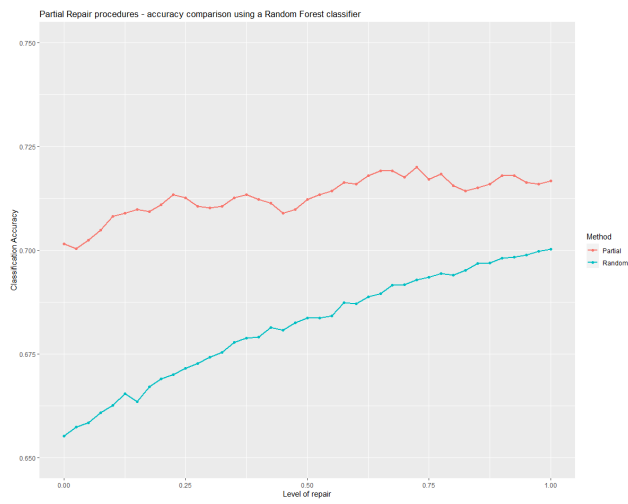


Figure 5.11: Classification accuracy of the random forest classifier of both partial repair methods, the classic method in red and random repair method in blue.

Chapter 6

Conclusion

6.1 Overview and Results

The ever expanding use of machine learning algorithms in everyday private life and especially in the professional world has shown how careful one needs to be when implementing such procedures as discriminatory behaviour in the data which may not be apparent at first glance can make its way in the decision making process of the algorithm. This opens up the discussion on fairness in machine learning and the need for techniques to detect, quantify and remove bias in a decision making algorithm if need be. This work sought to apply optimal transport theory to displace conditional distribution in order to repair data so as to remove hidden discriminatory pattern.

In Chapter 2 we have provided a an explanation motivating the need for the study of fairness in machine learning. In effect, decision making algorithms are used in many facets of professional life, many of them where a wrong decision can be harmful or prejudicial to the well-being of an individual such as is the case with the COMPAS algorithm which is used to assess the recidivism risk of inmates in the United States 2. This algorithm was found to wrongly attribute higher risks of recidivism to black people, urging the need for a multidisciplinary approach to detect and remove bias an machine learning to promote fairer treatment of individuals.

We also introduced the *Adult Income Dataset* 1, a database regrouping information on 48 842 individuals and whether each of them earned more or less than 50 000\$ per annum. This database is very interesting in the context of fair learning as it provides the user with the values of the sensitive and response variables. This dataset is also widely used in fair learning as it does contain discriminatory behaviour with respect to **sex** and **race**. In this work we will focus of the bias with respect to **sex**.

In order to be able to judge whether we have succeeded in reducing bias, we first need to accurately measure the level of fairness in our dataset. We introduced two methods of bias quantification: disparate impact and balanced error rate as they were defined in 3. Disparate impact is a measure of fairness in the sense that it compare the probabilities of success of two different classes of individuals while the balanced error rate is a measure of the predictability of the attribute of an individual.

Chapter 3 is a demonstration of the difficulty of making fair machine learning algorithms, especially when the data contains discriminatory patterns. Standard regulations dictate for

example that sensitive variables should be removed from any training data, however results show that this attempt to promote fairness does not significantly reduce the bias in the data. Other procedures such as positive discrimination were shown to have mitigated results. In effect, while this procedure did significantly reduced bias, it also significantly reduced the accuracy of classification. Moreover, as per its name, positive discrimination is a type of discrimination and thus hardly appropriate when trying to promote fairness in machine learning.

Chapter 4 was an introduction of optimal transport theory. Optimal transport is a mathematical field which specializes in comparing multiple distributions and looks at the multiple ways to morph and reshape one distribution into another while minimising the information lost because of the displacement.

Chapter 5 showed how optimal transport theory could be used in the context of fair learning to fracture the link between variables in a dataset. This is called repairing the data. This chapter focused on four data repair procedures and their implementation on both simulated data and the *Adult Income Dataset*. These repair procedures can be classified into two groups. Total repair procedures are when a dataset is divided according to the value of the sensitive variable and the resulting conditional distributions are all moved to a new common distribution. Partial repair on the other hand are procedures in which the conditional distributions are move towards a common distribution, without reaching it.

It was found that partial repair procedure offered on the whole more control over the data to the user. In effect, the legal disparate impact level threshold could be reached with a repair level as low as 0.325. This indicates that the dataset is only very slightly altered in order to significantly boost fairness and remove predictability of the attribute. We also found that a random repair procedure outperformed the classic partial repair procedure as it boasted better results for both disparate impact and balanced error rate at every level of repair.

6.2 Limitations and Future Work Avenues

When using the random repair procedure, the data was split using a pro rata of 95/5 instead of the usual 75/25 split. This is due to the fact that for random repair, 100 iterations of the repair procedure were computed to average out all the results so as to get consistant and robust results for the disparate impact and balanced error rate for every level of repair. When using the 75/25 split, the time taken to compute the necessary number of iterations was such that my computer would start overheating. Therefore, a conscious decision was made to keep the number of iterations at 100 but alter the split of the data so as to reduce the computation time using the test set.

This could explain the unusual behaviour of the classification accuracy for the random repair procedure in figures 5.8 and 5.11. In effect, as the level of repair increases, we would expect the accuracy of classification to decrease slightly since data is altered and the predictive outcome is also changed in some cases just as is observed in 8. The fact that the test data had been set to contain only 5% of the data could explain this anomaly.

Our work also focused on only two classifiers, logistic regression and random forest. Given the large amount of classifiers, it can be interesting to study whether certain types of repair procedures are more suited to certain types of classifiers. Moreover, the discussion of disparate impact is intrinsically tied to the legal definition of fairness as per United States law. A potential avenue of

improvement in the search for fairness in machine learning would be to work on other definition or fairness in other legal frameworks and find a unifying formulation. Finally, our data repair procedures use numerical or binary variables. Being able to extend the repair procedures to include categorical variables would be most beneficial for dealing with algorithmic bias in machine learning.

Bibliography

- [1] US Census Bureau. Adult income dataset. <https://www.kaggle.com/wenruli/adult-income-dataset>. Accessed: December 2020.
- [2] Surya Mattu Julia Angwin, Jeff Larson and Lauren Kirchner. Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks. <https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing>, 2016. Accessed: 07/07/2021.
- [3] Michael Feldman, Sorelle Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. 2015.
- [4] Gaspard Monge. *Mémoire sur la théorie des déblais et des remblais*. Histoire de l'Académie Royale des Sciences, 1781.
- [5] Robert Rothmann, Jaro Krieger-Lamina, and Walter Peissl. Credit scoring in Österreich. *Institut für Technikfolgen-Abschätzung (ITA) der Österreichischen Akademie der Wissenschaften (ÖAW); Studie in Kooperation mit der Bundesarbeitskammer (AK) Wien; ITA-Projektbericht Nr.: A66 ISSN: 1819-1320 ISSN-online: 1818-6556*, 07 2014.
- [6] Mikella Hurley and Julius Adebayo. Credit scoring in the era of big data. 2016.
- [7] State of California Fair Employment Practice Commission. Uniform guidelines on employee selection procedures. <https://www.govinfo.gov/content/pkg/CFR-2017-title29-vol4/xml/CFR-2017-title29-vol4-part1607.xml>, 1978. Accessed: 30/04/2021.
- [8] Eustasio del Barrio, Fabrice Gamboa, Paula Gordaliza, and Jean-Michel Loubes. Obtaining fairness using optimal transport theory. 2018.
- [9] Gabriel Peyré and Marco Cuturi. Computational optimal transport. 2020.
- [10] Cédric Villani. *Topics in Optimal Transportation*. American Mathematical Society, Providence, Rhode Island, 2003.
- [11] Philippe Besse, Eustasio del Barrio, Paula Gordaliza, and Jean-Michel Loubes. Confidence intervals for testing disparate impact in fair learning, 2018.
- [12] Sorelle A. Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P. Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. 2018.

- [13] Philippe Besse, Eustasio del Barrio, Paula Gordaliza, Jean-Michel Loubes, and Laurent Risser. A survey of bias in machine learning through the prism of statistical parity for the adult data set. 2020.
- [14] Matthew Thorpe. Introduction to optimal transport. 2018.
- [15] Weiwen Miao. Did the results of promotion exams have a disparate impact on minorities? using statistical evidence in *ricci v. DeStefano*. *Journal of Statistics Education*, 18(3), November 2010.
- [16] Emily Black, Samuel Yeom, and Matt Fredrikson. Fliptest. *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, Jan 2020.
- [17] Eustasio del Barrio, Paula Gordaliza, and Jean-Michel Loubes. Review of mathematical frameworks for fairness in machine learning. 2020.
- [18] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.

