Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

School of Engineering

A Study of Fairness Measures and Repairs in Artificial Intelligence

Daniel Smith

Supervisor: Associate Prof. Anthony Quinn

October 2, 2023

A dissertation submitted in partial fulfilment
of the requirements for the degree of
MAI (Electronic and Computer Engineering)

Declaration

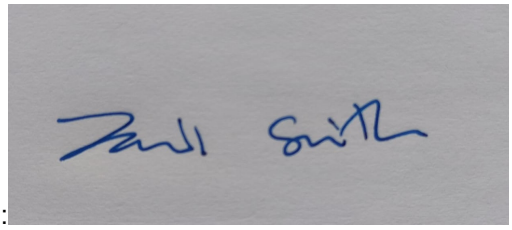
I hereby declare that this dissertation is entirely my own work and that it has not been submitted as an exercise for a degree at this or any other university.

I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at `http://www.tcd.ie/calendar`.

I have completed the Online Tutorial on avoiding plagiarism 'Ready Steady Write', located at `http://tcd-ie.libguides.com/plagiarism/ready-steady-write`.

I consent to the examiner retaining a copy of the thesis beyond the examining period, should they so wish (EU GDPR May 2018).

I agree that this thesis will not be publicly available, but will be available to TCD staff and students in the University's open access institutional repository on the Trinity domain only, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement. **Please consult with your supervisor on this last item before agreeing, and delete if you do not consent**

Signed: 
Date: 14/04/2023

Abstract

This dissertation proposes a novel approach for data repair in the context of fairness, using a new fairness measurement based on Kullback-Leibler divergence (KLD), and the concept of Conditional Independence (CI). This approach is compared against state-of-the-art data repair methods that use disparate impact (DI) as the fairness measurement.

The proposed repair method is based on the merging of Gaussian distributions, where as the novel measurement of fairness is inspired by the fully probabilistic design (FPD) framework, which considers all inputs to the system as probability distributions, and KLD is used as a measure for model comparison. The KLD-based fairness metric is designed with the goal of minimizing the divergence between the unfair existing model and the fair ideal, while achieving a fairness goal.

The proposed approach is then compared against existing work of Nicolas Courty and Paula Gordaliza et al., who developed Optimal Transport solutions for domain adaptation and the application of OT-based data repairs to correct for biases, respectively. The proposed method aims to learn a transport plan that minimizes the KLD loss between the current model and the fair model in a way that aligns with Optimal Transport theory.

This dissertation presents experimental results that demonstrate the effectiveness of the proposed approach and its comparability to state-of-the-art methods. It acknowledges that further research is required before it can be shown to be an alternative to an Optimal Transport approach. The proposed approach offers a promising potential alternative to traditional fairness measurements such as DI, and the use of KLD provides a flexible and powerful tool for measuring fairness in machine learning models.

Lay Abstract

The use of artificial intelligence to help governments, judges, doctors and educators make decisions, is increasing. So, when AI is used to make a decision, it should be possible to prove that the decision was based on information (data) that was correct or 'true' and there was no bias or (mistakes), involved.

The fairness of any decision usually depends on more than one factor. For example, when a person is offered a job, for that to be considered a 'fair' decision, it is accepted that the cv's of all the applicants applying for the job should be considered equally, those selected for interview should be selected on the basis of clear and reasonable rules and the interview panel should have open minds, be without prejudices, and capable of making good decisions.

In AI, computer programs (algothims) are written which can 'learn' from data. This is known as machine learning. The process is built on mathematical computations and different approaches have been developed by software engineers. As with the job offer example there are different aspects to ensuring an AI decision is fair. Despite ongoing research and development, no one way has been agreed on, to decide if an AI system, and all its component parts are fair and accurate and trustworthy.

This dissertation looks at one aspect of the process of AI decision making i.e. how to answer the question 'is the process is fair' and if not, how to fix that. This is done by firstly combining existing methods to define a new way of measuring what is fair, then developing data transformations to enable the model to 'learn' under different sets of conditions, and lastly comparing the results with state of the art or currently well known and accepted methods, to decide how the new method compares.

This dissertation, like most AI research, uses work done by others as its starting point. The building blocks used included the measuring fairness approach known as 'Kullback-Leibler divergence (KLD)', and a repair method developed by Nicolas Courty and Paula Gordaliza using optimal transport theory.

The results of the work were clear but not very decisive. The new method does work well for some sets of conditions, but not others. It produces similar but not better outcomes than the state of the art method that it was compared with. Before this could be used as a way to repair for fairness, there would need to be further research done.

Acknowledgements

Firstly, I want to thank my research supervisor, Assoc. Prof. Anthony Quinn. As well all the invaluable technical support he gave, which allowed the project to progress, I also what to thank him for the time he gave me and the patience he showed, especially given the other demands on his time.
Secondly, to my friends and family who supported me during this time.

Contents

List of Figures

Nomenclature

0.0.1 Acronyms

PDF	Probability Density Function
PMF	Probability Mass Function
MC	Monte Carlo
ARI	Adjusted Rand Index
KLD	Kullback-Leibler Divergence
GMM	Gaussian Mixture Model
DI	Disparate Impact
OT	Optimal Transport

0.0.2 Notational Conventions

X_s	Source Domain
X_t	Target Domain
k	the number of clusters
A_p	Protected Attribute
A_u	Unprotected Attribute
d	The dimension of the Source Domain
t	The dimension of the Target Domain
μ	Mean
r	Variance
ρ	Correlation Coefficient
Σ	Covariance matrix
f(x)	PDF of the synthesis distribution
$\hat{f}(x)$	PDF of the estimated distribution
$D(f\|\|\hat{f})$	The KLD from probability density functions f(x) and $\hat{f}(x)$
A_e	affine transformation
b_e	translation (b_e)
λ	Amount of repair $\in$ [0,1]
T	Transformation
W	W is the Wasserstein distance or Earthmovers distance
f_B	Wasserstein baycenter

1 Introduction

As of right now, 40% of Irish companies currently use Artificial Intelligence [1]. Decisions in the past that were made by people (judges, bank managers, doctors, police) are now being made using artificial intelligence. Examples include decisions made at passport control in airports all over the world; decisions around access to education, employment and public benefits; creditworthiness, administration of justice and medical decisions.

The increasing use of AI is of such significance that, in 2021, the European Commission published The 2021 Coordinated Plan on Artificial Intelligence [2] and a draft law to regulate artificial intelligence (AI) in the European Union.

However at this time, there is no cross cutting legislation in Ireland or the EU to control the 'trustworthiness' of AI. But it is clear that there will be a need for AI developers to be in a position to demonstrate that the AI tools available are capable of making unbiased and accurate decisions which can be verified and certified. [3]. To summarize, the EU and Irish government recognize the values but also the risks associated with AI, and are working towards legislation to manage those risks. There is still a lot of work to be done, including the development of standards and agreements on fundamentals, for example a definition of "what is fair" and an agreed methodology or methodologies "to repair for unfairness"

Against this background, the motivation of this dissertation was to explore the current state-of-art thinking on defining and assessing fairness and unfairness in AI, and to develop a method for repairing for unfairness when found.

1.0.1 Framework for defining fairness in Machine Learning

Different definitions and terminologies have evolved in the field of Machine Learning (ML) to describe the notion of what is a 'fair decision'. For the purposes of this dissertation, the idea of fairness in ML is formalised as follows:

Fairness in ML, is considered to exist when a decision made by ML model, is not seen to propagate or introduce unfair bias with respect to a sensitive attribute such as gender, leading for example to a decision by the model, which is independent of gender i.e. whether the subject is 'male' or 'not male'.

At this point it is import to note that unfairness and bias are not the same, as unfairness implies bias. However, bias does not necessarily imply unfairness. The literature often uses these words interchangeably.

Consider an example expressed in mathematical terms, where a binary classification task is being undertaken by a bank, i.e. the decision to award a loan or not, where $\hat{Y} \in 0, 1$ and 0 represents 'no loan' and 1 represents a 'loan'.

The task becomes to learn a mapping G: X $\rightarrow \hat{Y}$. Where $\mathbf{X} \in \mathbf{R}^d$ is the feature space of dimension d and $\mathbf{x}_i$ is a feature of the feature space X.

The task of deciding to approve a loan application or not, is a typical decision a bank might want to complete. The question then becomes what features $\mathbf{x}_i$ will the model used by the bank be trained on. The bank's objective will be to train on set of $\mathbf{x}_n$ that makes the most accurate and therefore most profitable decision $\hat{Y}$.

In law there exists two labels for Attributes A, Unprotected Attributes (A_u), such as 'education level', 'income level' and 'occupation'. The second is Protected Attributes (A_p) such as 'race', 'gender' and 'religion'.

Society requires that the bank's decision, $\hat{Y}$, be fair, i.e. not dependent on Protected Attributes (A_p). Therefore the study of fairness in ML, tries to answer the question 'how much the decision $\hat{Y}$ depends on A_p' and 'how do we measure and correct for it'.

2 Literature Review

As outlined in the introduction, this dissertation addresses fairness measurement. Therefore the first purpose of the literature review becomes, what are the current methods of quantifying and repairing for fairness in ML.The review also explored the datasets available and made some commentary on them.

These were the topics of the literature review, which is summarized below.

2.1 Fairness Measurements and Repairs

There are many ways of defining algorithmic fairness but no agreement on the best method. In "Measuring Algorithmic Fairness" by Hellman et al. [4], they argue that measures can be conceptual, normative or legal. According to Hellman, one measure frequently used requires that the score an algorithm produces should be equally accurate for members of legally protected groups, Caucasian and non Caucasian for example, and the second requires that the algorithm produces the same percentage of false positives or false negatives for each of the groups at issue.

It should be noted that the concept of fairness cannot be considered without also considering the concept of accuracy and the trade off that arises between them in machine learning.

2.1.1 Measurements of Algorithmic Bias

The notation is as follows: $\hat{Y}$, is the result of the decision function, Y is the ground truth result and A_p is the class of protected attributes. The following will all be binary classification, where $\hat{Y}$ is the decision from a bank to give a loan or not, $\hat{Y} \in {}'loan, noloan'$. Y can be seen as the ideal outcome with no unfair bias. A_p is this case being if the person is underprivileged or not.

The most frequently occurring measurements of Algorithmic bias in the literature, are as follows:

1. Disparate Impact

In "Fairness through awareness" by Hardt et al. [5] they proposed the use of disparate impact (DI) as a formal definition of fairness in classification tasks. DI requires a high ratio of positive prediction across different protected groups, given by:

$$\frac{P[\hat{Y} = 1/A_p = 1]}{P[\hat{Y} = 1/A_p = 0]} \geq 1 - \varepsilon \tag{2.1}$$

A higher value of DI represents similar results across Protected Groups and therefore an indication that a more fair decision is being made. For example, where the protected attribute is gender, a DI of 0.5 would indicate for every 5 females who get a loan, 10 males get a loan. DI is marginal with respect to A_u, therefore it will not account for a difference in underlying distributions of A_u given A_p. ϵ in equation 2.1, represents the allowed margin for unfairness. Typically, a value of ϵ = 0.8 is used to represent "80 percent rule" in disparate impact law [6].

Note: The term "disparate" means fundamentally different or distinct in quality or character. The ideal DI is 1, indicating an a equal ratio of positive predictions across different A_p. Therefore, this term is misleading as a lower DI indicates a higher level of unfairness.

In "Certifying and removing disparate impact" by Feldman et al. [6] they argue the strength of disparate impact as a measure of fairness in classification tasks, but further research is warranted.

In "Disparate Impact in Big Data Policing" by Selbst et al [7], the use of predictive policing algorithms was examined and the authors demonstrated how they can lead to a high disparate impact value for minority communities. The authors argued that, even if the algorithms are unbiased in terms of the features they use, they can still result in discriminatory outcomes, if they rely on data that reflects and reinforces historical biases. They suggest that policymakers consider disparate impact when evaluating the use of predictive policing tools.

2. Demographic Parity

Demographic Parity is a concept first mentioned in "On balanced sets and cores" by Shapley et al. [8], Demographic parity was then defined as a condition in which all balanced sets have the same proportion of individuals from each demographic group, regardless of their qualifications or characteristics. It has been used in many fields and has been adopted as a fairness metric.

In "Three naive Bayes approaches for discrimination-free classification" by Calders et al, [9], they propose three approaches for discrimination-free classification with Demographic Parity as the fairness goal. Similar to that of disparate impact, Demographic Parity quantifies the difference between positive outcomes for privileged and non privileged groups. Its computed as :

$$P[\hat{Y} = 1/A_p = 1] - P[\hat{Y} = 1/A_p = 0] \geq 1 - \varepsilon \tag{2.2}$$

A lower value of Demographic parity represents similar acceptance rates across groups and therefore more a more fair result. As with DI, Demographic Parity is also marginal w.r.t A_u. The issue of being marginal w.r.t A_u means that if a data or model repair is to occur with a Demographic Parity goal in mind there will be an fairness accuracy trade off.

3. Equalizing odds

In "Equality of opportunity" by Hardt et al. [10], they present another method. Equalizing odds measurement computes the difference in false positive rates (FPR) and false negative rates (FNR) of two groups (S, $C = \{1, 0\}$). This measurement is computed as:

$$|P[\hat{Y} = 1/A_p = 1, Y = 0] - P[\hat{Y} = 1/A_p = 0, Y = 0]| \leq \varepsilon \tag{2.3}$$

$$|P[\hat{Y} = 1/A_p = 1, Y = 1] - P[\hat{Y} = 1/A_p = 0, Y = 1]| \leq \varepsilon \tag{2.4}$$

Hardt [10], says that an outcome $\hat{Y}$ satisfies equalising odds with respect to the protected attribute A_p and outcome $\hat{Y}$, if $\hat{Y}$ and A_p are independent conditional on a ground truth Y. Setting requirements on both positive and negative predictions, this enables a more rounded way of measuring fairness. However Equalising Odds requires a ground truth (Y) which may not be available in many cases and restricts the ability to deploy with real data.

4. Equal Opportunity

A very similar method to Equalizing Odds, Equal Opportunity is a measurement of true positive rates. In "Equality of Opportunity in Supervised Learning" by Hardt et al. [11], Hardt continues his research into fairness measurements in supervised learning.

$$P[\hat{Y} = 1/A_p = 0, Y = 1] - P[\hat{Y} = 1/A_p = 1, Y = 1] \leq \varepsilon \tag{2.5}$$

As fairness is a new developing field in AI, many researchers are proposing papers for publication and it can be hard to determine which papers are worth considering in detail. In saying this Moritz Hardt, has many notable well cited papers in the field of fairness measurement and is a good starting point to access the field of fairness in AI.

2.1.2 Trade-offs

When repairing a dataset or a model for fairness, there will be a trade off between fairness and accuracy.

For example, if a DI of 1 is the goal in a data repair, and DI is marginal w.r.t. the unprotected attribute A_u, the feature vector. A DI fairness repair will require a loss of information about A_u, in such a way that the decision $\hat{Y}$ satisfies the DI criteria. The question then becomes how to minimise the lost of information of A_u while maximising a fairness criteria such as DI. We will see that the trade off can be seen as a function of the repair. Optimal Transport seems to align with these goals and leads to the work done in "Obtaining fairness using optimal transport theory" [12].
An interesting point is raised in "Algorithmic Fairness" by Pessach et al. [13], Pessach states that it is not possible to satisfy multiple notions of fairness at once. In 'Algorithmic decision making and the cost of fairness' by Corbett-Davies et al. [14], "The Measure and Mismeasure of Fairness" again by Corbett-Davies [15] and 'Fairness in Criminal Justice Risk Assessments' by Berk et al. [16], they all show that satisfying multiply measures of fairness at once is not possible.

2.1.3 Fairness Enhancing Mechanisms

It is important to distinguish between fair data and a fair model, but the literature is not always clear about this distinction. Unfair data contains biases, and if a model were trained on such data without any intervention, it would become an unfair model. To certify a company as practicing fair data processing methods, fairness-enhancing mechanisms must consider what is available, such as only having access to the model that a bank has trained, versus having access to the data before the model is trained. The ability to ensure that fairness criteria are met depends on what data and model access is available.

Therefore, Fairness Enhancing Mechanisms, can broadly been seen to fit into the following categories, 1) Pre-process: Transformation of the data to ensure a fair model is learnt, 2) In-process: During training time you deploy a fairness goal as a regularizer, and 3) Post-process : You

transform the outcomes of the model, in such a way that a fairness goal is achieved.

1.Pre-Process

The aim here is to eliminate biases before the model is trained. Early attempts at solving this problem are described in "Data preprocessing techniques for classification without discrimination" by Kamiran et al. [17], which includes methods such as suppressing the protected attribute, modifying class labels, and re-weighting the data to remove unwanted dependence without relabeling.

Current methods include modifying the features in the dataset to make the distribution across a given protected attribute more similar. In "Certifying and removing disparate impact" by Feldman et al. [6], pre-processing techniques are employed to mask bias while preserving relevant information in the data

2.In-Process Mechanisms

These Mechanisms are concerned with ML algorithms at training time[[18], [19], [20]], In "Learning Fair Classifiers" by Bechavod et al. [18], they were interested in trying to learn Fair classifiers with a regularization-inspired approach. They concluded that each approach to fair learning had its pro and cons, and in their case they were able to learn a fair classifier by using regularization, but they could not provide any theoretical guarantees for other datsets.

In "Penalizing Unfairness in Binary Classification" by Bechavod again [19], proposes adding a penalty term to the loss function to directly address unfairness. The current methods show that fair classifiers can be learnt, although with the big draw back that they need to be applied during training time, and may be specific to the data set the algorithm is being trained off.

3.Post-Process Mechanisms

These mechanisms are designed to modify the output scores of a classifier and change the decisions made by the classifier, to remove dependence between protected attributes and the decision. In "Equality of Opportunity in Supervised Learning" by Hardt et al. [11], the "Equalizing Odds" method of assessing fairness is also introduced. The main outcomes of their work are as follows:

1) Choosing reliable target variables requires access to observed outcomes, which is the same requirement as supervised learning.

2) Measuring unfairness, rather than proving fairness, is the main objective. Their method provides a framework for discovering and measuring potential concerns that require further investiga-

tion, but they do not claim to provide a definite answer for proving a model is fair or unfair. They also note that solving for fairness may be impossible without domain-specific knowledge.

3) The post-processing step is applied based on when their method is most appropriate.

In "Decoupled classifiers for fair and efficient machine learning" by Dwork in [21], a decoupling method is proposed that can be applied to any black box ML algorithm to learn different classifiers for different groups. Their work requires the designer to provide a loss function that trades off accuracy for fairness. They were able to demonstrate that for certain datasets, they could reduce the loss.

AI Fairness 360 toolkit Developed by IBM [22], this is an open-source library, providing a toolbox for measuring and repairing fairness in machine learning models. This library includes a range of fairness measurements, including disparate impact and equalizing odds, which allow users to evaluate the degree of bias present in their models. Additionally, the library offers a set of fairness enhancing techniques that can be applied to mitigate the effects of any bias identified in the model. By providing these tools, the library helps to ensure that machine learning models are fair and unbiased, promoting equity and inclusivity in the use of artificial intelligence.

2.2 Optimal Transport for Domain Adaption

As transfer learning has been shown to be a solution to problems of data bias, as part of the work in exploring fairness in AI, it was also necessary to research state of the art thinking around transfer learning. This required research into aspects such as domain adaptation and optimal transport.

For this reasoning, an OT based fairness repair was chosen as the benchmark to compare our approach against.

2.2.1 Domain Adaption

Today, Domain Adaption is a one of the most challenging tasks of machine learning and data analytics, as we seek to deploy models learnt in label rich environments to environments in which no or minimal labels are accessible. In "Domain adaptation problems: a DASVM classification technique and a circular validation strategy" by , Bruzzon et al. [23]. The authors provide several examples of domain adaptation problems, including: 1) Object recognition: A machine learning model trained to recognize objects in one environment may perform poorly when tested in a differ-

ent environment due to differences in lighting, background, or other factors. In "Learning to See in the Dark" by Chen et al. [24], Chen tackles the issue of a model trained with the images with the domain being the day, then testing this model in the dark to see the reduction in performance.

2)Text classification: A model trained on a specific type of text (e.g. news articles) may perform poorly when applied to a different type of text (e.g. social media posts) due to differences in language use and writing style. An example of this problem can be found in "Cross-Domain Sentiment Classification Using Sentiment Sensitive Embeddings" by Ghosh et al. [25].

3) Medical diagnosis: A model trained on medical data from one hospital or patient population may perform poorly when applied to a different hospital or patient population due to differences in demographics, disease prevalence, or treatment protocols. For example, "Deep Learning for Patient-Specific Kidney Graft Survival Analysis" by Esteban et al [26].

2.2.2 Optimal Transport for Domain Adaption

The theory was formalized by the French mathematician Gaspard Monge in 1781 [27]. Optimal Transport (OT) seeks to find a transport plan, that maps each point from the source distribution to the target distribution, with the lowest possible cost.

The field of Optimal Transport (OT) has gained significant attention in recent years due to its ability to compute distances between probability distributions. The distances, known by various names such as Wasserstein, Monge-Kantorovich, or Earth Mover distances, possess important properties that make them valuable for many applications. For instance, these distances can be evaluated directly on empirical estimates of the distributions without the need for non-parametric or semi-parametric approaches to smooth them. Additionally, they provide meaningful distances, even when the supports of the distributions do not overlap, thanks to the underlying metric space geometry.

Leveraging these properties, Courty et al. [28] introduced a novel framework for unsupervised domain adaptation, that involves learning optimal transportation based on empirical observations. The proposed approach includes several regularization terms that promote better transformation learning concerning the adaptation problem. These regularization terms can encode class information contained in the source domain' or promote the preservation of neighborhood structures.

To solve the resulting regularized optimal transport optimization problem, an efficient algorithm is proposed. The proposed framework can be easily extended to the semi-supervised case, where few labels are available in the target domain, by a simple and elegant modification in the optimal transport optimization problem.

Pip POT POT : Python Optimal Transport, is an open source Python library providing several solvers for optimization problems related to Optimal Transport. This tool box was created and is maintained by Remi Flamary

2.2.3 Obtaining fairness using optimal transport

The notation of seeing fairness as a domain adaption problem is becoming increasing popular, [29], [30] and [12]. The Observed data $\mathbf{X}$, which resides in the source domain, is seen to be unfair, hence we can also see this as the unfair Domain. The target domain being a fair representation of the source domain. In "Obtaining fairness using optimal transport theory" by Gordaliza et al [12], the authors propose a novel approach for fairness in binary classification problems by utilizing the optimal transport problem and the Wasserstein distance. They deal with the problem illustrated in Figure 2.1, where bias exists in the Outcome dependent on the protected class A_p.

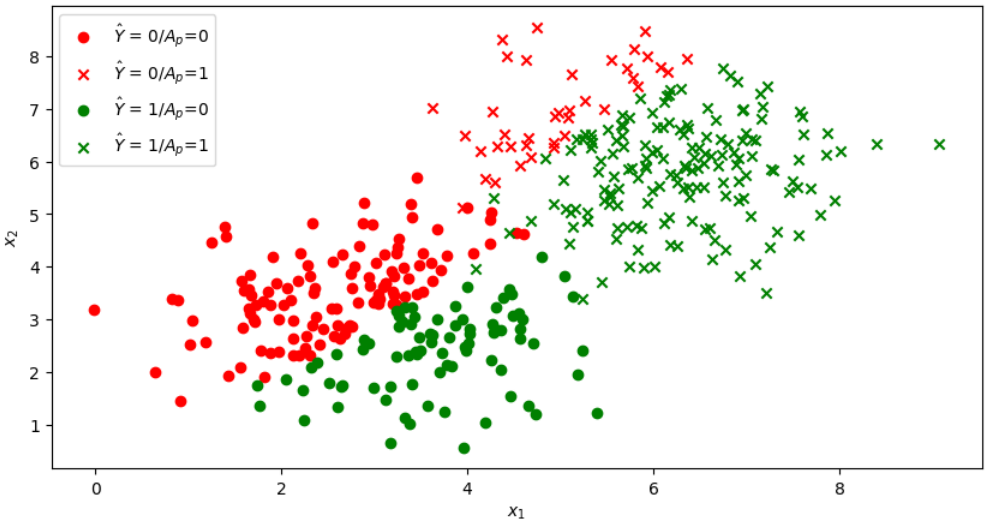


Figure 2.1: Observed Data $\mathbf{X}$, Unfair Domain

The authors first out lined the problem as Figure 2.2, where f_0 and f_1 represent the distributions of $\mathbf{X}|S = 0$ and $\mathbf{X}|S = 1$, and where $_f$ is the target distribution. Then they formulate the following two problems:

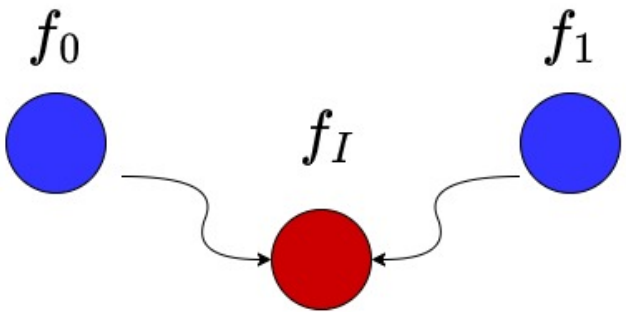


Figure 2.2: General repairing scheme

1) Choose a distribution f_I which can be as similar as possible to both distributions f_0 and f_1.
2) Having chosen f_I, next step is to find the optimal way of transporting f_S to f_I.

The Authors propose that a Wasserstein baycenter f_B, between f_0 and f_1, with weight of π_0 and π_1, should represent this target distribution f_I. They describe f_B as equation 2.6, where W is the Wasserstein distance or Earthmovers distance.

$$f_B \in= \text{argmin}_{f_I \in \mathcal{P}_2} \left\{ \pi_0 W_2^2 \left(f_0, f_I \right) + \pi_1 W_2^2 \left(f_1, f_I \right) \right\} (2.6)$$

Partial repair As mentioned previously, there exists an accuracy-fairness trade off with these transformations. For this reason Gordaliza et al. implement a partial repair in which the amount of repair desired can be achieved by weighting. $\lambda \in [0,1]$ is the parameter representing the amount of repair desired for $\mathbf{X}$, where $\lambda = 1$ is 'total repair' and $\lambda = 0$ is 'no repair'.

In Figure 2.3,λ can be seen to act as a slider for the amount of repair.

2.3 Fully Probabilistic Design

Fully probabilistic design (FPD) was a core for the inspiration of our alternative measurement of fairness. Therefore, it was a requirement of the research to understand and research the topic. FPD combats the limitation of traditional deterministic design models which are ignorant to uncertainty, achieving this goal by considering all inputs to the system as probability distributions .

In "Fully probabilistic design of hierarchical Bayesian models" by Quinn et al. [31], the use of fully probabilistic design (FPD) principles in hierarchical Bayesian models, is introduced. By min-

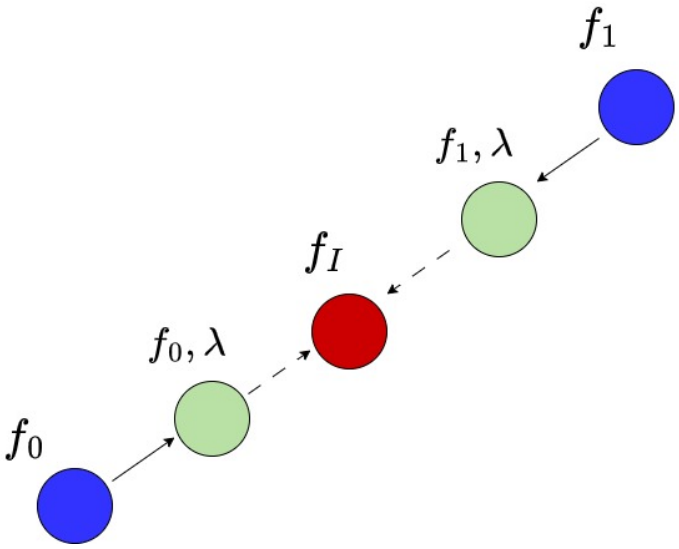


Figure 2.3: Original distributions (blue) and their partially repaired versions (green) towards the barycenter (red)

imizing the Kullback-Leibler divergence between the designer's ideal distribution and the chosen distribution, FPD enables optimal design of a stochastic model for the unknown distribution. This approach extends the applicability of FPD principles by allowing processing of non-linear functional constraints in the constructed distribution. The paper unifies currently available FPD procedures for merging external knowledge, approximate learning, stabilized forgetting, decision strategy design, and local adaptive control design within the hierarchical FPD framework.

Therefore in FPD, Kullback leiber divergence (KLD) is a go to measurement for model comparison. As seen [31] and [32], KLD is two argument measurement, the second argument seen as the zero-loss state, the theoretical state in which the system operates perfectly. The first argument of the KLD would be a current state of the design model given its operating conditions. The KLD then acts as a guide for adjusting the first argument to minimize the divergence between ideal state and the designed state.

Consider f_I as an ideal distribution and $f \in \mathbf{F}$ to be the set of candidate distributions. Then KLD acts as a similarity measure which is to be minimized between the set of $\mathbf{F}$ and f_I. The FPD is a generalization of the cross entropy loss measure which is used in many fields including loss functions in Stochastic gradient descent(SDG) algorithms which are the core of most modern Neural Networks.

This measure is given in equation 2.7. E_F refers to the expectation w.r.t the distribution f.

$$f^{\circ} = \arg\min_{f \in \mathbf{F}} D_{KL}(f \| f_I) = \arg\min_{f \in \mathbf{F}} E_f \left[\ln \frac{f}{f_I} \right] \tag{2.7}$$

2.3.1 Fairness in ML achieved with Fully Probabilistic Design

In recent years, there has been a growing interest in using probabilistic approaches to address fairness in machine learning. Although limted work has been published, noteworthy papers include "Fairness in Machine Learning with Tractable Models" by Varleya et all [33]. The focus of this paper is taking steps towards the application of tractable probabilistic models to fairness in machine learning. This aligns with FPD as they are considering probabilistic approach's to the fairness problem, though a true FPD approach, in which the goal to minimise the KLD from a ideal data distribution to a candidate distribution, is yet to be researched.

2.4 Kullback–Leibler divergence

In the search for an alternative measurement of fairness, we wanted to take advantage of the well established divergence measure of KLD, and so it was necessary to become more familiar with the properties and limitations of KLD measurements.

The Kullback–Leibler divergence, first proposed by S. Kullback and R. A. Leibler in 'On Information and Sufficiency' [34], is the measure of how one probability distribution P differs from a reference probability distribution Q. The KLD is denoted by $D_{KL}[P||Q]$. The formula for calculated the KLD is

$$D_{KL}[P||Q] = \sum P(x) log(P(x)/Q(X)) \tag{2.8}$$

where **X** is the sample space, P(x) is the probability of observing outcome x, under the approximate distribution P, and Q(x) is the probability of observing outcome x, under the true distribution Q.

In the case of continuous measurements, the summation in the above equation is replaced with an integral over the sample space:

$$D_{KL}[P||Q] = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx \tag{2.9}$$

where p(x) is the probability density function (PDF) of P, and q(x) is the PDF of Q.

Properties of KLD

1) Self similarity: The KLD is a measure of the divergence of two probability distributions, but it is not symmetric. In other words, $D_{KL}[P||Q]$ may not be equal to $D_{KL}[Q||P]$. However, the KLD is always non-negative and can be used to compare two distributions in either direction. Moreover, if $D_{KL}[P||Q] = 0$, then P and Q are identical.
The question for this research work is then which away around to take a KLD measurement.

2) Self identification : The KLD satisfies the property of self-identification, which means that $D_{KL}[P||Q]$ is equal to zero if and only if P and Q are the same distribution. This property makes the KLD a useful tool for model selection and hypothesis testing

3) Positivity : The KLD is always non-negative, which means that $D_{KL}(PQ) \geq 0$ for any probability distributions P and Q. This property is a consequence of the fact that the logarithm function is concave, and it implies that the KLD is a measure of how much more information is required to encode samples from P, using a code designed for Q.

2.4.1 Limitations of KLD

As we want to use KLD measurements we must know its limitations.

1) KLD is not symmetric: As mentioned earlier, KLD is not a symmetric measure, which means that $D_{KL}(P||Q)$ is not necessarily equal to $D_{KL}(Q||P)$. This is an issue as we do not know the correct way around of the arguments in our fairness measurements, though we suspect that the fair distrubution is the second argument. As in fully probilistic design (FPD), the second argument is the ideal, which in our case would be the fair case.

2) KLD is not a true distance metric: While KLD is often referred to as a "distance" or "divergence" measure, it does not satisfy all the properties of a true distance metric. For example, it violates the triangle inequality property, meaning that $D_{KL}[P||R]$ can be greater than the sum of $D_{KL}[P||Q]$ and $D_{KL}[Q||R$. This brings with it the issue of "How big is big" with KLD.

3) KLD is sensitive to outliers: KLD can be sensitive to outliers, meaning that a few

rare events, with very low probability, can have a disproportionate impact on the divergence measure.

4) KLD requires knowledge of the true distribution: KLD requires knowledge of the true distribution P, which may not always be available or known. In practice, it is often necessary to estimate P based on observed data, which can introduce additional uncertainty into the calculation. In a fairness context we would have to know the fair distribution. A true fair distribution is an ideal and doesn't exists so we must estimate it, which raises issues such as what is the ideal fair distribution.

2.4.2 The KLD from one Gaussian Mixture Model to another Gaussian Mixture Model

KLD is a widely used statistical method for assessing divergences in probability distributions, therefore we would like to investigate its use in evaluating divergences in GMM's, so that we can then apply it in a fairness context.

Unfortunately there exists no closed form solution for the KLD from one Gaussian Mixture Model to another Gaussian Mixture Model, though there are analytical approximations. The most accurate approximation method according to the literature [35] is by Monte Carlo simulations.

A Gaussian mixture model is given as:

$$f_a(x) = \sum_a \pi_a \mathcal{N}(\boldsymbol{\mu}_a; \boldsymbol{\Sigma}_a) \tag{2.10}$$

Where π_a is the prior probability of each state.

Therefore considering two GMM's f(x) and g(x), then to estimate D_{KL}[f||g], we draw a sample x_i from the pdf f(x) such that $E_f[\log f(x_i)/g(x_i)]$ = D(f||g). Using n i.i.d. samples $\{x_i\}_{i=1}^n$ we have the following equations

$$D_{\mathrm{MC}}(f\|g) = \frac{1}{n}\sum_{i=1}^{n} \log f(x_i)/g(x_i) \to D(f\|g) \tag{2.11}$$

as $n \to \infty$. This is the only method in [35] that yields a convergent method. To note as well, this approximation method does satisfy the similarity property, but not the positively property 2.4.

2.4.3 Jensen-Shannon Divergence

Researching possible ways in to overcome the limitations of the KLD, leads to alternative methods such as Jensen-Shannon Divergence.

The Jensen–Shannon divergence (JSD) is a symmetrized and smoothed version of the Kullback–Leibler divergence. It is given by $D_{JS}[P\|Q]$. It was first seen in [36]. It is defined by

$$D_{JS}[P\|Q] = \frac{1}{2}D_{KL}[P\|M] + \frac{1}{2}D_{KD}[Q\|M] \tag{2.12}$$

where $M = \frac{1}{2}(P + Q)$. The JSD is a possible solution to the limitation identified in 2.4.1 3). JSD overcomes this issue by using a smoothed version KLD avoids zero values. Specifically, JSD calculates the KLD divergence as the average distribution M = (P+Q)/2 and each of P and Q, which are normalized by their respective KLDs to M. This ensures that there are no zero values in the denominator of the logarithm function used in the KLD calculation, as follows:

$$JSD(P, Q) = \frac{1}{2}D_{KL}(P||M) + \frac{1}{2}D_{KL}(Q||M) \tag{2.13}$$

$$= \frac{1}{2}\sum_i P(i)\log_2\left(\frac{P(i)}{M(i)}\right) + \frac{1}{2}\sum_i Q(i)\log_2\left(\frac{Q(i)}{M(i)}\right) \tag{2.14}$$

$$= \frac{1}{2}\sum_i P(i)\log_2\left(\frac{2P(i)}{P(i)+Q(i)}\right) + \frac{1}{2}\sum_i Q(i)\log_2\left(\frac{2Q(i)}{P(i)+Q(i)}\right) \tag{2.15}$$

$$= \frac{1}{2}\sum_i \left[P(i)\log_2\left(\frac{2P(i)}{P(i)+Q(i)}\right) + Q(i)\log_2\left(\frac{2Q(i)}{P(i)+Q(i)}\right)\right] \tag{2.16}$$

Note that the denominator of equation 2.16 in the logarithm function is P(i)+Q(i), which is always greater than zero, since both P and Q are probability distributions, and therefore sum to 1. Therefore, there are no zero values in the denominator, and the "divide by zero" issue is avoided.

In summary, JSD avoids the "divide by zero" issue by using a smoothed version of P and Q that ensures there are no zero values in the denominator of the logarithm function used in the KLD calculation.

Another strength is the JSD is that it is bounded between 0-1, which makes its values easier to

interrupt.

Conclusion : Jensen-Shannon Divergence The Kullback-Leibler Divergence (KLD) and the Jensen-Shannon Divergence (JSD) are both measures of the difference between two probability distributions. However, there are some properties of KLD that are not present in JSD, which means that KLD is often used in fully probabilistic design, while JSD may not be.

1. Non-symmetry: KLD is not a symmetric measure. In contrast, JSD is a symmetric measure. Non-symmetry is often desirable in fully probabilistic design because it allows us to measure the difference between a reference distribution and a proposed distribution.
2. Lack of Triangle Inequality: The KLD does not satisfy the triangle inequality. This property can make KLD difficult to use in some optimization algorithms. JSD, on the other hand, satisfies the triangle inequality.
3. Unboundedness: KLD is an unbounded measure, meaning that it can take on any non-negative value. This property can be both a strength and a weakness, depending on the application. In contrast, JSD is bounded between 0 and 1

These properties of KLD make it a useful measure for fully probabilistic design, where non-symmetry, lack of triangle inequality, and unboundedness may be desirable. However, JSD is a more widely used divergence measure in many applications because of its symmetric and bounded properties. However, as KLD is the measurement adopted in FPD, this dissertation focus on KLD based Measurements.

2.5 Datasets

Real data was vital in this project as it provided motivation for our work and a way in which we could test both the State-of-the-art methods of data repair and our method of Data repair.

2.5.1 COMPAS Dataset

A frequently used data set used in ML fairness assessment, is the COMPAS data set collected by Probilica [37], which is open source. Propublica has collected features (both A_u and A_p) about people and the ML decisions ($\hat{Y}$) made for those people. These decisions are the result of a classification tool built by northpointe, designed to predict the likelihood of re-offence .

This dataset provided our first look into real data and this helped provide examples as to how to apply fairness measures and fairness repairs. The real data also helped in understanding the limitations of the state-of-the-art tools in fairness measurements and also to show that for different A_p, A_u will be distributed differently. For this example we will look at $A_p \in$ African American, Caucasian. For A_u we will look at three variables, age, number of priors, and number of Juvenile Misdemeanors. For $\hat{Y}$, we will take the Decile score $\in 0,\ldots,10$, This classification is created by northpointe, where 1 represents the lowest and 10 the highest chance of recidivism.

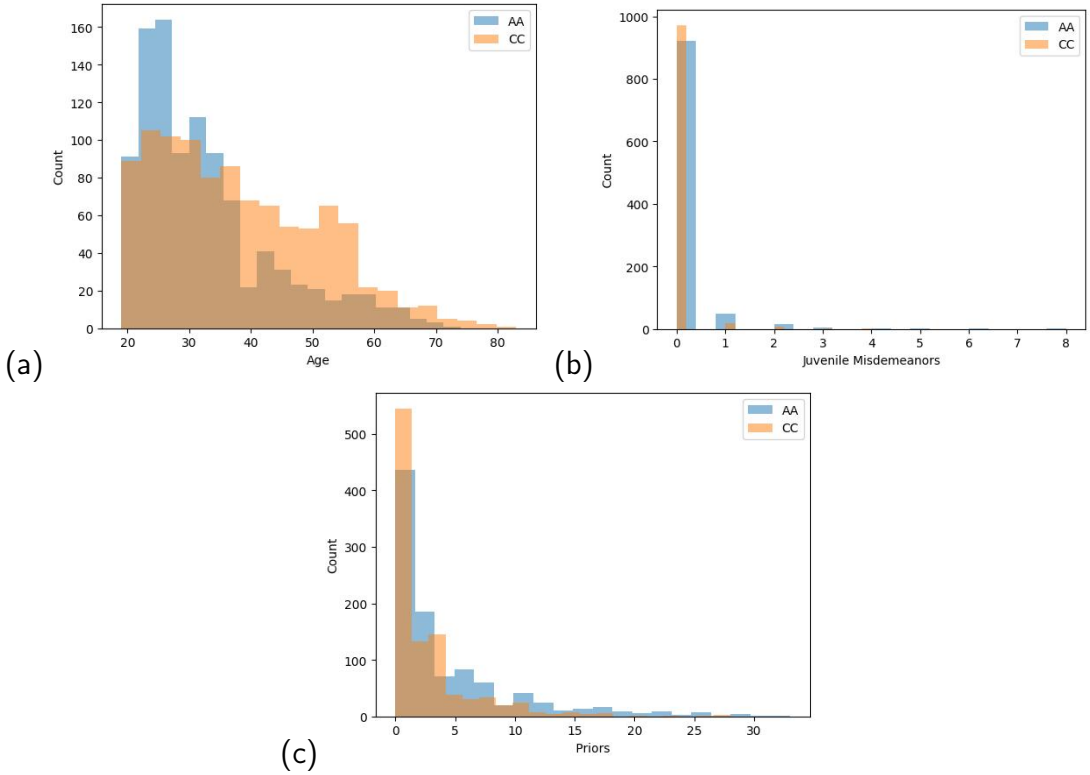


Figure 2.4: (a) A_{u_0}, Age (b) A_{u_1}, No. of Juveniles misdemeanor (c) A_{u_2}, No. of Priors

In Figure 2.4, There are three histograms of different Attributes given the protected Attributes. In the histograms, AA is to represent African American and CC is to represent Caucasian. From these three different plots of $\Pr[A_{u_n}/A_p]$ where n $\in \{0,1,2\}$, we can see that underlying distributions of $\Pr[A_u/A_p]$ vary both in , 1) $\Pr[A_u/A_p = 0]$ and $\Pr[A_u/A_p = 1]$, 2) for $\Pr[A_{u_1}/A_p]$ and $\Pr[A_{u_2}/A_p]$. This means that for 1) A_p is dependent on A_u, This may be due to social issues such as existing bias in the judicial system in the US. This dependence exists and must be recognised in a fairness assessment on any tool that may be trained on this data.

Next we look at the classifications by the northpointe software tool.

In Figure 2.5, it can been seen that there is a difference in $\Pr[\hat{Y}/A_p = 0]$ and $\Pr[\hat{Y}/A_p = 1]$,

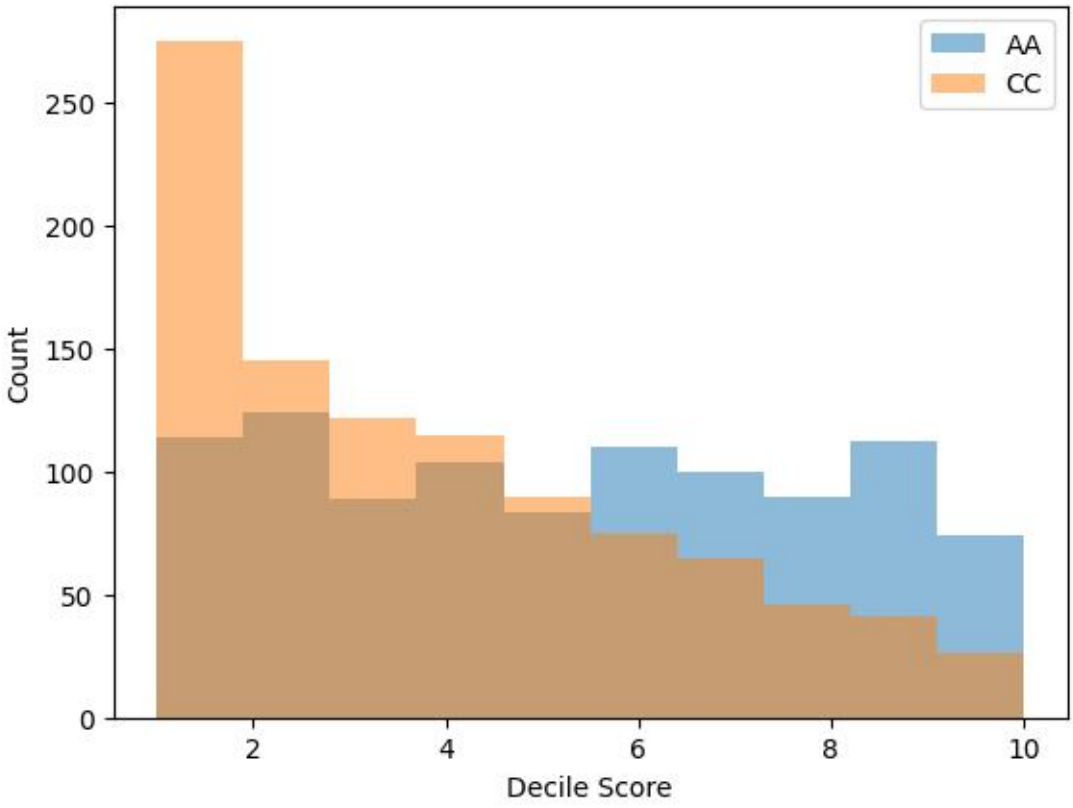


Figure 2.5: Northpointe Classier, $\hat{Y}$

which seems to indicate racial bias in the classifier. If we were to use Disparate impact (DI) for an example as a standard tool in fairness assessment. DI can be seen as equation 2.1

In DI, $\hat{Y}$ represents a positive or negative outcome; A_p though mostly in literature referred to as S, represents a privileged group and non privileged being permitted. Achieving a DI of 1 may be an unreasonable task.

In this case a low decile score would be a positive outcome ($\hat{Y} = 1$). So, in this example AA is non privileged ($A_p = 0$) and CC is privileged ($A_p = 1$). In calculating DI, we have DI = 0.49. This would suggest that the classifier is unfair, though DI does not speak to these underlying differences in distributions seen in Figure 2.4. Therefore, this provides motivation to build an alternative tool to help evaluate fairness, while acknowledging these underlying differences in $\Pr[A_u/A_p = 0]$ and $\Pr[A_u/A_p = 1]$.

2.5.2 Adult Dataset

The Adult Dataset, used in [12] also known as the Adult Income Dataset or the Census Income Dataset, is a popular dataset used in machine learning and data mining for binary classification tasks. It consists of approximately 32,000 instances, each representing a person's demographic and socioeconomic attributes, such as age, education level, occupation, and marital status. The task is to predict whether an individual's annual income is greater or less than $50,000 based on their attributes.

The Adult Dataset was created from the 1994 United States Census data and was preprocessed to remove missing values and encode categorical attributes as numeric values. It has been widely used in research on fairness, explainability, and privacy in machine learning, as it raises several important ethical and social issues related to discrimination, bias, and privacy.

The Adult Dataset has become a standard benchmark dataset for evaluating the performance of machine learning algorithms in binary classification tasks. It has been used in various studies on supervised learning, including logistic regression, decision trees, support vector machines, and neural networks, as well as studies on fairness, explainability, and privacy-enhancing technologies.

2.5.3 Fairness without access to protected attributes

It is often the case that the user or customer, request that decision maker i.e. the bank, has no access to the protected attribute. It does not seem unreasonable for people not to wish to disclose their gender or martial status when requesting a loan. Though unfortunately with out access to A_p, fairness tasks prove very difficult. For example, Fairness through unawareness is an approach which assumes that if we are unaware of protected attributes while making decisions, our decisions will be fair. However, this approach has been shown not to be effective in many cases, because protected variables could be correlated with other variables in the data. Therefore, we must have access to A_p to be able to quantify notions of fairness. From this we get the issue above, that we don't want banks to know are protected attributes. There exist methods that can deal with a lack of A_p in the datasets, some of them include 1) Variational auto encoder and 2) Weak proxies.

1) Training a variational autoencoder (vae) One approach to achieving fairness without using protected attributes is through the use of a causal variational autoencoder [38]. This

approach is based on the idea that protected information may not always be available in real-world scenarios, and thus seeks to mitigate against biases by assuming that the protected attribute is unobserved.

The approach involves training a variational autoencoder (vae) with a causal structure that models the underlying causal relationships between the observed features and the unobserved protected attribute. The vae is then used to generate a fair representation of the data by encoding the observed features into a latent representation and decoding it back into the original space.

2) Weak proxies The paper "Weak proxies are sufficient and preferable for fairness with missing protected attributes " [39] presents an approach to achieving fairness in decision-making when protected attributes are missing from the data. The authors argue that weak proxies, which are imperfectly correlated with the protected attribute, can still provide a sufficient and sometimes preferable basis for achieving fairness goals. They propose a method for learning fair decision rules based on weak proxies.

Some examples of weak proxies given in the paper[2] are zip code: zip code is often used as a weak proxy for race or income, as certain zip codes may be associated with particular racial or socioeconomic groups. Occupation: occupation can be used as a weak proxy for gender or race, as certain occupations may be more heavily represented by one group over another. Education level: education level can be used as a weak proxy for income, as higher levels of education are often associated with higher earning potential.

Conclusion on Requirement of A_p

Several methods have been proposed to address the issue of missing protected attributes in datasets. However, it is challenging to evaluate fairness using these substitute methods compared to scenarios where the actual protected attributes are available. It is yet to be determined whether biased algorithms can be accurately identified using these weak proxy methods. While these methods could be a viable substitute for missing protected attributes, our research only involved datasets where we had access to the actual protected attributes.

2.6 Literature Review Conclusion

The literature review provided insight into established methods for fairness measurement and mitigation, particularly in the context of optimal transport-based fairness repair. Disparate Impact

(DI) equation 2.1 was chosen as the fairness measurement for comparative purposes, as it is a widely-used and stable tool. The work of Gordaliza et al. in their study [12] was chosen as the benchmark method for comparing data repairs. The review of real-world data, such as in the COMPAS dataset, also suggested the need for an alternative fairness measure that could lead to a new fairness requirement.

3 Measuring and Repairing for Fairness in a Simulated Environment

3.1 Introduction

Having conducted a literature review to become familiar with the state of the art methods, the next step was the development of a novel method of assessing and repairing for fairness. The intention was to build a method to repair for fairness with a Conditional Independence (CI) fairness goal. But first it is necessary to explain briefly the concept of CI.

3.2 Fairness as Conditional Independence

Consider the three following random variables, A_p The Protected Attribute, A_u the Unprotected Attribute and $\hat{Y}$, the output of a classifier, where A_p,A_p and $Y \in \{1,0\}$. Now consider the dependence pathways in Figure 3.1. In the Unfair case there is a direct dependence of A_p on Y.

For example, take $A_p \in$ {male, non-male} and $A_u \in$ {tall, non-tall}, A_p could be predicted given only A_u, in this case, as there are underlying differences in distribution for A_u given A_p. In some cases, these underlying differences may be due to social conditions. In such cases, achieving a more fair outcome may require a social engineering rather than a computer engineering solution.

Though the dependence of A_p on A_u is expected and accepted, it can still be stated that any dependence of $\hat{Y}$ on A_p should come only from A_u. This would represent the fair case in Figure 3.1, i.e. that all the dependence of $\hat{Y}$ on A_p can be explained by A_u.

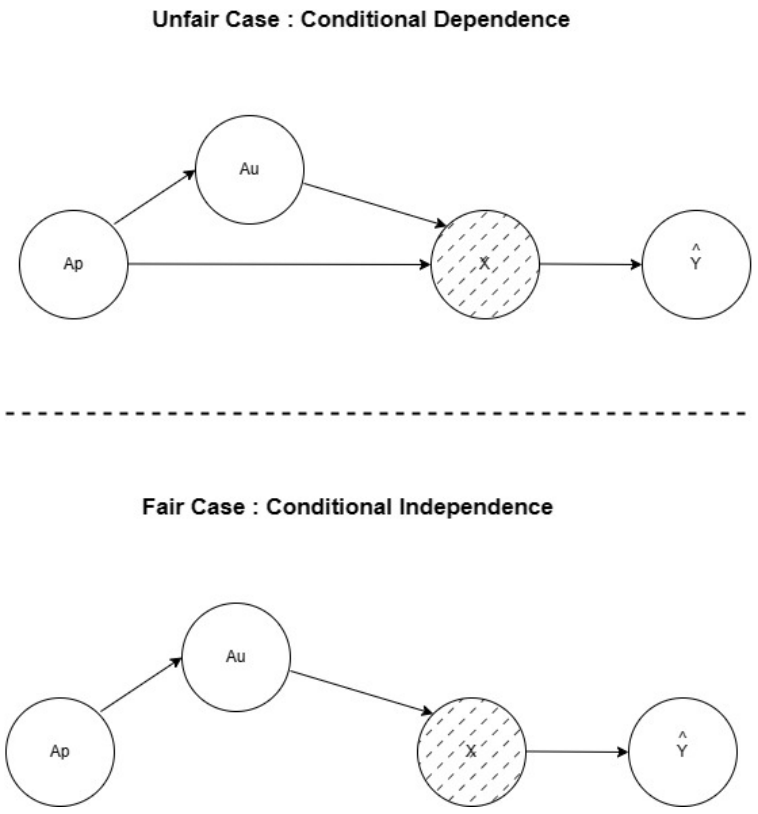


Figure 3.1: Probability graph for CI fairness and CD Unfairness

Fairness in terms of CI is expressed mathematically, using the equations 3.1 and 3.2. Equation 3.1, defines the unfair case, where not all dependence of A_p on $\hat{Y}$ can be explained by $\Pr[A_u/A_p]$. In equation 3.2, defines the case of Conditional Independence.

$$\Pr\left[\hat{Y}, A_u, A_p \mid X, \bar{F}\right] = \Pr\left[\hat{Y} \mid A_u, A_p, X, \bar{F}\right] \times \Pr\left[A_u \mid A_p, X, \bar{F}\right] \times \mathrm{P}\left[A_p \mid X, \bar{F}\right] \tag{3.1}$$

$$\Pr\left[\hat{Y}, A_u, A_p \mid X, F\right] = \Pr\left[\hat{Y} \mid A_u, X, F\right] \times \Pr\left[A_u \mid A_p, X, F\right] \times \mathrm{P}\left[A_p \mid X, F\right] \tag{3.2}$$

The equations 3.1 and 3.2 can be derived from the chain rule of probability, which states that the joint probability of a set of random variables can be decomposed as the product of the conditional probability of each variable given its predecessors. In the case of equation 3.1, the joint probability of $\hat{Y}$, A_u, and A_p given X and $\bar{F}$ can be decomposed as the product of the conditional probability of $\hat{Y}$ given A_u, A_p, X, and $\bar{F}$, the conditional probability of A_u given A_p, X, and $\bar{F}$, and the marginal probability of A_p given X and $\bar{F}$.

Similarly, in equation 3.2, the joint probability of $\hat{Y}$, A_u, and A_p given X and F, can be decomposed as the product of the conditional probability of $\hat{Y}$ given A_u, X, and F, the conditional probability of A_u given A_p, X, and F, and the marginal probability of A_p given X and F.

These equations formalize the concepts of conditional independence and dependence between the protected attribute A_p and the outcome variable $\hat{Y}$. In equation 3.2, the conditional probability of Y given A_u, X, and F represents the fairness criterion, since it implies that $\hat{Y}$ is independent of A_p given A_u, X, and F. In contrast, in equation 3.1, the dependence of A_p on $\hat{Y}$ cannot be fully explained by the conditional probability of A_u given A_p, X, and $\bar{F}$.

3.2.1 Fairness with Fully Probabilistic Design

Determining the KLD from one probability distribution to another does not speak to fairness directly. So, using the Fully Probabilistic Design approach, a method was established to use KLD to speak to fairness. In FPD, the target is to minimize KLD from a candidate distribution to it's ideal distribution. In this dissertation, this is the starting point for the use of KLD in ML fairness assessment, and so begins the process of creating another measurement for fairness.

But first it is necessary to define the following measurements :

1) **Cost-KLD** Using an FPD approach, f_F is considered to be the fair or ideal distribution and f_{UF} as the candidate or unfair distribution.

Therefore, the divergence from a unfair distribution to its 'ideal' or fair counterpart can be measured. This can also be seen as the cost of transforming f_{UF} to f_F and is given by equation 3.3,

$$D_{KL}[f_F||f_{UF}] \tag{3.3}$$

2) **Fair-KLD** This measure is used to quantify fairness after the decision is made i.e. a post classification decision ($\hat{Y}$). Using a FPD approach again, two post classification scenarios are considered, i) where the classifier is trained and tested on unfair data ii) the classifier is trained and tested on fair data. Two classification matrices are created using the results of the testing stage. The KLD between these two matrices represents the FDP measure for assessment of Fairness, given by:

2.1) **KLD not marginal w.r.t.** A_u

$$D_{KL}[M_{UF}||M_F] \tag{3.4}$$

2.2) **KLD not marginal w.r.t.** A_u

$$\Sigma_1^n D_{KL}[M_{UFi}||M_{Fi}] \tag{3.5}$$

where n is the total labels of A_u.

Building the arguments for the equations 3.4 and 3.5 Where M_{UF} and M_F are the result of two classification models. G_{UF} and G_F .
1) G_{UF}: $X_{UF} \rightarrow \hat{Y}_{UF}$
2) G_F: $X_F \rightarrow \hat{Y}_F$

Note: The model (G_{UF}) were not trained with the label A_p, this label was withheld and used to populate M_UF.

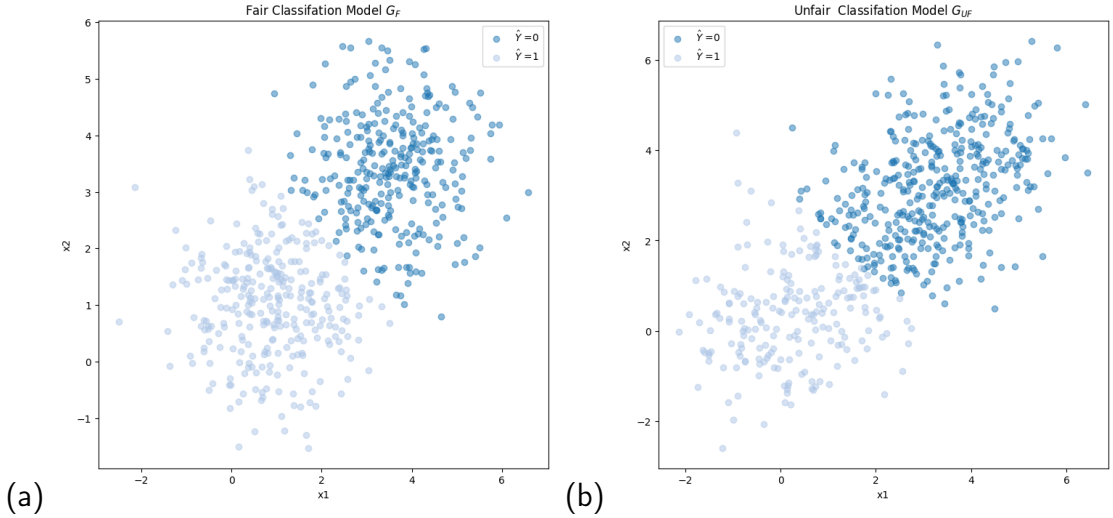


Figure 3.2: The result of G_{UF}: $X_{UF} \rightarrow \hat{Y}_{UF}$ (a)The result of G_F: $X_F \rightarrow \hat{Y}_F$(b)

1) For the unfair data, we collect the results into the following matrix,

	$(A_u, A_p) = (0,0)$	$(A_u, A_p) = (0,1)$	$(A_u, A_p) = (1,0)$	$(A_u, A_p) = (1,1)$
$\hat{Y}_{UF} = 1$	40	10	5	2
$\hat{Y}_{UF} = 0$	10	1	20	20

Then normalise the columns so they sum to 1.

	$(A_u, A_p) = (0,0)$	$(A_u, A_p) = (0,1)$	$(A_u, A_p) = (1,0)$	$(A_u, A_p) = (1,1)$
$\hat{Y}_{UF} = 1$	0.8	0.9	0.2	0.1
$\hat{Y}_{UF} = 0$	0.2	0.1	0.8	0.9

This matrix is to be called M_{UF} . Then the final step is to multiply M_{UF} by the probabilities of each state π_{UF}. $M_{UF} = M_{UF}.\pi_{UF}$

2) For the unfair data M_U is a 2 x 2 matrix, as there is no extra dependence of A_p

	$(A_u) = (0)$	$(A_u) = (1)$
$\hat{Y}_F = 1$	0.8	0.1
$\hat{Y}_F = 0$	0.2	0.9

A KLD argument requires the matrix's to be the same size, therefor to represent a fair 2x4 Matrix, the Matrix is dupliacted.

	$(A_u, A_p) = (0,0)$	$(A_u, A_p) = (0,1)$	$(A_u, A_p) = (1,0)$	$(A_u, A_p) = (1,1)$
$\hat{Y}_F = 1$	0.8	0.8	0.1	0.1
$\hat{Y}_F = 0$	0.2	0.2	0.9	0.9

Then as with M_UF, M_U is multiplied by the probabilities of each state π_U. $M_U = M_U.\pi_U$

Therefore, M_{UF} and M_F become the two arguments for the equations 3.4 and 3.5

3.2.2 Learning about FPD through Simulation work

Before setting to work with these two measurements, it is necessary to test the KLD's ability to measure the success of a ML task. The task chosen was clustering. The clustering function C: X $\rightarrow$ 1, 2, ..., k maps x $\in$ **X** to 1 of k clusters, where **X** $\in \mathbf{R}^{nxm}$, n is the number of rows, m the number of columns, and 1, 2, ..., k represents the set of possible cluster labels (in this case, k is a fixed positive integer).

The clustering method chosen was Expectation-maximization (E-M) clustering. The simulations were implemented in Python using Scikit-learn's Gaussian Mixture [40].

The goal of EM clustering is to find the optimal parameters of the Gaussian Mixture Model that maximizes the likelihood of the observed data. The likelihood function is given by equation

3.6:

$$\mathcal{L}(\theta|X) = \prod_{i=1}^{n} \sum_{j=1}^{k} \pi_j \mathcal{N}(x_i|\mu_j, \Sigma_j) \tag{3.6}$$

where $\theta = \{\pi_1, \dots, \pi_k, \mu_1, \dots, \mu_k, \Sigma_1, \dots, \Sigma_k\}$ are the parameters of the Gaussian Mixture Model, $X = \{x_1, \dots, x_n\}$ is the set of observed data points, π_j is the mixing coefficient for component j, μ_j is the mean of component j, Σ_j is the covariance matrix of component j, and $\mathcal{N}(x|\mu, \Sigma)$ is the Gaussian Probability Density Function with mean μ and covariance Σ. The EM algorithm iteratively maximizes the log-likelihood function $\log \mathcal{L}(\theta|X)$ with respect to the parameters θ.

KLD estimations with Monte Carlo Simulations

Having selected the clustering method the next step is to measure the divergence (KLD) between synthetic and learnt model.

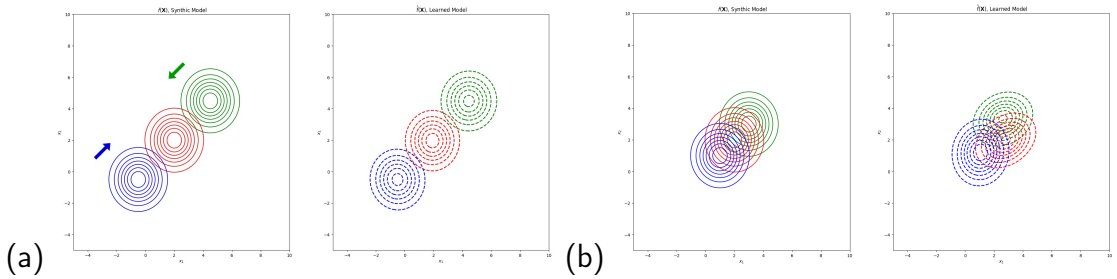
The Kullback-Leibler divergence (KLD) will be measured from the synthesis model f(X) to a learnt/estimated model $f(\hat{X})(\mathbf{X})$ while varying the operating conditions of the synthesis model. The synthesis model is a Gaussian Mixture Model (GMM) represented by $\boldsymbol{\mu} = [\mu_1, \mu_2, \dots, d]$ for the means, $\boldsymbol{\Sigma} \in \mathbf{R}^{dxd}$ for the covariance matrices, $\boldsymbol{\pi} = [\pi_1, \pi_2, \dots, k]$ for the weighting of the components, and N for the number of sampled points. Calculating the KLD divergence of two GMMs is not possible using a closed-form solution, so it will be estimated empirically via Monte Carlo simulations, as described in the algorithms section. Our objective is to calculate $D_{KL}[\hat{f}(x)||f(x)]$ under different operating conditions.

JUMP - SUGGEST SUMMARY TABLE/DIAGRAM OF WORK?

Inter-mean distance For the first simulation, for which the output is given in Figure 3.3, the operating condition of choice was a inter-mean distance between two outer components (Blue and Green) and the inner component (Red). See equation 3.7. Note: All operating conditions, other than the mean, remained constant in the simulations.

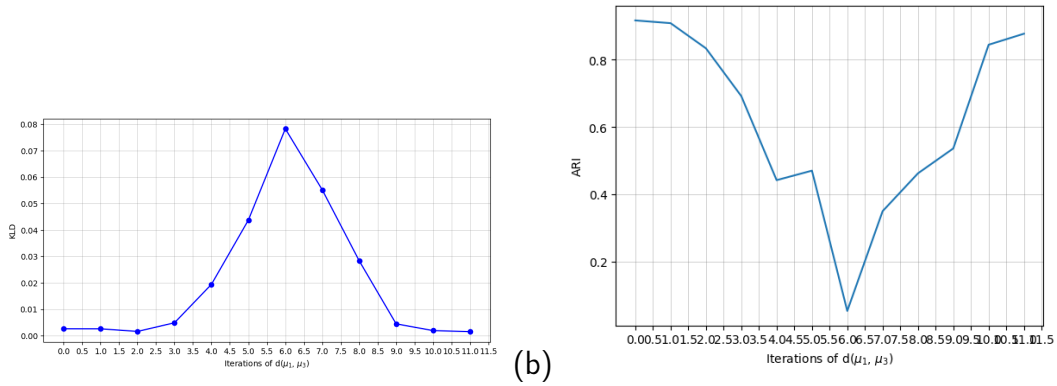
$$d(\mu_1, \mu_3), = \sqrt{(\mu_{1_0} - \mu_{3_0})^2 + (\mu_{1_1} - \mu_{3_1})^2} \tag{3.7}$$

Using Monte Carlo simulations to calculate long run averages, an empirical estimate of $D_{KL}[\hat{f}(x)||f(x)]$ was obtained, for each of the different operating conditions.



(a) (b)

Figure 3.3: Increasing $d(\mu_1, \mu_3)$ as (a)→(b)



(a) (b)

Figure 3.4: $D_{KL}[\hat{f}(X)||f(X)]$ over Inter-mean Distance (a) ARI over Inter-mean Distance (b)

Results: Inter-Mean In Figure 3.4 (a) , it is observed that the KLD increases as the Inter-mean distance decreases. Specifically, the inter-mean distance increases from 0 to 6, and then decreases from 6 to 0. This trend highlights that as the components get closer, the clustering difficulty increases, and the KLD is able to capture this clustering error.

To control for this trend, we used the Adjusted Rand Index, a measure of similarity between two clustering results. The ARI ranges between -1 and 1, where -1 indicates complete disagreement between the two clustering results, 0 indicates that the clustering results are no more similar than expected by chance, and 1 indicates complete agreement between the two clustering results. In Figure 3.4 b), the ARI is inverted, but it still displays the same trend in clustering difficulty as the KLD. It is important to note that E-M, the probabilistic clustering algorithm used in this study, provides a probability of each point **x** belonging to a cluster **k**, rather than a discrete clustering decision. Therefore, to implement the ARI, we converted the E-M results to a discrete clustering result.

Rotation of Components Having established that changing the means in the synthesis model increases the clustering difficulty, the next step is to explore alternative operating conditions.

In the context of the synthesis model, changing the correlation coefficient ρ for two of the three components can be thought of as changing the relationship between these components. More specifically, it changes how they are oriented with respect to each other in the feature space. In Figure 3.5 (a), the contour plots of the simulate to be performed

When ρ is positive, it indicates that the two components are positively correlated, meaning that they tend to occur together in the same data points. When ρ is negative, it indicates that the two components are negatively correlated, meaning that they tend to occur in different data points. When ρ is zero, it indicates that the two components are uncorrelated, meaning that there is no relationship between them.

By changing the correlation coefficient ρ for two of the three components, the components can effectively be rotated about a common axis. This rotation changes the relationship between the components and can affect how they are grouped into clusters. For example, if two components are originally positively correlated, but decrease the correlation coefficient between them is decreased , they may no longer cluster together as strongly as before. Conversely, if two components are originally uncorrelated, but the correlation coefficient is increased between them, they may start to cluster together more strongly than before. Therefore, changing the correlation coefficient ρ is a useful tool for exploring how the orientation of the components affects the clustering results.

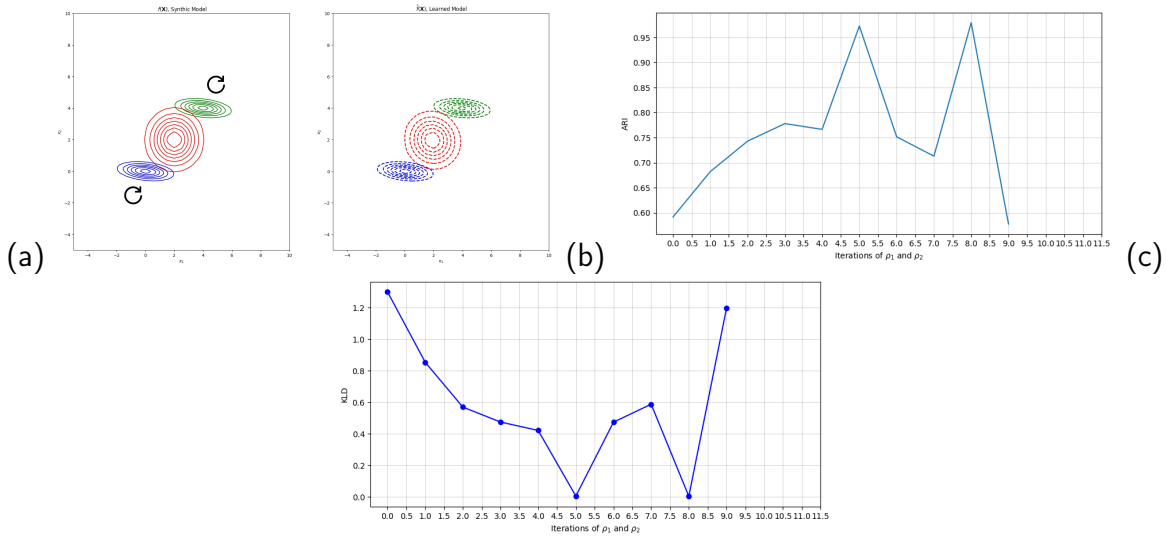


Figure 3.5: Contour of Simulation changing ρ_1 and ρ_2 (a) $D_{KL}[\hat{f}(X)||f(X)]$ over changing ρ_1 and ρ_2 (b) ARI over changing ρ_1 and ρ_2 (c)

Results : Rotation of Components

These simulations showed that by varying $\boldsymbol{\rho}$ in the synthesis model, the ability of E-M clustering

algorithm to learn the components of the synthesis model was affected. This shows the relationship between the value of $\boldsymbol{\rho}$ and the ability of a learnt model to estimate a synthesis model. In Figure 3.5, ARI is again shown to exhibit the same trend as KLD.

Conclusion

By varying the operating conditions of the synthesis model, this affects the ability of the learnt model to predict this true model. This is the first indication that KLD measurements can be used to track these clustering regimes.

Simulating Fair and Unfair Data

Having gained confidence with KLD measurements, the next step is to align the idea of conditional independence as a fairness goal with the output of an ML clustering task. To achieve this, we propose two data representations:

1. To represent a scenario where Conditional Dependence is present.
2. To represent a scenario where Conditional Independence is present.

Let A_u and A_p both be binary, $\in$ 0,1. We obtain four joint probability distributions: f($A_u = 0$,$A_p = 0$), ... , f($A_u = 1$,$A_p = 1$). A_p represents a privileged and a non-privileged group, where $A_p = 0$ is the non-privileged group and $A_p = 1$ is the privileged group. Similarly, A_u represents a desirable and a non-desirable unprotected attribute, where $A_u = 0$ is the non-desirable attribute and $A_u = 1$ is the desirable attribute.

To create an Unfair mixture, a split in A_u which is dependent on A_p is forced, as shown in the left hand plot in Figure 3.6. The two shades of green represent the split in $A_u = 1$ dependent on A_p, and the two shades of red represent the split in $A_u = 0$ dependent on A_p . This is achieved by giving each of the states a separate μ and Σ. To create a fair mixture, A_u is sampled with no dependence on A_p, as shown in the right plot in Figure 3.6. Conversely a merging of the split components, is taken as representing the fair case.

Two key questions then arise:

1. Which two Gaussian components to merge?
2. How to merge two Gaussian components?

For 1), we believe that the closest components represent the ideal merge candidates. However, the weakness of this approach is the assumption that the A_p split on A_u is not as great as the split on $A_u = 0$ and $A_u = 1$.

1) Find closest pairs of Gaussian's This is done by

---

1. Creating a distance matrix D $\in R^{n \times n}$ between all pairs of Gaussian components, where the distance metric is the Mahalanobis distance found in equation 3.9 :
2. Setting the diagonal elements of D to infinity, so that each Gaussian component is not considered the closest to itself: $D_{i,i} = \infty$.

3. Initializing the two closest pairs with the first pair of indices: pair1 $= [0, 1]$, pair2 $= [2, 3]$.

4. Initializing the closest distance with the distance between the first pair of Gaussian's:

$$\text{closest\_dist} = D_{\text{pair1}[0],\text{pair1}[1]} + D_{\text{pair2}[0],\text{pair2}[1]}. \tag{3.8}$$

5. Looping through all pairs of Gaussian components to find the two closest pairs, by checking all combinations of indices i, j, k, l where $i \neq k, l$ and $j \neq k, l$. For each combination, the distance between the two pairs of Gaussians is computed: dist $= D_{i,j} + D_{k,l}$.
   If the distance is smaller than the current closest distance, update the closest pairs and distance: closest_dist $=$ dist, pair1 $= [i, j]$, pair2 $= [k, l]$.

6. Returning the two closest pairs: return pair1, pair2.

$$D_{i,j} = \sqrt{(m_i - m_j)^T \Sigma_j^{-1} (m_i - m_j)}, \tag{3.9}$$

where m_i and Σ_i are the mean and covariance of the i-th Gaussian component, respectively, and n is the total number of Gaussian components.

II) Merge of two Gaussians 1) Single variate :

$$\mu = E[x] = \frac{\pi_1\mu_1 + \pi_2\mu_2}{\pi_1 + \pi_2}$$

$$\sigma^2 = E\left[x^2\right] - E[x]^2$$

$$= \frac{\sigma_1^2\pi_1^2 + \sigma_2^2\pi_2^2 + \left(\pi_1\mu_1 + \pi_2\mu_2\right)^2}{\left(\pi_1 + \pi_2\right)^2} - \mu^2$$

$$= \frac{\sigma_1^2\pi_1^2 + \sigma_2^2\pi_2^2}{\left(\pi_1 + \pi_2\right)^2} + \frac{\left(\pi_1\mu_1 + \pi_2\mu_2\right)^2}{\left(\pi_1 + \pi_2\right)^2} - \mu^2$$
$$= \frac{\sigma_1^2\pi_1^2 + \sigma_2^2\pi_2^2}{\left(\pi_1 + \pi_2\right)^2} + \mu^2 - \mu^2$$
$$= \frac{\pi_1^2}{\left(\pi_1 + \pi_2\right)^2}\sigma_1^2 + \frac{\pi_2^2}{\left(\pi_1 + \pi_2\right)^2}\sigma_2^2$$
$$\alpha = \frac{\pi_1^2}{\left(\pi_1 + \pi_2\right)^2}$$
$$= \alpha\sigma_1^2 + (1 - \alpha)\sigma_2^2$$

Here, μ represents the mean of the merged Gaussian distribution, and σ^2 represents its variance. α is a weight factor that determines the contribution of each Gaussian component to the merged distribution.

2) Multivariate

For the multivariate condition, the following formula is used:

$$\Sigma = E\left[XX^T\right] - E[X]E[X]^T$$

$$= \frac{\pi_1^2\Sigma_1 + \pi_2^2\Sigma_2 + (\pi_1\mu_1 + \pi_2\mu_2)(\pi_1\mu_1 + \pi_2\mu_2)^T}{(\pi_1 + \pi_2)^2} - \mu\mu^T$$

$$= \frac{\pi_1^2\Sigma_1 + \pi_2^2\Sigma_2}{(\pi_1 + \pi_2)^2} + \frac{(\pi_1\mu_1 + \pi_2\mu_2)(\pi_1\mu_1 + \pi_2\mu_2)^T}{(\pi_1 + \pi_2)^2} - \mu\mu^T$$

$$= \frac{\pi_1^2\Sigma_1 + \pi_2^2\Sigma_2}{(\pi_1 + \pi_2)^2} + \mu\mu^T - \mu\mu^T$$

$$= \frac{\pi_1^2}{(\pi_1 + \pi_2)^2}\Sigma_1 + \frac{\pi_2^2}{(\pi_1 + \pi_2)^2}\Sigma_2$$

$$\alpha = \frac{\pi_1^2}{(\pi_1 + \pi_2)^2}$$

$$= \alpha\Sigma_1^2 + (1 - \alpha)\Sigma_2^2$$

Here, Σ represents the covariance matrix of the merged Gaussian distribution and α is a weight factor that determines the contribution of each Gaussian component to the merged distribution.

Merge Two Multivariate Gaussian's

Using equations 3.10 and 3.12 , the transformation the 4 component GMM to a 2 component GMM, is carried out by calling the merge function twice.

$$\mu = E[x] = \frac{\pi_1\mu_1 + \pi_2\mu_2}{\pi_1 + \pi_2} \tag{3.10}$$

$$\alpha = \frac{\pi_1^2}{(\pi_1 + \pi_2)^2}, \tag{3.11}$$

$$\Sigma = \alpha\Sigma_1^2 + (1 - \alpha)\Sigma_2^2 \tag{3.12}$$

Simulations 2: Inter-mean distance Having established how to merge a 4 component Unfair Gaussian Mixture into a Fair 2 component one, the next step is to explore how increasing levels of split would affect the Clustering performance. By varying the operating conditions of the Unfair Mixture, increasing and decreasing levels of fairness can be forced. Building on previous simulations an Inter-mean distance is chosen.

Note : A 70/30 training test split was used.

The process is the following :

1. Sample from the unfair and the fair mixtures, with operating Conditions $d_1(\mu_0, \mu_1)$
2. Cluster both mixtures with number of Clusters set to 2, calculate $D_K L[M_{UF}||M_F]$
3. Repeat the process with increasing $d(\mu_0, \mu_1)$

starting with Figure 3.6 (a) and moving to a case such as Figure 3.6 (b).

Results: Inter mean In Figure 3.7 (a), the KLD measurement that is conditional on A_u is measured over an increasing Inter-mean distance. The source of divergence can be tracked by this measurement, the higher of the 4 KLD's in the graph being the result of the split of the two components.

In Figure 3.7 (b), KLD is marginal w.r.t A_u and is shown to align with measurements of DI. Note again that DI of 1 or a KLD of 0 represents the ideal.

In summary, these simulations demonstrate that a marginal KLD can show similar results to that of DI. A conditional KLD can show which components are the source of the divergence.

Simulations 2: Rotation of Components Again following earlier simulations, a rotation of the components in the unfair Gaussian represented the next sequence of operating conditions to

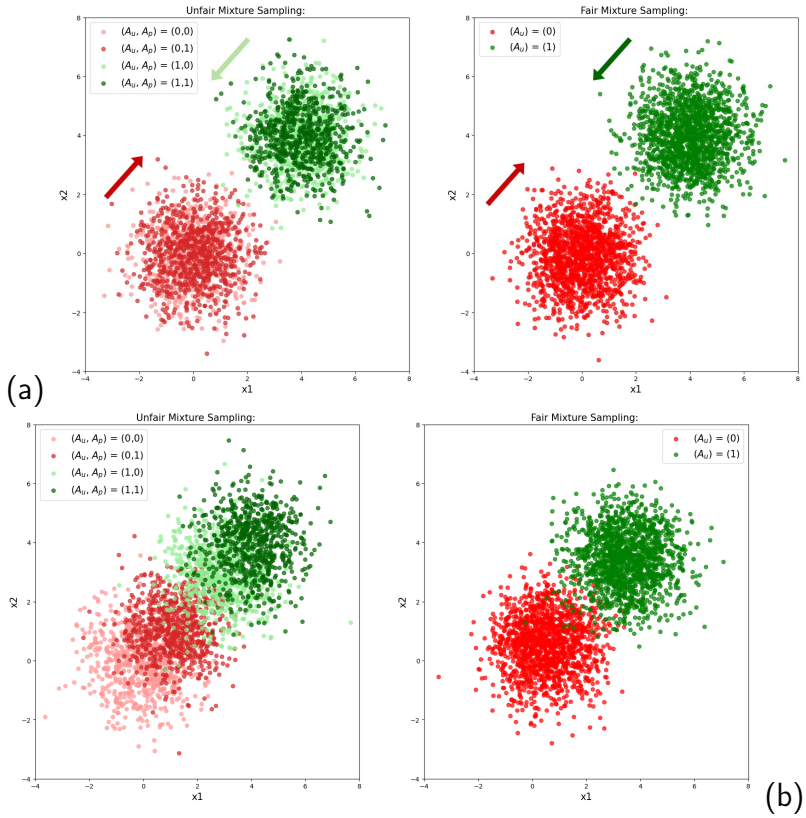


Figure 3.6: Starting with no Split (a) Split (b)

be performed. The results are shown in Figure 3.8. One of each the green and red components are rotated and thus the complexity of the clustering task is increased.

Results : Correlation Coefficient The output gives similar, but noisier, results as that of the inter mean simulations, as shown in Figure 3.9. This is because the merged or fair representation is challenged by a more difficult clustering task. However, the marginal KLD in this case seem to be more robust than that of DI.

Simulations 3: Increasing Complexity

An important aspect of the simulation work, was to examine how the model would scale, by increasing the number of components in the Unfair Gaussian Mixture. There now is another case of A_u Figure 3.10 , that could be seen as an average attribute, such $A_u \in$ {'bad', 'average', 'good'}, which would correspond to {'red', 'blue', 'green'}.

In this case, again an Inter-mean distance between components of the same A_u was used. In Figure 3.11, the marginal KLD was compared again with the DI. The results, although a little

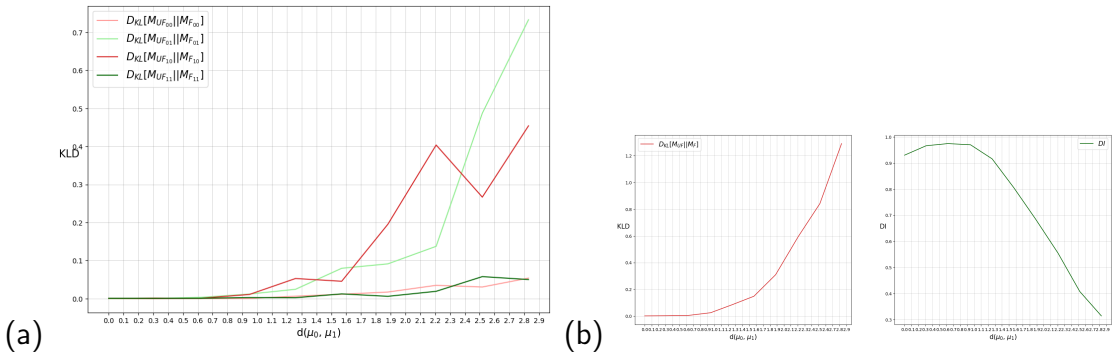


Figure 3.7: KLD fair (a) KLD fair vs DI (b)

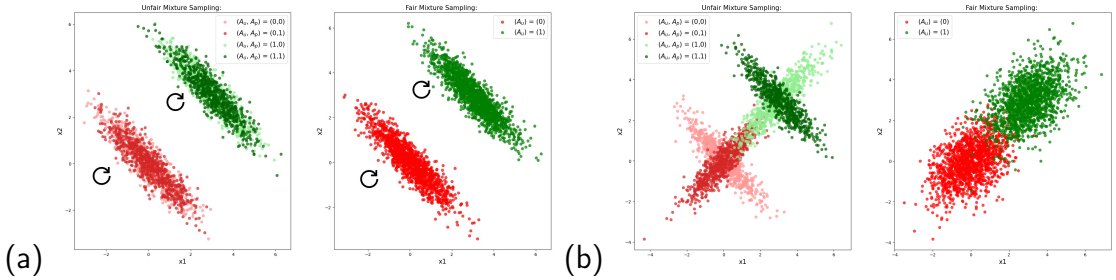


Figure 3.8: Starting with no Split (a) Split (b)

noisy, indicate both methods are capable of tracking increasing levels of unfairness.

To summarize, increasing complexity via increasing labels for A_u , can provide results similar to that with lower complexity.

3.3 Repair for CI

In the previous section, it was observed how, by changing the operating conditions of a Gaussian Mixture, it would increase/decrease the levels of unfairness, when measured with DI and the new KLD measurement. Now the task becomes to repair this simulated data.

In the creation of our fair data, the two closest Gaussian's in the unfair data were identified and then merged. This informed the following transport plan :

3.3.1 Merge Repair

1. Split the data set by protected attribute, X_0 and X_1
2. Learning the means $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ of both X_0 and X_1 .

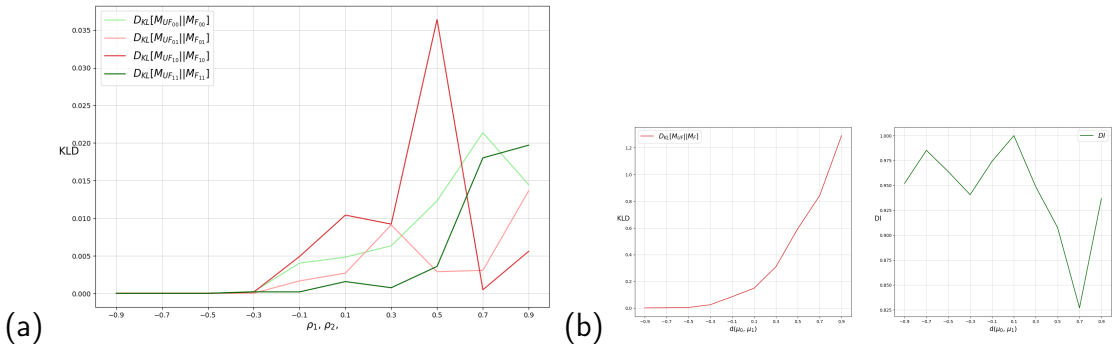


Figure 3.9: KLD fair (a) KLD fair vs DI (b)

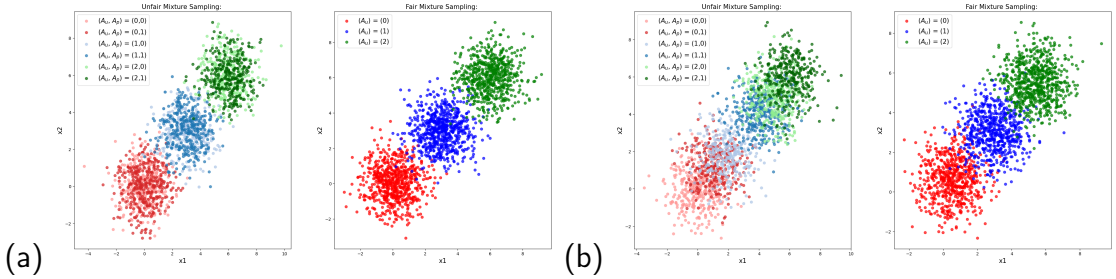


Figure 3.10: Starting with no Split (a) Split (b)

3. Find the merge equivalent, μ_{merge} and Σ_{merge}

$$\boldsymbol{\mu}_{\text{merged}} = \frac{\pi_i \boldsymbol{\mu}_i + w_j \boldsymbol{\mu} j}{\pi_i + \pi_j}, \quad \boldsymbol{\Sigma}_{\text{merged}} = \alpha \Sigma_i^2 + (1-\alpha)\Sigma_j^2 \tag{3.13}$$

where π_i and π_j are the weights of the i-th and j-th Gaussian components, respectively.

4. Apply a full or partial transformation:

   - **Full repair**: For a full repair, for a point x_i from $X \in \mathbf{R}^d$,

     (a) $x_i = x_i$ - $\mu_{orignal}$.

     (b) $\Sigma = LL^T$, where L is a lower triangular matrix with positive diagonal entries.

     (c) Then $\mathbf{x} = \mathbf{x} \cdot A_c$.

     (d) Then $x_i^{repaired} = \mu_{merged} + x_i$.

   - **Partial repair**: Frr a partial repair, Set $\lambda \in [0,1]$ as the amount of repair. Set $r \sim$ B(n, p), where p $=\lambda$. If $r < \lambda$, repair; otherwise, don't repair.

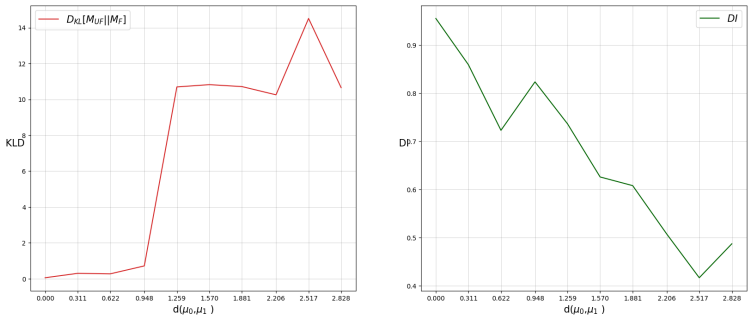


Figure 3.11: $D_{KL}[\hat{f}(X)||f(X)]$ of a inter mean Distance

3.3.2 Optimal Transport based Repair

Having established the goal of developing a merge based transformation repair method, it was then necessary to examine how results using this method compared to the results using a state-of-the-art method. The OT based repair was implemented using the methods established by Gordaliza et al [12] and the Github repository [41]. Two OT based repairs were implemented, 1) A Geometric based repair (OT: Geometric) and 2) a Random based repair (OT: Random)

Geometric Repair: For a geometric based repair the steps are as follows:

1. First, split the data by protected attribute into X_0 and X_1. Then, calculate an OT linear mapping in both directions:

$$A_{e_{01}}, b_{e_{01}} = T(X_0, X_1)$$
$$A_{e_{10}}, b_{e_{10}} = T(X_1, X_0)$$

2. Next, calculate the weights w_0 and w_1 of X_0 and X_1, respectively. Then, calculate the two barycenters f_{b_0} and f_{b_1}:

$$f_{b_0} = w_1(X_0 A_{e_{01}} + b_{e_{01}}) + w_0 X_0$$
$$f_{b_1} = w_0(X_1 A_{e_{10}} + b_{e_{10}}) + w_1 X_1$$

3. Finally, calculate the repaired datasets $X_0^{repaired}$ and $X_1^{repaired}$ as follows:

$$X_0^{repaired} = \lambda f_{b_0} + (1 - \lambda)X_0$$
$$X_1^{repaired} = \lambda f_{b_1} + (1 - \lambda)X_1$$

Here, A_e and b_e are the affine transformation (A_e) and the translation (b_e) respectively. The affine transformation is a matrix which deals with the transformation of the second-order moments, while the translation is a vector which transforms of first-order moments.

Random Repair: The random based repair is carried out as follows

1. First split the data by protected attribute into X_0 and X_1, then calculate an OT linear mapping in both directions:

$$A_{e_{01}},\ b_{e_{01}} = \mathrm{T}(X_0,\ X_1)$$
$$A_{e_{10}},\ b_{e_{10}} = \mathrm{T}(X_1,\ X_0)$$

2. Next, calculate the weights w_0 and w_1 of X_0 and X_1, respectively. Then, calculate the two barycenters f_{b_0} and f_{b_1}:

$$f_{b_0} = w_1(X_0 A_{e_{01}} + b_{e_{01}}) + w_0 X_0$$
$$f_{b_1} = w_0(X_1 A_{e_{10}} + b_{e_{10}}) + w_1 X_1$$

3. Generate two Bernoulli random variables $\mathrm{Ber}(\lambda)$ of dimensions $(n_0, 1)$ and $(n_1, 1)$ respectively:

$$\mathrm{B0}, \mathrm{B1} \sim \mathrm{Ber}(\lambda)$$

4. Randomly repair the data using the Bernoulli random variables to create $X_0^{repaired}$ and $X_1^{repaired}$:

$$X_0^{repaired} = \mathrm{Ber}_0 \cdot f b_0 + (1 - \mathrm{Ber0}) \cdot X0$$
$$X_1^{repaired} = \mathrm{Ber}_1 \cdot f b_1 + (1 - \mathrm{Ber1}) \cdot X1$$

3.3.3 Repair for CI with OT and Merge

Now having established the method "Merge" and two OT based methods Geometric and Random, next step is to examine how they compare, when applied to fairness repair task. Simulating X_{UF} as Gaussian Mixture Model with 4 components to represent the Conditional Dependence between, A_u and A_p. Then sampling X_F the fair or merged equivalent of X_{UF}.

In these simulations we aim to track the following:
1) KLD that is conditional w.r.t A_u.

$$D_{KL}[M_{UF}||M_F] \tag{3.14}$$

2) Disparate Impact and KLD marginal w.r.t A_u

$$\sum_{i=1}^{n} D_{KL}[M_{UF}||M_F] \tag{3.15}$$

where n represents the number of unique A_u labels.

3) The Cost of Transportation, the KLD between the original GMM and the repaired GMM.

$$\hat{D_{KL}}[f_{repaired}||f_{orignal}] \tag{3.16}$$

Note: This is a estimation of KLD via Monte Carlo simulations.

First the plot of the data must be repaired. Simulating the data as explained in the above, we obtain the results shown in Figure 3.12, and observe again that we have conditional dependence on A_p, which will cause classification dependent on A_p. The amount of repair is determined by varying $\lambda \in [0, 1]$. In order to simulate different levels of repair λ , so we track all these measurements for a range of λ.

1) KLD that is conditional w.r.t A_u The conditional KLD was tracked using equation 3.14, over a range of λ. The plot for both the Merge and OT repair is given in Figure 3.13. At 0 λ, the KLD's are not the same, due to different components causing different levels of divergence.

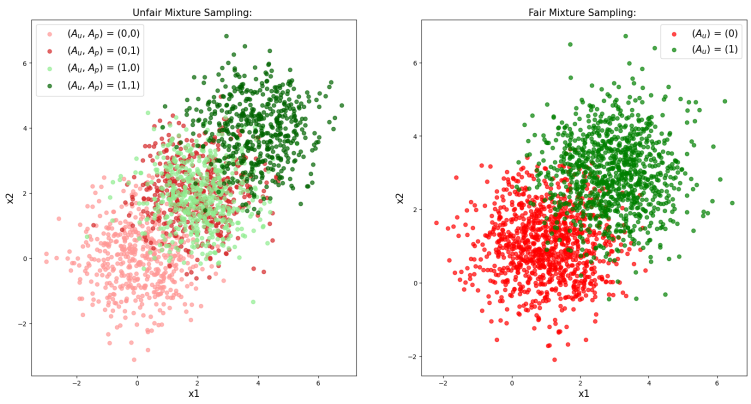


Figure 3.12: Simulated data to be repaired

The use of KLD here appears to offer more insight into the location of unfairness in the data than a DI measure or marginal KLD.

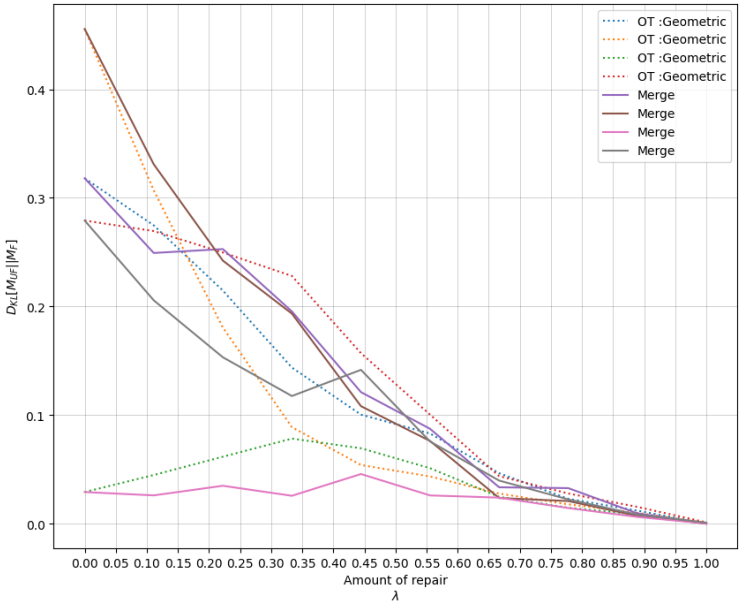


Figure 3.13: KLD conditional w.r.t A_u

2) Disparate Impact and KLD marginal w.r.t A_u For the same repair, the tracking of marginal KLD and DI is carried out using equation 3.15.

In Figure 3.14, KLD converges to zero as whilst repairing for the Conditional Dependence in the data X_{UF}, and DI converges to one, demonstrating that the Conditional Dependence is being removed. The clustering algorithm is no longer clustering dependent on A_p, which is showing that the clustering can now be considered fair.

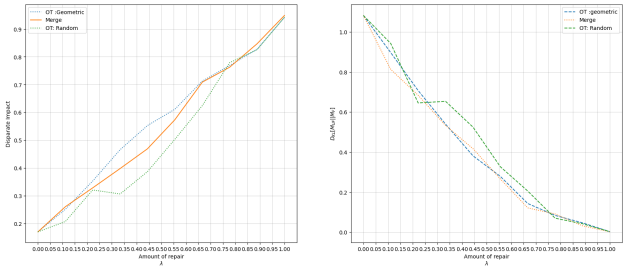


Figure 3.14: KLD conditional w.r.t A_u

3) The Cost of Transportation To implement equation 3.16, it was necessary to learn all the components of $f_{original}$, an Unfair GMM which is a 4 component GMM, then learn a 2 component GMM at every stage of repair $(f_{repaired}, \lambda_i)$, then calculating the KLD from the original to the current model at every stage of repair λ.

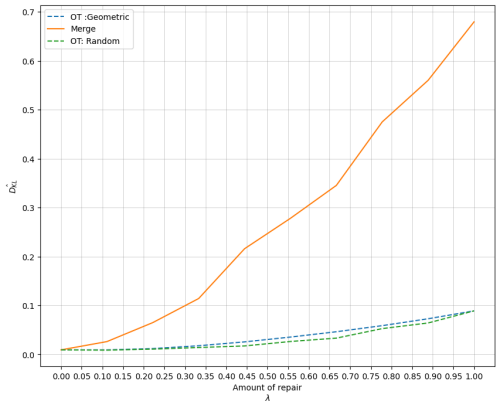


Figure 3.15: KLD conditional w.r.t A_u

From Figure 3.15, it apparent that the two OT methods are significantly more efficient than the merge methods, $\hat{D}_{KL-OT}[f_{repaired}||\ f_{original}]$ being much smaller than that of the merge repair $\hat{D}_{KL-Merge}[f_{repaired}||\ f_{original}]$. This could also be considered an accuracy/fairness trade off.

4 Repairing Real Data

After seeing the effectiveness of the methodologies using simulated data, the next step is to apply them with real data. There are many commonly used datasets in fairness research with access to the Protected Attributes, which is vital for our research. These can be seen as research datasets. The features in these datasets are often non-Gaussian categorical features . For example, in the COMPAS dataset "Prior Counts" is a categorical non-Gaussian feature.

Relaxations working with Real data Although the simulations deal with Gaussian Mixture Models, as transformations are being applying via first and second-order moments, a merger repair is valid, although acknowledging that this knowledge constraint means that only a partial repair is implementable.

In the simulation work, the ML task of interest was clustering as this was appropriate as the task at the time was to learn GMM's. However, most of the datasets deal with prediction tasks, where classification models are more relevant.

4.1 Implementation with Real Data

The steps in implementing the methodology with real data are sourcing and cleaning the data, exploring issues with the data, feature engineering, and model comparison.

The motivation for a repair is to ensure two things: 1) that there exists unfair bias within the data and 2) that the feature vectors can predict the target.

Flow of the ML Task To give a overview of the following repair process, Figure 4.1 is provided.

The process is as follows:

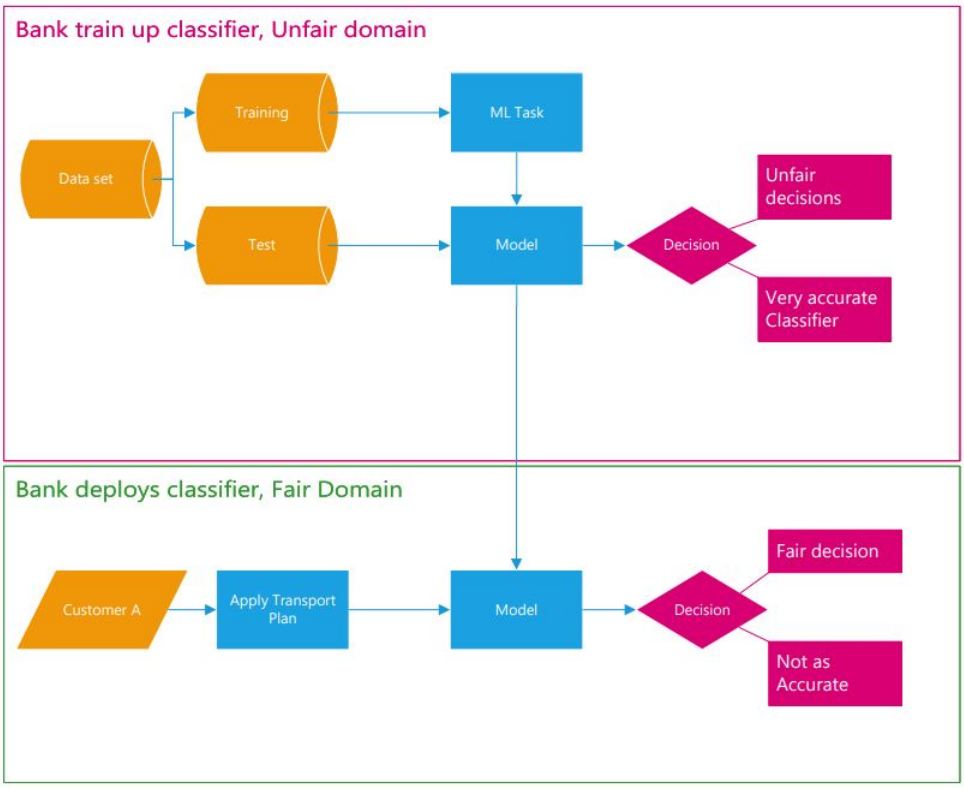


Figure 4.1: ML Flow of the Data Repairs

Train up a model on data that has not be repaired for, then import that model into a new domain. In this domain, apply a transformation, OT's and Merge transformation separately. Perform these transformations for a range of repair λ and predict with them using the imported model.

Motivation for a Repair

1. Split the dataset $\mathbf{X}$ into two datasets $\mathbf{X0} = \mathbf{X}/\mathbf{A_p} = \mathbf{0}$ and $\mathbf{X1} = \mathbf{X}/\mathbf{A_p} = \mathbf{1}$.
2. Combine the $\mathbf{X0}$ and $\mathbf{X1}$, keeping track of the label A_p
3. Split the dataset into test and train data with a (70/30) split
4. Train the data with a logistic regression model
5. Predict with the holdout data, calculating the DI and the accuracy/misclassification

If there is motivation for repair, the repair steps are:

Repair

1. Split the dataset $\mathbf{X}$ into two datasets $\mathbf{X}_0$, $\mathbf{X}/\mathbf{A_p} = \mathbf{0}$ and $\mathbf{X}_1$, $\mathbf{X}/\mathbf{A_p} = \mathbf{1}$.
2. Apply a full repair, obtaining $\mathbf{X}_{0_{Full}}^{Repaired}$ and $\mathbf{X}_{1_{Full}}^{Repaired}$
3. Apply the transformation (Merge and OT: Geometric) with the weighing of λ_i
4. Split the Repaired datasets into test and train data with a (70/30) split
5. Train the data with a logistic regression model for $\mathbf{X}_{Full}^{Repaired}$ $\mathbf{X}_{Partial}^{Repaired}$
6. Predict with the hold out data, record the DI, accuracy and take the KLD from the confusion matrix of the Prediction with $\mathbf{X}_{Partial}^{Repaired}$ from the confusion matrix of $\mathbf{X}_{Full}^{Repaired}$

4.2 COMPAS data

As has been noted in the literature review, the COMPAS dataset is a dataset collected and published by Propublica. The Protected Attributes in the dataset are 'sex', 'age', and 'race'. 'Caucasian' and 'non Caucasian' was the A_p of choice here, Caucasian is $A_p = 1$ and 'non Caucasian' is $A_p = 0$. The feature vectors (Unprotected Attributes) are 'juvenile felony count', 'juvenile misdemeanor count', 'Prior Counts', 'Charge degree', 'score text' and ' violent score text'.
The target variable is 'Two year recisvism', which is the recisvism rate for defendants for which there are recorded feature vectors.

Motivation for Repair: COMPAS The DI was around 0.5, which indicates a biased classifier. Therefore, the is motivation for repair. The Misclassification is 0.322, which shows that the feature vectors can predict two year recisvism (the target). The repair process is a follows:

Data Repair: COMPAS

In Figure 4.2, it can be seen that the merge repair is effective for removing the unfair bias in that data.

4.3 Adult Dataset

The Protected Attributes were 'Gender' and 'race', and the target variables were 'age', 'Education level', 'Capital gain', 'Capital loss', and 'Working Hours per week'. The Protected Attribute chosen for repair was 'race', where 'Caucasian' is $A_p = 1$ and 'non Caucasian' is $A_p = 0$.

Motivation for Repair: Adult The starting DI was 0.6, which indicates a unfair biased classifier. The misclassification for the logistic regression model was 0.18, showing a strong ability

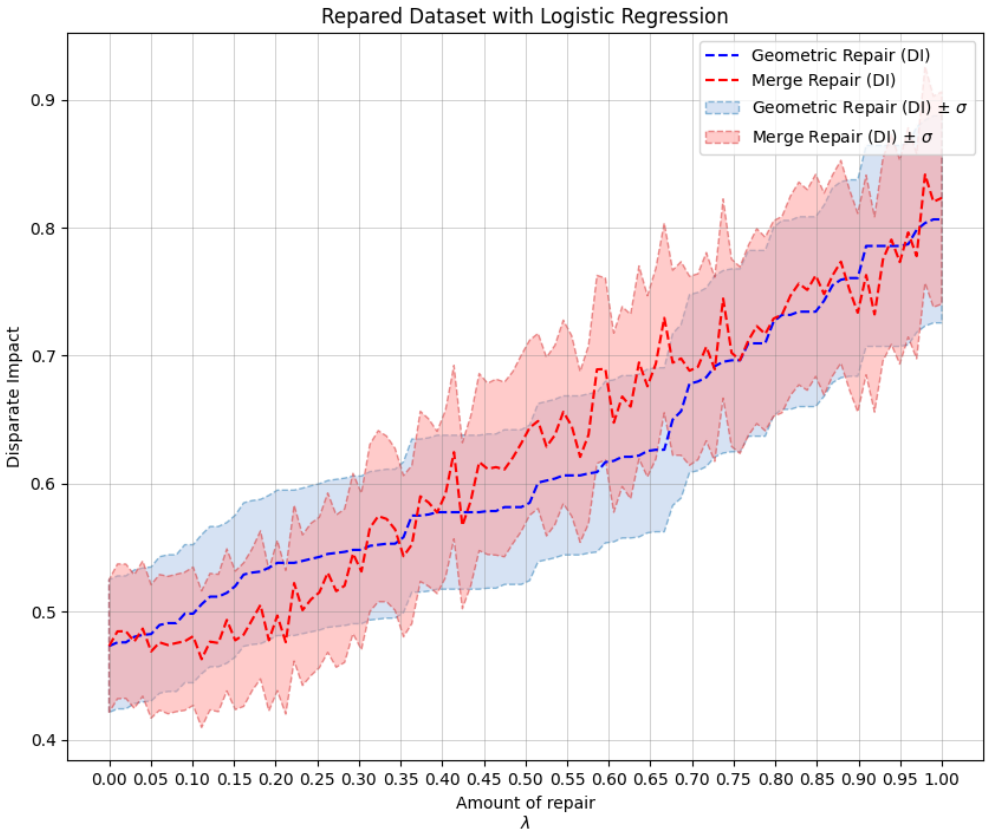


Figure 4.2: Repairing COMPAS Dataset, with OT: Geometric and Merger method

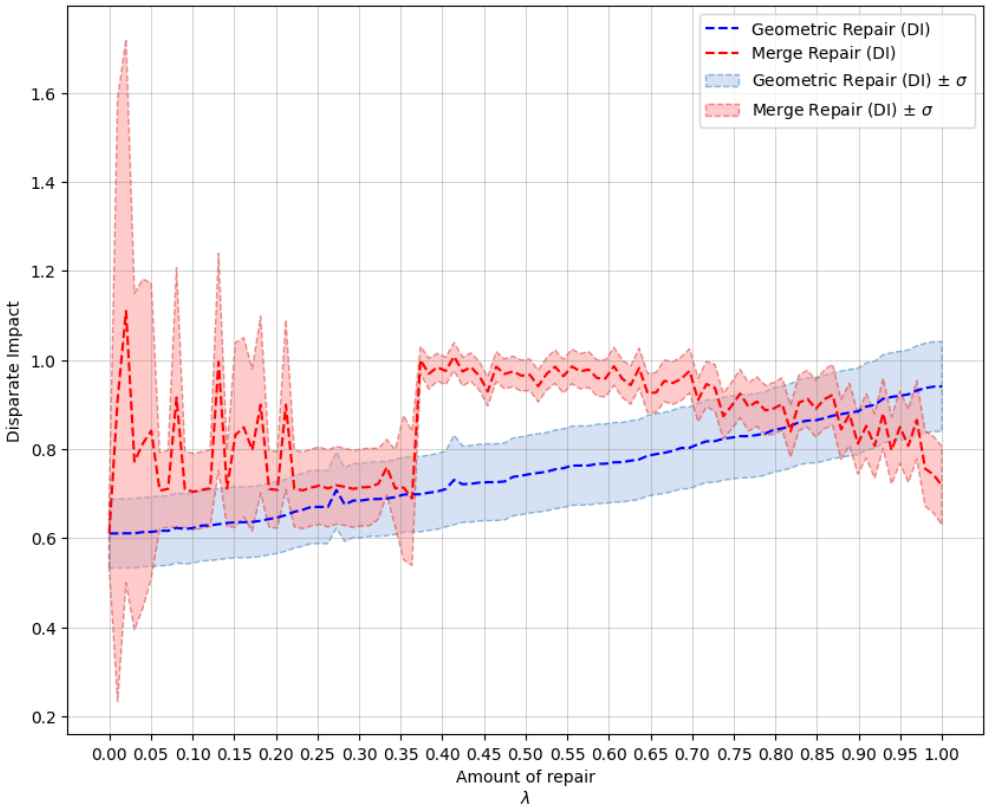


Figure 4.3: Repairing Adult Dataset, with OT: Geometric and Merger method

of the feature vectors to predict the target variable 'income'. Therefore, there is motivation for repair.

Data Repair: Adult In Figure 4.3, Merge repair is seen to take a completely different path to that OT Geometric. These results are hard to interpret and the early stages of merger repair for the Adult Dataset are seen to be very noisy.

4.4 Bank Dataset

The Protected Attribute was 'age'. The feature vectors were 'job', 'marital', 'education', 'housing', 'loan', 'contact', 'month', 'day of week', and 'previous'.

Motivation for Repair: Bank

The initial DI was 0.65, and the Misclassification was 0.12, indicating the presence of bias in the classifier. Therefore, there is motivation for repair.

Data Repair: Bank

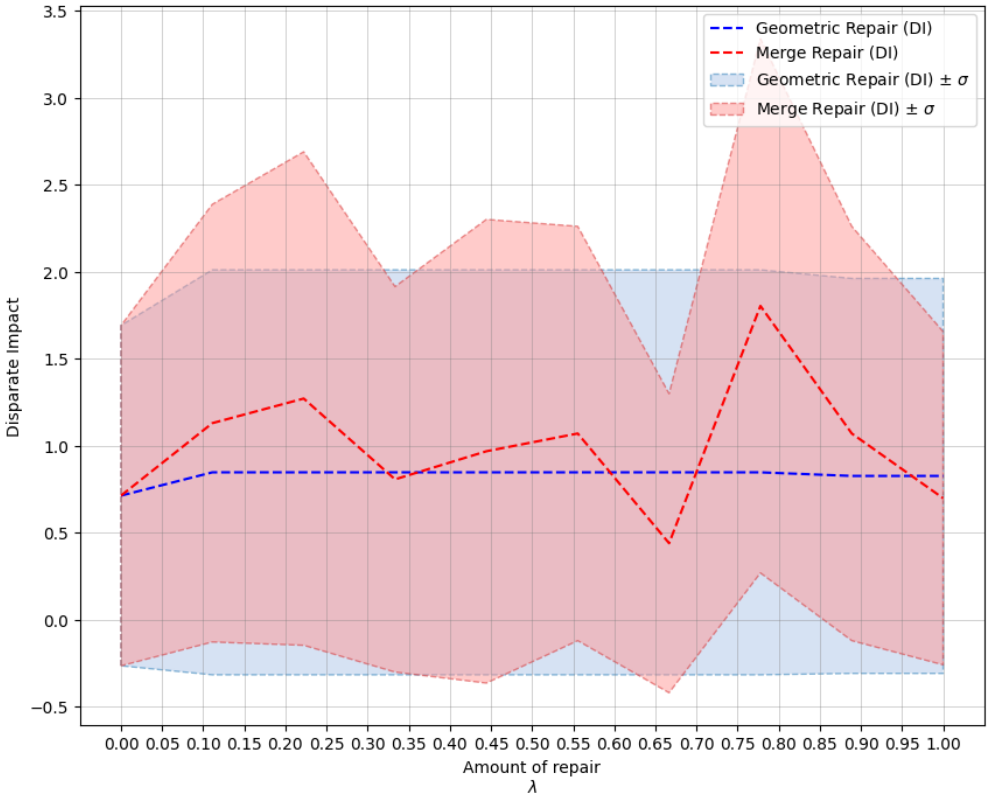


Figure 4.4: Repairing Bank Dataset, with OT: Geometric and Merger method

In Figure 4.4, it is observed that OT: Geometric struggles to repair the dataset, as seen by the decreasing levels of confidence as indicated by the standard interval ($\mu_{DI} \pm |\sigma|$). However, Merge repair takes a different path from Geometric repair, as seen by the divergent trends in the repaired DI values.

4.4.1 Conclusions

The simulation work, showed much promise for a Merge based repair. However when applied to real data, the results of the merge repair preform to the same level as the state-of-art-methods of OT.

Altering the method for the merge repair by moving from an E-M clustering model to a logistic regression model, compromised much of the learning from the simulation work.

This showed that there is much work needed if a Merge based data repair was to be established as a reliable approach for data repair in the future. Though the success of Merge based approach in the simulated data and COMPAS data shows there is motivation for further development.

Additionally if time had permitted, the merger method could have been applied with a clustering goal.

5 Conclusion

The importance of defining and addressing fairness in AI is paramount, as current methods are new and there is no single agreed upon method.

The aim of this research was to propose an alternative measure of fairness in AI, and to explore if conditional independence (CI) could be viewed as a fairness goal.

The main findings of the study contribute to this field of research in several ways. Firstly, the research indicates that Kullback-Leibler Divergence (KLD) based measurements are comparable to Disparate Impact (DI) and therefore have potential for use as an alternative measure of fairness, notwithstanding limitations noted when used with real data. Secondly the results show that CI can be viewed as a fairness goal. Thirdly, it was observed that treating fairness measurement and repair as one system could allow for more effective fairness repairs.

Positive outcomes of the research were firstly, support for the argument that CI has value in AI deployment, as a realistic, representative and achievable fairness goal as an alternative for example, to a disparate impact fairness goal. Additionally, it was found that the KLD Conditional measurement, in providing a deeper qualitative knowledge of fairness measurement, could inform and allow for the optimization and efficiency of repair methodologies.

The study was limited by the research design and choices made during the work, which limit applicability of the findings to real data. Due to insufficient work done on CI in real data, the value of the results is also limited. Finally although many simulations were carried out, time limitations applied and so areas for further research are suggested.

5.1 Project Goals Assessment

With regards to objectives outlined in the interim report in the early stages of this project, the following objectives were successfully completed.

1. Research current methods of measuring algorithmic fairness
2. Research current approach to correction/repair for this bias
3. Design alternative method of assessing and repairing for fairness
4. Selecting criteria for assessing the robustness of the framework and Determine the robustness of the framework when applied to the transfer learning

As outlined in the above conclusions, tasks 1-4 were completed to a degree.

The following objectives were partially completed:

1. Developing a framework for applying a FPD approach
2. Applying the framework through transfer learning

On point 1) the approach taken took inspiration from FPD, however the dissertation did not take advantage of the currently available FPD procedures such as, merging of external knowledge or approximate learning and stabilized forgetting.

The dissertation did deal with the transfer learning scenario, by taking a previously trained unfair model, and transforming data that the unfair model was predicted on, in such a way that the resulting decision was fair. However, the merge method was far from optimal and learning about data in the source domain was not explored as well as it should have been.

5.2 Recommendations for the future

During the dissertation work, many simulations were carried out but it was not possible to explore all the avenues of interest. The following could form the basis for further research.

5.2.1 Optimise the Merge Repair

The simulation work with repairs showed that compared to the two Optimal transport approach's, the merging repair transports more features to achieve the same level or fairness, which compromises the accuracy post-transport, see Figure 3.15 .

Future research into a more optimal way in which to merge two or more distributions would be useful. The research acknowledges that Optimal Transport represents the current state-of-the-art,

but these research outcomes indicate that further research into this new methodology is justified.

In this research, the data representation chosen was a Gaussian Mixtures. Further research could include work on developing merge plans for alternative data representation e.g. Gaussian and Gamma distributions. This could be done in such a way that a merge rule having prior knowledge of the data representation, could apply a merge condition based on that knowledge.

A possible next step to optimise a merge based transport plan could be through inclusion of stabilized forgetting, such as described by Quinn et al. in [31], potentially delivering an Optimal-FPD transport plan comparable with that delivered by Courty et al. [28] in optimal transport.

5.2.2 Measuring fairness with a conditional KLD

In the simulation work carried out in this project a conditional KLD was shown to indicate where unfair bias is occurring and shows that repairs targeting these unfair biases are unseen by DI. Due to the time constraints of this project, implementation of this idea with real data was not done. This could represent a nice tool for evaluating which features in a dataset are causing the most unfair bias and inform whether to remove them or not.

5.2.3 Clustering Real Data

The simulation work involved clustering models, and then aligning the work done in this dissertation to the work done by Gordaliza et al. [12]. When working with real data, classification models were used instead of clustering models. Future work could implement merge based fairness repairs with fair clustering as the goal.

5.2.4 Final statement

In conclusion, the proposed alternative measure of fairness and the findings from the study have the potential to inform and improve fairness in AI deployment and the researcher has gained a better understanding of, and the ability to contribute usefully, in the field of AI fairness.

Bibliography

[1] "Four in ten irish companies currently use artificial intelligence (AI)," https://www.nsai.ie/about/news/four-in-ten-irish-companies-currently-use-artificial-intelligence-ai/, accessed: 2023-4-11.

[2] https://www.eciia.eu/wp-content/uploads/2021/05/1_en_annexe_autre_acte_part1_v8_vf_C4B261EB-ABA4-5C30-1555482869410384_75787.pdf, accessed: 2023-4-11.

[3] M. Cannarsa, "Ethics guidelines for trustworthy AI," in *The Cambridge Handbook of Lawyering in the Digital Age*. Cambridge University Press, Nov. 2021, pp. 283–297.

[4] D. Hellman, "Measuring algorithmic fairness," Jul. 2019.

[5] T. P. O. R. ynthia Dwork, Moritz Hardt and R. Zemel, "Fairness through awareness," 2012.

[6] J. M. C. S. S. V. Michael Feldman, Sorelle A. Friedler, "Certifying and removing disparate impact," 2015.

[7] A. D. Selbst, "Disparate impact in big data policing," *SSRN Electron. J.*, Feb. 2017.

[8] L. S. Shapley, "On balanced sets and cores," *Nav. Res. Logist. Q.*, vol. 14, no. 4, pp. 453–460, 1967.

[9] T. Calders and S. Verwer, "Three naive bayes approaches for discrimination-free classification," 2010.

[10] E. P. Moritz Hardt and N. Srebro., "Equality of opportunity in supervised learning. in advances in neural information processing systems," 2016.

[11] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," Oct. 2016.

[12] E. del Barrio, F. Gamboa, P. Gordaliza, and J.-M. Loubes, "Obtaining fairness using optimal transport theory," Jun. 2018.

[13] E. S. DANA PESSACH, "\"algorithmic fairness\"," "2010".

[14] A. F. S. G. A. H. Sam Corbett-Davies, Emma Pierson, "Algorithmic decision making and the cost of fairness," 2018.

[15] S. Corbett-Davies and S. Goel., "The measure and mismeasure of fairness: A critical review of fair machine learning," 2018.

[16] S. J. M. K. A. R. Richard Berk, Hoda Heidari, "Fairness in criminal justice risk assessments: The state of the art," 2017.

[17] F. Kamiran and T. Calders, "Data preprocessing techniques for classification without discrimination," 2012.

[18] K. L. Yahav Bechavod, "Learning fair classifiers: A regularization-inspired approach," 2017.

[19] ——, "Penalizing unfairness in binary classification," 2018.

[20] V. S. Novi Quadrianto, "Recycling privileged learning and distribution matching for fairness."

[21] A. T. K. Cynthia Dwork, Nicole Immorlica and M. Leiserson, "Decoupled classifiers for fair and efficient machine learning," July 21, 2017.

[22] "Aif360," https://pypi.org/project/aif360/, accessed: 2023-4-9.

[23] L. Bruzzone and M. Marconcini, "Domain adaptation problems: a DASVM classification technique and a circular validation strategy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 770–787, May 2010.

[24] C. Chen, Q. Chen, J. Xu, and V. Koltun, "Learning to see in the dark," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Jun. 2018, pp. 3291–3300.

[25] D. Bollegala, T. Mu, and J. Y. Goulermas, "Cross-Domain sentiment classification using sentiment sensitive embeddings," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 2, pp. 398–410, Feb. 2016.

[26] M. Luck, T. Sylvain, H. Cardinal, A. Lodi, and Y. Bengio, "Deep learning for patient-specific kidney graft survival analysis," May 2017.

[27] G. Peyré and M. Cuturi, "Computational optimal transport," Mar. 2018.

[28] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy, "Optimal transport for domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 9, pp. 1853–1865, Sep. 2017.

[29] S. Chiappa and A. Pacchiano, "Fairness with continuous optimal transport," Jan. 2021.

[30] N. Si, K. Murthy, J. Blanchet, and V. A. Nguyen, "Testing group fairness via optimal transport projections," Jun. 2021.

[31] A. Quinn, M. Kárný, and T. V. Guy, "Fully probabilistic design of hierarchical bayesian models," *Inf. Sci. (Ny)*, vol. 369, pp. 532–547, Nov. 2016.

[32] M. Kárný, "Approximate bayesian recursive estimation," *Inf. Sci. (Ny)*, vol. 285, pp. 100–111, Nov. 2014.

[33] M. Varley and V. Belle, "Fairness in machine learning with tractable models," May 2019.

[34] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Stat.*, vol. 22, no. 1, pp. 79–86, Mar. 1951.

[35] J. R. Hershey and P. A. Olsen, in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*. IEEE, Apr. 2007.

[36] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, 1991.

[37] J. Larson, J. Angwin, L. Kirchner, and S. Mattu, "How we analyzed the COMPAS recidivism algorithm," https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm, May 2016, accessed: 2023-4-14.

[38] V. Grari, S. Lamprier, and M. Detyniecki, "Fairness without the sensitive attribute via causal variational autoencoder," Sep. 2021.

[39] Z. Zhu, Y. Yao, J. Sun, H. Li, and Y. Liu, "Weak proxies are sufficient and preferable for fairness with missing sensitive attributes," Oct. 2022.

[40] "Sklearn.Mixture.GaussianMixture," https://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html, accessed: 2023-4-13.

[41] "optimal-transport-fairness: Python implementation and theoretical summary of the paper "obtaining fairness using optimal transport theory" (eustasio del barrio, fabrice gamboa, paula gordaliza, Jean-Michel loubes - https://arxiv.org/abs/1806.03195)."