

國立臺灣大學電機資訊學院電子工程學研究所
碩士論文

Graduate Institute of Electronics Engineering
College of Electrical Engineering and Computer Science
National Taiwan University
Master Thesis

Fairness Model in Neural Network Framework
神經網路架構下的公平模型

Divya Jain
狄雅茵

Advisor: Jie-Hong Roland Jiang, Ph.D.

指導教授:江介宏 博士

中華民國 109 年 8 月

August, 2020

國立臺灣大學碩士學位論文
口試委員會審定書

神經網路架構下的公平模型

Fairness Model in Neural Network Framework

本論文係狄雅茵君 (R07943158) 在國立臺灣大學電子工程學研究所完成之碩士學位論文，於民國109年08月14日承下列考試委員審查通過及口試及格，特此證明

口試委員：

江介宏

(指導教授)

王柏宏

邵子

系主任、所長

林孝男

Fairness Model in Neural Network Framework

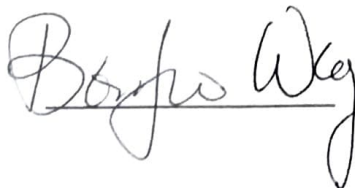
By
Divya Jain

THESIS

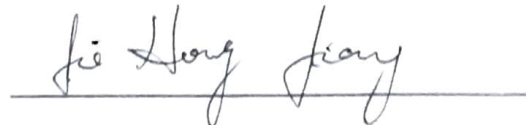
Submitted in partial fulfillment of the requirement
for the degree of Master of Science in Electronics Engineering
at National Taiwan University
Taipei, Taiwan, R.O.C.

August, 2020

Approved by :

A handwritten signature in black ink, appearing to read 'Bo-fu Wang', written over a horizontal line.A handwritten signature in black ink, appearing to read 'Key Yen', written over a horizontal line.

Advised by :

A handwritten signature in black ink, appearing to read 'Jie Hong Jang', written over a horizontal line.

Approved by Director :

A handwritten signature in black ink, appearing to read 'J. H. Jang', written over a horizontal line.

Acknowledgements

First of all, I would like to thank my thesis advisor Jie-Hong Roland Jiang from National Taiwan University. Without his support, I could not have thought of doing this independent research so smoothly. He guided me with his careful advice and valuable suggestions due to which I could carry out some interesting research in the neural network domain.

I would also like to thank Prof. Fang Yu and Prof. Bow-Yaw Wang for their kind agreement to be on my oral defense committee, for spending their valuable time and giving worthful suggestions regarding the content of this work. Another word of appreciation goes to all the members of ALCom Lab for their insightful discussions and knowledge that made my work and life interesting and easy during my stay in the lab.

Last, but most, I would like to thank my family for their support in my MS journey. The time came when I felt low, but the immense support of my parents Devendra Jain, Babita Jain, sister Rajul Bothra and brother in law Rahul Bothra always kept me motivated. My brother Veer Jain and nephew Shaurya Bothra made me smile whenever I felt lost in my journey. I also thank my friends for all their great support.

Sincerely

Divya Jain

National Taiwan University

August 14th, 2020

Fairness Model in Neural Network Framework

Student: Divya Jain Advisor: Jie-Hong Roland Jiang Ph.D

Graduate Institute of Electronics Engineering

National Taiwan University

Abstract

The new area of research in algorithmic fairness has gained attention over the past few years. In recent years, it is noticeable that Artificial Intelligence(AI) replaces humans at many important decision points, such as who will get hired or who will get a loan. One might assume that these decision-taking algorithms are bias-free but that is not the case. For example, a travel aggregator steers Mac users to more expensive hotels, and risk assessment software employed in criminal justice exhibits race-related issues. Hence, fairness in machine learning models is the demand of time.

Broadly speaking, fairness measures can be divided into two categories - Individual fairness and Group Fairness. Many methods have been proposed to enforce fairness in machine learning tasks while not compromising with accuracy. Three widely used techniques are pre-processing, in-processing and post processing methods which results in fairness. Many fairness metrics such as Disparate Impact, Disparate Mistreatment, Equalised odds are taken into account to ensure fairness and remove unwanted bias. Fairness aware machine learning algorithms seek to provide methods under which the predicted outcome is fair or non-discriminatory

for certain protected attributes such as race, gender, religion, etc. also known as sensitive attributes.

In this thesis, we propose a model in the neural network framework i.e gradient-based learning approach using mathematical definitions of fairness metrics in our optimization goal to enforce fairness in our model. We also consider multiple sensitive attributes to ensure fairness for each of them simultaneously. To ensure the validity of our approach, we draft another neural network model with the constraints used in previous works. This is done to show that it is not necessary to introduce new definitions of fairness as done in previous works until they differ fundamentally from the existing ones. With this work, we hope that it could lay down a new approach to achieving fairness in classification through neural networks.

Keywords: Disparate Impact; Disparate Mistreatment; Neural Network; Algorithmic Fairness; Sensitive Attribute; Equalized Odds

神經網路架構下的公平模型

研究生:狄雅茵

指導教授: 江介宏

博士

國立臺灣大學電子工程學研究所

摘要

過去數年間，演算法公平性的研究逐漸受到關注。近年來，我們可以發現到人工智慧(AI)逐漸在許多重要的工作中取代人類進行決策，像是決定誰會被雇用或是誰可以成功貸款等。人們可能認為這些演算法所進行的決策都是無偏差的，然而事實不然。舉例來說，線上旅遊網站會引導Mac用戶前往較貴的飯店，以及刑事司法所採用的風險評估軟體存在著種族相關的偏差。因此，機器學習中的公平性是當前的一大需求。大體來說，公平性可分為個體公平性以及群體公平性。前人已經提出很多方法可以在機器學習任務中達到公平性，同時保持的正確率。許多公平性的指標，例如差別影響、差別對待、均等概率，皆需要被考慮以達到公平性。公平性感知機器學習演算法試圖找出方法，在某些特定的保護屬性或敏感屬性下，如種族、性別、宗教等，得以使習得模型產生公平、無偏差的預測結果。

在這篇論文中，我們提出了2種神經網路架構下的模型，利用替代限制及直接的公平性限制進行梯度學習，並達到公平性。我們也考慮了多個敏感屬性，並確保了它們的公平性。此模型也達到了群體公平性的不同標準。透過這項研究，我們希望可以提出一個新的方法達成神經網路分類任務之公平性。

關鍵字: 差別影響、差別對待、神經網路、演算法公平性、敏感屬性、均等概率

Contents

Verification Letter from the Oral Examination Committee	i
Acknowledgements	iii
Abstract	iv
Chinese Abstract	vi
List of Figures	x
List of Tables	xii
1 Introduction	1
1.1 Motivation	1
1.1.1 Considering Multiple Sensitive Attributes	2
1.2 Cause of Unfairness	3
1.3 Our Contributions	4
1.4 Thesis Organization	5
2 Background	6
2.1 Machine Learning Nomenclature	6
2.2 Fairness Nomenclature	8

2.3	Mathematical Notation	9
2.4	Types of Fairness	10
2.5	Notions of Fairness	12
2.6	Bias Mitigation Strategy	15
3	Neural Network for Classification	18
3.1	Architecture of Neural Network	18
3.1.1	Define Neural Network Model Parameters	18
3.2	Forward Propagation	22
3.3	Backward Propagation	23
4	Fairness in Classification	27
4.1	Classifier with Low Disparate Impact	28
4.2	Classifier with Low Disparate Mistreatment	29
4.3	Fairness Constraints for Classification	30
4.3.1	Mitigating Disparate Impact	31
4.3.2	Mitigating Disparate Mistreatment	31
4.3.3	L2 Regularization Technique and Total Loss Function	32
4.4	Direct Constraints Loss function - Our Model of Fairness	34
4.4.1	Loss Function for Disparate Impact	35
4.4.2	Loss Function for Disparate Mistreatment	36
4.4.3	Intuition behind Having Direct Fairness Constraints	38
4.5	Training Algorithm	38
5	Fairness Model for Multiple Sensitive Attributes	40
5.1	Fairness Approach for Multiple Sensitive Attributes	41

5.1.1	Mitigating Disparate Impact and Disparate Mistreatment Si-	
	multaneously	43
6	Experimental Results	45
6.1	Datasets	47
6.2	Results	48
6.2.1	Validity of Our Fairness Model	50
6.2.2	Adult Dataset	50
6.2.3	COMPAS data set:	55
6.2.4	NYPD data set:	57
6.2.5	Bank data set:	58
6.2.6	German Credit data set:	59
6.2.7	Multiple Sensitive Attribute Results	61
6.2.8	Mitigating Disparate Impact and Disparate Mistreatment Si-	
	multaneously	63
7	Discussions and Analysis	64
8	Conclusions and Future Work	69
	Bibliography	71

List of Figures

2.1	Overview of Classification.	7
2.2	Fairness and Classification.	9
2.3	Illustration of Individual Fairness ¹	11
2.4	Illustration of Group Fairness.	12
3.1	A 4-layer Neural Network Model	19
4.1	Overview of different Misclassification Definition	29
6.1	Comparison of Our fairness model with baseline model and covariance based fairness model for three fairness metrics on Adult Dataset . . .	51
6.2	Comparison of Our fairness model with baseline model and covariance based fairness model for three fairness metrics on COMPAS Dataset	51
6.3	Comparison of Our fairness model with baseline model and covariance based fairness model for three fairness metrics on NYPD Dataset . .	52
6.4	Comparison of Our fairness model with baseline model and covariance based fairness model for three fairness metrics on Bank Dataset . . .	52
6.5	Comparison of Our fairness model with baseline model and covari- ance based fairness model for five fairness metrics on German Credit Dataset	53

List of Figures

6.6	Curve between accuracy and disparate Impact for Adult Data set for Fairness Model 1 and Fairness Model 2	54
6.7	Curve between accuracy and disparate Impact for COMPAS Data set for Fairness Model 1 and Fairness Model 2	56
6.8	Curve between accuracy and disparate Impact for NYPD Data set for Fairness Model 1 and Fairness Model 2	57
6.9	Curve between accuracy and disparate Impact for Bank Data set for Fairness Model 1 and Fairness Model 2	59
6.10	Curve between accuracy and disparate Impact for German Credit Data set for Fairness Model 1 and Fairness Model 2	61

List of Tables

2.1	Notions of Fairness	13
2.2	Decision of three classifiers based on sensitive and non sensitive at- tributes	15
6.1	Negative (low income) and Positive (high income) labels with respect to gender attribute for adult dataset	53
6.2	BM model results on adult dataset	54
6.3	Comparison of three models of this work with Previous Works on Adult dataset	55
6.4	Positive (Yes) and Negative (No) labels with respect to race attribute for COMPAS dataset	55
6.5	Comparison of three models of this work with Previous Works on COMPAS dataset	56
6.6	Positive (Yes) and Negative (No) labels in original NYPD dataset . .	57
6.7	Comparison of three models of this work with Previous Works on NYPD dataset	58
6.8	Positive (Yes) and Negative (No) labels in Bank dataset	59

List of Tables

6.9	Comparison of three models of this work with previous works on Bank dataset	60
6.10	Positive (Yes) and Negative (No) labels in German Credit dataset . .	60
6.11	Comparison of three models of this work with previous works on German Credit dataset	61
6.12	Comparison of baseline model with fairness constrained model in multiple sensitive attribute case	62
6.13	Results on five datasets for mitigating DI and DM simultaneously. . .	63
7.1	Capabilities of different methods in handling Disparate Impact (DI), Disparate Mistreatment (DM) and multiple sensitive attributes separately.	65

Chapter 1

Introduction

1.1 Motivation

The knowledge discovery process is continuously more relying on data-driven approaches. It is nothing new that Artificial Intelligence (AI) has replaced humans at many important decision-making places, from something as small as how to grant free tickets of a movie to more consequential ones like deciding whether to hire a person or grant a loan, a large number of decisions are automated.

However, despite reduced manual labor, improved efficiency, and user experiences, some concerns such as fairness seem to arise. Some day to day life examples tells us the extent of unfairness caused by machines for example the algorithms used by Goldman Sachs to approve applications for Apple cards show alarming gender inequality [32]. Industrial facial recognition systems developed by Microsoft,

1.1. Motivation

Face++, and IBM, all show a significant disparity in accuracy across different races, where darker skin colors could cause up to 30% performance degradation [5]. Further examples include gender bias in word embeddings [4], racial bias in recidivism prediction [3], or prejudice in New York City’s stop and frisk policy [16] [15].

The requisite to enforce fairness to any algorithmic decision-making process is to pinpoint the component to which we believe to be critical or in other words protected or sensitive. Fairness in machine learning has not already raised concerns but brings along with it many ethical issues.

Fairness is motivated in many fields by national and international legislation. It is well known that the Universal Declaration of Human Rights, a milestone document in the history of human rights, emphasizes freedom from discrimination, and equality as basic human rights. Now, with increasing AI applications, it is an urge to have classifiers that are fair or non-discriminatory for certain features of data such as gender, race, marital status, or color of skin.

1.1.1 Considering Multiple Sensitive Attributes

A stark example of multiple attribute bias in deployed systems was discovered by Buolamwini and Gebru [5] who showed that several commercially available gender classification systems from facial image data had substantial intersectional accuracy disparities when considering gender and race, with darker-skinned women being the most misclassified group - having an accuracy drop of over 30% compared to lighter-skinned men.

In another example, Hart [19] notes that medical data, e.g, from randomized control

1.2. Cause of Unfairness

trials, are often biased in favor of white men and therefore any model trained on this data may exhibit healthcare inequalities. Ameh and Van Den Broek [2] conducted a study on the increasing risk of maternal death among ethnic minority women in the UK. They found that there was very limited data specifically for black and ethnic minority women who were born in the UK and hence emphasized the need for reliable statistics to understand the scale of the problem. So there are many times when we need to consider many attributes simultaneously. Hence, we consider multiple sensitive attributes in our model.

1.2 Cause of Unfairness

It is assumed that machine learning algorithms are objective since decision making is entirely based on user data. In a typical workflow procedure, an algorithm is shown a large amount of data from where it can learn, and then the decision-making process is defined by what it sees. However, any data given to the algorithm is describing, directly or indirectly, the choices that have already been made in society. If black defendants are at higher risk to be determined as false than white defendants, then an algorithm will learn the same from the data assuming it as true. This bias present in available training data creates a feedback mechanism where the algorithm will give unfair decisions based on what it has learned, enhance discrimination further in the society, and thus further taint the data used in the future.

1.3 Our Contributions

In this thesis, we present a novel method to obtain fairness in the neural network architecture. Since many decisions are made using AI in our day to day life, hence it is important to ensure fairness in these machine learning algorithms. We train neural network classifiers to enforce fairness as neural networks are flexible models and can handle large volumes of data. Also, very little work has been done to impose fairness in neural network classification. Our work can be described as:

1. We construct a loss function for neural network architecture which includes fairness and accuracy loss terms. We directly incorporate the mathematical definitions of fairness metrics as regularisation terms in our optimization goal to achieve fairness.
2. Many popular works use an indirect relationship between sensitive attribute information and outcome such as covariance, mutual information, Wasserstein distance, etc. to enforce fairness. We have a strong emphasis that these measures are just proxy to fairness and it is hard to formulate such relationships for every fairness metric. To prove this, we model a neural network with covariance as our regularisation term and compare with our model to show that using direct definitions of fairness constraints perform better.
3. We build an architecture that can be used to ensure maximum fairness when considering multiple sensitive attributes.
4. We evaluated our model on different metrics of fairness which could ensure

that our model is fair. Basically, in total we have considered five definitions of fairness.

1.4 Thesis Organization

This thesis is organized as follows.

Chapter 2 provides essential background knowledge of this work. It illustrates the important definitions related to machine learning and basic concepts of fairness. Chapter 3 describes the working of neural network, forward propagation and backward propagation steps in detail. In chapter 4, we formulate the fairness constraints and determine fairness loss function in detail. In chapter 5, we have considered multiple sensitive attributes and proposed an approach to calculate loss function and impose fairness with respect to each attribute. Chapter 6 shows the experimental results conducted on various datasets and their comparison with previous existing works. Chapter 7 gives a brief analysis of the algorithms previously used and how our framework stands out. Finally, Chapter 8 concludes this thesis.

Chapter 2

Background

2.1 Machine Learning Nomenclature

A machine learning algorithm processes a set of input data and aims to distinguish the patterns inside those data. Given a new data, it can detect those found patterns in the samples and give the predictions. Those predictions might be a classification, where input is assigned an assumed outcome class, or regression, where a continuous value is assigned.

Classifier A machine learning predictor which assigns a class to each data sample is known as a classifier. In the scope of fairness, this work is mostly focused on binary classification, i.e., predicting only two distinct classes.

Prediction and Ground Truth This machine learning nomenclature differentiates between what an individual belongs to and what the classifier predicts.

2.1. Machine Learning Nomenclature

Ground truth is the class to which a data sample belongs and prediction is the class predicted by the classifier.

True Positives and False Positive A true positive is a sample that was correctly classified to belong to the positive class, i.e. the ground truth corresponds to the positive class as well. If the ground truth would actually have been the negative class, then it is called a false positive.

True Negative and False Negative A true negative is a sample that was correctly classified to belong to the negative class i.e the ground truth also belongs to the negative class. If the ground truth would have been belong to the positive class, then it is called false negative.

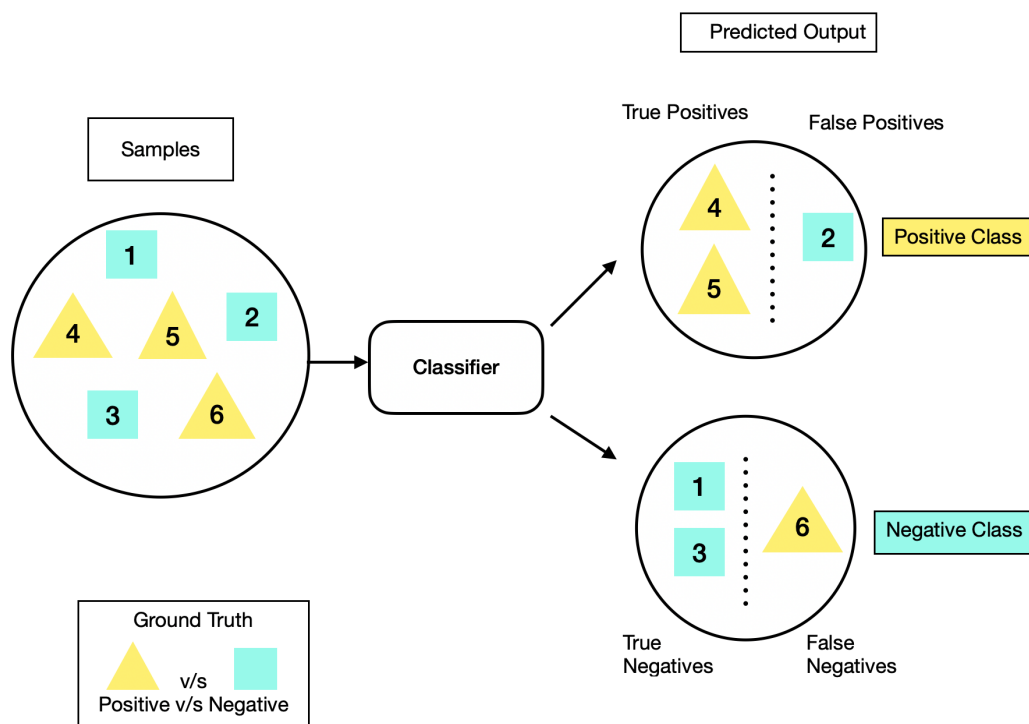


Figure 2.1: Overview of Classification.

2.2 Fairness Nomenclature

The common terminologies used in context of fairness are listed below -

Sensitive Attribute: A property of an individual that must not influence the decision process of the machine learning algorithm is called a sensitive attribute.

Typical examples include sex, race, religion, age, or caste.

Privileged and Unprivileged Group: Given a binary protected attribute like sex (male, female), the individuals over which decisions are made are divided into two demographic groups: sex is either male or female. Assuming discrimination against one group (i.e. females), the other group experiences a favorable treatment. The latter group is referenced as the privileged group, whereas the group experiencing discrimination is known as the unprivileged group.

Favourable and Unfavourable Outcome: In a binary classification scenario, the positive class corresponds to the favourable outcome the individuals wish to achieve, whereas the negative class corresponds to an unfavourable outcome respectively.

Qualified and Unqualified Individuals: The individual may be qualified for the favourable outcome if the ground truth of that individual belongs to the positive class.

2.3. Mathematical Notation

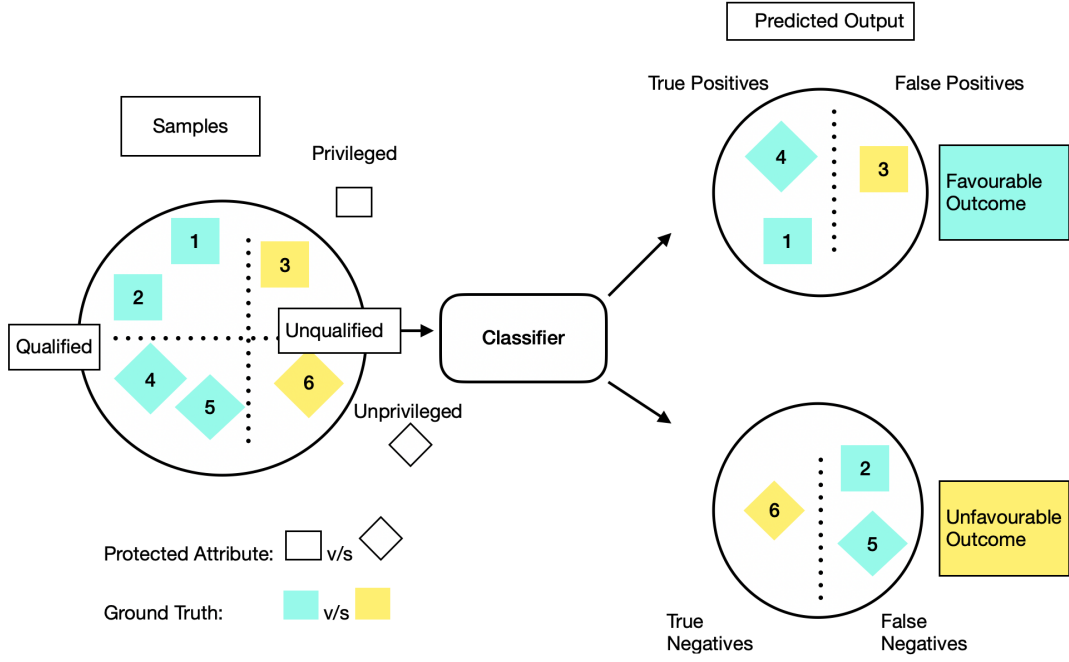


Figure 2.2: Fairness and Classification.

2.3 Mathematical Notation

We use many mathematical notations in this work. We denote $P(A|B) = P(A \cap B)/P(B)$ as the conditional probability of the event A happening given that B occurs. In the following work, we assume a finite dataset of n individuals D in which each individual is defined as a triple (X, Y, Z) :

-X are all attributes used for predictions regarding the data sample.

-Y is the corresponding ground-truth of the sample.

-Z is a binary protected attribute, $Z \in \{0, 1\}$, which is included in X and used by the classifier.

The favorable and unfavorable outcomes correspond to $Y = 1$ and $Y = 0$ accordingly.

A classifier is a mapping $f : X \rightarrow [0, 1]$, giving a score $S = f(X)$ which corresponds

2.4. Types of Fairness

to the predicted probability of an individual to belong to a positive class. For a given threshold value α the individual is predicted to belong to the positive class Y if $f(X) > \alpha$. The final prediction based on the threshold value is denoted as \hat{Y} with

$$\hat{Y} = 1 \Leftrightarrow f(X) > \alpha.$$

In this work, we assume binary classifiers and represent the probability that the favourable outcome will be predicted for the individuals from the privileged group (here we denote individuals with a feature belongs to privileged group) as

$$P(\hat{Y} = 1 \mid Z = a)$$

The another notation is,

$$P(Y = 0 \mid Z = b, \hat{Y} = 1)$$

which means that probability of a positively classified individual from the unprivileged group (individuals with sensitive feature b) is actually unqualified.

In the followings sections, we present the notions and algorithms of fairness using the mathematical notations described here.

2.4 Types of Fairness

Broadly fairness could be divided into two categories. Individual fairness which ensures similar decision outcomes for two individuals belonging to two different groups with respect to the sensitive feature and yet sharing similar non-sensitive

2.4. Types of Fairness

features. The other notion is of group fairness which requires different sensitive groups to receive beneficial outcomes in similar proportions.

Individual Fairness: It was first proposed by Cynthia Dwork in 2012 in his work Fairness Through Awareness [36], which is one of the most important contributions in the fairness domain. The notion of individual fairness is based on metrics above the individuals themselves, formulating a (D, d) -Lipschitz property. Let D be a distance metric over the room of possible classifications, and let d be a distance metric over individuals. Let $M: X \rightarrow \Delta(O)$ be a map that maps each individual to distributed outcomes. Then, a classifier is said to fulfill individual fairness if

$$D(M(x), M(x')) \leq d(x, x')$$

where x and x' denote individuals. That is, the distance of predicted outcomes must not be greater than the distance between the respective individuals in the first place was. In other words: similar individuals need to be similarly classified. .

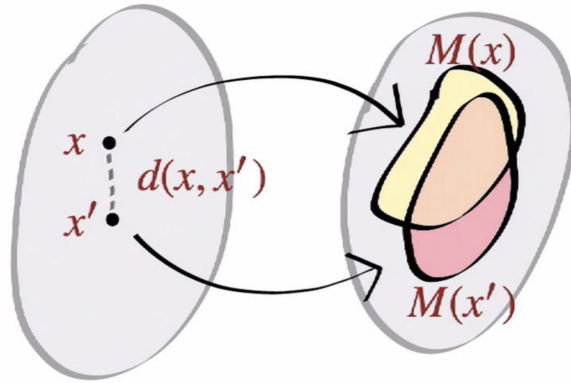


Figure 2.3: Illustration of Individual Fairness¹

Group Fairness: Group Fairness [11] requires the probability for an individual to

















¹<https://mrtz.org/nips17/#/94>


2.5. Notions of Fairness

be assigned the favourable outcome to be equal across privileged and unprivileged groups.

$$P_0[Y = 1] = P_1[Y = 1]$$

Here P_0 and P_1 refers to probability of individuals belonging to group with sensitive attribute $Z=0$ and 1 respectively.

Favourable Outcome				Unfavourable Outcome				Fair ?
								FAIR
								UNFAIR

 Privileged Group


 Unprivileged Group

Figure 2.4: Illustration of Group Fairness.

In our work, we focus mainly on group fairness.

2.5 Notions of Fairness

To achieve a fair classifier, a metric on how fairness is measured is needed first. In all cases based on sensitive attribute $Z \in [a, b]$, the fairness notions can be classified under three categories:

It should be noted that the first definition under Unawareness title cannot be used in our work as Disparate Treatment does not take into account the sensitive attribute information but our model needs this information during the training of neural

2.5. Notions of Fairness

S.no	Definition	Formula
Unawareness - not taking into account the sensitive attribute		
1	Disparate Treatment [31]	$P[Y'=y] = P[Y'=y Z=a]=P[Y'=y Z=b]$
Definitions Based on Predicted Outcome		
1	Disparate Impact [13]	$P[Y'=1 Z=a]-P[Y'=1 Z=b]$
Definitions Based on Predicted and Actual Outcome		
1	Predictive Parity (True Positive Rate) [31]	$P[Y'=1 Y=1, Z=a] = P[Y'=1 Y=1, Z=b]$
2	Predictive equality (False Positive Rate) [31]	$P[Y'=1 Y=0, Z=a] = P[Y'=1 Y=0, Z=b]$
3	Equalized Opportunity (False Negative Rate) [31]	$P[Y'=0 Y=1, Z=a] = P[Y'=0 Y=1, Z=b]$
4	Equalised Odds (True Positive Rate and False Positive Rate) [31]	$P[Y'=1 Y=1, Z=a] = P[Y'=1 Y=1, Z=b]$ $P[Y'=1 Y=0, Z=a] = P[Y'=1 Y=0, Z=b]$
5	Conditional Use Accuracy Equality (True Positive Rate and True Negative Rate) [31]	$P[Y'=1 Y=1, Z=a] = P[Y'=1 Y=1, Z=b]$ $P[Y'=0 Y=0, Z=a] = P[Y'=0 Y=0, Z=b]$

Table 2.1: Notions of Fairness

network. Based on table 2.1, we can explicitly define two main definitions widely used in other research works. These two definitions are basically included in our work as fairness constraints and they try to achieve most of the definitions stated above in table 2.1. The definitions under title predicted and actual outcome are basically covered as disparate mistreatment in our work and definitions under the title predicted outcome are covered in Disparate Impact. So broadly speaking, we try to meet five standard definitions which covers different notions of fairness. They are - Disparate Impact, True Positive Rate, True Negative Rate, False Positive Rate and False Negative Rate.

Disparate Impact: a decision making process suffers from disparate impact

2.5. Notions of Fairness

if it grants disproportionately large fraction of beneficial (or positive classification) outcomes to certain sensitive feature groups (e.g., men, women).

Disparate Mistreatment: a decision making process suffers from disparate mistreatment if its accuracy (or error rate) is different for different sensitive feature groups. The definitions listed in 2.1 under group-conditioned accuracy fall under disparate mistreatment.

Table 2.2 provides example of binary classifiers with respect to the two notions of fairness discussed above. This example is related to the granting of loan. Gender is the sensitive attribute in this case. In all cases, classifiers need to decide whether to grant loan or not based on employment details and age. The ground truth is shown which tells us that if actually the loan was granted or not.

C1 is unfair with respect to disparate impact because the fraction of males and females granted loan are different (1.0 and 0.66, respectively), where the latter gets discriminated from a decision of granting loan. C2 and C3 does not suffer from disparate impact because the fractions of males and females granted loan are same(0.66).

User Attributes			Ground Truth - Loan Granted	Classifier Decision to Grant Loan		
Sensitive	Non- Sensitive			Classifier 1	Classifier 2	Classifier 3
Gender	Employed	Age >25				
Male 1	1	1		1	1	1
Male 2	1	0		1	1	0
Male 3	0	1		0	1	1
Female 1	1	1		1	1	1
Female 2	1	0		0	1	1
Female 3	0	0		1	0	0

We deem classifiers C1 and C2 to be unfair due to disparate mistreatment since their rate of wrong decisions or accurate decisions for males and females are different.

2.6. Bias Mitigation Strategy

	Disparate Impact	Disparate Mistreatment
Classifier 1	Yes	Yes
Classifier 2	No	Yes
Classifier 3	No	No

Table 2.2: Decision of three classifiers based on sensitive and non sensitive attributes

C1 has different false negative rates for males and females (0.0 and 0.5, respectively), whereas C2 has different false positive rates (0.0 and 1.0) as well as different false negative rates (0.0 and 0.5) for males and females. Similarly, C1 has different true positive rate (1.0 and 0.5) for males and females, whereas C2 has different true negative rates (1.0 and 0.0) as well true positive rates (1.0 and 0.5) for males and females. Finally, classifier C3 does not suffer from disparate mistreatment because it has same false positive and false negative rates for males and females.

In this work, we consider disparate impact and disparate mistreatment as a metric of fairness in our models. In the coming chapters, we would introduce the mathematical equations for these notions.

2.6 Bias Mitigation Strategy

There have been different strategies to mitigate bias in different components in the machine learning pipeline, which are called pre-processing, in-processing, and post-processing, respectively.

Pre-processing: Usually, the classifier is not the only problem; the dataset is also biased. In this technique, a transformation is applied to input data, so that most of the task-specific information is maintained while removing sensitive information.

2.6. Bias Mitigation Strategy

The goal is to pre-process the training data such that any classification algorithm trained on this data would generate unfairness-free outcomes. For example, Kamiran and Calders (2010) [24] propose a pre-processing technique that operates by first training an unconstrained classifier, and then duplicating the data points from the group with lower acceptance rate (as compared to the other group). This is done until the classification outcomes obtained are free of any disparate impact. Other famous works are by Feldman [13], Luong [27] etc.

In-processing: In-processing algorithms add additional fairness constraints to existing algorithms so that the learned models could simultaneously satisfy some performance measure (for example, accuracy) and fairness requirements. For example, the technique by Kamishima et al. (2011) [25] [12] which is only limited to a logistic regression classifier works by adding a regularization term in the objective that penalizes the mutual information between the sensitive feature and the classifier decisions. Some other famous works are Calders and Verwer [6], Goh [17], Zafar [34] etc.

Post-processing: Post processing maintains fairness by adjusting the predictive labels given by potentially unfair models so that the adjusted label distribution could satisfy fairness requirements. These strategies require the information regarding sensitive feature at the decision time, hence they cannot be used in cases where sensitive feature information is unavailable (e.g., due to privacy reasons) or prohibited from being used due to disparate treatment laws (Barocas and Selbst, 2016) [1]. Hardt et al. (2016) [18] and Corbett-Davies et al. (2017) [7] have used this method and they require access to the sensitive feature information at the decision time.

2.6. Bias Mitigation Strategy

In this work, we use in-processing technique to mitigate the biasness and obtain a fair model.

Chapter 3

Neural Network for Classification

In this chapter, we have introduced the neural network architecture, it's working, and the binary classification approach. Artificial Neural networks draw inspiration from biological neural networks in human brains. Same as the neural system in humans, ANN has the nodes and neurons which transmit information. In this chapter, we would define a simple four-layer neural network and realize the equations for the feed-forward network and backpropagation method.

3.1 Architecture of Neural Network

3.1.1 Define Neural Network Model Parameters

The Neural Network primarily consists of three layers -an input layer, hidden layer, output layer. [9]

3.1. Architecture of Neural Network

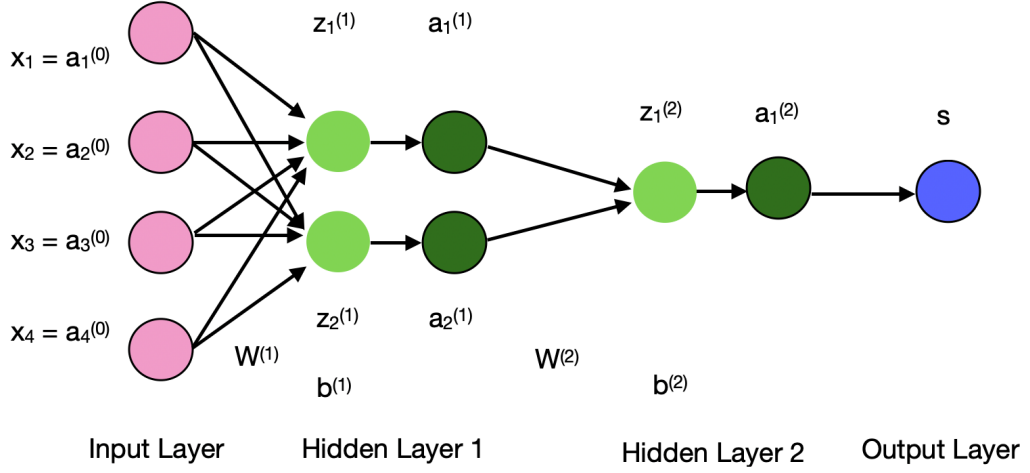


Figure 3.1: A 4-layer Neural Network Model

Input Layer: The neurons in pink color represent the input data. These can be scalar or vector matrices.

$$a_i^{(0)} = x_i$$

where $i \in [1, 2, 3, 4]$. The first set of activation are same as input.

Hidden Layer: The W^1 is the weight matrix for the first hidden layer. There are two neurons present in the first hidden layer. a^2 is the activation function for the first hidden layer or the second layer of our network. The equations of this layer can be written as :

$$z^{(1)} = W^{(1)} * a^{(0)} + b^{(1)}$$

$$a^{(1)} = f(z^{(1)})$$

Similarly, the same equations can be obtained for the second hidden layer.

$$z^{(2)} = W^{(2)} * a^{(1)} + b^{(2)}$$

3.1. Architecture of Neural Network

$$a^{(2)} = f(z^{(2)})$$

W^1 and W^2 are the weights in layers 2 and 3 respectively while b^1 and b^2 are the biases in those layers. Activations a^2 and a^3 are computed using an activation function f . Typically, this function f is non-linear (e.g. ReLU, sigmoid, tanh) and allows the network to learn complex patterns in data.

Let's pick layer 2 and its parameters as an example. The same operations can be applied to any layer in the network.

- W^1 is a weight matrix. It is of shape (n, m) where m is the number of output neurons i.e. neurons in the next layer and n is the number of input neurons i.e neurons in the previous layer. For us, $n = 2$ and $m = 4$.

$$W^1 = \begin{bmatrix} W_{11}^{(1)} & W_{12}^{(1)} & W_{13}^{(1)} & W_{14}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} & W_{23}^{(1)} & W_{24}^{(1)} \end{bmatrix} \quad (3.1)$$

- x is the input vector of shape $(m, 1)$ where m is the number of input neurons.

For us, $m = 4$.

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad (3.2)$$

- b^1 is a bias vector of shape $(n, 1)$ where n is the number of neurons in the

3.1. Architecture of Neural Network

current layer. Here we have $n = 2$.

$$b^1 = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \end{bmatrix} \quad (3.3)$$

Now we can compute the value of z^1 , we get

$$z^1 = \begin{bmatrix} W_{11}^{(1)} * x_1 + W_{12}^{(1)} * x_2 + W_{13}^{(1)} * x_3 + W_{14}^{(1)} * x_4 \\ W_{21}^{(1)} * x_1 + W_{22}^{(1)} * x_2 + W_{23}^{(1)} * x_3 + W_{24}^{(1)} * x_4 \end{bmatrix} + \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \end{bmatrix} \quad (3.4)$$

Output Layer: The last layer of the neural network is the output layer which predicts the output. In this example, we have one neuron which is blue colored to represent the predicted output.

$$s = f(W^{(3)} * a^{(3)})$$

3.2 Forward Propagation

The equations formed above form a forward propagation network. [33] [29]

$$a^{(0)} = x \quad \text{input layer} \quad (3.5)$$

$$z^{(1)} = W^{(1)} * a^{(0)} + b^{(1)} \quad \text{neuron value at hidden layer 1} \quad (3.6)$$

$$a^{(1)} = f(z^{(1)}) \quad \text{activation value at hidden layer 1} \quad (3.7)$$

$$z^{(2)} = W^{(2)} * a^{(1)} + b^{(2)} \quad \text{neuron value at hidden layer 2} \quad (3.8)$$

$$s = f(z^{(2)}) \quad \text{output layer} \quad (3.9)$$

The output or the ground truth y is part of the training dataset (x, y) . Here, x is the input. The final step in a forward propagation is to evaluate the predicted output s against an expected output y .

Evaluation between predicted output s and ground truth y is done through cost function. In our work, we use a Cross-Entropy loss function as this is preferred loss function under the inference framework of maximum likelihood. This loss function is intended for use where predicted labels $\in [0,1]$. We name it as Cost function C and denote it as:

$$C = \text{cost}(s, y)$$

The Cross Entropy Loss function is defined as:

$$C = -(y * \log(s) + (1 - y) * \log(1 - s)) \quad (3.10)$$

3.3. Backward Propagation

where y is ground truth and s is predicted label.

3.3 Backward Propagation

The backward propagation aims to minimize the cost function of the network by adjusting its weights and biases. The parameters (weights and biases) are adjusted by computing gradients of the cost function with respect to these parameters. One basic intuition beside this is to measure the sensitivity of cost function with respect to those parameters. The gradients show how much parameters need to change to minimize the cost function. Gradients are computed using the chain rule. [26]

In the following part of this section, we would compute the equations for gradients and use them for back-propagating to obtain the best parameters which could guarantee the low value of cost function. The update equations for the parameters look like:

$$W^{(l)} = W^{(l)} - \alpha * \frac{\partial C}{\partial W^{(l)}} \quad (3.11)$$

$$b^{(l)} = b^{(l)} - \alpha * \frac{\partial C}{\partial b^{(l)}} \quad (3.12)$$

Here alpha α is the learning rate of the network, which decides how big updates we perform to reach local minima.

Equation 3.11 and 3.12 tells us to compute derivative of cost function with respect to Weights and Biases. Hence, we can write derivative of cost function with respect to weights as:

$$\frac{\partial C}{\partial W^{(l)}} = \frac{\partial C}{\partial z^{(l)}} * \frac{\partial z^{(l)}}{\partial W^{(l)}} \quad (3.13)$$

3.3. Backward Propagation

We can write $z^{(l)}$ as

$$z^{(l)} = W^{(l)} * a^{(l-1)} + b^{(l)} \quad (3.14)$$

Taking derivative of 3.14 with respect to W , gives

$$\frac{\partial z^{(l)}}{\partial W^{(l)}} = a^{(l-1)} \quad (3.15)$$

Hence the 3.13 can be rewritten as:

$$\frac{\partial C}{\partial W^{(l)}} = \frac{\partial C}{\partial z^{(l)}} * a^{(l-1)} \quad (3.16)$$

Similar set of rules can be applied while calculating the derivative of cost function with respect to bias.

$$\frac{\partial C}{\partial b^{(l)}} = \frac{\partial C}{\partial z^{(l)}} * \frac{\partial z^{(l)}}{\partial b^{(l)}} \quad (3.17)$$

$$\frac{\partial z^{(l)}}{\partial b^{(l)}} = 1 \quad (3.18)$$

$$\frac{\partial C}{\partial b^{(l)}} = \frac{\partial C}{\partial z^{(l)}} * 1 \quad (3.19)$$

So, derivative of cost function with respect to z is the common terms in both the cases. The $\frac{\partial C}{\partial z}$ can be computed as follow:

$$\frac{\partial C}{\partial z} = \frac{\partial C}{\partial a} * \frac{\partial a}{\partial z} \quad (3.20)$$

Here a is activation layer and in our case, we use sigmoid function as the activation function. Hence, we need to calculate the derivative of sigmoid function with respect to z .

3.3. Backward Propagation

The mathematical expression for sigmoid function is:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (3.21)$$

If we notice, the input to sigmoid function is z . Hence, the derivative of sigmoid can be expressed as:

$$\frac{\partial a^l}{\partial z^l} = a^l * (1 - a^l) \quad (3.22)$$

Now, we can compute the derivative of cost function with respect to activation function. Our cost function as been defined in 3.10. We replace s with $a^{(2)}$ as s is the output from sigmoid function. We derive its derivative step by step in following equations.

$$\begin{aligned} C &= -(y * \log(a^{(2)}) + (1 - y) * \log(1 - a^{(2)})) \\ \frac{\partial C}{\partial a} &= -(y * \frac{\partial \log(a^{(2)})}{\partial a^{(2)}} + (1 - y) * \frac{\partial \log(1 - a^{(2)})}{\partial 1 - a^{(2)}}) \\ &= -(\frac{y}{a^{(2)}} - \frac{1 - y}{1 - a^{(2)}}) \end{aligned} \quad (3.23)$$

Where $*$ represents element-wise multiplication of the matrices, also known as the Hadamard product. Thus, 3.20 can be re-written as:

$$\frac{\partial C}{\partial z} = -(\frac{y}{a^{(l)}} - \frac{1 - y}{1 - a^{(l)}}) * a^l * (1 - a^l) = a^{(l)} - y \quad (3.24)$$

We can also obtain this derivative in the form of z . Equation 3.20 can be re-written

3.3. Backward Propagation

as:

$$\frac{\partial C}{\partial z^{(l)}} = \frac{\partial C}{\partial z^{(l+1)}} * \frac{\partial z^{(l+1)}}{\partial a^{(l)}} * \frac{\partial a^{(l)}}{\partial z} \quad (3.25)$$

$$z^{(l+1)} = W^{(l+1)} * a^{(l)} + b^{(l)} \quad (3.26)$$

$$\frac{\partial z^{(l+1)}}{\partial a^{(l)}} = W^{(l+1)} \quad (3.27)$$

Putting all the values together in eq. 3.25, we get

$$\frac{\partial C}{\partial z} = (W^{(l+1)T} \cdot \frac{\partial C}{\partial z^{(l+1)}}) * a^l * (1 - a^l) \quad (3.28)$$

Here '.' represents the matrix multiplication operation and '*' represents the element-wise product as above.

Chapter 4

Fairness in Classification

We will be dealing with two main definitions of fairness discussed in section 2.5. In the previous chapter, we have already seen how the backpropagation algorithm minimizes the cost function during the training of neural networks. We noticed that minimizing the cost function would lead us to obtain the parameters which could guarantee the maximum accuracy. When we talk about fairness in the machine learning, we need to look for such a model which could guarantee fairness but not at the cost of accuracy. So our cost function in the neural network should comprise of accuracy loss and fairness loss. We would then aim to find the best model parameters using the backpropagation algorithm to achieve our above task.

For a binary classifier, one aims to find a mapping between input data $x \in \mathbb{R}$ and predicted labels $Y' \in [0,1]$. We learned to develop the neural network model with the best parameters. When this model is used on the unseen test set, it is expected to achieve high accuracy.

4.1. Classifier with Low Disparate Impact

In the context of fairness in binary classification, each user is attached to a sensitive attribute $Z \in [a,b]$, and goal is to find the parameters which could have good accuracy and be devoid of any unfairness. Formally, we are looking to develop a classifier with no disparate impact and disparate mistreatment. In our model we use the mathematical definitions of disparate impact and disparate mistreatment as constraints to our optimization goal. We will see the approach to introduce these constraints in our neural network cost function in the following sections.

4.1 Classifier with Low Disparate Impact

A binary classifier is said to be free of any Disparate Impact (DI) if the probability of predicting the favorable output is the same for both values of a sensitive attribute.

Mathematically,

$$P(Y' = 1|Z = a) = P(Y' = 1|Z = b) \quad (4.1)$$

or in other words

$$DI = P(Y' = 1|Z = a) - P(Y' = 1|Z = b) \quad (4.2)$$

The lower the value of DI, the more fair the model is. If DI is equal to zero then we can say the probability that model classifies the favorable outcome is same for both the groups. We must also note that DI value lies in the range $[0,1]$. In our work, we aim to minimize the value of DI and bring it closer to zero.

4.2 Classifier with Low Disparate Mistreatment

A binary classifier is said to be free of any Disparate Mistreatment (DM) if the rate of misclassification (accuracy) is the same for both of the sensitive attribute. Figure 4.1 briefly gives an overview of different definitions of misclassification.

		PREDICTED LABEL		
		Y'=1	Y'=0	
True Label	Y =1	True Positive	False Negative	P(Y'≠Y Y=1) False Negative Rate
	Y =0	False Positive	True Negative	P(Y'≠Y Y=0) False Positive Rate
		P(Y'≠Y Y'=1) False Discovery Rate	P(Y'≠Y Y'=0) False Omission Rate	P(Y'≠Y) Overall Misclass. Rate

Figure 4.1: Overview of different Misclassification Definition

The different measures for misclassification can be expressed as follows:

Overall Misclassification Rate (OMR) :

$$P(Y' \neq Y|z = a) - P(Y' \neq Y|z = b) \quad (4.3)$$

False Positive Rate (FPR) :

$$P(Y' \neq Y|Y = 0, z = a) - P(Y' \neq Y|Y = 0, z = b) \quad (4.4)$$

4.3. Fairness Constraints for Classification

False Negative Rate (FNR) :

$$P(Y' \neq Y|Y = 1, z = a) - P(Y' \neq Y|Y = 1, z = b) \quad (4.5)$$

False Omission Rate (FOR) :

$$P(Y' \neq Y|Y' = 0, z = a) - P(Y' \neq Y|Y' = 0, z = b) \quad (4.6)$$

False Discovery Rate (FDR):

$$P(Y' \neq Y|Y' = 1, z = a) - P(Y' \neq Y|Y' = 1, z = b) \quad (4.7)$$

In our work, we are more focused on false positive rates and false negative rates. It must be noted that making the false positive rate close to zero also brings the true negative rate closer to zero. The same is the case with a false negative rate and a true positive rate. In the next sections, we would come up with the fairness constraints for these definitions.

4.3 Fairness Constraints for Classification

After going through the mathematical notations of fairness metrics, it makes it clear that our goal is to reduce the value of DI or DM_{FPR} or DM_{FNR} if we want to achieve a classifier free of any disparate impact or disparate mistreatment. In the last chapter, we learned that our main goal was to minimize the cost function. Hence, we can conclude that to obtain a fair classifier we need to minimize cost function

4.3. Fairness Constraints for Classification

subject to fairness constraints. Let's see how we can draft it in mathematical form.

4.3.1 Mitigating Disparate Impact

Equation 4.2 states that to obtain classifier free of disparate impact, we need to have low value of DI. The goal of optimization is:

$$\begin{aligned} &\textbf{minimize} \quad \textit{Loss Function} \quad L(W) \\ &\textbf{subject to} \quad \textit{DI constraints} \end{aligned}$$

or with reference to eq. 3.10

$$\begin{aligned} &\textbf{min} \quad C = -(y * \log(s) + (1 - y) * \log(1 - s)) \\ &\textbf{s.t.} \quad P(Y' = 1|Z = a) - P(Y' = 1|Z = b) \approx 0 \end{aligned}$$

When considering the neural network architecture, we need to include this constraint into our cost function so that we can minimize it all together. For this purpose, we used L2 regularisation norm and obtained a single loss function. In sub-section 4.3.3, we give a brief overview of L2 regularization.

4.3.2 Mitigating Disparate Mistreatment

Eq 4.3 , eq 4.4 and eq 4.5 gives us the definition of overall misclassification, false positive rates and false negative rates respectively explicitly. The goal of optimization

4.3. Fairness Constraints for Classification

is:

$$\begin{aligned} & \textbf{minimize} \quad \textit{Loss Function} \quad L(W) \\ & \textbf{subject to} \quad \textit{Disparate Mistreatment constraints} \end{aligned}$$

or with reference to eq. 3.10, eq.4.3 , eq 4.4 and eq 4.5

$$\begin{aligned} & \textbf{min} \quad C = -(y * \log(s) + (1 - y) * \log(1 - s)) \\ & \textbf{s.t.} \quad P(Y' \neq Y | Y = 0, z = a) - P(Y' \neq Y | Y = 0, z = b) \approx 0 \quad \textbf{FPR constraint} \\ & \textbf{s.t.} \quad P(Y' \neq Y | Y = 1, z = a) - P(Y' \neq Y | Y = 1, z = b) \approx 0 \quad \textbf{FNR constraint} \end{aligned}$$

When we are designing a classifier free of FPR, we can include FPR constraint in our optimization problem and when we looking for classifier free of FNR, we need to consider FNR constraint. In our work, we combine both FPR and FNR constarints together to get the classifier with less Disparate Mistreatment. We will learn later that how we can include more than one constraint in our optimization problem.

4.3.3 L2 Regularization Technique and Total Loss Function

Let's look at the main motivation behind the L2 regularization method. The most common form of regularization is based on Tikhonov regularization [30], with Euclidean norms, which results in a (convex) quadratic optimization problem. This is naturally described as a (convex) vector optimization problem with two objectives.

4.3. Fairness Constraints for Classification

We can understand this with a simple example shown below:

$$\begin{aligned} \mathbf{min:} \quad & L(w) \\ \text{st.} \quad & F(w) \leq t \end{aligned}$$

Using a Lagrange multiplier we can rewrite the problem as:

$$L(w) + \lambda * (F(w))^2 \tag{4.8}$$

where $\lambda \geq 0$.

A constrained optimization problem is said to be a convex optimization if both the objective function and the constraint are convex functions. In our case, the first objective function is the cross-entropy loss function which is a convex function, and we will calculate the double differentiation of our second term to see if it is convex or not. If we get it greater than zero, then we can prove it to be convex.

$$\begin{aligned} y &= (F(w))^2 \\ \frac{\partial y}{\partial w} &= 2 * F(w) \\ \frac{\partial^2 y}{\partial w} &= 2 > 0 \end{aligned}$$

This proves that our regularisation term is convex. We find for the value of parameter λ such that

$$\nabla L = \lambda * \nabla F$$

Hence, L2 regularization is a technique where the sum of squared parameters, or weights, of a model (multiplied by some coefficient) is added into the loss function

4.4. Direct Constraints Loss function - Our Model of Fairness

as a penalty term to be minimized.

$$\text{LossFunction} \quad L = L_A + \lambda * (L_F)^2 \quad (4.9)$$

We learned in the last chapter that how the back propagation algorithm works.

Since the Loss function is convex, so the gradient-based learning approach is used.

We noted that in Eq.3.11,

$$\begin{aligned} W^{(l)} &= W^{(l)} - \alpha * \frac{\partial L}{\partial W^{(l)}} \\ W^{(l)} &= W^{(l)} - \alpha * \frac{\partial (L_A + \lambda * (L_F)^2)}{\partial W^{(l)}} \\ W^{(l)} &= W^{(l)} - \alpha * \frac{\partial L_A}{\partial W^{(l)}} - \alpha * \lambda * \frac{\partial (L_F)^2}{\partial W^{(l)}} \end{aligned} \quad (4.10)$$

So, we already have computed $\frac{\partial L_A}{\partial W^{(l)}}$ in Eq.3.28. The second term is fairness loss and we will learn to calculate its derivative in the coming section. So the effect of the L2 penalty term (fairness loss term), is that on each optimization step in addition to stepping based on a gradient to better fit the training data.

4.4 Direct Constraints Loss function - Our Model of Fairness

In the last section, we learned that how we can add the fairness loss term (L_F) to our accuracy loss function (L_A) and obtain the total Loss function (L) which needs to be minimized during the training of the neural networks. In this section, we will learn to obtain an equation for fairness loss function by introducing the mathematical definitions of fairness notions as constraints in our optimization goal.

4.4.1 Loss Function for Disparate Impact

We have learnt that

$$DI = P(Y' = 1|Z = a) - P(Y' = 1|Z = b)$$

Broadly speaking, we have two terms in the above equation, the first term is the probability of users present in a group with sensitive feature a producing favourable outcome and the second term is the probability of users present in a group with sensitive feature b producing a favourable outcome. Let us suppose that

$$P(Y' = 1|Z = a) = p$$

$$P(Y' = 1|Z = b) = q$$

The logistic model has the form

$$p = \frac{1}{1 + e^{-(W^T \cdot X_a + b)}}$$

$$q = \frac{1}{1 + e^{-(W^T \cdot X_b + b)}}$$

where W is the weight parameters, b is the bias parameters, X_a is the group of all users with sensitive feature a and X_b is the group of all users with sensitive feature b .

Our job is to find a maximum likelihood estimate [20] [8] of the model parameters W and b . The p and q value above a threshold is considered to be a favourable

4.4. Direct Constraints Loss function - Our Model of Fairness

outcome. Here, we are considering the value of p and q to be either 0 or 1. So let there be n sample in X_a and m samples in X_b , the equation for Disparate Impact can be rewritten as:

$$DI = \frac{\sum_{n=1}^n p}{n} - \frac{\sum_{m=1}^m q}{m}$$

So our fairness loss L_F is defined as:

$$L_F = \frac{\sum_{n=1}^n \frac{1}{1+e^{-(W^T \cdot X_a + b)}}}{n} - \frac{\sum_{m=1}^m \frac{1}{1+e^{-(W^T \cdot X_b + b)}}}{m}$$

Since we had already learned the gradient descent of sigmoid function in the last chapter, so we will not discuss it here. But we can note that the partial derivative of L_F with respect to weight W and bias b is the same as 3.22.

4.4.2 Loss Function for Disparate Mistreatment

We have learnt the following definitions of false positive rate and false negative rates above.

$$DM_{FPR} = P(Y' \neq Y | Y = 0, z = a) - P(Y' \neq Y | Y = 0, z = b)$$

or

$$DM_{FPR} = P(Y' = 1 | Y = 0, z = a) - P(Y' = 1 | Y = 0, z = b)$$

Similarly,

$$DM_{FNR} = P(Y' \neq Y | Y = 1, z = a) - P(Y' \neq Y | Y = 1, z = b)$$

4.4. Direct Constraints Loss function - Our Model of Fairness

or

$$DM_{FNR} = P(Y' = 0|Y = 1, z = a) - P(Y' = 0|Y = 1, z = b)$$

In the above subsection, we have seen the method to get the probability of the users with certain sensitive attributes. In the case of FPR, we are looking for all those cases in which outputs have been misclassified to be favourable for the users with different sensitive attributes. Let X_a is the group of all users with sensitive feature a with n samples and X_b is the group of all users with sensitive feature b with m samples. Let p denotes the predicted output of all users belonging to group X_a and q denotes the predicted output of all users belonging to group X_b . Then, FPR can be written as:

$$DM_{FPR} = \frac{\sum_{n=1}^n p * (1 - Y)}{n} - \frac{\sum_{m=1}^m q * (1 - Y)}{m}$$

Similarly, FNR can be written as:

$$DM_{FNR} = \frac{\sum_{n=1}^n (1 - p) * (Y)}{n} - \frac{\sum_{m=1}^m (1 - q) * Y}{m}$$

It should be noted that ground truth information is needed when dealing with disparate mistreatment. However, this information is not needed when including the constraint for disparate impact. Also, since we have calculated the equations for FPR and FNR separately, hence, we can include them in our loss term either separately or as a sum of both.

$$L_F = DM_{FPR} + DM_{FNR}$$

4.5. Training Algorithm

4.4.3 Intuition behind Having Direct Fairness Constraints

Neural networks provide us with the flexibility to learn and implement various tasks. Since we are dealing with the definition of group fairness, hence we can approximate the fairness measure using gradient descent batch learning. So if we have any loss function which can be minimized using a gradient descent approach, it makes the task easier. In this work, we have used direct constraints that are convex in nature and can be included in our loss function. The cross-entropy function is convex and we used the L2 regularization method to include penalty term i.e. fairness loss to obtain total loss function.

$$\text{LossFunction} \quad L = L_A + \lambda * (L_F)^2 \quad (4.11)$$

Section 4.3.3 clearly states that the above equation can easily be subjected to gradient based learning. Further we also draft another neural network with covariance between sensitive attribute and predicted outcome [35] as the constraint to our cross entropy loss function. We note that introducing new measures of fairness just serve as proxy and cannot guarantee fairness.

4.5 Training Algorithm

We define our training procedure in Algorithm 1 below. We performed our experiments in python. We computed the total loss function as stated above.

4.5. Training Algorithm

Algorithm 1 Training Procedure for Neural Network with Fairness Constraints as Regularization Terms

Input: Training observations (X_i, \mathbf{Z}, Y_i) , $i \in (1, n)$, $\mathbf{Y} \in (0, 1)$

Output: return a trained neural network f_w

```
1: for e in Epochs do
2:   for b in batch do
3:     Do Forward Propagation as described in 3.2
4:     Compute Fairness Loss  $F_L$ 
5:     Compute Cross Entropy Loss  $F_A$ 
6:     Compute Total Loss according to eq. 4.9
7:     Back Propagate using details in subsection 4.4.1 and 3.22
8:     Update the network parameters  $\mathbf{W}$  and  $\mathbf{b}$  according to eq. 4.10
9:   end for
10: end for
11: Return the neural network  $f_w$ 
```

Chapter 5

Fairness Model for Multiple Sensitive Attributes

It is well-known fact that only a few fairness-aware algorithms can formally handle multiple sensitive attributes directly. However, it must be noted that all existing algorithms can handle them if only pre-processed and combined into a single sensitive attribute (e.g., race-sex).

Consider the following example: Let us consider a dataset with two binary features, one is gender (say male and female) and the other is race (say white and black), both of which are distributed independently and evenly at random in a dataset. Consider a classifier that gives a favorable outcome if and only if it corresponds to a white man or a black woman. Then the classifier will appear to be fair when only one protected attribute is considered, in the sense that it labels both men and women as positive with equal probability, and the same for labels of black and

5.1. Fairness Approach for Multiple Sensitive Attributes

white individuals. But if one looks at two attributes (such as black women) at the same time, then the classifier maximally violates the fairness constraint. Hence, we need to look for fairness criteria when handling multiple attributes simultaneously.

Previously, the conjunction of two attributes has been done to handle multiple sensitive attributes. However, we might expect combining the attributes in this way to degrade performance under some metrics, especially in the case of algorithms that can only handle binary sensitive attributes, or when there are too many combinations for the dataset to provide a large group of people with each new combined sensitive value. The comparative study done by [14] on the Adult dataset by combining two attributes gives varying results on different algorithms. They showed that sex is especially predictive of the Adult Income data set, so the value of DI for sex is quite low. Race generally receives a higher DI value from these algorithms. When combining both at once, all of the algorithms find that the DI value is somewhere in between that for the race and that for sex. Hence, in this chapter, we provide a method to handle multiple sensitive attributes simultaneously without pre-processing them.

5.1 Fairness Approach for Multiple Sensitive Attributes

In the last chapter, we learned that we include fairness constraints when we deal with the neural networks to achieve accuracy along with fairness. Let us recap the fairness constraints learned previously. Equation 4.2 states that to obtain a classifier

5.1. Fairness Approach for Multiple Sensitive Attributes

free of disparate impact, we need to have a low value of DI. The goal of optimization is:

$$\textbf{minimize} \quad \textit{Loss Function} L(W)$$

$$\textbf{subject to} \quad \textit{DI constraints}$$

or with reference to eq. 3.10

$$\textbf{min} \quad C = -(y * \log(Y) + (1 - y) * \log(1 - Y))$$

$$\textbf{s.t.} \quad P(Y' = 1|Z1 = a) - P(Y' = 1|Z1 = b) \approx 0$$

where, y is ground truth, Y is predicted label and $Z1$ is sensitive attribute having two values a and b . Now, we can simply add another constraint to our above optimization problem. So our second constraint becomes

$$\textbf{s.t.} \quad P(Y' = 1|Z2 = c) - P(Y' = 1|Z2 = d) \approx 0$$

where, $Z2$ is our second sensitive attribute with values c and d . This way we have included another constraint in our optimization problem. Previously we used L2 regularization to obtain our loss function. Now we can also relate our problem as multi-objective problem which has three objectives - minimize the cross-entropy loss function, minimize the disparate impact of sensitive attribute $Z1$ and minimize the disparate impact of sensitive attribute $Z2$, and for this, we need to obtain a hybrid loss function. This is simply a Lagrangian problem with two Lagrangian multipliers.

5.1. Fairness Approach for Multiple Sensitive Attributes

Let L_A , L_{F1} and L_{F2} be the three objective functions - cross-entropy loss function, fairness loss due to the first sensitive attribute, and fairness loss due to the second sensitive attribute. We introduce new variables λ and α for each constraint, these are called the Lagrangian multipliers. The generalized Lagrangian is then defined as:

$$L(\lambda, \alpha) = L_A + \lambda * L_{F1} + \alpha * L_{F2} \quad (5.1)$$

The optimal values of λ and α can be computed by using following equation:

$$\nabla L_A = \nabla \lambda * L_{F1} + \nabla \alpha * L_{F1}$$

We can now solve a constrained multi-objective minimization problem using the unconstrained optimization of the generalized Lagrangian. Deciding the correct values of λ and α gives us the best neural network parameters for L_A .

So, with this approach, we can include more sensitive attributes in our optimization problem. However, in this work, we are concentrated only on two sensitive attribute.

5.1.1 Mitigating Disparate Impact and Disparate

Mistreatment Simultaneously

In this chapter, we have learned that we can optimize a function subject to multiple constraints. In this work, we have tried to formulate an approach to remove

5.1. Fairness Approach for Multiple Sensitive Attributes

mitigating disparate impact and disparate mistreatment simultaneously.

$$\begin{aligned} &\textbf{minimize} && \textit{Loss Function} && L(W) \\ &\textbf{subject to} && \textit{Disparate Mistreatment constraints} \\ &\textbf{subject to} && \textit{Disparate Impact constraints} \end{aligned}$$

We had already learned the disparate impact loss function and disparate mistreatment loss function in the last chapter. Let us represent disparate impact loss function and disparate mistreatment loss function as L_{DI} and L_{DM} respectively.

Then using the Lagrangian multiplier, we can write our optimized goal as:

$$L(\lambda, \alpha) = L_A + \lambda * L_{DI} + \alpha * L_{DM} \tag{5.2}$$

Here, L_A represents the accuracy loss of the classifier denoted by cross-entropy loss function used here.

Chapter 6

Experimental Results

In this chapter, we perform experiments on different real world data sets where we consider single and multiple sensitive attribute. We consider a simple 2 hidden layer feed forward network. We consider two different models on the basis of fairness loss term. The first model is based on using covariance as constraints. This model is drafted just to show that using surrogate functions (covariance, mutual information) just serves as proxy to fairness model. It must be noted that many existing in-processing techniques use such surrogate loss functions. Hence, through the experiments and implementation of first model we have tried to prove an upper edge of our model (Direct Fairness Constraint Model) over other existing models. The second model consists of fairness loss with constraints based on definitions of fairness metrics as described in section 4.4. The experiment results consists of primarily three models:

Baseline-BM: This model evaluates the dataset without considering fairness con-

straints.

Covariance Fairness Constraint -CFM: This model considers covariance as the fairness constraints i.e first model stated above.

Direct Fairness Constraint -DFM: This model considers mathematical definitions of fairness metrics introduced as constraints in our optimization goal i.e second model stated above.

The main challenge in above approach is that it involves many hyper-parameters. It is an exhaustive task to tune all of them. Hyper-parameters are sometimes specific to the data set used. They also change with the degree of strictness of the constraint enforced.

In this chapter, we show the comparison of our model with well known fairness aware machine learning models - Zafar¹ in-processing method, Feldman² Method and Kamishima³ Method. Every comparison is done with two models of Zafar - one is baseline model which is unconstrained and other is Zafar sensitive attribute model where fairness constraints are taken into consideration, fairness constrained model of Feldman and Kamishima and adversarial method by Zhang ⁴. To ensure fairness in comparison, we run the models used here for comparison in their default settings on our computer. We included Wasserstein Model trained on Neural Network in our experimental results. However, due to unavailability of their code, we cite the results from their paper [28] for adult dataset.

¹<https://github.com/mbilalzafar/fair-classification>

²<https://github.com/algofairness/fairness-comparison/tree/master/fairness/algorithms/feldman>

³<https://github.com/algofairness/fairness-comparison/tree/master/fairness/algorithms/kamishima>

⁴https://github.com/Trusted-AI/AIF360/blob/master/aif360/algorithms/inprocessing/adversarial_debiasing.py

6.1 Datasets

We conducted experiments on five real world data sets described below:

Adult Dataset: The Adult Data (Census Income Dataset) [10] consists of 45222 samples. It has 14 features which include information such as relationship status, education level, or occupation, as well as the sensitive attributes race, sex, and age. The associated prediction task is to classify whether an individual’s income exceeds \$50,000 annually. The sensitive attribute used here are gender for single sensitive attribute model and gender and race when considering sensitive multiple attributes.

ProPublica Recidivism: The ProPublica recidivism data set [21] includes data from the COMPAS risk assessment tool and was analysed by Angwin et al. [3] to show COMPAS race bias. It contains 5728 individuals and encodes the sensitive attributes race, sex, and age. The prediction task outcome is whether an individual was rearrested within two years of the first arrest. A violent recidivism version exists where the outcome is a rearrest within two years on basis of a violent crime.

NYPD stop-question-and-frisk (SQF) dataset: The NYPD SQF dataset [23] consists of 84,869 pedestrians who were stopped in the year 2012 on the suspicion of having a weapon. The dataset also contains over 100 features (e.g., gender, height, reason for stop) and a binary label which indicates whether (negative class) or not (positive class) a weapon was discovered. For our analysis, we consider the race to be the sensitive feature with values blacks and whites.

German Credit Dataset: The German Credit Data consist of 1000 samples

6.2. Results

with 20 features of each user sample. It contains features such as employment time, current credits, or marital status. The prediction task is to determine whether an individual has good or bad credit risk. Sensitive attributes are sex and age.

Bank Dataset: This dataset [22] has 41,188 data points, each with 20 attributes. The label is binary which indicates whether that person (data point) has subscribed or not to a term deposit. The age is considered as the binary sensitive variable, individuals between 25 and 60 years of age form one group the rest form the other group.

6.2 Results

We have seen different fairness metrics in 2.1 and will use them to evaluate the fairness. Z is the sensitive attribute with binary values a and b . Y' and Y are predicted outcome and ground truth respectively.

- **Disparate Impact (DI):**

$$|P(Y' = 1|Z = a) - P(Y' = 1|Z = b)|$$

Lower the value of DI, the more fair the model is.

- **Predictive Parity (True Positive Rate):**

We calculate the difference of favorable outcomes of each group given their

6.2. Results

ground truth is true.

$$P[Y' = 1|Y = 1, S = a] - P[Y' = 1|Y = 1, S = b]$$

It is clear that the difference of true positive rate should be more close to zero

to make the model fairer. It should also be noted that

Predictive Parity (True Positive Rate) + Equalized Opportunity(False Negative Rate) =0

Hence, we only show $DM_{(FNR)}$ value in the experimental result table.

- **Predictive Equality(False Positive Rate):**

We calculate the difference of favorable outcomes of each group given their ground truth is false.

$$P[Y' = 1|Y = 0, S = a] - P[Y' = 1|Y = 0, S = b]$$

In our work, we denote it as Disparate Mistreatment (FPR) or DM_{FPR} . Lower the value of DM_{FPR} , more fair is the model.

- **Equalized Opportunity(False negative Rate):**

We calculate the difference of unfavorable outcomes of each group given their ground truth is true.

$$P[Y' = 0|Y = 1, S = a] - P[Y' = 0|Y = 1, S = b]$$

In our work, we denote it as Disparate Mistreatment (FNR) or DM_{FNR} . Lower the value of DM_{FNR} , more fair is the model.

6.2. Results

- **Equalised Odds:** Equalised Odds is the conjunction of true positive rate and false positive rates. Since we have already taken in account the FPR and TPR, therefore we don't take into account this in our table.
- **Conditional Use Accuracy Equality:** It is the conjunction of true positive rate and true negative rates. True Negative Rate is defined as:

$$P[Y' = 0|Y = 0, S = a] - P[Y' = 0|Y = 0, S = b]$$

. It should be noted that sum of True Negative rate difference for both attributes and False Positive Rate difference for both the attributes is equal to zero. Hence, we just show $DM_{(FPR)}$ in the experimental result table.

6.2.1 Validity of Our Fairness Model

In this section we prove the validity of our approach. As we discussed before that defining new terms for fairness and using them as regularisation terms or fairness penalty may seem to be a just proxy for fairness. To prove above statement, we train a neural network model and use covariance as regularisation term. We compare this with our model and evaluate it on five datasets on five different fairness metrics.

6.2.2 Adult Dataset

The adult dataset is divided into train and test set. Table 6.1 shows the distribution of sensitive features and class labels in adult dataset used in the evaluation. The detailed results of my BM model listed in table 6.2 below: We can easily calculate

6.2. Results

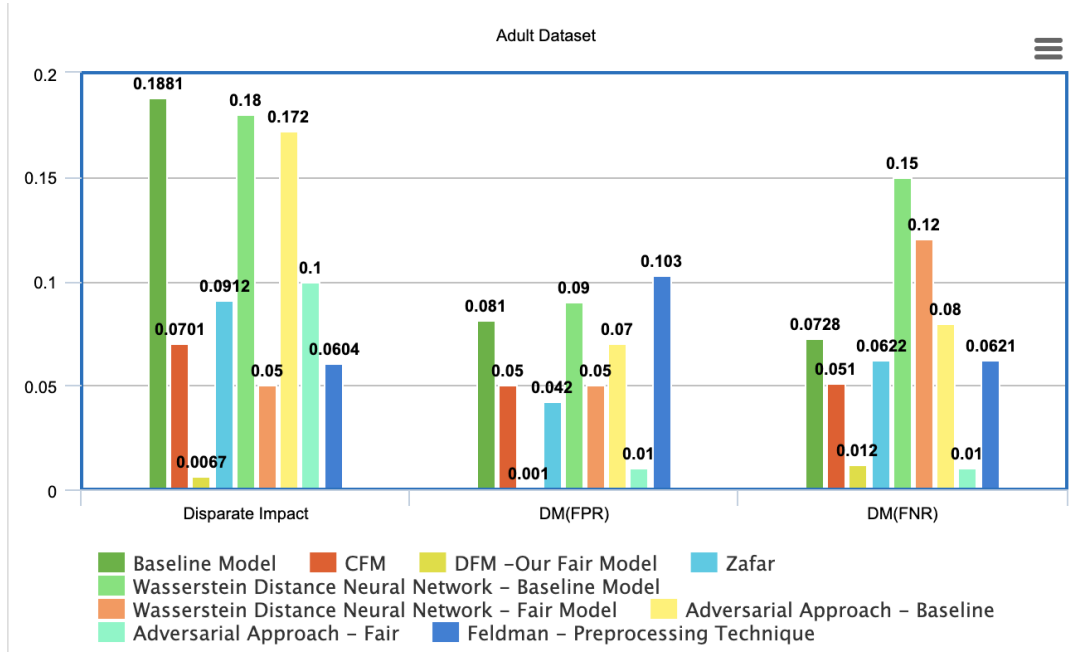


Figure 6.1: Comparison of Our fairness model with baseline model and covariance based fairness model for three fairness metrics on Adult Dataset

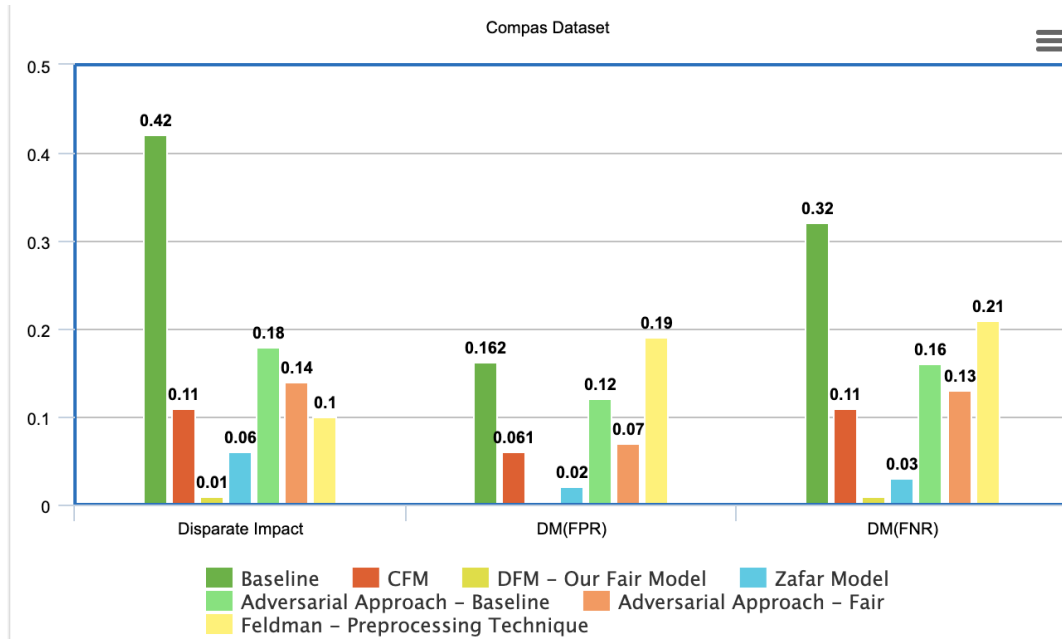


Figure 6.2: Comparison of Our fairness model with baseline model and covariance based fairness model for three fairness metrics on COMPAS Dataset

the value of disparate impact and disparate mistreatment from the above data.

So for Baseline model, the value of Disparate Impact is 0.1881. Also the value of Disparate Mistreatment (FPR) is 0.0502 and Disparate Mistreatment (FNR) is

6.2. Results

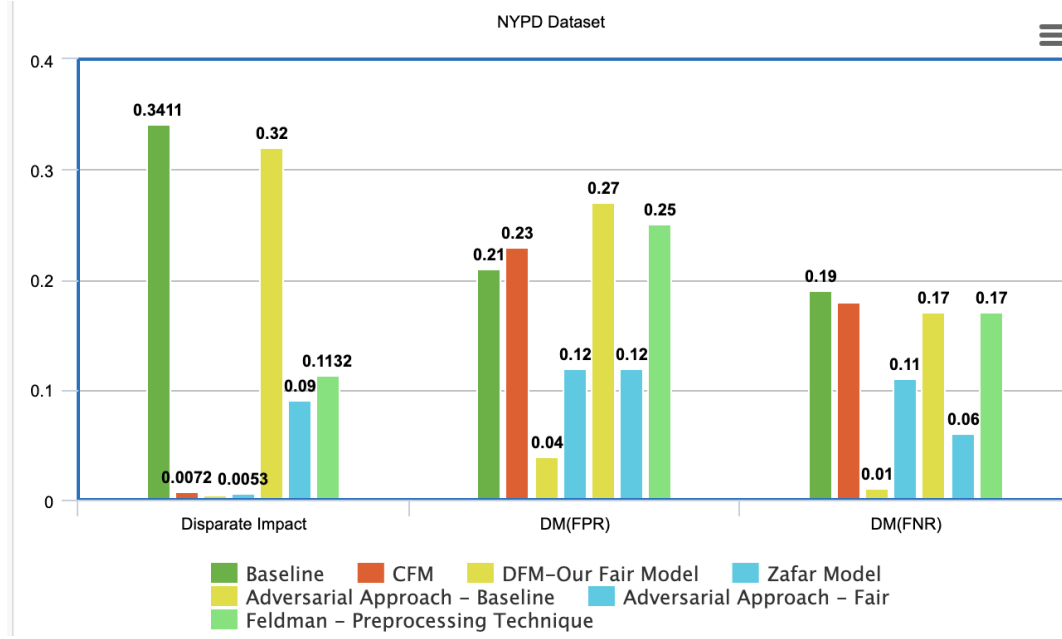


Figure 6.3: Comparison of Our fairness model with baseline model and covariance based fairness model for three fairness metrics on NYPD Dataset

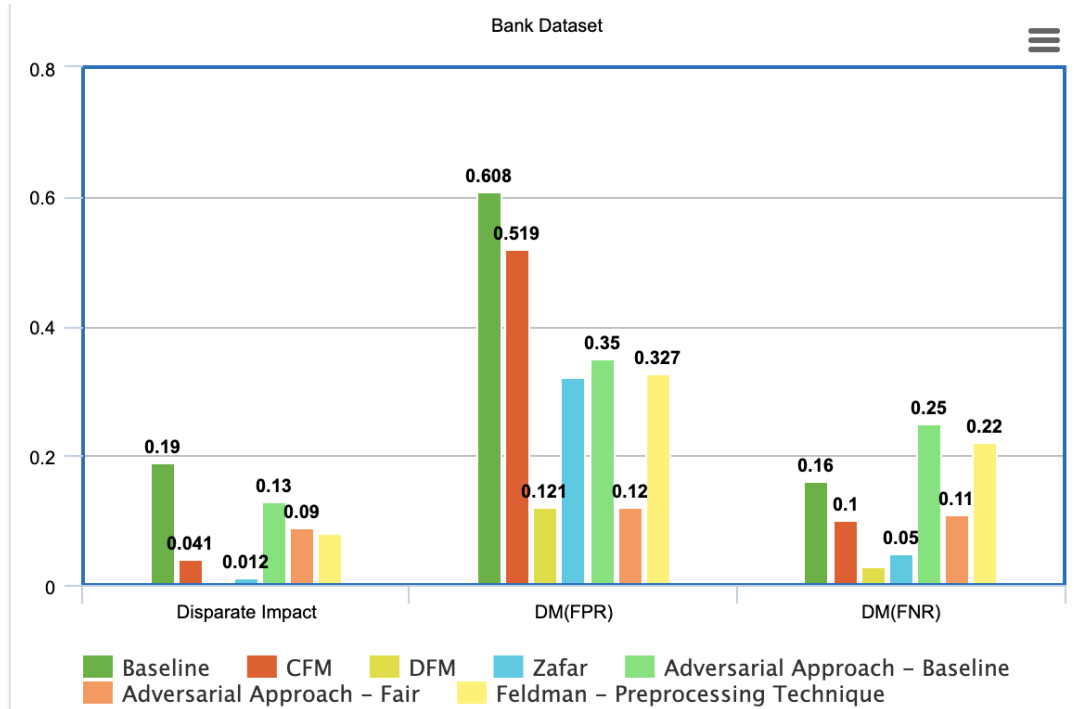


Figure 6.4: Comparison of Our fairness model with baseline model and covariance based fairness model for three fairness metrics on Bank Dataset

0.0728. We also show the comparison of our work with the previous works in table

6.3. By comparing the results of our three models, we can conclude that our two

6.2. Results

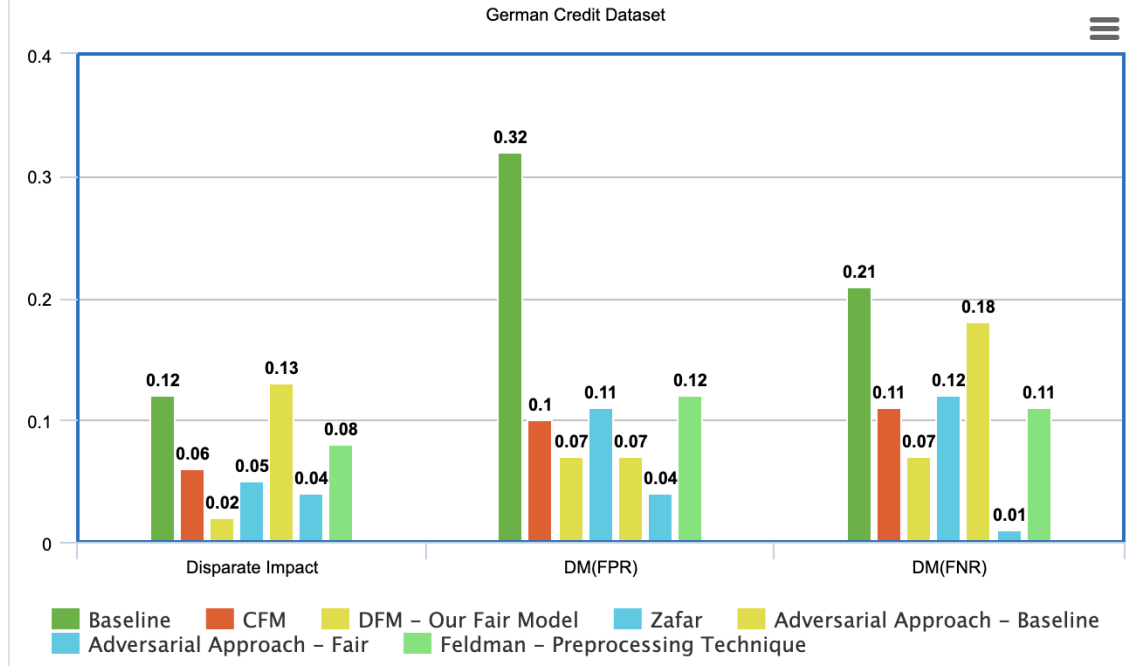


Figure 6.5: Comparison of Our fairness model with baseline model and covariance based fairness model for five fairness metrics on German Credit Dataset

Gender	Low Income	High Income	Total
Males	20988 (69%)	9352 (21%)	30527 (100%)
Females	13026(88%)	1669(12%)	14695(100%)
Total	34014 (75%)	112018 (25%)	45222 (100%)

Table 6.1: Negative (low income) and Positive (high income) labels with respect to gender attribute for adult dataset

models which have fairness constraints perform better than the baseline model in terms of fairness metric at a small drop in accuracy. Also, the second model, i.e, DFM model performs better than CFM model. The DI dropped by (0.1881 -0.0067) 0.1814 i.e reduced by 96.4% in the DFM model as compared to baseline model (BM), while DI got decreased in second Model - DFM by 63.7% compared to first model.

We show a curve between accuracy and disparate impact (fairness measures) for different values of regularizers constant λ . The DFM model shows a reduc-

6.2. Results

Label	Y=1				Y= 0			
Prediction	Y' =1		Y' =0		Y' =1		Y' =0	
Gender	Male	Female	Male	Female	Male	Female	Male	Female
Count	1202	165	771	151	437	60	3749	2510
Accuracy	$\frac{1202 + 165 + 3749 + 2510}{9045} = 84.31\%$							

Table 6.2: BM model results on adult dataset

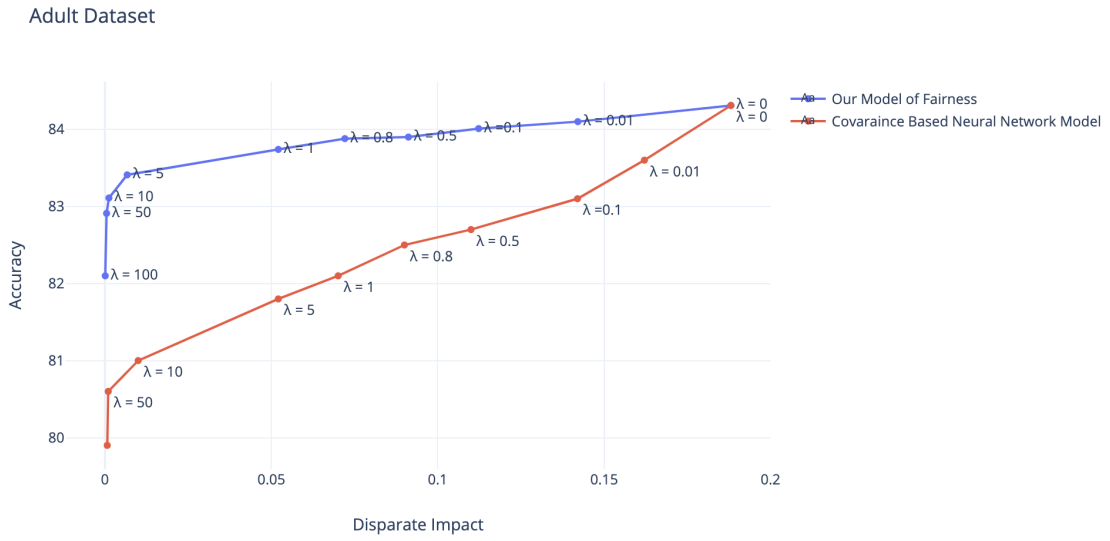


Figure 6.6: Curve between accuracy and disparate Impact for Adult Data set for Fairness Model 1 and Fairness Model 2

tion of 92.5% in Disparate Impact as compared to Zafar sensitive attribute model.

Our model performs better than other two models as well. In terms of Disparate Mistreatment as well, our DFM model shows a drastic reduction in its values as compared to our baseline model with an accuracy of 80.13%.

6.2. Results

Model	Acc with DI cons	DI	DM_{FPR}	DM_{FNR}
BM	84.31	0.1881	0.081	0.0728
CFM	82.10	0.0701	0.050	0.0510
DFM	84.12	0.0067	0.001	0.0120
Zafar baseline	84.00	0.1900	0.051	0.0811
Zafar sens. attr.	83.31	0.0912	0.042	0.0622
Wasserstein NN-BM	82.00	0.1800	0.090	0.1500
Wasserstein NN-FairModel	78.00	0.1100	0.0500	0.1200
Feldman	82.12	0.0604	0.103	0.0621
Kamishima	82.54	0.1033	0.101	0.1201
Adversarial NN- BM	86.40	0.1720	0.0669	0.0825
Adversarial NN- Fair	80.50	0.1000	0.0054	0.0109

Table 6.3: Comparison of three models of this work with Previous Works on Adult dataset

6.2.3 COMPAS data set:

The dataset is divided into train and test set. Table 6.4 shows the distribution of sensitive features and class labels in COMPAS dataset used in evaluation. Figure

Race	Yes	No	Total
White	1661 (52%)	1514 (48%)	3175 (100%)
Black	822 (39 %)	1281 (61%)	2103 (100%)
Total	2843 (47%)	2795 (53%)	5728 (100 %)

Table 6.4: Positive (Yes) and Negative (No) labels with respect to race attribute for COMPAS dataset

6.7 shows a tradeoff curve between Disparate Impact and Accuracy. Table 6.5 shows the accuracy, disparate impact and disparate mistreatment values for baseline model, DFM model and CFM model. We also include the previous results to show a better

6.2. Results

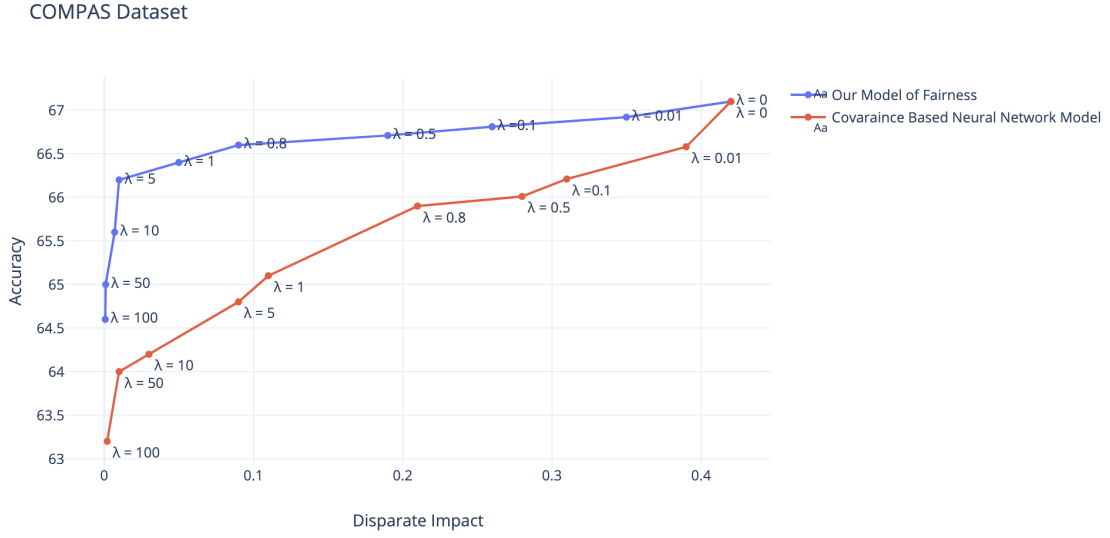


Figure 6.7: Curve between accuracy and disparate Impact for COMPAS Data set for Fairness Model 1 and Fairness Model 2

comparison between our work and previous works.

Model	Acc with DI cons	DI	DM_{FPR}	DM_{FNR}
BM	67.1	0.42	0.162	0.32
CFM	65.1	0.11	0.061	0.11
DFM	66.2	0.01	0.002	0.01
Zafar baseline	66.4	0.32	0.180	0.29
Zafar sens. attr.	63.3	0.06	0.020	0.03
Feldman	64.1	0.10	0.190	0.21
Kamishima	64.7	0.12	0.110	0.31
Adversarial NN- BM	66	0.18	0.120	0.16
Adversarial NN- Fair	65	0.14	0.070	0.13

Table 6.5: Comparison of three models of this work with Previous Works on COMPAS dataset

We note that DFM model shows a reduction of 97.6% in DI as compared to our baseline model. It also shows a reduction in DM_{FPR} and DM_{FNR} with an accuracy rate of 63.2 %.

6.2. Results

6.2.4 NYPD data set:

The dataset is divided into train and test set. Table 6.6 shows the distribution of sensitive features and class labels in NYPD dataset used in evaluation. The original NYPD dataset is quite imbalanced hence we subsample it to have equal number of subjects from each class.

Race	Yes	No	Total
Black	2113 (3%)	77337 (97%)	79450
White	803 (15%)	4616 (85%)	5419
Total	2916 (3%)	81953 (97%)	84869

Table 6.6: Positive (Yes) and Negative (No) labels in original NYPD dataset

We show a curve between accuracy and disparate impact (fairness measures) for different values of regularizers constant λ .

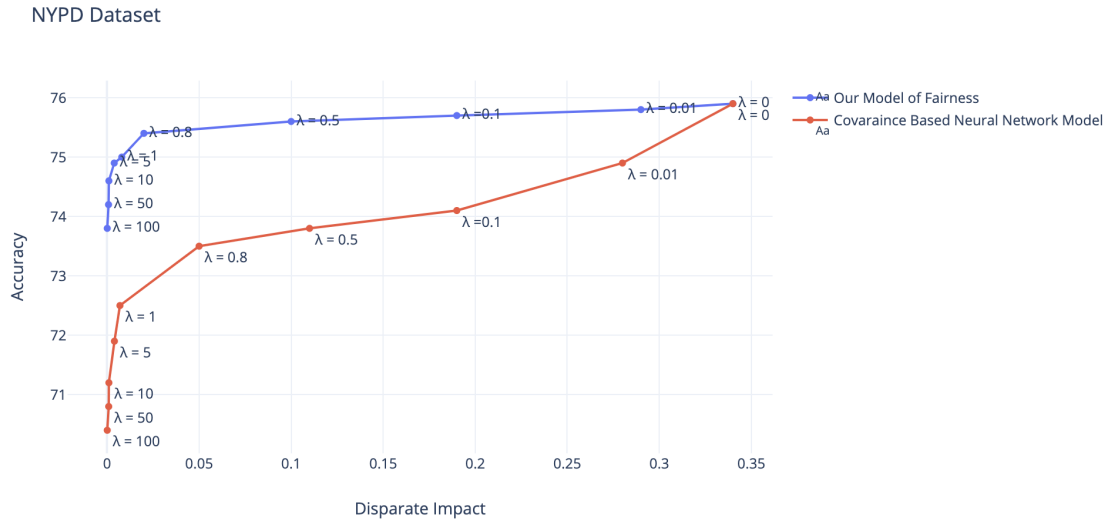


Figure 6.8: Curve between accuracy and disparate Impact for NYPD Data set for Fairness Model 1 and Fairness Model 2

6.2. Results

Model	Acc with DI cons	DI	DM_{FPR}	DM_{FNR}
BM	75.9	0.3411	0.21	0.19
CFM	72.5	0.0072	0.23	0.18
DFM	74.9	0.0039	0.04	0.01
Zafar Baseline	75.1	0.2104	0.27	0.12
Zafar Sensitive	72.6	0.0053	0.12	0.11
Feldman	72.1	0.1132	0.25	0.17
Kamishima	71.4	0.1519	0.2	0.19
Adversarial NN- BM	76.0	0.3200	0.27	0.17
Adversarial NN- Fair	72.0	0.0900	0.12	0.06

Table 6.7: Comparison of three models of this work with Previous Works on NYPD dataset

Table 6.7 shows the accuracy, disparate impact and disparate mistreatment values for baseline model, DFM model and CFM model. We also include the previous results to show a better comparison between our work and previous works. We note that DFM model shows a reduction of 98.8%, 80.8% and 94.7% in DI , DM_{FPR} and DM_{FNR} respectively, as compared to our baseline model. The model achieved accuracy of 72 % with DM constraints.

6.2.5 Bank data set:

The dataset is divided into train and test set. Table 6.8 shows the distribution of sensitive features and class labels in data set used in evaluation. We show a curve between accuracy and disparate impact (fairness measures) for different values of regularizers constant λ .

Table 6.9 shows the accuracy, disparate impact and disparate mistreatment values

6.2. Results

Age	Yes	No	Total
$25 \leq \text{age} \leq 60$	3970 (10%)	35240 (90%)	39210
$\text{age} < 25 \text{ or } \text{age} > 60$	670 (34%)	1308 (66 %)	1978
Total	4640 (11%)	36548 (89%)	41188

Table 6.8: Positive (Yes) and Negative (No) labels in Bank dataset

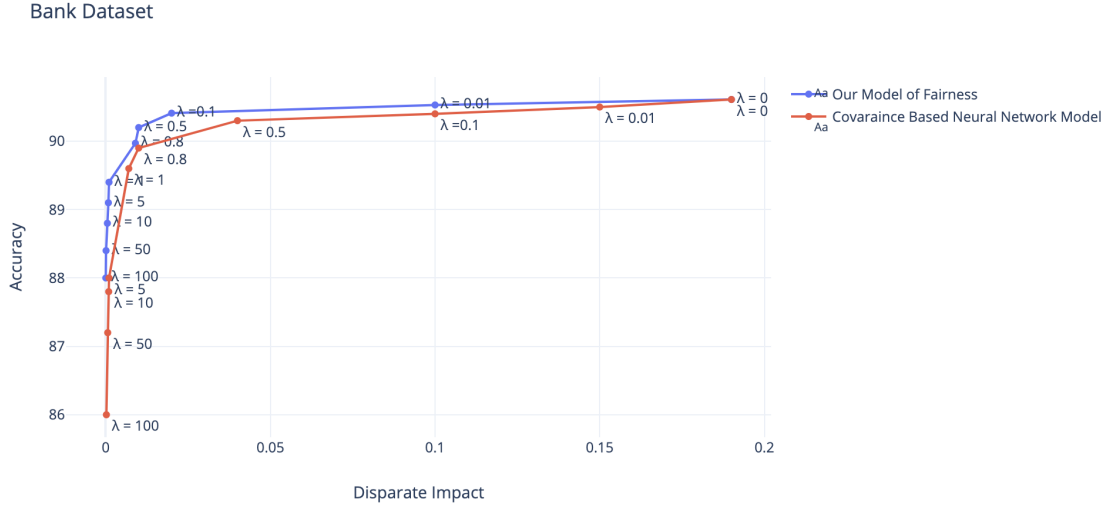


Figure 6.9: Curve between accuracy and disparate Impact for Bank Data set for Fairness Model 1 and Fairness Model 2

for baseline model, DFM model and CFM model. We also include the previous results to show a better comparison between our work and previous works. We note that DFM model shows a reduction of 98.9 % , 80.2% and 81.25% in DI, DM_{FPR} and DM_{FNR} respectively as compared to our baseline model. The accuracy observed in our model with DM constraints is 89.2%.

6.2.6 German Credit data set:

The dataset is divided into train and test set. Table 6.10 shows the distribution of sensitive features and class labels in German Credit dataset used in evaluation.

6.2. Results

Model	Accuracy	DI	DM_{FPR}	DM_{FNR}
BM	90.61	0.190	0.608	0.16
CFM	90.31	0.041	0.519	0.10
DFM	90.41	0.002	0.121	0.03
Zafar Baseline	90.10	0.191	0.672	0.15
Zafar Sensitive	90.02	0.012	0.32	0.05
Feldman	89.82	0.081	0.327	0.22
Kamishima	90.13	0.110	0.104	0.21
Adversarial NN- BM	90.00	0.13	0.35	0.25
Adversarial NN- Fair	89	0.09	0.12	0.11

Table 6.9: Comparison of three models of this work with previous works on Bank dataset

Gender	Yes	No	Total
Male	512 (74%)	178 (26 %)	690 (100 %)
Female	188 (61%)	122 (39%)	310 (100 %)
Total	700 (70 %)	300 (30%)	1000 (100 %)

Table 6.10: Positive (Yes) and Negative (No) labels in German Credit dataset

Figure 6.10 shows the curve between Disparate Impact and Accuracy and it is notable that there is accuracy drop for lower values of Disparate Impact. Table 6.11 shows the accuracy, disparate impact and disparate mistreatment values for baseline model, DFM model and CFM model. The DFM model with DM constraints gives accuracy of 72%.

6.2. Results

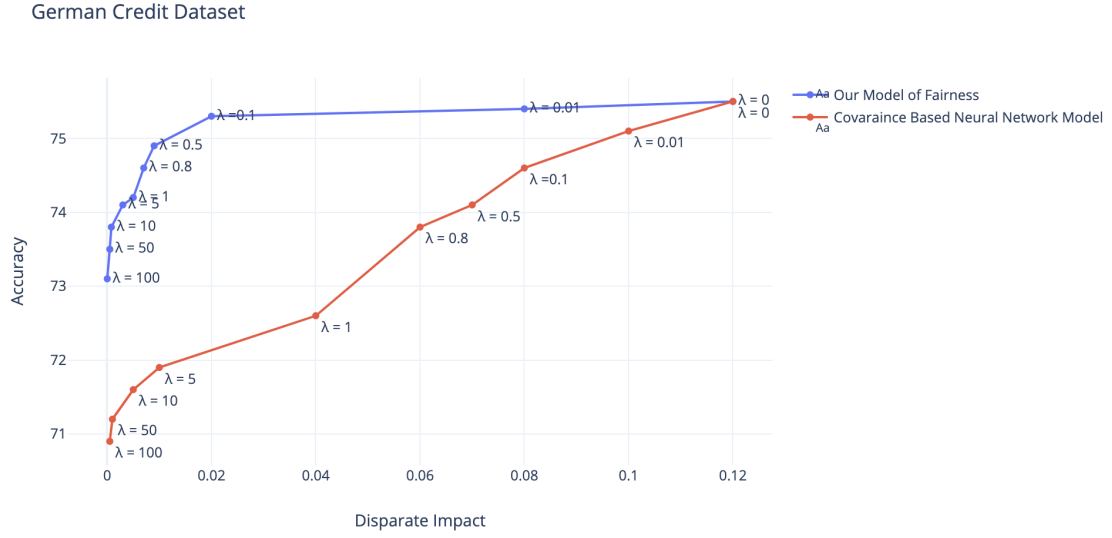


Figure 6.10: Curve between accuracy and disparate Impact for German Credit Data set for Fairness Model 1 and Fairness Model 2

Model	Accuracy	DI	DM_{FPR}	DM_{FNR}
BM	75.5	0.12	0.32	0.21
CFM	73.8	0.06	0.10	0.11
DFM	75.3	0.02	0.07	0.07
Zafar Baseline	73.6	0.15	0.29	0.22
Zafar Sensitive	72.1	0.05	0.11	0.12
Feldman	72.3	0.08	0.21	0.14
Kamishima	72.5	0.08	0.12	0.11
Adversarial NN- BM	75	0.13	0.07	0.18
Adversarial NN- Fair	70	0.04	0.04	0.01

Table 6.11: Comparison of three models of this work with previous works on German Credit dataset

6.2.7 Multiple Sensitive Attribute Results

We considered multiple sensitive attribute for two datasets - Adult dataset, COMPAS. For the adult dataset we considered sex and race as our two sensitive attributes.

6.2. Results

For COMPAS also race and sex are considered. Following table 6.12 shows Disparate Impact values for the two datasets when multiple attributes are considered. DI1 and DI2 represents the Disparate Impact caused with respect sensitive attribute sex and race respectively. For this problem, we have two models one is Baseline Model (BM) which does not take into account any fairness constraint while the other model is Direct Fairness Model (DFM) which involves the fairness constraints as discussed in 5. We note that the Disparate impact reduces significantly by 95.2% and 77.2%

Dataset	Model	Accuracy	DI -1	DI-2
Adult	BM	83.02	0.1955	0.5463
	DFM	81.07	0.0093	0.1243
COMPAS	BM	66.3	0.412	0.345
	DFM	63.2	0.056	0.081

Table 6.12: Comparison of baseline model with fairness constrained model in multiple sensitive attribute case

for sensitive attribute gender and race, respectively in Adult dataset. Also, we can note that disparate impact has reduced in COMPAS dataset as well with a drop of 86.4% and 76.8 % for sensitive attribute race and gender, respectively.

We also notice that there is a small drop in accuracy when multiple constraints are applied. However, this drop is more than the accuracy drop when only single sensitive attribute was considered.

6.2.8 Mitigating Disparate Impact and Disparate Mistreatment Simultaneously

We consider two fairness metrics simultaneously and note the results on five datasets.

We have two models for each dataset. One is baseline model which does not take into account any fairness measure into account and another is fair model in which we mitigate DI and DM simultaneously. We tune the parameters for lambda and alpha such that we could get the best model with maximum accuracy and minimum DI and DM.

Dataset	Model	Accuracy	DI	DM(FPR)	DM(FNR)
Adult Dataset	Baseline	84.31	0.1881	0.081	0.07
Adult Dataset	Fair	81.2	0.0912	0.051	0.04
COMPAS Dataset	Baseline	67.1	0.4200	0.162	0.32
COMPAS Dataset	Fair	64.2	0.1200	0.093	0.12
NYPD Dataset	Baseline	75.9	0.3411	0.210	0.19
NYPD Dataset	Fair	71.2	0.1200	0.020	0.10
Bank Dataset	Baseline	90.61	0.1900	0.608	0.16
Bank Dataset	Fair	89.2	0.0120	0.423	0.08
German Credit	Baseline	75.5	0.1200	0.320	0.21
German Credit	Fair	72.1	0.0600	0.360	0.15

Table 6.13: Results on five datasets for mitigating DI and DM simultaneously.

In table 6.13, we note that though there is drop in accuracy in fair model as compared to baseline model but DI and DM is reduced in almost all of them by a good margin.

Chapter 7

Discussions and Analysis

Many works have been done in the fairness awareness machine learning domain which lays down different approaches. We already learned some techniques introduced in section 2.6 regarding the same. In this chapter, we show some existing methods and give a contrast that how our method differs from their method. We also show that what different our method has tried to achieve.

Table 7.1 shows a comparison of different fairness strategies in handling three parameters - disparate impact, disparate mistreatment, and multiple sensitive attributes separately. A cross indicates that they do not solve the particular issue whereas tick indicates that the issue has been taken into account.

So it is clear that only our model works on multiple sensitive attribute things. We also show the difference in the techniques applied to achieve fairness. The approach suggested by Kamiran uses decision tree induction and splits the dataset D based

Method	DI	DM	Mult. sens var	Machine learning method
Our framework	✓	✓	✓	Neural Network
Kamiran and Calders Method [24]	✓	✗	✗	Any score based
Calders and Verwer Method [6]	✓	✗	✗	Naive Bayes
Kamishima Method [25]	✓	✗	✗	Logistic Regression
Feldman Method	✓	✗	✗	Any score based
Wasserstein Method [28]	✓	✗	✗	Neural Network
Dwork Method [11]	✓	✓	✗	Any score based
Zafar Method [35]	✓	✓	✗	SVM Classifier and Logistic Regression
Zhang Adversarial Method [37]	✓	✓	✗	Neural Network

Table 7.1: Capabilities of different methods in handling Disparate Impact (DI), Disparate Mistreatment (DM) and multiple sensitive attributes separately.

on the attribute leading to the highest information gain until only leafes in which all datapoints share the same ground truth remain. The information gain over ground truth is defined as:

$$IG_Y = H_Y(D) - \sum_{i=1}^k \frac{|D_i|}{|D|} H_Y(D_i) \quad (7.1)$$

where H_Y denotes the entropy with respect to the ground truth. Similarly, by accounting for the entropy for the entropy H_Z over the protected attribute, the discrimination gain can be measured by

$$IG_Z = H_Z(D) - \sum_{i=1}^k \frac{|D_i|}{|D|} H_Z(D_i) \quad (7.2)$$

For determining the most suitable split during training time, three different combinations of IG_Y , IG_Z are possible: $IG_Y - IG_Z$ to only allow non-discriminatory splits, IG_Y / IG_Z to make the trade-off between accuracy and fairness, and $IG_Y + IG_Z$ to

increase both, accuracy and unfairness.

Calders and Verwer introduce a fairness-aware algorithm popularly known as Two Naive Bayes. They trained separate models for the different groups and iteratively measures the fairness of the combined model using the CV measure, make small changes to the observed probabilities in the direction of reducing the measure, and retrains their two models. The models M_z for $z \in \{0,1\}$ are trained on $D_z = \{(X,Y,Z) \in D \mid Z=z\}$ only and, the overall outcome is determined for an individual by the outcome of the model corresponding to the individual's respective protected attribute. As the overall model depends on Z , the model can be formalized as

$$P(X, Y, Z) = P(Y|Z) \prod_{i=1}^m P(X^{(i)}|Y, Z) \quad (7.3)$$

Kamishima introduces a fairness-focused regularization term and applies it to a logistic regression classifier. It aims to reduce the prejudice learned by the model. To measure the prejudice, Kamishima defined prejudice index:

$$PI = \sum_{Y,Z} P(Y, Z) \ln \frac{P(Y, Z)}{P(Y)P(Z)} \quad (7.4)$$

Let ϕ denote the parameters of the prediction model f . The prejudice removing regularisation term (for fairness) is defined as -

$$R_{PR}(D, \phi) = \sum_{(x,z) \in D} \sum_{y \in \{0,1\}} h(x; \phi) \ln \frac{P(Y' = y|Z = z)}{P(Y' = y)} \quad (7.5)$$

Feldman gives a preprocessing approach that modifies each attribute so that the marginal distributions based on the subsets of that attribute with a given sensitive

value are all equal; it does not modify the training labels.

In Wasserstein Distance Method learn fair representations through neural networks by introducing a regularisation term known as Wasserstein Distance. The Wasserstein-2 distance between the two conditional distributions is defined as

$$W_2^2(\mu_{(\theta 0)}, \mu_{\theta 1}) = \int_0^1 (H_0^{-1}(\tau) - H_1^{-1}(\tau))^2 d\tau \quad (7.6)$$

Here, $H_g^{-1}(\tau)$ is the τ 'th quantile of the observed values of feature g with favourable outcomes. Dwork proposes a decoupled classifier that is they train different classifiers for different subgroups and obtains a joint loss to penalize unfairness. The framework starts by obtaining a set of classifiers $C_z = \{C_z^{(1)}, \dots, C_z^{(k)}\}$ for each group $z \in Z$, in which the $C_z^{(i)}$ for each group $z \in Z$ differ in the number of positively classified individuals from the group. The decoupled training step outputs a single element yielding one classifier per group by minimizing a joint loss function. The joint loss needs to penalize unfairness as well as model the explicit trade-off between accuracy and fairness.

Zafar re-expresses fairness constraints (which can be nonconvex) via convex relaxation. The author proposes covariance as a proxy to measure unfairness and formulate the problem as:

$$\begin{aligned} \min \quad & L(\theta) \\ \text{st.} \quad & \text{cov}(z, f(\theta)) \leq c \end{aligned} \quad (7.7)$$

They formulate the problem using the SVM loss to achieve this.

Zhang trained an adversarial network to mitigate the impact of unfairness. They

trained different models for different definitions of fairness separately. The adversary is trained for fairness metric. For Disparate Impact, their loss function is:

$$H(Z|Y') = E[-\log(Z = z|Y' = y)] \quad (7.8)$$

Our framework for the fair model introduces constraints for DI, DM through a gradient-based learning approach. We propose fairness focused regularization term and apply it to a neural network classifier. We note that our method achieves a state of the art performance in all cases and performs better in some other cases. Also, the drop in accuracy in our fairness constrained model is quite less as compared to other models. We also noted that we meet the business necessity clause i.e achieving fairness but not at the cost of fairness.

Chapter 8

Conclusions and Future Work

We have developed a neural network framework for fair classification when considering multiple sensitive attributes at the same time. Also, we added fairness constraints directly into our neural network loss function and removed the unfairness caused in classification. We basically concentrated on disparate impact and disparate mistreatment. We noted that introducing mathematical definitions as constraints to our model perform better than introducing some other definitions of fairness. Experiments were conducted on real-world data sets and compared with already existing strategies to showcase the performance of our model.

In the future, we would like to extend our model to a multi-valued sensitive variable. Also, we would work upon other measures of fairness. Since fairness cannot be achieved on the cost of accuracy, hence in our future work, our aim would be to develop the model which is fairer as well as accurate as compared to the baseline model. Also, right now we are concentrated only on binary labels, hence we can

work upon multi-class data. Developing a fair model for the multi-valued sensitive variable is also an important task to be done in near the future.

Bibliography

- [1] A. Altman. *The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University*. 2016.
- [2] C. A. Ameh and N. V. D. Broek. Increased risk of maternal death among ethnic minority women in the uk. In *The Obstetrician Gynaecologist*, pages 177–182, 2008.
- [3] J. Angwin, J. Larson, and L. Mattu, S.and Kirchner. Machine bias: There’s software used across the country to predict future criminals. and it’s biased against blacks. May 2016.
- [4] T. Bolukbasi, K.-W. Chang, J. Zou, V. Saligrama, and A. Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS’16, pages 4356–4364, Red Hook, NY, USA, 2016. Curran Associates Inc.
- [5] J. Buolamwini and T. Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In S. A. Friedler and C. Wilson, editors, *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*,

- volume 81 of *Proceedings of Machine Learning Research*, pages 77–91, New York, NY, USA, 23–24 Feb 2018. PMLR.
- [6] T. Calders and S. Verwer. Three naive bayes approaches for discrimination free classification. *Journal of Data Mining and Knowledge Discovery*, 21(2):277–292, 2010.
- [7] S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, and A. Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’17, pages 797–806, New York, NY, USA, 2017. Association for Computing Machinery.
- [8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.
- [9] S. E. Dreyfus. Artificial neural networks, back propagation, and the kelley-bryson gradient procedure. *Journal of Guidance, Control, and Dynamics*, 13(5):926–928, 1990.
- [10] D. Dua and C. Graff. UCI machine learning repository, 2017.
- [11] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. S. Zemel. Fairness through awareness. volume abs/1104.3913, 2011.
- [12] T. C. Faisal Kamiran and M. Pechenizkiy. Discrimination aware decision tree learning. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on. IEEE,*, pages 869–874, 2010.

- [13] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 259–268, New York, NY, USA, 2015. Association for Computing Machinery.
- [14] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, S. Choudhary, E. P. Hamilton, and D. Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, pages 329–338, New York, NY, USA, 2019. Association for Computing Machinery.
- [15] S. Goel, J. M. Rao, and R. Shroff. Personalized risk assessments in the criminal justice system. *American Economic Review*, 106(5):119–23, May 2016.
- [16] S. Goel, J. M. Rao, and R. Shroff. Precinct or prejudice? understanding racial disparities in new york city’s stop-and-frisk policy. *Ann. Appl. Stat.*, 10(1):365–394, 03 2016.
- [17] G. Goh, A. Cotter, M. Gupta, and M. P. Friedlander. Satisfying real-world goals with dataset constraints. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2415–2423. Curran Associates, Inc., 2016.
- [18] M. Hardt, E. Price, and N. Srebro. Equality of opportunity in supervised learning. *CoRR*, abs/1610.02413, 2016.
- [19] R. D. Hart. If you’re not a white male, artificial intelligence’s use in healthcare could be dangerous. In *Quartz*, july 2017.

- [20] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, 2001.
- [21] <https://www.propublica.org/datastore/dataset/compas-recidivism-risk-score-data-and-analysis>. Compas dataset.
- [22] <http://tinyurl.com/UCI Bank>. Bank dataset. 2014.
- [23] <http://www1.nyc.gov/site/nypd/stats/reports-analysis/stopfrisk>. Nypd stop-question and frisk dataset. 2017.
- [24] F. Kamiran and T. Calders. Classification with no discrimination by preferential sampling. In *Informal proceedings of the 19th Annual Machine Learning Conference of Belgium and The Netherlands, BENELEARN*, 2010.
- [25] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Proceedings of the 2012th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part II, ECMLPKDD'12*, pages 35–50, Berlin, Heidelberg, 2012. Springer-Verlag.
- [26] Y. Li, Y. Fu, H. Li, and S. W. Zhang. The improved training algorithm of back propagation neural network with self-adaptive learning rate. In *International Conference on Computational Intelligence and Natural Computing*, volume 94, pages 73–76, June 2009.
- [27] B. T. Luong, S. Ruggieri, and F. Turini. K-nn as an implementation of situation testing for discrimination discovery and prevention. In *Proceedings of the 17th*

- ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 502–510, New York, NY, USA, 2011. Association for Computing Machinery.
- [28] L. Risser, Q. Vincenot, N. Couellan, and J.-M. Loubes. Using wasserstein-2 regularization to ensure fair decisions with neural-network classifiers. *arXiv*, abs/1908.05783, 08 2019.
- [29] P. Tahmasebi and A. Hezarkhani. A hybrid neural networks-fuzzy logic-genetic algorithm for grade estimation. *Computers Geosciences*, 42:18 – 27, 2012.
- [30] A. N. Tikhonov and V. Y. Arsenin. Solutions of ill-posed problems. v. h. winston sons. *Journal of Biosciences and Medicines*, 1977.
- [31] S. Verma and J. Rubin. Fairness definitions explained. In *ACM/IEEE International Workshop on Software Fairness*, 2018.
- [32] N. Vigdor. Apple card investigated after gender discrimination complaints. *Newyork Times*, Nov 2019.
- [33] B. Widrow, A. Greenblatt, Y. Kim, and D. Park. The no-prop algorithm: A new learning algorithm for multilayer neural networks. *Journal of the International Neural Network Society*, 18(7):182–188, 2013.
- [34] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and P. Krishna. Fairness constraints: A flexible approach for fair classification. *Journal of Machine Learning Research*, 20(75):1–42, 2019.
- [35] M. B. Zafar, I. Valera, M. G. Rodriguez, , and K. P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *AISTATS*, 2017b.

Bibliography

- [36] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning fair representations. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 325–333, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

- [37] B. H. Zhang, B. Lemoine, and M. Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 2018.